# How Humans Affect Machine Learning: Privacy, Efficiency, and Biases

# How Humans Affect Machine Learning: Privacy, Efficiency, and Biases

**Dissertation**

der Mathematisch-Naturwissenschaftlichen Fakultät

der Eberhard Karls Universität Tübingen

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

vorgelegt von
Mohammad-Amin Charusaie
aus Mahshahr/Iran

Tübingen
2025

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen

| | |
|---|---|
| Tag der mündlichen Qualifikation: | 19.05.2025 |
| Dekan: | Prof. Dr. Thilo Stehle |
| 1. Berichterstatterin: | Dr. Samira Samadi |
| 2. Berichterstatter: | Prof. Dr. Philipp Hennig |

To the people I found and to the ones I lost in life

# Abstract

Artificial intelligence increasingly intersects with human lives, both by processing personal data and by influencing societal systems, such as decision-making. These interactions necessitate careful considerations of privacy preservation and the mitigation of societal impacts, such as algorithmic biases in decision-making. Achieving these objectives requires leveraging both computational and human resources effectively during the decision-making process. This thesis addresses these challenges, focusing on the private generation of data that preserves statistical characteristics while maintaining privacy, as well as optimizing models that integrate human input in decision-making processes.

In the realm of privacy preservation, this work introduces a novel approach to summarize and privatize data distributions while enhances the quality of the generated data by compressing the distribution in an embedding space. For decision-making systems involving human resources, referred to in this thesis as learn-to-defer (L2D) methods, this thesis introduces methods to train these models in an active and offline manner and analyzes and compares their sample complexity. This work further achieves a uniquely optimal solution for L2D systems with secondary objectives such as algorithmic fairness. Finally, it extends the idea of L2D to systems where the human expert and model prediction can be combined.

Through these contributions, this thesis advances the state of the art in privacy-preserving data generation and human-AI collaboration, addressing technical and societal challenges in the deployment of machine learning systems.

# Kurzfassung

Künstliche Intelligenz überschneidet sich zunehmend mit dem menschlichen Leben, sowohl durch die Verarbeitung persönlicher Daten als auch durch die Beeinflussung gesellschaftlicher Systeme, wie beispielsweise Entscheidungsfindung. Diese Interaktionen erfordern sorgfältige Überlegungen zur Wahrung der Privatsphäre und zur Minderung gesellschaftlicher Auswirkungen, wie etwa algorithmischer Verzerrungen in Entscheidungsprozessen. Um diese Ziele zu erreichen, ist es notwendig, sowohl rechnerische als auch menschliche Ressourcen effektiv im Entscheidungsprozess einzusetzen. Diese Dissertation befasst sich mit diesen Herausforderungen, wobei der Schwerpunkt auf der privaten Generierung von Daten liegt, die statistische Eigenschaften bewahren und gleichzeitig die Privatsphäre schützen, sowie auf der Optimierung von Modellen, die menschliche Beiträge in Entscheidungsprozesse integrieren.

Im Bereich der Wahrung der Privatsphäre stellt diese Arbeit einen neuartigen Ansatz vor, um Datenverteilungen zusammenzufassen und zu privatisieren, wobei die Qualität der generierten Daten durch die Komprimierung der Verteilung in einem Einbettungsraum verbessert wird. Für Entscheidungssysteme, die menschliche Ressourcen einbeziehen, in dieser Arbeit als ”Learn-to-Defer”(L2D)-Methoden bezeichnet, werden Methoden vorgestellt, um diese Modelle auf aktive und offline Weise zu trainieren, sowie deren Stichprobenkomplexität analysiert und verglichen. Darüber hinaus wird eine einzigartig optimale Lösung für L2D-Systeme mit sekundären Zielen wie algorithmischer Fairness erzielt. Schließlich wird die Idee von L2D auf Systeme erweitert, bei denen die Vorhersagen von menschlichen Experten und Modellen kombiniert werden können.

Durch diese Beiträge trägt diese Dissertation zur Weiterentwicklung des aktuellen Stands der Technik in der datenschutzfreundlichen Datengenerierung und der Zusammenarbeit zwischen Mensch und KI bei und adressiert sowohl technische als auch gesellschaftliche Herausforderungen bei der Implementierung von maschinellen Lernsystemen.

# Acknowledgments

I would like to celebrate writing this thesis with everyone who has helped me, intrigued me, challenged me, discussed with me and kept me excited during the course of my PhD. First and foremost, I owe what I earned to my father, mother, and brother. My brother, Hadi, made my PhD possible by covering the ticket to Germany and the rent of the first month in Tübingen. My mother and her constant care and concern has reminded me of my relevance and I am thankful for her presence. My father, with his eager enthusiasm for learning, has inspired me daily to persist and seek knowledge. I thank him for all that and hope for his continued well-being.

I further thank my advisor, Dr. Samira Samadi, for offering me this incredible opportunity and for her guidance and support during this time.

I would like to extend my appreciation to Lea Braüner for the deep and enthusiastic scientific and philosophical discussions we shared upon my arrival to Germany and beyond which made me more of a critical thinker. I thank Clemens Sauter for the joyful and musical moments we shared in the WG and the intriguing discussions regarding human neuroscience and biology. I further thank Alejandra Leyva who accompanied me in sickness and in health and was a constant source of emotional support during the challenging times of my PhD. I am grateful to Nicolò Zottino for the great friendship we built, and for helping me better understand and adapt to the new culture. A special thank to Georg Grab for his thoughtful feedbacks on Defer-and-Fusion method, the many laughs we shared in the office, and the unforgettable chant he taught me. I am further thankful to Anja Trumpp and the Bieberach crew (Alex, Cari, Ela, Linus, and Philipp) for adopting me as their non-German kid. I would also like to express my gratitude to the Jürgensenstrasse WG (Andrés, Camill, Cathy, Fabio, Iliana, Pari, Serhat, Sotiris, and Susanne) for being the coolest family I could have wished for, by my move to Tübingen. I am especially thankful to Cathy (Si Yi) Meng for the hepful feedbacks regarding the generalized Neyman-Pearson lemma and its use in learn-to-defer systems.

I deeply appreciate the fruitful collaboration I had with Hussein Mozannar and David Sontag on my first work on learn-to-defer, and I thank them for inviting me to the MIT in fall of 2022. I would like to further thank my office-mates Omri Ben-Dov, Ahmad Ehyaei, and Tom Sühr for the everyday fun in the office and the interesting scientific discussions. I express my gratitude to the members of Social Foundations of Computation department - Ana-Andreea, Doro, Guanhua, Jiduan, Mina, Mila, Vivian, and Yatong - for their warm and welcoming presence. They changed my work environment to a significantly better place. I am especially thankful to André Cruz for our timely discussions on algorithmic fairness and for presenting my work in NeurIPS 2024, Florian Dorner

for the discussions we had in various areas of statistics, math and machine learning, and Ricardo Dominguez-Olmedo for the great times we shared both during and outside of work. A special thank to my older brother, Michael Muehlebach for his unwavering scientific, administrative, and emotional supports during my PhD, which played a vital role in its completion.

I thank International Max-Planck Research School and Tübingen AI Center for funding my PhD and my advisor. I thank my thesis committee - Prof. Dr. Moritz Hardt, Prof. Dr. Philipp Hennig, and Prof. Dr. Ulrike von Luxburg - for their scientific feedbacks during my PhD.

Lastly, a special note of gratitude goes to the immigration officers and policymakers of the United States and Canada, whose dedication in enforcing systemic barriers ensured that none of the materials in this thesis could be presented at any conference by their authors.

# Contents

# Chapter 1

# Introduction

The field of machine learning has gone through a drastic change in the last few years, from a rather simple take on regression and classification problems, which were used on image or tabular datasets, to the current discourse in which artificial intelligence (AI) assistants are largely prevalent to the extent that 2.5% of the world population are only subscribed to the ChatGPT. AI is more than ever is connected to the daily life of individuals. Such a shift in applications of AI seeks for further considerations on responsibility, privacy, and efficient and constructive interaction with humans. This thesis is devoted to how we can address each of these concepts in a fundamental manner.

In a nutshell, this thesis is divided into two sections, addressing two of these notions:

1. In the first section, the matter of **privacy** for synthetic data generation is discussed. This problem, which is one of the main problems in the field of privacy, aims to generate a synthetic dataset that replicates the statistics of a real dataset, while preserving the privacy of each individuals. Plainly put, by observing the synthetic dataset, we should not be able to single out an individual that participated in the real dataset. The content of this section is brought from the article:

   Vinaroz*, M., Charusaie*, M. A., Harder, F., Adamczewski, K., & Park, M. J. (2022, June). Hermite polynomial features for private data generation. In International Conference on Machine Learning 2022

2. In the second section, we discuss the **interaction of human expert and machine learning models** through the lens of efficiency and possible **algorithmic and human biases** within such interaction. In the systems that we discuss, the model can incorporate human expert prediction for a task where it is in-confident or to improve upon its inherent biases. The content of this section is brought from the three publications:

   Charusaie*, M. A., Mozannar*, H., Sontag, D., & Samadi, S. (2022, June). Sample efficient learning of predictors that complement humans. In International Conference on Machine Learning 2024

   Charusaie, M. A., & Samadi, S. (2024). A Unifying Post-Processing Framework for Multi-Objective Learn-to-Defer Problems. In Thirty-Eighth Advances in Neural Information Processing Systems

Charusaie, M. A., Fesharaki, A. J., & Samadi, S. Defer-and-Fusion: Optimal Predictors that Incorporate Human Decisions. (Under Review)

In the following, we will review the basic notions that are used in each section. We further summarize the main contributions for each problem and clarify the overall thread that connects each piece of this thesis.

## 1.1 Differentially Private Data Generation

The first part of this thesis is devoted to a method for generating samples from a distribution in a private manner. This is a problem that is emerged in a variety of applications that include learning based on sensitive records. In such cases, record holders are expected to release a dataset that replicates the statistics of the records while having a guarantee that the record of an individual is not compromised after the release.

A well-recognized measure for ensuring such guarantee is introduced in Dwork *et al.* (2006) as *differential privacy*. This measure of privacy controls the changes in a learning system, given that a particular record is within its training data or not. Formally, it ensures that for two neighboring datasets $\mathcal{D}$ and $\mathcal{D}'$, i.e., have at most one different element, and for all choices of the set $\mathcal{S}$, we have

$$\frac{\Pr(A(\mathcal{D}) \in \mathcal{S})}{\Pr(A(\mathcal{D}') \in \mathcal{S}')} \leq e^{\varepsilon}, \tag{1.1}$$

for the learning algorithm $A(\cdot)$ and a fixed small scalar $\varepsilon \in \mathbb{R}$.

This property provides us with a leakage guarantee that older privacy notions such as $k$-anonymity Sweeney (2002) do not hold. One such advantage is avoiding linkage attacks that could lead to drastic results, as of the Netflix competition privacy compromise Narayanan and Shmatikov (2008). A linkage attack uses the public data to acquire information about a single record in an anonymized dataset. By a comparison of public and anonymized data, the adversary figures to whom does a record belong. The differential privacy ensures that the auxiliary information about presence of all but one member would not disclose much information about presence of that member and therefore is robust to such attacks. Robustness to linkage attacks, however, is not the only advantage of using differential privacy. This measure further ensures the privacy of groups in the data, the privacy of composition of private algorithms Dwork *et al.* (2010), and the privacy of post-processing of the outcome of the algorithm.

Despite its theoretical promises, the notion of differential privacy is poorly implemented on modern learning methods such as deep learning. Fundamentally, having a differential privacy guarantee requires controlling the influence of each training data on the output of the learning system. However, estimating such influence (also called *influence function*) on neural networks is not readily calculable due to the effect of many

steps of stochastic gradient descent (SGD), and therefore ensuring differential privacy in such networks directly is not realizable.

The seminal work in this direction is DP-SGD (Abadi *et al.* (2016a)) that instead of privatizing the overall network privatizes the gradients in each SGD step. Formally, the weight $w_i$ of the network is obtained as the sum $w_i = w_i^0 + \frac{1}{m}\sum_{j=1}^{T}\eta_j\sum_{k=1}^{m}\nabla_{w_i}f_j(x_k)$ where $w_i^0$ is the initial weight, $\eta_j$ is the stepsize, $f_j(\cdot)$ is the network in time $j$, and $\{x_k\}_{k=1}^m$ is the training dataset. Therefore, the influence function of such network is $\max_k\left|\sum_{j=1}^{T}\eta_j\nabla_{w_i}f_j(x_k)\right|$. Hence, the perturbation that is needed for privatization of such network is of the same scale. However, this value can be arbitrarily large, since this is a worst-case measure and only one instance can change the influence function in its entirety. The DP-SGD method instead clips the gradients in each step by a value $c$ and adds the perturbation accordingly. The larger is the value $c$, the more accurate is the obtained gradient, however the larger is the required perturbation for privatization. Therefore, by such over-estimation of the influence function, in each case the final network loses the accuracy during the many times cut-off and added noise.

The DP-SGD was the primary method for training most promising generative methods at the time (such as GANs Goodfellow *et al.* (2014)) and therefore the most basic methods to privatize such networks led to inaccurate results Papernot *et al.* (2017); Xie *et al.* (2018a). Such issues with privatization of SGD-based methods, sought for replacing them with learning methods that do not use SGD for their training. One such method is a kernel method as in Harder *et al.* (2021a) that encompasses the distribution of a training dataset into a vector, called *mean embedding* without use of the SGD algorithm.

To clarify the notion of mean embedding, we first need to define *maximum mean discrepancy* (MMD), which is a distance function between two distributions. This distance function is defined as

$$MMD(P,Q) := \sup_{f\in\mathcal{F}} \mathbb{E}_{X\sim P}\big[f(X)\big] - \mathbb{E}_{X\sim Q}\big[f(X)\big], \tag{1.2}$$

where $\mathcal{F}$ is a unit ball in a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$, which is a metric space of functions with distance that is induced by an inner product. An RKHS has a useful property that is resulted from Riesz representation theorem (Reed and Simon, 1980, Theorem II.4) that for each $x$, there exists a unique function $K_x$ such that the value of all members of $\mathcal{H}$ on $x$ can be re-defined as

$$f(x) = \langle f, K_x\rangle. \tag{1.3}$$

Such set of functions form a symmetric and positive-definite *reproducing kernel* $k(x,y)$ as

$$k(x,y) = \langle K_x, K_y\rangle,$$

for each pair of $x$ and $y$.

Getting back to the definition of MMD in ((1.2)) and using ((1.3)), we observe that

$$\begin{aligned} MMD(P,Q) &= \sup_{f:\|f\|_{\mathcal{H}}\leq 1} \big\langle f, \mathbb{E}_{X\sim P}\big[K_X\big] - \mathbb{E}_{X\sim Q}\big[K_X\big]\big\rangle \\ &= \big\|\mathbb{E}_{X\sim P}\big[K_X\big] - \mathbb{E}_{X\sim Q}\big[K_X\big]\big\|_{\mathcal{H}}, \end{aligned} \tag{1.4}$$

where the last identity holds because the norm is defined as $\|f\|_{\mathcal{H}} := \langle f, f\rangle^{1/2}$.

Therefore, the MMD can be obtained using the two values

$$\mu_P := \mathbb{E}_{X\sim P}\big[K_X\big] \tag{1.5}$$

and $\mu_Q$ that are called mean embeddings. As a result, if MMD is a characteristic distance of probability distributions, i.e., if $MMD(P,Q) = 0$ if and only if $P = Q$, then, to replicate $P$, one only needs to have a distribution with the same mean embedding $\mu_P$, and therefore the distribution $P$ is compressed into the mean embedding $\mu_P$. This is the cornerstone idea of the private data generation that is brought in this thesis.

A very first issue to compress a distribution into its mean embedding is that inherently $\mu_P$ itself is a function in $\mathcal{H}$, and therefore cannot be memorized and further privatized. As a result, if such function represents an infinite-dimensional vector, we need to further compress it into a finite-dimensional vector with a similar set of properties. The main step for approximation of a mean embedding, as is expressed in this thesis, is the approximation of its corresponding reproducing kernel. As an instance, an early work of Rahimi and Recht (2007) uses random Fourier features to approximate such function. This work is based on Bochner's theorem (Unser and Tafti, 2014, Theorem B.1) that shows each shift-invariant positive-definite kernel is the characteristic function of a distribution, or equivalently for the kernel $k(\cdot,\cdot)$, there exists a distribution $S$ such that

$$K(x,y) = \mathbb{E}_{\omega\sim P}\big[e^{j\omega(x-y)}\big]. \tag{1.6}$$

As a result, a method for approximating the kernel is to form a multi-dimensional function

$$\phi(x) = [e^{j\omega_1 x}, \ldots, e^{j\omega_T x}], \tag{1.7}$$

where $\omega_i$ for $i = 1:T$ is drawn from the probability distribution $P$. Next, using some mathematical analysis, we can show that the kernel can be approximated as

$$K(x,y) \simeq \phi(x)\phi(y)^{\dagger}, \tag{1.8}$$

and therefore the MMD can be approximated as

$$MMD(P,Q) \simeq \big\|\mathbb{E}_{X\sim P}\big[\phi(X)\big] - \mathbb{E}_{X\sim Q}\big[\phi(X)\big]\big\|_2. \tag{1.9}$$

This means that $\mathbb{E}_{X\sim P}\big[\phi(X)\big]$ that is a finite-dimensional vector is a proper approximation of the mean embedding.

The above approximation of the mean embedding equips us with a method to compress a distribution into a vector without any training and therefore without using the SGD algorithm. This is the key feature that enables Harder *et al.* (2021a) to achieve a private data generation method. Although smart, this method cannot prioritize the elements of the mean embedding approximation that contain the most information regarding the distribution. This is where the contribution of this thesis lays.

In this thesis, we replace the above approximation of the mean embedding with an approximation that is based on Hermite polynomials, a sequence of polynomials that generate all eigenfunctions of a Gaussian kernel using Mehler's formula (see (2.5)). This formula shows that a Hermite polynomial of order $k$ generates an eigenfunction of such kernel, where the eigenvalue is decreasing in terms of $k$. Therefore, by forming a mean embedding by truncation of the first $k$ order of the polynomials, one will have more information regarding a distribution than the Fourier mean embedding. This is further theoretically shown in Proposition A.2.1.

The above compression of mean embedding elements further has an application in privacy of the generative model. In the early works in differential privacy and Gaussian mechanism (Dwork *et al.*, 2014, Theorem 2.22) it is shown that the overall variance of the perturbation in a vector is linearly related to the dimension of the vector. In other words, the more the number of elements of a vector, the more the information contained in that vector about an individual, and therefore the more perturbation we need to preserve the privacy of those individuals. As a result, a more compressed mean embedding, as discussed above, leads to less perturbation of that vector and therefore more accurate data generation. This and further discussions regarding the optimal dimension of the mean embedding vector is presented in Chapter 2.

The main challenge of implementing our method to compress a distribution arises due to the curse of dimensionality. As we will further discuss in Chapter 2, capturing the interdependence of different dimensions in a dataset requires computing the outer product of the aforementioned series of Hermite polynomials for each dimension, an operation that requires exponential time in terms of the dimension of the datapoints. Moreover, although further truncation of the polynomial series in each dimension decreases the base of such exponential, as discussed above, it quickly leads to a reduced quality of the generated samples.

As a solution to the above issue, in this thesis, we use two mechanisms: (i) we capture the marginal distribution of all dimensions using a mean embedding with the length that is linear in terms of datapoint dimensions, and (ii) during the training of the distribution, in each epoch, we sub-sample from all the dimensions. The first mechanism enables us to capture marginal distributions with a high quality, since there is no computational cost to force early truncation of the mean embedding in this case. The second mechanism, however, ensures that the resulting distribution is close to the original distribution in each random subset of dimensions, while reducing the base of the exponential computational

cost. We will elaborate the effect of each of these mechanisms on overall accuracy and computational costs in Chapter 2.

As mentioned, in this thesis, we study the privacy of individuals in generative models from theoretical aspects to the implementation aspects of privatizing algorithms. However, the role of human in machine learning system goes beyond their connection with the training data. As an instance, the presence of a human in the process of decision-making of a machine learning system, at least up to this point of time and for a majority of tasks, is inevitable. This has lead us to devote a large proportion of this thesis to the collaboration of human and learning systems in decision-making. An introduction to the contribution of this thesis on the analysis of such collaborative settings is brought in the next section.

## 1.2 Human and AI Collaboration

It is not hard to think of tasks that we cannot easily assign to the learning models in isolation due to sensitivity of the task and the consequences of such assignments. This is resulted in a sub-field of machine learning that studies the effect of human and AI collaboration. We can imagine such collaboration to occur in various settings, each of which having a different, and possibly complex, effect on the outcome of the overall system predictions. A simplified version of such collaboration that aims to reduce the task assignment to the human, while optimizing the overall accuracy is introduced in Madras *et al.* (2018). In this setting, which we refer to as *Learn-to-Defer (L2D)* setting, the final prediction is made either by human expert or the machine learning model and there is no further interaction between the two agents.

A L2D method assumes that the human expert is a stationary decision-maker conditioned on each task (input). As a result of such assumption, the aim of L2D method is to find regions of the input features for which the human is more accurate than the machine learning model, and vice versa, and to assign tasks within those regions accordingly. The existence of such regions, further is known as the complementarity of human experts and learning models, is studied and exhibited in some early studies, including Wilder *et al.* (2020).

To explain how training such regions of accuracy plays a role in training an L2D system, let us first define the primary objective of a L2D system as the minimization the $0-1$ loss

$$L_{\text{def}}^{0-1}(h,r) := \mathbb{E}_{X,Y,M}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{h(X)\neq Y} + \mathbb{I}_{r(X)=1}\mathbb{I}_{M\neq Y}\right],$$

where $X$ is the instance feature, $Y$ is the true label, and $M$ is the human prediction. Furthermore, $h(\cdot)$ and $r(\cdot)$ are the machine learning model and the rejection function, respectively. Equivalently, we aim to minimize a loss that is equal to the machine learning loss whenever the instance is assigned to that model, i.e., the rejection function takes

$r(x) = 0$, and is equal to the human loss otherwise.

A joint minimization of the above loss for $h(\cdot)$ and $r(\cdot)$, based on a variation of Chow's rule Chow (1970) leads to the optimizers

$$r^*(x) = \mathbb{I}_{\Pr(h(X)=Y|x) \leq \Pr(M=Y|x)},$$

and

$$h^*(x) = \arg\max_y \Pr(Y = y|x).$$

Therefore, it seems that the machine learning model in this case is the Bayes optimal classifier in isolation, and the confidence of human expert and the classifier indicates whether an instance should be deferred to the human expert.

While it seems that, theoretically, the optimal classifier in isolation is also optimal in an L2D system, Wilder *et al.* (2020) conjectures that this is not the case in practice. In fact, the inherent assumption of Chow's rule is that the hypothesis class of models are infinitely complex. However, in practice, we always induce a certain amount of smoothness in models to avoid overfitting, and therefore we restrain the hypothesis class of trained models, violating the assumption of Chow's rule.

The two training methods for finding the classifier and the rejection function are namely (i) the staged learning method Madras *et al.* (2018), in which the classifier is trained in isolation, and the rejection function optimizes the corresponding L2D loss, and (ii) the joint learning method Mozannar and Sontag (2020a); Wilder *et al.* (2020), in which the classifier and the rejection function are trained simultaneously to optimize L2D loss. In Chapter 3, we analytically show the effect of model complexity on system performance of the two above methods. This is a proof to the conjecture that is posed in Wilder *et al.* (2020). The claim of that conjecture was that the sub-optimality of staged learning compared to the joint learning is inversely related to the complexity of the models. In Chapter 3 we show that such relationship exists where model complexity is defined as the Vapnik-Chervonenkis dimension Vapnik and Chervonenkis (1971), a classical measure of richness of a hypothesis class of predictors.

Another important restraint that we study in this thesis, besides the model complexity, is the limitation of the human data acquirement. Depending on the expertise that is needed for resolving a task, the human predictions for training an L2D system can be of high cost. In Chapter 3 we show how staged learning can possibly improve over this issue. In a staged learning setting, the predictor can be trained on a dataset that does not include human expert predictions. As a result, a dataset that is not labeled by human expert and has a size larger than the labeled dataset can improve the accuracy of the predictor, and in in turn overall accuracy, compared to the joint learning method that cannot use the unlabeled dataset.

We move to extend over this result to an active learning setting, in which the human expert prediction is sought only when it is needed. To that aim, we assume that the human

expert correctness $\mathbb{I}_{M=Y}$ is a deterministic function of the predicator $x$. Then, in Chapter 3 we devise an algorithm that in each iteration updates a set of possible predictors of this correctness. More formally, this algorithm finds an instance $x$ for which there are two possible predictors in such set that lead to different predictions for that instance. Then, it acquires human expert prediction for that instance and removes the incorrect predictor from the set. Such algorithm that is a disagreement-based active learning method Hanneke (2014) leads to the complexity of $O(\log \frac{1}{\varepsilon})$, where $\varepsilon$ is the target error, as opposed to $O(\frac{1}{\varepsilon^2} \log \frac{1}{\varepsilon})$ for offline learning methods.

The limitations on the human expert budget can be further studied through the lens of constrained learning. Equivalently, we can reformulate the L2D problem by penalizing the deferral to the human in order to reach a certain deferral proportion for the dataset. This approach was primarily studied by Okati *et al.* (2021a) and the Bayes optimal deferral system was obtained by a thresholding rule over the L2D system scores. This observation that a thresholding rule is an optimal solution to a constrained L2D problem is the motivating observation that initiates Chapter 4. This chapter answers the question that to what extent such a result holds for other losses and constraints.

Thresholding rules are repetitively discovered in modern machine learning literature, from the basic maximization of the scores as the Bayes optimal classifier, to the optimal fair classifiers Hardt *et al.* (2016); Cruz and Hardt (2023). The basic idea of these rules is that finding probability scores of some kind is essential for obtaining an optimal predictor. As an instance, for finding the optimal classifier for a general cost-sensitive loss on a dataset we need to (i) find an estimation of the probability score $\Pr(Y = 1|X = x)$, and (ii) threshold that score based on the definition of the loss. We observe that the first step is independent of the loss function, a phenomenon that is similarly occured for obtaining an optimal fair classifier Hardt *et al.* (2016) for which the scores are primarily estimated and are thresholded consequently and based on the measure of algorithmic fairness. The foundation of such methods, also referred to as *post-processing* methods, are extended in Chapter 4 to a large class of losses and constraints.

In this thesis, we argue that the post-processing methods are by and large explained by an extension of the fundamental Neyman-Pearson lemma Neyman and Pearson (1933), a lemma that is primarily studied to find a hypothesis testing method that is optimal in terms of false positive error, while keeps the false negative error under control. This lemma that obtains such testing method by thresholding the likelihood ratio of the null hypothesis and the alternative hypothesis is extended in Chapter 4 to obtain optimal solution to a set of classification as well as L2D problems with constraints on algorithmic fairness, anomaly detection, human expert budget, long-tail classification, and Type-I/II error. This chapter explains that for each loss function as well as each constraint that is dependent on the input features and the outcome of a classification or an L2D system, there exist an embedding function. Next, the optimal constrained solution is obtained via an ensemble of such embedding functions, a method that we name $d$-dimensional generalized Neyman-Pearson, or *$d$-GNP*, method

We further study a series of negative results regarding constrained L2D problem in Chapter 4. We start off by the hardness of the constrained L2D problem when policies are deterministic and show that this problem and the $0-1$ Knapsack problems are equivalent, an evidence that the deterministic constrained L2D problem is NP-Hard. Next, we take in-processing algorithms off the table by introducing a negative result on their generalization. To elaborate, we show that if we treat the deferral to the human expert as a label and make an attempt to learn that label, we face the ambiguity in reconstruction of the optimal deferral policy. Indeed, we show that there are two distributions for which the deferral labels are equal in distribution, while the optimal deferral policy is different and not interchangeable. We argue that such issue occurs because of an attempt to compress the two bits of information regarding the human expert and the classifier accuracy into one bit of deferral label.

In Chapter 4, we further show statistical generalization results for the $d$-GNP method. This results ensure that if we have an estimation of the actual scores in a dataset, then the ensembling method based on such estimations lead to solutions that firstly preserve the constraints approximately, and secondly are close in terms of the primary objective to the Bayes optimal solution. We further explain that the closeness of the empirical solution and the Bayes optimal solution depends on (i) the size of the validation dataset using which the rule of ensembling is extracted, (ii) the accuracy of the score estimations, and (iii) the smoothness of the constraints in terms of the variation in the prediction rule, a value that is merely dependent on the data distribution.

The final chapter of this thesis is an antithesis to the idea of learn-to-defer in which the final decision is made either by the predictor or the human expert. We start this chapter by two examples that show sub-optimality of a L2D system for finding the optimal predictor based on the human expert and the classifier. These examples suggest that the information within the mere prediction of each of these agents is not sufficient to obtain a final prediction, and a fusion of these predictions is needed. To resolve that issue, we introduce a system, named Defer-and-Fusion, using which the prediction is either made by classifier, human expert, or a fusion of both predictions.

In Chapter 5, we mainly study two problems, firstly how to optimally fuse the two predictions, and secondly when to fuse the predictions. We argue that the solution to the first issue is to treat the prediction of either of the agents as the context of the classification, and find a classifier using such contexts and typical classification methods. We further explain that the hardship in the second issue is that, as opposed to the L2D problems for which the estimation of human confidence is needed, in the Defer-and-Fusion method we need the confidence of the fusion component to obtain the deferral policy. To obtain the estimation of such confidence, we propose two methods: (i) we simulate the human expert for each input, i.e., we find the probability of the decisions for which the human expert takes in average for a certain input, and find the mean confidence of the fusion component in response to each of these decisions, and (ii) we learn the confidence of the fusion component directly from the training dataset.

Moreover, in Chapter 5, we discuss that a mere combination of the human expert and

the classifier prediction, as studied in Kerrigan *et al.* (2021), is not as accurate as L2D or Defer-and-Fusion methods. Such combination methods miss the main idea behind L2D methods that certain regions of the input space induce various confidence for human expert and the classifier, and averages the effect of those regions by treating the predictions of these agents equally for all input features. However, we show that the use of the Defer-and-Fusion method for synthetic and real datasets significantly improves over the L2D and combination baselines in terms of accuracy as well as cost-sensitive losses.

The issues regarding the implementation of the L2D methods is extensively studied in this thesis. In Chapter 3, a family of surrogate methods is defined for joint training of the predictor and rejection function. This family surrogate functions are introduced to tackle the discontinuity of the $0-1$ loss function. Minimization of each member of this family, which extends logistic, exponential, and squared loss function, leads to a set of scores that could produce the Bayes optimal solution. This property that is named Fisher consistency is further followed by its stronger approximate variation for the introduced family of surrogates, which connects the convergence of the trained model in surrogate loss to the convergence in $0-1$ loss. The introduced family of surrogates were of particular interest in the past few years and has been further studied and extended Cao *et al.* (2022); Liu *et al.* (2024).

Moreover, the main objective of Chapter 4 is obtaining a practical solution for the constrained learning problem. We argue that by training two neural networks, one for estimating the embedding function of the objective and one for estimating the embedding function of the constraint, one can achieve the optimal solution of that problem. This is an alternative to the primal-dual methods (Chamon *et al.*, 2022) that require training based on a regularized loss and updating the regularization cost, where takes longer to converge and could possibly lead to an oscillation due to sub-optimality of the primal parameters after each update.

Furthermore, in Chapter 5 a variety of joint training of classifier, rejection function, and fusion is introduced. This includes methods that train these components using One-vs-All losses Verma and Nalisnick (2022a) or train them by softmax losses and updating each component iteratively. We discuss the calibration of each of these methods in Chapter 5.

Finally, as mentioned, the material in this thesis consider a simplified view on the systems that keep human experts in the loop. In this view, the human expert is observed as a stationary source of prediction that neither shifts over the time, nor is affected by the overall predictions or the error of the system. This leaves a set of problems for future studies: first, to verify the above assumption across an extensive set of decision-making settings; second, to identify examples where such assumptions need to be relaxed to non-stationary and performative ones; and finally, to develop a general algorithmic framework that optimizes the collaboration between human experts and machine learning systems under such relaxations.

# Chapter 2

# Hermite Polynomial Features for Private Data Generation

*This chapter is based on the joint work with Margarita Vinaroz, Frederik Harder, Kamil Adamczewski, and Mijung Park. I thank their contributions in shaping ideas, implementation of the method, and preparation of the paper.*

## 2.1 Introduction

One of the popular distance metrics for generative modelling is *Maximum Mean Discrepancy* (MMD) Gretton *et al.* (2012). MMD computes the average distance between the realizations of two distributions mapped to a reproducing kernel Hilbert space (RKHS). Its popularity is due to several facts: (a) MMD can compare two probability measures in terms of all possible moments (i.e., infinite-dimensional features), resulting in no information loss due to a particular selection of moments; and (b) estimating MMD does not require the knowledge of the probability density functions. Rather, MMD estimators are in closed form, which can be computed by pair-wise evaluations of a kernel function using the points drawn from two distributions.

However, using the MMD estimators for training a generator is not well suited when *differential privacy* (DP) of the generated samples is taken into consideration. In fact, the generated points are updated in every training step and the pair-wise evaluations of the kernel function on generated and true data points require accessing data multiple times. One of the key properties of DP is composability that implies each access of data causes privacy loss. Hence, privatizing the MMD estimator in every training step – which is necessary to ensure the resulting generated samples are differentially private – incurs a large privacy loss.

A recent work Harder *et al.* (2021b), called *DP-MERF*, uses a particular form of MMD via a *random Fourier feature* representation Rahimi and Recht (2008) of kernel mean embeddings for DP data generation. Under this representation, one can approximate the MMD in terms of two finite-dimensional mean embeddings (as in eq. (2.3)), where the approximate mean embedding of the true data distribution (data-dependent) is detached from that of the synthetic data distribution (data-independent). Thus, the data-dependent

term needs privatization only once and can be re-used repeatedly during training of a generator. However, DP-MERF requires an excessively high number of random features to approximate the mean embedding of data distributions.

We propose to replace[1] the random feature representation of the kernel mean embedding with the *Hermite polynomial* representation. We observe that Hermite polynomial features are ordered where the features at the low orders contain more information on the distribution than those at the high orders. Hence, the required order of Hermite polynomial features is significantly lower than the required number of random features, for the similar quality of the kernel approximation (see Fig. 2.1). This is useful in reducing the *effective sensitivity* of the data mean embedding. Although the sensitivity is $\frac{1}{m}$ in both cases with the number of data samples $m$ (see Sec. 2.3), adding noise to a vector of longer length (when using random features) has a worse signal-to-noise ratio, as opposed to adding noise to a vector of shorter length (when using Hermite polynomial features), if we require the norms of these vectors to be the same (for a limited sensitivity). Furthermore, the Hermite polynomial features maintain a better signal-to-noise ratio as it contains more information on the data distribution, even when Hermite polynomial features are the same length as the random Fourier features

To this end, we develop a private data generation paradigm, called *differentially private Hermite polynomials* (DP-HP), which utilizes a novel kernel which we approximate with Hermite polynomial features in the aim of effectively tackling the privacy-accuracy trade-off. In terms of three different metrics we use to quantify the quality of generated samples, our method outperforms the state-of-the-art private data generation methods at the same privacy level. What comes next describes relevant background information before we introduce our method.

## 2.2 Background

In the following, we describe the background on kernel mean embeddings and differential privacy.

### 2.2.1 Maximum Mean Discrepancy

Given a positive definite kernel $k\colon \mathcal{X} \times \mathcal{X}$, the MMD between two distributions $P, Q$ is defined as Gretton *et al.* (2012): $\mathrm{MMD}^2(P,Q) = \mathbb{E}_{x,x'\sim P} k(x,x') + \mathbb{E}_{y,y'\sim Q} k(y,y') - 2\mathbb{E}_{x\sim P}\mathbb{E}_{y\sim Q} k(x,y)$. According to the Moore–Aronszajn theorem Aronszajn (1950), there exists a unique reproducing kernel Hilbert space of functions on $\mathcal{X}$ for which k is a reproducing kernel, i.e., $k(x,\cdot) \in \mathcal{H}$ and $f(x) = \langle f, k(x,\cdot)\rangle_{\mathcal{H}}$ for all $x \in \mathcal{X}$ and $f \in \mathcal{H}$, where $\langle\cdot,\cdot\rangle_{\mathcal{H}} = \langle\cdot,\cdot\rangle$ denotes the inner product on $\mathcal{H}$. Hence, we can find a *feature map*,

[1] There are efforts on improving the efficiency of randomized Fourier feature maps, e.g., by using quasi-random points in Avron *et al.* (2016).

$\phi\colon \mathcal{X} \to \mathcal{H}$ such that $k(x,y) = \langle \phi(x), \phi(y) \rangle_{\mathcal{H}}$, which allows us to rewrite MMD as Gretton *et al.* (2012):

$$\mathrm{MMD}^2(P,Q) = \left\| \mathbb{E}_{x\sim P}[\phi(x)] - \mathbb{E}_{y\sim Q}[\phi(y)] \right\|_{\mathcal{H}}^2, \tag{2.1}$$

where $\mathbb{E}_{x\sim P}[\phi(x)] \in \mathcal{H}$ is known as the (kernel) mean embedding of $P$, and exists if $\mathbb{E}_{x\sim P}\sqrt{k(x,x)} < \infty$ Smola *et al.* (2007). If $k$ is *characteristic* Sriperumbudur *et al.* (2011), then $P \mapsto \mathbb{E}_{x\sim P}[\phi(x)]$ is injective, meaning $\mathrm{MMD}(P,Q) = 0$, if and only if $P = Q$. Hence, the MMD associated with a characteristic kernel (e.g., Gaussian kernel) can be interpreted as a distance between the mean embeddings of two distributions.

Given the samples drawn from two distributions: $X_m = \{x_i\}_{i=1}^m \sim P$ and $X'_n = \{x'_i\}_{i=1}^n \sim Q$, we can estimate[2] the MMD by sample averages Gretton *et al.* (2012):

$$\begin{aligned} \widehat{\mathrm{MMD}}^2(X_m, X'_n) = \tfrac{1}{m^2} \sum_{i,j=1}^{m} k(x_i, x_j) \\ + \tfrac{1}{n^2} \sum_{i,j=1}^{n} k(x'_i, x'_j) - \tfrac{2}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} k(x_i, x'_j). \end{aligned} \tag{2.2}$$

However, at $O(mn)$ the computational cost of $\widehat{\mathrm{MMD}}(X_m, X'_n)$ is prohibitive for large-scale datasets.

### 2.2.2 Kernel approximation

By approximating the kernel function $k(x,x')$ with an inner product of finite dimensional feature vectors, i.e., $k(x,x') \approx \hat{\boldsymbol{\phi}}(x)^\top \hat{\boldsymbol{\phi}}(x')$ where $\hat{\boldsymbol{\phi}}(x) \in \mathbb{R}^A$ and $A$ is the number of features, the MMD estimator given in eq. (2.2) can be computed in $O(m+n)$, i.e., linear in the sample size:

$$\widehat{\mathrm{MMD}}^2(P,Q) = \left\| \tfrac{1}{m} \sum_{i=1}^{m} \hat{\boldsymbol{\phi}}(x_i) - \tfrac{1}{n} \sum_{i=1}^{n} \hat{\boldsymbol{\phi}}(x'_i) \right\|_2^2. \tag{2.3}$$

This approximation is also beneficial for private data generation: assuming $P$ is a data distribution and $Q$ is a synthetic data distribution, we can summarize the data distribution in terms of its kernel mean embedding (i.e., the first term on the right-hand side of eq. (2.3)), which can be privatized only once and used repeatedly during training of the generator which produces samples from $Q$.

[2] This particular MMD estimator is biased.

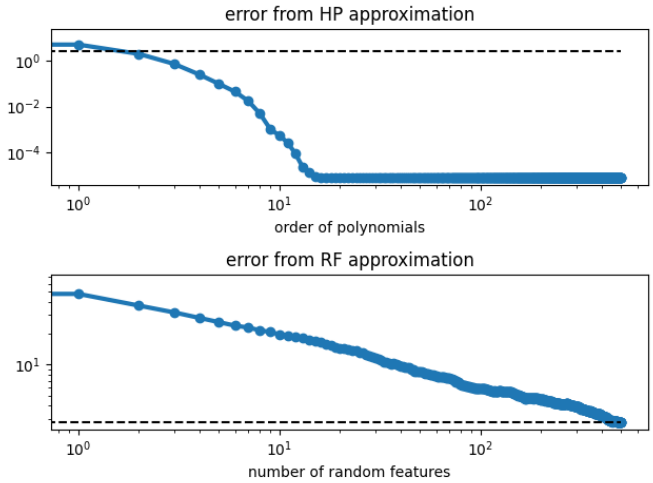



Figure 2.1: **HP VS. RF features.** Dataset $X$ contains $N = 100$ samples drawn from $\mathcal{N}(0,1)$ and $X'$ contains $N = 100$ samples drawn from $\mathcal{N}(1,1)$. The error is defined by: $\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}|k(x_i,x'_j) - \hat{\boldsymbol{\phi}}(x_i)^\top \hat{\boldsymbol{\phi}}(x'_j)|$ where $\hat{\boldsymbol{\phi}}$ is either RF or HP features. **Top**: The error decays fast when using HP features (eq. (2.6)). **Bottom**: The plot shows the average error over 100 independent draws of RF features (eq. (2.4)). The error decays slowly when using RF features. The best error (black dotted line) using 500 RF features coincides with the error using HP features with order 2 only.

### 2.2.3 Random Fourier features.

As an example of $\hat{\boldsymbol{\phi}}(\cdot)$, the random Fourier features Rahimi and Recht (2008) are derived from the following. Bochner's theorem Rudin (2013) states that for any translation invariant kernel, the kernel can be written as $k(x,x') = \widetilde{k}(x-x') = \mathbb{E}_{\omega\sim\Lambda}\cos(\omega^\top(x-x'))$. By drawing random frequencies $\{\omega_i\}_{i=1}^{A} \sim \Lambda$, where $\Lambda$ depends on the kernel, (e.g., a Gaussian kernel $k$ corresponds to normal distribution $\Lambda$), $\widetilde{k}(x-x')$ can be approximated with a Monte Carlo average. The resulting vector of random Fourier features (of length $A$) is given by

$$\hat{\boldsymbol{\phi}}_{RF,\boldsymbol{\omega}}(x) = (\hat{\phi}_{1,\boldsymbol{\omega}}(x),\ldots,\hat{\phi}_{A,\boldsymbol{\omega}}(x))^\top \tag{2.4}$$

where $\hat{\phi}_{j,\boldsymbol{\omega}}(x) = \sqrt{2/A}\,\cos({\omega_j}^\top x)$, $\hat{\phi}_{j+A/2,\boldsymbol{\omega}}(x) = \sqrt{2/A}\sin(\omega_j^\top x)$, for $j = 1,\cdots,A/2$.

DP-MERF Harder *et al.* (2021b) uses this very representation of the feature map given in eq. (2.4), and minimizes eq. (2.3) with a privatized data mean embedding to train a generator.

### 2.2.4 Hermite polynomial features.

For another example of $\hat{\boldsymbol{\phi}}(\cdot)$, one could also start with the *Mercer's theorem* (See Appendix Sec. A.3), which allows us to express a positive definite kernel $k$ in terms of the eigen-values $\lambda_i$ and eigen-functions $f_i$: $k(x,x') = \sum_{i=1}^{\infty} \lambda_i f_i(x) f_i^*(x')$, where $\lambda_i > 0$ and complex conjugate is denoted by $*$. The resulting *finite-dimensional* feature vector is simply $\hat{\boldsymbol{\phi}}(x) = \hat{\boldsymbol{\phi}}_{HP}(x) = [\sqrt{\lambda_0} f_0(x), \sqrt{\lambda_1} f_1(x), \cdots, \sqrt{\lambda_C} f_C(x)]$, where the cut-off is made at the $C$-th eigen-value and eigen-function. For the commonly-used Gaussian kernel, $k(x,x') = \exp(-\frac{1}{2l^2}(x-x')^2)$, where $l$ is the length scale parameter, an analytic form of eigen-values and eigen-functions are available, where the eigen-functions are represented with Hermite polynomials (See Sec. 2.3 for definition). This is the approximation we will use in our method.

### 2.2.5 Differential privacy

Given privacy parameters $\varepsilon \geq 0$ and $\delta \geq 0$, a mechanism $\mathcal{M}$ is ($\varepsilon$, $\delta$)-DP if the following equation holds: $\Pr[\mathcal{M}(\mathcal{D}) \in S] \leq e^{\varepsilon} \cdot \Pr[\mathcal{M}(\mathcal{D}') \in S] + \delta$, for all possible sets of the mechanism's outputs $S$ and all neighbouring datasets $\mathcal{D}$, $\mathcal{D}'$ differing by a single entry. In this paper, we use the *Gaussian mechanism* to ensure the output of our algorithm is DP. Consider a function $h : \mathcal{D} \mapsto \mathbb{R}^p$, where we add noise for privacy and the level of noise is calibrated to the *global sensitivity* Dwork *et al.* (2006), $\Delta_h$, defined by the maximum difference in terms of $L_2$-norm $||h(\mathcal{D}) - h(\mathcal{D}')||_2$, for neighbouring $\mathcal{D}$ and $\mathcal{D}'$ (i.e. $\mathcal{D}$ and $\mathcal{D}'$ have one sample difference by replacement). where the output is denoted by $\widetilde{h}(\mathcal{D}) = h(\mathcal{D}) + n$, where $n \sim \mathcal{N}(0, \sigma^2 \Delta_h^2 \mathbf{I}_p)$. The perturbed function $\widetilde{h}(\mathcal{D})$ is $(\varepsilon, \delta)$-DP, where $\sigma$ is a function of $\varepsilon$ and $\delta$ and can be numerically computed using, e.g., the auto-dp package by Wang *et al.* (2019).

## 2.3 Our method: DP-HP

### 2.3.1 Approximating the Gaussian kernel using Hermite polynomials (HP)

Using the *Mehler formula*[3] Mehler (1866), for $|\rho| < 1$, we can write down the Gaussian kernel[4] as a weighted sum of Hermite polynomials

$$\exp\left(-\frac{\rho}{1-\rho^2}(x-y)^2\right) = \sum_{c=0}^{\infty} \lambda_c f_c(x) f_c(y) \tag{2.5}$$

[3] This formula can be also derived from the Mercer's theorem as shown in Zhu *et al.* (1997); Rasmussen and Williams (2005).

[4] The length scale $l$ in terms of $\rho$ is $\frac{1}{2l^2} = \frac{\rho}{1-\rho^2}$.

where the $c$-th eigen-value is $\lambda_c = (1-\rho)\rho^c$ and the $c$-th eigen-function is defined by $f_c$, where $f_c(x) = \frac{1}{\sqrt{N_c}} H_c(x) \exp\left(-\frac{\rho}{1+\rho}x^2\right)$, and $N_c = 2^c c! \sqrt{\frac{1-\rho}{1+\rho}}$. Here, $H_c(x) = (-1)^c \exp(x^2) \frac{d^c}{dx^c} \exp(-x^2)$ is the $c$-th order physicist's Hermite polynomial.

As a result of the Mehler formula, we can define a $C$-th order Hermite polynomial features as a feature map (a vector of length $C+1$):

$$\hat{\boldsymbol{\phi}}^{(C)}_{HP}(x) = \left[\sqrt{\lambda_0} f_0(x), \cdots, \sqrt{\lambda_C} f_C(x)\right], \tag{2.6}$$

and approximate the Gaussian kernel via $\exp\left(-\frac{\rho}{1-\rho^2}(x-y)^2\right) \approx \hat{\boldsymbol{\phi}}^{(C)}_{HP}(x)^\top \hat{\boldsymbol{\phi}}^{(C)}_{HP}(y)$.

This feature map provides us with a uniform approximation to the MMD in eq. (2.1), for every pair of distributions $P$ and $Q$ (see Theorem 14 and Lemma 1 in Appendix Sec. A.3).

We compare the accuracy of this approximation with random features in Fig. 2.1, where we fix the length scale to the median heuristic value[5] in both cases. Note that the bottom plot shows the average error across 100 independent draws of random Fourier features. We observe that the error decay is significantly faster when using HPs than using RFs. For completeness, we derive the kernel approximation error under HP features and random features for 1-dimensional data in Appendix Sec. A.2. Additionally, we visualize the effect of length scale on the error further in Appendix Sec. A.1.

**Computing the Hermite polynomial features.** Hermite polynomials follow the recursive definition: $H_{c+1}(x) = 2xH_c(x) - 2cH_{c-1}(x)$. At high orders, the polynomials take on large values, leading to numerical instability. So we compute the re-scaled term $\phi_c = \sqrt{\lambda_c} f_c$ iteratively using a similar recursive expression given in Appendix Sec. A.5.

### 2.3.2 Handling multi-dimensional inputs

#### Tensor (or outer) product kernel

The Mehler formula holds for 1-dimensional input space. For $D$-dimensional inputs $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^D$, where $\mathbf{x} = [x_1, \cdots, x_D]$ and $\mathbf{x}' = [x'_1, \cdots, x'_D]$, the *generalized Hermite Polynomials* (Proposition A.3.1 and Remark 1 in Appendix Sec. A.3) allows us to represent the multivariate Gaussian kernel $k(\mathbf{x}, \mathbf{x}')$ by a tensor (or outer) products of the Gaussian kernel defined on each input dimension, where the coordinate-wise Gaussian kernel is

[5] Median heuristic is a commonly-used heuristic to choose a length scale, which picks a value in the middle range (i.e., median) of $\|x_i - x_j\|$ for $1 \le i, j \le n$ for the dataset of $n$ samples.

approximated with Hermite polynomials:

$$k(\mathbf{x},\mathbf{x}') = k_{X_1} \otimes k_{X_2} \cdots \otimes k_{X_D} = \prod_{d=1}^{D} k_{X_d}(x_d, x'_d),$$
$$\approx \prod_{d=1}^{D} \hat{\boldsymbol{\phi}}_{HP}^{(C)}(x_d)^\top \hat{\boldsymbol{\phi}}_{HP}^{(C)}(x_d), \tag{2.7}$$

where $\hat{\boldsymbol{\phi}}_{HP}^{(C)}(.)$[6] is defined in eq. (2.6). The corresponding feature map, from $k(\mathbf{x},\mathbf{x}') \approx \widehat{v}_p(\mathbf{x})^\top \widehat{v}_p(\mathbf{x}')$, is written as

$$\widehat{v}_p(\mathbf{x})$$
$$= \text{vec}\left[\hat{\boldsymbol{\phi}}_{HP}^{(C)}(x_1) \otimes \hat{\boldsymbol{\phi}}_{HP}^{(C)}(x_2) \otimes \cdots \hat{\boldsymbol{\phi}}_{HP}^{(C)}(x_D)\right] \tag{2.8}$$

where $\otimes$ denotes the tensor (outer) product and vec is an operation that vectorizes a tensor. The size of the feature map is $(C+1)^D$, where $D$ is the input dimension of the data and $C$ is the chosen order of the Hermite polynomials. This is prohibitive for the datasets we often deal with, e.g., for MNIST ($D = 784$) with a relatively small order (say $C = 10$), the size of feature map is $11^{784}$, impossible to fit in a typical size of memory.

In order to handle high-dimensional data in a computationally feasible manner, we propose the following approximation. First we subsample input dimensions where the size of the selected input dimensions is denoted by $D_{prod}$. We then compute the feature map only on those selected input dimensions denoted by $\mathbf{x}^{D_{prod}}$. We repeat these two steps during training. The size of the feature map becomes $(C+1)^{D_{prod}}$, significantly lower than $(C+1)^D$ if $D_{prod} \ll D$. What we lose in return is the injectivity of the Gaussian kernel on the full input distribution, as we compare two distributions in terms of selected input dimensions. We need a quantity that is more computationally tractable and also helps distinguishing two distributions, which we describe next.

### Sum kernel

Here, we define another kernel on the joint distribution over $(x_1, \cdots, x_D)$. The following kernel is formed by defining a 1-dimensional Gaussian kernel on each of the input

[6] One can let each coordinate's Hermite Polynomials $\phi_{HP,d}^{(C)}(x_d)$ take different values of $\rho$, which determine a different level of fall-offs of the eigen-values and a different range of values of the eigen-functions. Imposing a different cut-off $C$ for each coordinate is also possible.

dimensions:

$$\begin{aligned}\widetilde{k}(\mathbf{x},\mathbf{x}') &= \tfrac{1}{D}\left[k_{X_1}(x_1,x_1') + \cdots + k_{X_D}(x_D,x_D')\right], \\ &= \tfrac{1}{D}\sum_{d=1}^{D} k_{X_d}(x_d,x_d'), \\ &\approx \tfrac{1}{D}\sum_{d=1}^{D} \hat{\boldsymbol{\phi}}_{HP}^{(C)}(x_d)^\top \hat{\boldsymbol{\phi}}_{HP}^{(C)}(x_d), \end{aligned} \tag{2.9}$$

where $\hat{\boldsymbol{\phi}}_{HP,d}^{(C)}(.)$ is given in eq. (2.6). The corresponding feature map, from $\widetilde{k}(\mathbf{x},\mathbf{x}') \approx \widehat{v}_s(\mathbf{x})^\top \widehat{v}_s(\mathbf{x}')$, is represented by

$$\widehat{v}_s(\mathbf{x}) = \begin{bmatrix} \hat{\boldsymbol{\phi}}_{HP,1}^{(C)}(x_1)/\sqrt{D} \\ \hat{\boldsymbol{\phi}}_{HP,2}^{(C)}(x_2)/\sqrt{D} \\ \vdots \\ \hat{\boldsymbol{\phi}}_{HP,D}^{(C)}(x_D)/\sqrt{D} \end{bmatrix} \in \mathbb{R}^{((C+1)\cdot D)\times 1}, \tag{2.10}$$

where the features map is the size of $(C+1)D$. For the MNIST digit data ($D = 784$), with a relatively small order, say $C = 10$, the size of the feature map is $11 \times 784 = 8624$ dimensional, which is manageable compared to the size ($11^{784}$) of the feature map under the generalized Hermite polynomials.

Note that the sum kernel does not approximate the Gaussian kernel defined on the joint distribution over all the input dimensions. Rather, the assigned Gaussian kernel *on each dimension is characteristic*. The Lemma 2 in Appendix Sec. A.4 shows that by minimizing the approximate MMD between the real and synthetic data distributions based on feature maps given in eq. (2.10), we assure that the marginal probability distributions of the synthetic data converges to those of the real data.

### Combined Kernel

Finally we arrive at a new kernel, which comes from a sum of the two fore-mentioned kernels:

$$k_c(\mathbf{x},\mathbf{x}') = k(\mathbf{x},\mathbf{x}') + \widetilde{k}(\mathbf{x},\mathbf{x}'), \tag{2.11}$$

where $k(\mathbf{x},\mathbf{x}') \approx \widehat{v}_p(\mathbf{x}^{D_{prod}})^\top \widehat{v}_p(\mathbf{x}'^{D_{prod}})$ and $\widetilde{k}(\mathbf{x},\mathbf{x}') \approx \widehat{v}_s(\mathbf{x})^\top \widehat{v}_s(\mathbf{x}')$, and consequently the corresponding feature map is given by

$$\widehat{v}_c(\mathbf{x}) = \begin{bmatrix} \widehat{v}_p(\mathbf{x}^{D_{prod}}) \\ \widehat{v}_s(\mathbf{x}) \end{bmatrix} \tag{2.12}$$

where the size of the feature map is $\mathbb{R}^{\left((C+1)^{D_{prod}}+(C+1)\cdot D)\right)\times 1}$.

**Why this kernel?** When $D_{prod}$ goes to $D$, the product kernel itself in eq. (2.11) becomes characteristic, which allows us to reliably compare two distributions. However, for computational tractability, we are restricted to choose a relatively small $D_{prod}$ to subsample the input dimensions, which forces us to lose information on the distribution over the un-selected input dimensions. The use of sum kernel is to provide extra information on the un-selected input dimensions at a particular training step. Under our kernel in eq. (2.11), every input dimension's marginal distributions are compared between two distributions in all the training steps due to the sum kernel, while some of the input dimensions are chosen to be considered for more detailed comparison (e.g., high-order correlations between selected input dimensions) due to the outer product kernel.

### 2.3.3 Approximate MMD for classification

For classification tasks, we define a mean embedding for the joint distribution over the input and output pairs $(\mathbf{x},\mathbf{y})$, with the particular feature map given by $\mathbf{g}$

$$\widehat{\boldsymbol{\mu}}_{P_{\mathbf{x},\mathbf{y}}}(\mathcal{D}) = \frac{1}{m}\sum_{i=1}^{m}\mathbf{g}(\mathbf{x}_i,\mathbf{y}_i). \tag{2.13}$$

Here, we define the feature map as an outer product between the input features represented by eq. (2.12) and the output labels represented by one-hot-encoding $\mathbf{f}(\mathbf{y}_i)$:

$$\mathbf{g}(\mathbf{x}_i,\mathbf{y}_i) = \widehat{v}_c(\mathbf{x}_i)\mathbf{f}(\mathbf{y}_i)^T. \tag{2.14}$$

Given eq. (2.14), we further decompose eq. (2.13) into two, where the first term corresponds to the outer product kernel denoted by $\widehat{\boldsymbol{\mu}}_P^p$ and the second term corresponds to the sum kernel denoted by $\widehat{\boldsymbol{\mu}}_P^s$:

$$\widehat{\boldsymbol{\mu}}_{P_{\mathbf{x},\mathbf{y}}} = \begin{bmatrix}\widehat{\boldsymbol{\mu}}_P^p \\ \widehat{\boldsymbol{\mu}}_P^s\end{bmatrix} = \begin{bmatrix}\frac{1}{m}\sum_{i=1}^{m}\widehat{v}_p(\mathbf{x}_i^{D_{prod}})\mathbf{f}(\mathbf{y}_i)^T \\ \frac{1}{m}\sum_{i=1}^{m}\widehat{v}_s(\mathbf{x}_i)\mathbf{f}(\mathbf{y}_i)^T\end{bmatrix}. \tag{2.15}$$

Similarly, we define an approximate mean embedding of the synthetic data distribution by $\widehat{\boldsymbol{\mu}}_{Q_{\mathbf{x}',\mathbf{y}'}}(\mathcal{D}'_{\boldsymbol{\theta}}) = \frac{1}{n}\sum_{i=1}^{n}\mathbf{g}(\mathbf{x}'_i(\boldsymbol{\theta}),\mathbf{y}'_i(\boldsymbol{\theta}))$, where $\boldsymbol{\theta}$ denotes the parameters of a synthetic data generator. Then, the approximate MMD is given by: $\widehat{\mathrm{MMD}}^2_{HP}(P,Q) = ||\widehat{\boldsymbol{\mu}}_{P_{\mathbf{x},\mathbf{y}}}(\mathcal{D}) - \widehat{\boldsymbol{\mu}}_{Q_{\mathbf{x}',\mathbf{y}'}}(\mathcal{D}'_{\boldsymbol{\theta}})||_2^2 = ||\widehat{\boldsymbol{\mu}}_P^p - \widehat{\boldsymbol{\mu}}_{Q_{\boldsymbol{\theta}}}^p||_2^2 + ||\widehat{\boldsymbol{\mu}}_P^s - \widehat{\boldsymbol{\mu}}_{Q_{\boldsymbol{\theta}}}^s||_2^2$. In practice, we minimize the augmented approximate MMD:

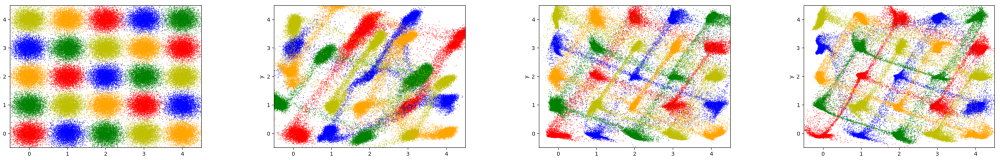
$$\min_{\boldsymbol{\theta}}\ \gamma||\widehat{\boldsymbol{\mu}}_P^p - \widehat{\boldsymbol{\mu}}_{Q_{\boldsymbol{\theta}}}^p||_2^2 + ||\widehat{\boldsymbol{\mu}}_P^s - \widehat{\boldsymbol{\mu}}_{Q_{\boldsymbol{\theta}}}^s||_2^2. \tag{2.16}$$

where $\gamma$ is a positive constant (a hyperparameter) that helps us to deal with the scale

difference in the two terms (depending on the selected HP orders and subsampled input dimensions) and also allows us to give a different importance on one of the two terms. We provide the details on how $\gamma$ plays a role and whether the algorithm is sensitive to $\gamma$ in Sec. 2.5. Minimizing eq. (2.16) yields a synthetic data distribution over the input and output, which minimizes the discrepancy in terms of the particular feature map eq. (2.15) between synthetic and real data distributions.

### 2.3.4 Differentially private data samples

For obtaining privacy-preserving synthetic data, all we need to do is privatizing $\widehat{\boldsymbol{\mu}}_P^p$ and $\widehat{\boldsymbol{\mu}}_P^s$ given in eq. (2.15), then training a generator. We use the Gaussian mechanism to privatize both terms. See Appendix Sec. A.6 for sensitivity analysis. Unlike $\widehat{\boldsymbol{\mu}}_P^s$ that can be privatized only and for all, we need to privatize $\widehat{\boldsymbol{\mu}}_P^p$ every time we redraw the subsampled input dimensions. We split a target $\varepsilon$ into two such that $\varepsilon = \varepsilon_1 + \varepsilon_2$ (also the same for $\delta$), where $\varepsilon_1$ is used for privatizing $\widehat{\boldsymbol{\mu}}_P^s$ and $\varepsilon_2$ is used for privatizing $\widehat{\boldsymbol{\mu}}_P^p$. We further compose the privacy loss incurred in privatizing $\widehat{\boldsymbol{\mu}}_P^p$ during training by the analytic moments accountant Wang *et al.* (2019), which returns the privacy parameter $\sigma$ as a function of $(\varepsilon_2, \delta_2)$. In the experiments, we subsample the input dimensions for the outer product kernel in every epoch as opposed to in every training step for an economical use of $\varepsilon_2$.



Data Samples
NLL $\approx 3.1 \times 10^5$

DP-CGAN ($\epsilon = 1$)
NLL $\approx 4.7 \times 10^5$

DP-MERF ($\epsilon = 1$)
NLL $\approx 4.1 \times 10^5$

**DP-HP** ($\epsilon = 1$)
NLL $\approx 3.7 \times 10^5$

Figure 2.2: Simulated example from a Gaussian mixture. **Left**: Data samples drawn from a Gaussian Mixture distribution with 5 classes (each color represents a class). NLL denotes the negative log likelihood of the samples given the true data distribution. **Middle-Left**: Synthetic data generated by DP-CGANs at $\varepsilon = 1$, where some modes are dropped, which is reflected in poor NLL. **Middle-Right**: Synthetic data samples generated by DP-MERF at $\varepsilon = 1$. **Right**: Synthetic data samples generated by DP-HP at $\varepsilon = 1$. Our method captures all modes accurately at $\varepsilon = 1$, and achieves better NLL thanks to a smaller size of feature map than that of DP-MERF (see text).

## 2.4 Related Work

Approaches to differentially private data release can be broadly sorted into three categories. One line of prior work with background in learning theory aims to provide theoretical guarantees on the utility of released data Snoke and Slavković (2018); Mohammed *et al.* (2011); Xiao *et al.* (2010); Hardt *et al.* (2012); Zhu *et al.* (2017). This usually requires strong constraints on the type of data and the intended use of the released data.

A second line of work focuses on the sub-problem of discrete data with limited domain size, which is relevant to tabular datasets Zhang *et al.* (2017); Qardaji *et al.* (2014); Chen *et al.* (2015); Zhang *et al.* (2021). Such approaches typically approximate the structure in the data by identifying small sub-sets of features with high correlation and releasing these lower order marginals in a private way. Some of these methods have also been successful in the recent NIST 2018 Differential Privacy Synthetic Data Challenge National Institute for Standards and Techonologies (2018), while these methods often require discretization of the data and do not scale to higher dimensionality in arbitrary domains.

The third line of work aims for broad applicability without constraints on the type of data or the kind of downstream tasks to be used. Recent approaches attempt to leverage the modeling power of deep generative models in the private setting. While work on VAEs exists Acs *et al.* (2018), GANs are the most popular model Xie *et al.* (2018b); Torkzadehmahani *et al.* (2019); Frigerio *et al.* (2019); Yoon *et al.* (2019); Chen *et al.* (2020), where most of these utilize a version of DP-SGD Abadi *et al.* (2016b) to accomplish this training, while PATE-GAN is based on the private aggregation of teacher ensembles (PATE) Papernot *et al.* (2017).

The closest prior work to the proposed method is DP-MERF Harder *et al.* (2021b), where kernel mean embeddings are approximated with random Fourier features Rahimi and Recht (2008) instead of Hermite polynomials. Random feature approximations of MMD have also been used with DP Balog *et al.* (2018); Sarpatwar *et al.* (2019). A recent work utilizes the Sinkhorn divergence for private data generation Cao *et al.* (2021), which more or less matches the results of DP-MERF when the regularizer is large and the cost function is the L2 distance. To our knowledge, ours is the first work using Hermite polynomials to approximate MMD in the context of differentially private data generation.

## 2.5 Experiments

Here, we show the performance of our method tested on several real world datasets. Evaluating the quality of generated data itself is challenging. Popular metrics such as inception score and Fréchet inception distance are appropriate to use for evaluating color images. For the generated samples for tabular data and black and white images, we use the following three metrics: (a) Negative log-likelihood of generated samples given a

ground truth model in Sec. 2.5.1; (b) $\alpha$-way marginals of generated samples in Sec. 2.5.2 to judge whether the generated samples contain a similar correlation structure to the real data; (c) Test accuracy on the real data given classifiers trained with generated samples in Sec. 2.5.3 to judge the generalization performance from synthetic to real data.

As comparison methods, we tested PrivBayes Zhang *et al.* (2017), DP-CGAN Torkzadehmahani *et al.* (2019), DP-GAN Xie *et al.* (2018b) and DP-MERF Harder *et al.* (2021b). For image datasets we also trained GS-WGAN Chen *et al.* (2020). Our experiments were implemented in PyTorch Paszke *et al.* (2019) and run using Nvidia Kepler20 and Kepler80 GPUs. Our code is available at `https://github.com/ParkLabML/DP-HP`.

Table 2.1: $\alpha$-way marginals evaluated on generated samples with discretized Adult and Census datasets.

| *Adult* | PrivBayes | | DP-MERF | | **DP-HP** | |
|---|---|---|---|---|---|---|
| | $\varepsilon$=0.3 | $\varepsilon$=0.1 | $\varepsilon$=0.3 | $\varepsilon$=0.1 | $\varepsilon$=0.3 | $\varepsilon$=0.1 |
| $\alpha$=3 | 0.446 | 0.577 | 0.405 | 0.480 | **0.332** | **0.377** |
| $\alpha$=4 | 0.547 | 0.673 | 0.508 | 0.590 | **0.418** | **0.467** |
| *Census* | PrivBayes | | DP-MERF | | DP-HP | |
| | $\varepsilon$=0.3 | $\varepsilon$=0.1 | $\varepsilon$=0.3 | $\varepsilon$=0.1 | $\varepsilon$=0.3 | $\varepsilon$=0.1 |
| $\alpha$=2 | 0.180 | 0.291 | 0.190 | 0.222 | **0.141** | **0.155** |
| $\alpha$=3 | 0.323 | 0.429 | 0.302 | 0.337 | **0.211** | **0.232** |

### 2.5.1 2D Gaussian mixtures

We begin our experiments on Gaussian mixtures, as shown in Fig. 2.2 (left). We generate 4000 samples from each Gaussian, reserving 10% for the test set, which yields 90000 training samples from the following distribution: $p(\mathbf{x},\mathbf{y}) = \prod_i^N \sum_{j\in C_{\mathbf{y}_i}} \frac{1}{C}\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_j, \sigma\mathbf{I}_2)$ where $N = 90000$, and $\sigma = 0.2$. $C = 25$ is the number of clusters and $C_y$ denotes the set of indices for means $\boldsymbol{\mu}$ assigned to class $y$. Five Gaussians are assigned to each class, which leads to a uniform distribution over $\mathbf{y}$ and 18000 samples per class. We use the negative log likelihood (NLL) of the samples under the true distribution as a score[7] to measure the quality of the generated samples: $\text{NLL}(\mathbf{x},\mathbf{y}) = -\log p(\mathbf{x},\mathbf{y})$. The lower NLL the better.

We compare our method to DP-CGAN and DP-MERF at $(\varepsilon,\delta) = (1, 10^{-5})$ in Fig. 2.2. Many of the generated samples by DP-CGAN fall out of the distribution and some modes are dropped (like the green one in the top right corner). DP-MERF preserves all modes. DP-HP performs better than DP-MERF by placing fewer samples in low density regions as indicated by the low NLL. This is due to the drastic difference in the size of the feature map. DP-MERF used $30,000$ random features (i.e., $30,000$-dimensional feature map).

[7]Note that this is different from the other common measure of computing the negative log-likelihood of the true data given the learned model parameters.

DP-HP used the 25-th order Hermite polynomials on both sum and product kernel approximation (i.e., $25^2+25=650$-dimensional feature map). in this example, as the input is 2-dimensional, it was not necessary to subsample the input dimensions to approximate the outer product kernel.

### 2.5.2 $\alpha$-way marginals with discretized tabular data

We compare our method to PrivBayes Zhang *et al.* (2017) and DP-MERF. For PrivBayes, we used the published code from McKenna *et al.* (2019), which builds on the original code with Zhang *et al.* (2018) as a wrapper. We test the model on the discretized Adult and Census datasets. Although these datasets are typically used for classification, we use their inputs only for the task of learning the input distribution. Following Zhang *et al.* (2017), we measure $\alpha$-way marginals of generated samples at varying levels of $\varepsilon$-DP with $\delta = 10^{-5}$. We measure the accuracy of each marginal of the generated dataset by the total variation distance between itself and the real data marginal (i.e., half of the L1 distance between the two marginals, when both of them are treated as probability distributions). We use the average accuracy over all marginals as the final error metric for $\alpha$-way marginals. In Table 2.1, our method outperforms other two at the stringent privacy regime. See Appendix Sec. A.7.1 for hyperparameter values we used, and Appendix Sec. A.7.2 for the impact of $\gamma$ on the quality of the generated samples. We also show how the selection of $D_{prod}$ affects the accuracy in Appendix Sec. A.7.5.

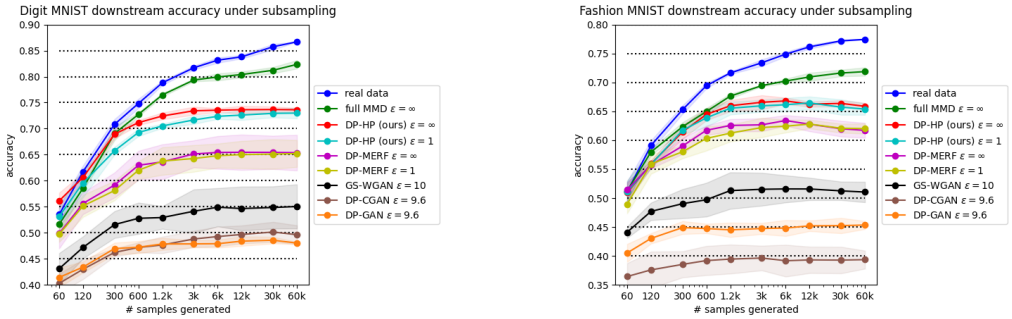
### 2.5.3 Generalization from synthetic to real data

Following Chen *et al.* (2020); Torkzadehmahani *et al.* (2019); Yoon *et al.* (2019); Chen *et al.* (2020); Harder *et al.* (2021b); Cao *et al.* (2021), we evaluate the quality of the (private and non-private) generated samples from these models using the common approach of measuring performance on downstream tasks. We train 12 different commonly used classifier models using generated samples and then evaluate the classifiers on a test set containing *real* data samples. Each setup is averaged over 5 random seeds. The test accuracy indicates how well the models generalize from the synthetic to the real data distribution and thus, the utility of using private data samples instead of the real ones. Details on the 12 models can be found in Table A.8.

**Tabular data.** First, we explore the performance of DP-HP algorithm on eight different imbalanced tabular datasets with both numerical and categorical input features. The numerical features on those tabular datasets can be either discrete (e.g. age in years) or continuous (e.g. height) and the categorical ones may be binary (e.g. drug vs placebo group) or multi-class (e.g. nationality). The datasets are described in detail in Appendix Sec. A.7. As an evaluation metric, we use ROC (area under the receiver characteristics curve) and PRC (area under the precision recall curve) for datasets with binary labels,

Table 2.2: Performance comparison on Tabular datasets. The average over five independent runs.

| | Real | | DP-CGAN $(1, 10^{-5})$-DP | | DP-GAN $(1, 10^{-5})$-DP | | DP-MERF $(1, 10^{-5})$-DP | | **DP-HP** $(1, 10^{-5})$-DP | |
|---|---|---|---|---|---|---|---|---|---|---|
| **adult** | 0.786 | 0.683 | 0.509 | 0.444 | 0.511 | 0.445 | 0.642 | 0.524 | **0,688** | **0,632** |
| **census** | 0.776 | 0.433 | 0.655 | 0.216 | 0.529 | 0.166 | 0.685 | 0.236 | **0,699** | **0,328** |
| **cervical** | 0.959 | 0.858 | 0.519 | 0.200 | 0.485 | 0.183 | 0.531 | 0.176 | **0,616** | **0,312** |
| **credit** | 0.924 | 0.864 | 0.664 | 0.356 | 0.435 | 0.150 | 0.751 | 0.622 | **0,786** | **0,744** |
| **epileptic** | 0.808 | 0.636 | 0.578 | 0.241 | 0.505 | 0.196 | 0.605 | 0.316 | **0,609** | **0,554** |
| **isolet** | 0.895 | 0.741 | 0.511 | 0.198 | 0.540 | 0.205 | 0.557 | 0.228 | **0,572** | **0,498** |
| | F1 | | F1 | | F1 | | F1 | | F1 | |
| **covtype** | 0.820 | | 0.285 | | 0.492 | | 0.467 | | **0.537** | |
| **intrusion** | 0.971 | | 0.302 | | 0.251 | | **0.892** | | 0.890 | |




(a) MNIST (b) FashionMNIST

Figure 2.3: We compare the real data test accuracy as a function of training set size for models trained on synthetic data from DP-HP and comparison models. Confidence intervals show one standard deviation.

and F1 score for dataset with multi-class labels. Table 2.2 shows the average over the 12 classifiers trained on the generated samples (also averaged over 5 independent seeds), where overall DP-HP outperforms the other methods in both the private and non-private settings, followed by DP-MERF.[8] See Appendix Sec. A.7.3 for hyperparameter values we used. We also show the non-private MERF and HP results in Table A.5 in Appendix.

**Image data.** We follow previous work in testing our method on image datasets MNIST LeCun *et al.* (2010) (license: CC BY-SA 3.0) and FashionMNIST Xiao *et al.* (2017) (license: MIT). Both datasets contain 60000 images from 10 different balanced classes. We test both fully connected and convolutional generator networks and find that the former

[8]For the Cervical dataset, the non-privately generated samples by DP-MERF and DP-HP give better results than the baseline trained with real data. This may be due to the fact that the dataset is relatively small which can lead to overfitting. The generating samples by DP-MERF and DP-HP could bring a regularizing effect, which improves the performance as a result.

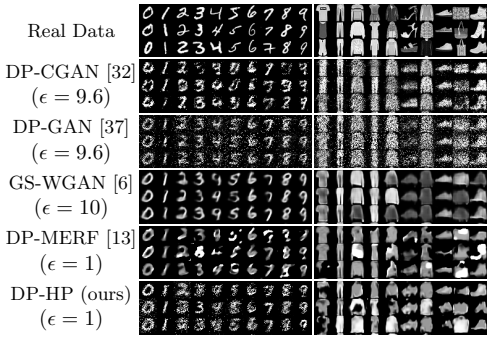



Figure 2.4: Generated MNIST and FashionMNIST samples from DP-HP and comparison models

works better for MNIST, while the latter model achieves better scores on FashionMNIST. For the experimental setup of DP-HP on the image datasets see Table A.7 in Appendix Sec. A.8.2. A qualitative sample of the generated images for DP-HP and comparison methods is shown in Fig. 2.4. While qualitatively GS-WGAN produces the cleanest samples, DP-HP outperforms GS-WGAN on downstream tasks. This can be explained by a lack of sample diversity in GS-WGAN shown in Fig. 2.3.

In Fig. 2.3, we compare the test accuracy on real image data based on private synthetic samples from DP-GAN, DP-CGAN, GS-WGAN, DP-MERF and DP-HP generators. As additional baselines we include performance of real data and of *full MMD*, a non-private generator, which is trained with the MMD estimator in eq. (2.2) in a mini-batch fashion. DP-HP gives the best accuracy over the other considered methods followed by DP-MERF but with a considerable difference especially on the MNIST dataset. For GAN-based methods, we use the same weak privacy constraints given in the original papers, because they do not produce meaningful samples at $\varepsilon = 1$. Nonetheless, the accuracy these models achieve remains relatively low. Results for individual models for both image datasets are given in Appendix Sec. A.8.

Finally, we show the downstream accuracy for smaller generated datasets down to 60 samples (or 0.1% of original dataset) in Fig. 2.3. The points, at which additional generated data does not lead to improved performance, gives us a sense of the redundancy present in the generated data. We observe that all generative models except *full MMD* see little increase in performance as we increase the number of synthetic data samples to train the classifiers. This indicates that the *effective dataset size* these methods produce lies only at about 5% (3k) to 10% (6k) of the original data. For DP-GAN and DP-CGAN this effect is even more pronounced, showing little to no gain in accuracy after the first 300 to 600 samples respectively on FashionMNIST.

## 2.6 Summary and Discussion

We propose a DP data generation framework that improves the privacy-accuracy trade-off using the Hermite polynomials features thanks to the orderedness of the polynomial features. We chose the combination of outer product and sum kernels computational tractability in handling high-dimensional data. The quality of generated data by our method is significantly higher than that by other state-of-the-art methods, in terms of three different evaluation metrics. In all experiments, we observed that assigning $\varepsilon$ more to $\varepsilon_1$ than $\varepsilon_2$ and using the sum kernel's mean embedding as a main objective together with the outer product kernel's mean embedding as a constraint (weighted by $\gamma$) help improving the performance of DP-HP.

As the size of mean embedding grows exponentially with the input dimension under the outer product kernel, we chose to subsample the input dimensions. However, even with the subsampling, we needed to be careful not to explode the size of the kernel's mean embedding, which limits the subsampling dimension to be less than 5, in practice. This gives us a question whether there are better ways to approximate the outer product kernel than random sampling across all input dimensions. We leave this for future work.

# Chapter 3

# Sample Efficient Learning of Predictors that Complement Humans

*This chapter is based on the joint work with Hussein Mozannar and David Sontag. I thank their contributions in shaping ideas and preparation of the paper.*

## 3.1 Introduction

How do we combine AI systems and human decision makers to both reduce error and alleviate the burden on the human? AI systems are starting to be frequently used in combination with human decision makers, including in high-stakes settings like healthcare (Beede *et al.*, 2020) and content moderation Gillespie (2020). A possible way to combine the human and the AI is to learn a 'rejector' that queries either the human or the AI to predict on each input. This allows us to route examples to the AI model, where it outperforms the human, so as to simultaneously reduce error and human effort. Moreover, this formulation allows us to jointly optimize the AI so as to complement the human's weaknesses, and to optimize the rejector to allow the AI to defer when it is unable to predict well. This type of interaction is typically referred to as *expert deferral* and the learning problem is that of jointly learning the AI classifier and the rejector. Empirically this approach has been shown to outperform either the human or the AI when predicting by their own Kamar *et al.* (2012); Tan *et al.* (2018). One hypothesis is that humans and machines make different kinds of errors. For example humans may have bias on certain features Kleinberg *et al.* (2018) while AI systems may have bounded expressive power or limited training data. On the other hand, humans may outperform AI systems as they may have side information that is not available to the AI, for example due to privacy constraints.

Existing deployments tend to ignore that the system has two components: the AI classifier (henceforth, the classifier) and the human. Typically the AI is trained without taking int account the human—and deferral is done using post-hoc approaches like model confidence Raghu *et al.* (2019). The main problem of this approach, that we refer to as *staged learning*, is that it ignores the possibility of learning a better combined system by accounting for the human (and its mistakes) during training. More recent work has de-

veloped joint training strategies for the AI and the rejector based on surrogate losses and alternating minimization Mozannar and Sontag (2020a); Okati *et al.* (2021a). However, we lack a theoretical understanding of the fundamental merits of joint learning compared to the staged approach. In this work, we study three main challenges in expert deferral from a theoretical viewpoint: 1) *model capacity* constraints, 2) lack of *data of human expert's prediction* and 3) optimization using *surrogate losses*.

When learning a predictor and rejector in a limited hypothesis class, it becomes more valuable to allocate model capacity to complement the human. We prove a bound on the gap between the approach that learns a predictor that complements human and the approach that learns the predictor ignoring the presence of the human in Section 3.3. To practically learn to complement the human, the literature has shown that surrogate loss functions are successful Madras *et al.* (2018); Mozannar and Sontag (2020a). We propose a family of surrogate loss functions that generalizes existing surrogates such as the surrogate in Mozannar and Sontag (2020a), and we further prove surrogate excess risk bounds and generalization properties of these surrogates in Section 3.4. Finally, a main limitation of being able to complement the human is the availability of samples of human predictions. For example, suppose we wish to deploy a system for diagnosing pneumonia from chest X-rays in a new hospital. To be able to know when to defer to the new radiologists, we need to understand their specific strengths and weaknesses. We design a provable active learning scheme that is able to first understand the human expert error boundary and learn a classifier-rejector pair that adapts to it in Section 3.5. To summarize, the contributions of this paper are the following:

- **Understanding the gap between joint and staged learning**: we prove bounds on the gap when learning in bounded capacity hypothesis classes and with missing human data.
- **Theoretical analysis of Surrogate losses:** we propose a novel family of consistent surrogates that generalizes prior work and analyze asymptotic and sample properties.
- **Actively learning to defer:** we provide an algorithm that is able to learn a classifier-rejector pair by minimally querying the human on selected points.

## 3.2 Problem Setting

We study classification problems where the goal is to predict a target $Y \in \{1, \cdots, K\}$ based on a set of features $X \in \mathcal{X}$, or via querying a human expert opinion $M \sim \mu_{M|XY}$ that has access to a domain $\mathcal{Z}$. Upon viewing the input $X$, we decide first via a rejector function $r : \mathcal{X} \to \{0,1\}$ whether to defer to the expert, where $r(\mathbf{x}) = 1$ means deferral and $r(\mathbf{x}) = 0$ means predicting using a classifier $h : \mathcal{X} \to [K]$. The expert domain may contain side information beyond $X$ to classify instances. For example, when diagnosing diseases

from chest X-rays the human may have access to the patient's medical records while the AI only has access to the X-ray. We assume that $X, Y, M$ have a joint probability measure $\mu_{XYM}$.

We let deferring the decision to the expert incur a cost equal to the expert's error and an additional penalty term: $\ell_{\text{exp}}(\mathbf{x}, y, m) = \mathbb{I}_{m \neq y} + c_{\text{exp}}(\mathbf{x}, y, m)$ that depends on the features $\mathbf{x}$, the value of target $Y = y$, and the expert's prediction $M = m$. Moreover, we assume that predicting without querying the expert incurs a different cost equal to the classifier error and an additional penalty: $\ell_{\text{AI}}(\mathbf{x}, y, m) = \mathbb{I}_{h(\mathbf{x}) \neq y} + c_{\text{AI}}(\mathbf{x}, y, m)$ where $h(\mathbf{x})$ is the prediction of the classifier. With the above in hand, we write the true risk as

$$L_{\text{def}}(h, r) = \mathbb{E}_{X,Y,M}\left[\, \ell_{\text{AI}}\big(X, Y, h(X)\big) \cdot \mathbb{I}_{r(X)=0} + \ell_{\text{exp}}(X, Y, M) \cdot \mathbb{I}_{r(X)=1} \,\right] \tag{3.1}$$

In the setting when we only care about misclassification costs with no additional penalties, the deferral loss becomes a $0-1$ loss as follows:

$$L^{0-1}_{\text{def}}(h, r) = \mathbb{E}\left[\, \mathbb{I}_{h(X) \neq Y} \mathbb{I}_{r(X)=0} + \mathbb{I}_{M \neq Y} \mathbb{I}_{r(X)=1} \,\right] \tag{3.2}$$

We focus primarily on the $0-1$ loss for our analysis; it is also possible to extend parts of the analysis to handle additional cost penalties. We restrict our search to classifiers within a hypothesis class $\mathcal{H}$ and a rejector function within a hypothesis class $\mathcal{R}$. The optimal joint classifier and rejector pair is the one that minimizes (3.2):

$$h^*, r^* = \operatorname*{argmin}_{h \in \mathcal{H}, r \in \mathcal{R}} L^{0-1}_{\text{def}}(h, r) \tag{3.3}$$

To approximate the optimal classifier-rejector pair, we have to handle two main obstacles: (i) *optimization* of the non-convex and discontinuous loss function and (ii) availability of the *data* on human's predictions and the true label.

In the following section 3.3 and in section 3.5, we restrict the analysis to binary labels $Y = \{0, 1\}$ for a clearer exposition. The theoretical results in the following section are shown to apply further for the multiclass setting in a set of experimental results. However, in section 3.4, where we discuss practical algorithms, we switch back to the mutliclass setting for full generality. In the following section, we compare two strategies for expert deferral across these two dimensions.

## 3.3 Staged Learning of Classifier and Rejector

### 3.3.1 Model Complexity Gap

**Staged learning.** The optimization problem framed in (3.3) requires joint learning of the classifier and rejector. Alternatively, a popular approach comprises of first learning a classifier that minimizes average misclassification error on the distribution, and then,

learning a rejector that defers each point to either classifier or the expert, depending on who has a lower estimated error Raghu *et al.* (2019); Wilder *et al.* (2020).

Formally, we first learn $h$ to minimize the average misclassification error:

$$\hat{h} = \operatorname*{argmin}_{h \in \mathcal{H}} \mathbb{E}_{X,Y}[\mathbb{I}_{h(X) \neq Y}] \tag{3.4}$$

and in the second step we learn the rejector $r$ to minimize the joint loss (3.2) with the now fixed classifier $\hat{h}$:

$$\hat{r} = \operatorname*{argmin}_{r \in \mathcal{R}} L_{\text{def}}^{0-1}(\hat{h}, r) \tag{3.5}$$

This procedure is particularly attractive as the two steps (3.4) and (3.5) could be cast as classification problems, and approached by powerful known tools that are devised for such problems. Despite its convenience, this method is not guaranteed to achieve the optimal loss (as in (3.3)), since it decouples the joint learning problem. Assuming that we are able to optimally solve both problems on the true distribution, let $(h^*, r^*)$ denote the solution of joint learning and $(\hat{h}, \hat{r})$ the solution of staged learning. To estimate the extent to which staged learning is sub-optimal, we define the following minimax measure $\Delta(d_1, d_2)$ for the binary label setting:

$$\Delta(d_1, d_2) = \inf_{\mathcal{H}, \mathcal{R} \in \mathfrak{H}_{d_1, d_2}} \sup_{\mu_{XYM}} L_{\text{def}}^{0-1}(\hat{h}, \hat{r}) - L_{\text{def}}^{0-1}(h^*, r^*)$$

To disentangle the above measure, the supremeum $\sup_{\mu_{XYM}}$ is a worst-case over the data distribution and expert pair, while the infimum $\inf_{\mathcal{H}, \mathcal{R} \in \mathfrak{H}_{d_1, d_2}}$ is the best-case classifier-rejector model classes with specified complexity $d_1$ and $d_2$ where $\mathfrak{H}_{d_1, d_2} = \{(\mathcal{H}, \mathcal{R}) : d(\mathcal{H}) = d_1, d(\mathcal{R}) = d_2\}$ and $d(\cdot)$ denotes the VC dimension of a hypothesis class. As a result, this measure expresses the worst-case gap between joint and staged learning when learning from the optimal model class given complexity of the predictor and rejector model classes. The following theorem provides a lower- and upper-bound on $\Delta(d_1, d_2)$.

**Theorem 1.** *For every set of hypothesis classes $\mathcal{H}, \mathcal{R}$ where $d(\cdot)$ denotes the VC-dimension of a hypothesis class, the minimax difference measure between joint and staged learning is bounded between:*

$$\frac{1}{d(\mathcal{H}) + 1} \leq \Delta(d(\mathcal{H}), d(\mathcal{R})) \leq \frac{d(\mathcal{R})}{d(\mathcal{H})} \tag{3.6}$$

Proof of the theorem can be found in Appendix B.1. The theorem implies that for any classifier and rejector hypothesis classes, we can find a distribution and an expert such that the gap between staged learning and joint learning is at least 1 over the VC dimension of the classifier hypothesis class. Meaning the more complex our classifier hypothesis class is, the smaller the gap between joint and staged learning is. On the other hand, the gap is no larger than the ratio between the rejector complexity over the the classifier complexity. Which again implies if our hypothesis class is comparatively much richer

than the rejector class, the gap between the joint and staged learning reduces. What this does not mean is that deferring to the human is not required for optimal error when the classifier model class is very large, but that training the classifier may not require knowledge of the human performance.

### 3.3.2 Data Trade-offs

Current datasets in machine learning are growing in size and are usually of the form of feature $X$ and target $Y$ pairs. It is unrealistic to assume that the human expert is able to individually provide their predictions for all of the data. In fact, the collection of datasets in machine learning often relies on crowd-sourcing where the label can either be a majority vote of multiple human experts, e.g. in hate-speech moderation Davidson *et al.* (2017), or due to an objective measurement, e.g. a lab test result for a patient medical data. In the expert deferral problem, we are interested in the predictions of a particular human expert and thus it is infeasible for that human to label all the data and perhaps unnecessary.

In the following analysis, we assume access to fully labeled data $S_l = \{(\mathbf{x}_i, y_i, m_i)\}_{i=1}^{n_l}$ and data without expert labels $S_u = \{(\mathbf{x}_i, y_i)\}_{i=n_l+1}^{n_l+n_u}$. This is a realistic form of the data we have available in practice. We now try to understand how we can learn a classifier and rejector from these two datasets. This is where we expect the staged learning procedure can become attractive as it can naturally exploit the two distinct datasets to learn.

**Joint Learning.** Learning jointly requires access to the dataset with the entirety of expert labels, thus we can only use $S_l$ to learn

$$\widetilde{h}, \widetilde{r} = \operatorname*{argmin}_{h,r} \sum_{i \in S_l} \mathbb{I}_{h(\mathbf{x}_i) \neq y_i} \mathbb{I}_{r(\mathbf{x}_i)=0} + \mathbb{I}_{y_i \neq m_i} \mathbb{I}_{r(\mathbf{x}_i)=1}$$

**Staged learning.** On the other hand, for staged learning we can exploit our expert unlabeled data to first learn $h$:

$$\hat{h} = \operatorname*{argmin}_{h} \sum_{i \in S_u} \mathbb{I}_{h(\mathbf{x}_i) \neq y_i}$$

and in the second step we learn $\hat{r}$ to minimize the joint loss with the fixed $\hat{h}$ but only on $S_l$.

**Generalization.** Given that staged learning exploits both datasets, we expect that if we have much more expert unlabeled data than labeled data, i.e. $n_u \gg n_l$, then it maybe possible to obtain better generalization guarantees from staged learning. The following proposition shows that when the Bayes optimal classifier is in the hypothesis class, then staged learning can possibly improve sample complexity over joint learning.

**Proposition 1.** *Let $\mathcal{S}_l = \{(\mathbf{x}_i, y_i, m_i)\}_{i=1}^{n_l}$ and $\mathcal{S}_u = \{(\mathbf{x}_{i+n_l}, y_{i+n_l})\}_{i=1}^{n_u}$ be two iid sample sets that are drawn from the distribution $\mu_{XYM}$ and are labeled and not labeled by the human, respectively. Assume that the optimal classifier $\bar{h} = \underset{h}{argmin}\, \mathbb{E}_{X,Y\sim\mu_{XY}}[\mathbb{I}_{h(X)\neq Y}]$ is a member of $\mathcal{H}$ (i.e., realizability).*

*Let $(\hat{h},\hat{r})$ be the staged solution and let $(\widetilde{h},\widetilde{r})$ be the joint solution obtained by learning only on $S_L$. Then, with probability at least $1-\delta$ we have for staged learning*

$$L_{def}^{0-1}(\hat{r},\hat{h}) \leq L_{def}^{0-1}(h^*,r^*) + \mathfrak{R}_{n_u}(\mathcal{H}) + 2\mathcal{R}_{n_l}(\mathcal{R}) + 2\min\left\{P(M\neq Y), \mathcal{R}_{n_l Pr(M\neq Y)/2}(\mathcal{R})\right\} + C\sqrt{\frac{\log 1/\delta}{\min(n_l,n_u)}} + P(M\neq Y)e^{-n_l\frac{Pr(M\neq Y)^2}{2}} \tag{3.7}$$

*while for joint learning we have:*

$$L_{def}^{0-1}(\widetilde{r},\widetilde{h}) \leq L_{def}^{0-1}(h^*,r^*) + \mathfrak{R}_{n_l}(\mathcal{H}) + 2\mathcal{R}_{n_l}(\mathcal{R}) + 2\mathcal{R}_{n_l Pr(M\neq Y)/2}(\mathcal{R}) + C\sqrt{\frac{\log 1/\delta}{n_l}} + P(M\neq Y)e^{-n_l\frac{Pr(M\neq Y)}{2}} \tag{3.8}$$

Proof of the proposition can be found in Appendix B.2. From the above proposition, when the Bayes classifier is in the hypothesis class, the upper bound for the sample complexity required to learn the classifier and rejector is reduced by only paying the Rademacher complexity of the hypothesis class on the unlabeled data compared to on the potentially smaller labeled dataset. The Rademacher complexity is a measure of model class complexity on the data and can be related to the VC dimension.

While in this case study staged learning may improve the generalization error bound comparing to that of joint learning, the number of labeled samples for both to achieve $\varepsilon$-upper-bound on the true risk is of order $O(\frac{\log 1/\varepsilon}{\varepsilon^2})$. As we can see, there exist computational and statistical trade-offs between joint and staged learning. While joint learning leads to more accurate systems, it is computationally harder to optimize than staged learning. In the next section, we investigate whether it is possible to more efficiently solve the joint learning problem while still retaining its favorable guarantees in the multiclass setting.

## 3.4 Surrogate Losses For Joint Learning

A common practice in machine learning is to propose surrogate loss functions, which often are continuous and convex, that approximate the original loss function we care about Bartlett *et al.* (2006). The hope is that these surrogates are more readily optimized and minimizing them leads to predictors that also minimize the original loss. In their work on expert deferral, Mozannar and Sontag (2020a) reduces the learning to defer problem

to cost-sensitive learning which enables them to use surrogates for cost-sensitive learning in the expert deferral setting. We follow the same route in deriving our novel family of surrogate losses. We now recall the reduction in Mozannar and Sontag (2020a): define the random costs $\mathbf{c} \in \mathbb{R}_+^{K+1}$ where $c(i)$ is the $i'$th component of $\mathbf{c}$ and represents the cost of predicting label $i \in [K+1]$. The goal of cost sensitive learning is to build a predictor $\widetilde{h} : \mathcal{X} \to [K+1]$ that minimizes the cost-sensitive loss $\mathbb{E}[c(\widetilde{h}(X))]$. The reduction is accomplished by setting $c(i) = \ell_{\text{AI}}\big(X, Y, i\big)$ for $i \in [K]$ while $c(K+1)$ represents the cost of deferring to the expert with $c(K+1) = \ell_{\text{exp}}(X, Y, M)$. Thus, the predictor $\widetilde{h}$ learned in cost-sensitive learning implicitly defines a classifier-rejector pair $(h, r)$ with the following encoding:

$$h(\mathbf{x}), r(\mathbf{x}) = \begin{cases} h(\mathbf{x}) = i, r(\mathbf{x}) = 0, & \text{if } \widetilde{h}(\mathbf{x}) = i \in [K] \\ h(\mathbf{x}) = 1, r(\mathbf{x}) = 1 & \text{if } \widetilde{h}(\mathbf{x}) = K+1 \end{cases} \tag{3.9}$$

Note that when $\widetilde{h}(\mathbf{x}) = K+1$ the classifier $h$ is left unspecified and thus we assign it a dummy value of 1. Cost-sensitive learning is a non-continuous and non-convex optimization problem that makes it computationally hard to solve in practice. In order to approximate it, we propose a novel family of cost-sensitive learning loss functions that extend any multi-class loss function to the cost-sensitive setting.

First we parameterize our predictor $\widetilde{h}$ with $K+1$ functions $f_i : \mathcal{X} \to \mathbb{R}$ and define the predictor to be the max of these $K+1$ functions: $\widetilde{h}_{\mathbf{f}}(\mathbf{x}) = \arg\max_i f_i(\mathbf{x})$. Note that $\widetilde{h}_{\mathbf{f}}$ gives rise to the classifier-rejector pair $(h_{\mathbf{f}}, r_{\mathbf{f}})$ according to the decoding rule (3.9).

Formally, let $\ell_\phi(y, \mathbf{f}(\mathbf{x})) : [K+1] \times \mathbb{R}^{K+1} \to \mathbb{R}$ be a surrogate loss function of the zero-one loss for multi-class classification. We define the extension of this surrogate to the cost-sensitive setting as:

$$\widetilde{\ell}_\phi(\mathbf{c}, \mathbf{f}(\mathbf{x})) = \sum_{i=1}^{K+1} \big[ \max_{j \in k+1} c(j) - c(i) \big] \ell_\phi(i, \mathbf{f}(\mathbf{x})) \tag{3.10}$$

Note that if $\ell_\phi$ is continuous or convex, then because $\ell_{\mathbf{c},\phi}$ is a finite positively-weighted sum of $\ell_\phi$'s, then $\ell_{\mathbf{c},\phi}$ is also continuous or convex, respectively. We show in the following proposition, that if $\ell_\phi$ is a consistent surrogate for multi-class classification, then $\widetilde{\ell}_\phi$ is consistent for cost-sensitive learning and by the reduction above is also consistent for learning to defer.

**Proposition 2.** *Suppose $\ell_\phi(y, \mathbf{f}(\mathbf{x}))$ is a consistent surrogate for multi-class classification, meaning if the surrogate is minimized over all functions then it also minimizes the misclassification loss:*

*let* $\boldsymbol{f}^* = \text{arginf}_{\mathbf{f}} \mathbb{E}\left[\ell_\phi(Y, \mathbf{f}(\mathbf{x}))\right]$, *then:* $\widetilde{\mathbf{h}}_{f^*} = \text{arginf}_{\mathbf{h}} \mathbb{E}[\mathbb{I}_{Y \neq \mathbf{h}(X)}]$, *where* $\widetilde{\mathbf{h}}_{f^*}$ *is defined as above.*

*Then, our surrogate $\widetilde{\ell}_\phi(\mathbf{c},\mathbf{g}(\mathbf{x}))$ defined in (3.10) is a consistent surrogate for cost-sensitive learning and thus for learning to defer:*

*let $\widetilde{\boldsymbol{f}}^* = \text{arginf}_{\mathbf{g}}\mathbb{E}\left[\widetilde{\ell}_\phi(\mathbf{c},\mathbf{f}(\mathbf{x}))\right]$, then: $h^*_{\mathbf{f}}, r^*_{\mathbf{f}} = \text{arginf}_{h,r} L^{0-1}_{def}(h,r)$, with $(h^*_{\mathbf{f}}, r^*_{\mathbf{f}})$ defined in (3.9)*

Proof of the proposition can be found in Appendix B.3. To illustrate the family of surrogates implied by Proposition 2, we first start by recalling a family of surrogates for multi-class classification. Theorem 4 of Zhang (2004) shows that there is a family of consistent surrogates for $0-1$ loss in multi-class classification parameterized by three functions $(u,s,t)$ and takes the form $\ell_\phi(y,\mathbf{f}(\mathbf{x})) = u(f_y(\mathbf{x})) + s\Big(\sum_{j=1}^{K+1} t\big(f_j(x)\big)\Big)$. This family is consistent under certain conditions of the aforementioned functions.

Now we show with a few of examples that this family encompasses some popular surrogates used in cost sensitive learning:

**Examples.** *(1) If we set $u(x) = -2x$, $s(x) = x$, and $t(x) = x^2$, then we can obtain a weighted quadratic loss:*

$$\widetilde{L}_2 = \mathbb{E}[\pi \sum_{i=1}^{K+1} |f_i - q(i)|^2], \tag{3.11}$$

*where $q(i)$ is the normalized expected value of $\max_{j\in[K+1]} c(j) - c(i)$ given $X = x$, and $\pi$ represents the normalization term.*

*(2) If we set $u(x) = -x$, and $s(x) = \log(x)$ and $t(x) = e^x$, then we have $a\psi'(x) + t'(x) = -a + e^x = 0$, and as a result $x = \log a$, which is an increasing function of a. As a result, the surrogate loss*

$$\widetilde{L}_{CE}(\mathbf{f}) = \mathbb{E}[-\sum_{i=1}^{K+1} \big(\max_j c(j) - c(i)\big) \log \frac{e^{f_i(X)}}{\sum_{k=1}^{K+1} e^{f_k(X)}}] \tag{3.12}$$

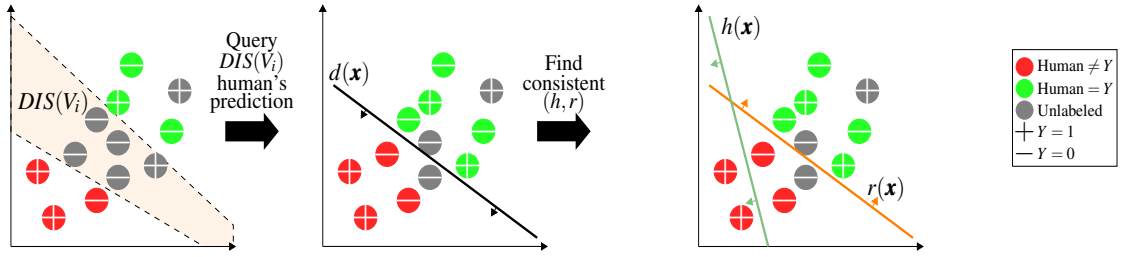
*which is the loss defined in Mozannar and Sontag (2020a) and used for learning to defer.*

### 3.4.1 Theoretical Properties of Surrogate

**Goodness of a Surrogate.** Given a surrogate, how can we quantify how well it approximates our original loss? One avenue is through the surrogate excess-risk bound as follows. Let $\widetilde{L}$ be a surrogate for the loss function $L$, and let $\widetilde{h}^*$ be the minimizer of the surrogate and $h^*$ the minimizer of $L$. We call the *excess surrogate risk* Bartlett *et al.* (2006) the following quantity if we can find a *calibration function* $\psi$ such that for any $h$ we have:

$$\psi(L(h) - L(h^*)) \leq \widetilde{L}(h) - \widetilde{L}(\widetilde{h}^*) \tag{3.13}$$

The excess surrogate risk bound tells us if we are $\varepsilon$-close to the minimizer of the surrogate, then we are $\psi^{-1}(\varepsilon)$-close to the minimizer of the original loss.




**Step 1:** Identify points to query for human's prediction

**Step 2:** Query for human's predictions and learn their error boundary $d$

**Step 3:** Learn classifier-rejector pair given human's predictions

Figure 3.1: Illustration of our active learning algorithm Disagreement on Disagreements (DoD) (1). At each round, we compute the disagreement set for our predictors of the human label disagreement, we then query the human for their prediction on these points. After we learn the expert error boundary, we then learn a consistent classifier-rejector pair.

We now show that the family of surrogates defined in (3.10) has a polynomial excess-risk bound and furthermore prove an excess-risk bound for the surrogate loss function $\widetilde{L}_{CE}$ defined in Mozannar and Sontag (2020a).

**Theorem 2.** *Suppose that $\psi(x) = Cx^{\varepsilon}$, for $\varepsilon \in [1,\infty)$ is a calibration function for the multiclass surrogate $\ell_\phi$ and if $|c(i)| \leq M$ for all $i$, then $\psi'(x) = \frac{C}{M^{\varepsilon-1}}x^{\varepsilon}$ is a calibration function of $\widetilde{\ell}_\phi(\mathbf{c},\cdot)$.*

*As an example, for the surrogate $\widetilde{L}_{CE}$ (3.12) the calibration function is $\psi(x) = \frac{1}{16MK}x^2$.*

Proof of the theorem can be found in Appendix B.4. Note, that Nowak-Vila *et al.* (2019) proved that the for the cross-entropy loss the calibration function $\phi$ is of order $\Theta(\varepsilon^2)$ which is in accordance with our results.

**Generalization Error.** Equipped with the excess surrogate risk bound, we can now study the sample complexity properties of minimizing the surrogates proposed. For concreteness, we focus on the surrogate $\widetilde{L}_{CE}$ of Mozannar and Sontag (2020a) when reduced to the learning to defer setting. The following theorem proves a generalization error bound when minimizing the surrogate $\widetilde{L}_{CE}$ for learning to defer.

**Theorem 3.** *Let $K$ denote the number of classes, and let $\mathcal{F}$ be a hypothesis class of functions $f_i : \mathcal{X} \to \mathbb{R}$ with bounded infinity norm $\|f_i\|_\infty < C$. Given $\hat{\mathbf{f}} \in \mathcal{F}^{k+1}$ the empirical*

*minimizer of the surrogate loss $\widetilde{L}_{CE}$, then we have with probability at least $1-\delta$, we have*

$$\psi\big(L^{0-1}_{def}(h_{\hat{\mathbf{f}}}, r_{\hat{\mathbf{f}}}) - \min_{\mathbf{f}} L^{0-1}_{def}(h_{\mathbf{f}}, r_{\mathbf{f}})\big) \leq 2(K+1)\mathcal{R}_n\big(\mathcal{F}\big) + \sqrt{\frac{(8C - 4\log(K+1))\log 2/\delta}{n}} + 2(K+2)\min\{Pr(M \neq Y), \mathcal{R}_{nPr(M\neq Y)/2}\big(\mathcal{F}\big)\} + CPr(M \neq Y)(k+2)e^{-nPr^2(M\neq Y)/2} + e_{\phi-appr}, \tag{3.14}$$

*where $e_{\phi-appr}$ is the approximation error for the $\phi$-surrogate, which is defined as*

$$e_{\phi-appr} = \min_{\mathbf{f}\in\mathcal{F}^{k+1}} \widetilde{L}_{CE}(\mathbf{f}) - \min_{\mathbf{f}} \widetilde{L}_{CE}(\mathbf{f}). \tag{3.15}$$

Proof of the theorem can be found in Appendix B.5. Comparing the sample complexity estimate for minimizing the surrogate to that of minimizing the 0-1 loss as computed by Mozannar and Sontag (2020a), we find that we pay an additional penalty for the complexity of the hypothesis class in addition to the higher sample complexity that scales with $O(\frac{\log \varepsilon}{\varepsilon^4})$ due to the calibration function. To compensate for such increase in sample complexity, in the next section we seek to design active learning schemes that reduce the required amount of human labels for learning.

## 3.5 Active Learning for Expert Predictions

### 3.5.1 Theoretical Understanding

In Section 3.3, we assumed that we have a randomly selected subset of data that is labeled by the human expert. In a practical setting, we may assume that we have the ability to choose which points we would like the human expert to predict on. For example, when we deploy an X-ray diagnostic algorithm to a new hospital, we can interact with each radiologist for a few rounds to build a tailored classifier-rejector pairs according to their individual abilities.

Therefore, we assume that we have access to the distribution of instances $\mathbf{x}$ and their labels and we could query for the expert's prediction on each instance. The goal is to query the human expert on as few instances as possible while being able to learn an approximately optimal classifier-rejector pair. To make progress in theoretical understanding, we assume that we can achieve zero loss with an optimal classifier-rejector pair:

**Assumption 1** (Realizability)**.** *We assume that the data is realizable by our joint class $(\mathcal{H},\mathcal{R})$: there exists $h^*, r^* \in \mathcal{H}\times\mathcal{R}$ that have zero error $L(h^*, r^*) = 0$.*

In this section, the algorithms we develop apply to the multiclass setting but we restrict the theoretical analysis to binary labels. The fundamental algorithm in active learning in the realizable case for classification is the CAL algorithm Hanneke (2014). The algorithm keeps track of a version class of the hypothesis space that is consistent with the data so far and then at each step computes the disagreement set: the points on which there exists two classifiers in the hypothesis class with different predictions, and then picks at random a set of points from this disagreement set. We start by initializing our version space by taking advantage of Assumption 1:

$$V_0 = \{h, r \in \mathcal{H} \times \mathcal{R} : \forall \mathbf{x}, r(\mathbf{x}) = 0 \rightarrow\ h(\mathbf{x}) = y\} \tag{3.16}$$

The above initialization of the version space assumes we know the label of all instances in our support. Alternatively, one could collect at most $O\big(\frac{1}{\varepsilon}(d(\mathcal{H})\log\frac{1}{\varepsilon} + \log\frac{1}{\delta})\big)$ labels of instances and that would be sufficient to test for realizability of our classifier with error $\varepsilon$ (see Lemma 3.2 of Hanneke (2014)).

The main difference with active learning for classification is that we are not able to compute the disagreement set for the overall prediction of the deferral system as it requires knowing the expert predictions. However, we know that a necessary condition for disagreement is that there exists a feasible pair of classifiers-rejectors where the rejectors disagree. Suppose $(h_1, r_1)$ and $(h_2, r_2)$ are in our current version space. These two pairs can only disagree when on an instance $\mathbf{x}$: $r_1(\mathbf{x}) \neq r_2(\mathbf{x})$, since otherwise when both defer, the expert makes the same prediction, and when both do not defer, both classify the label correctly by the realizability assumption. Thus, we define the disagreement set in terms of only the rejectors that are in the version space at each round $j$:

$$DIS(V_{j-1}) = \{x \in \mathcal{X} \mid \exists (h_1, r_1), (h_2, r_2) \in V_{j-1} s.t.\ r_1(x) \neq r_2(x)\} \tag{3.17}$$

Then we ask for the labels of $k$ instances in $DIS(V_{j-1})$ to form $\mathcal{S}_j = \{(\mathbf{x}_i, y_i, m_i) : \mathbf{x}_i \in DIS(V_j)\}$ and we update the version space as

$$V_j = \{(h, r) \in V_{j-1} \mid \forall (\mathbf{x}, y, m) \in \mathcal{S}_j, \text{if } r(\mathbf{x}) = 1 \rightarrow y = m\} \tag{3.18}$$

Now, we prove that the above rejector-disagreement algorithm will converge if the optimal unique classifier-rejector pair is unique:

**Proposition 3.** *Assume that there exists a unique pair $(h^*, r^*) \in \mathcal{H} \times \mathcal{R}$ that have zero error $L(h^*, r^*) = 0$. Let $\Theta$ be defined as:*

$$\Theta = \sup_{t>0} \frac{Pr(X \in DIS(B((h^*, r^*), t)))}{t} \tag{3.19}$$

*where $B\big((h, r), t\big) = \{(h', r') \in \mathcal{H} \times \mathcal{R} : Pr(r(X)M + (1 - r(X))h(X) \neq r'(x)M + (1 - r'(X))h'(X)) \leq t\}$.*

*Then, running the rejector-disagreement algorithm with $k = O\big(\Theta^2((d(\mathcal{H}) + d(\mathcal{R})\log\Theta +$*

$\log\frac{1}{\delta}+\log\log\frac{1}{\varepsilon}$)) *for* $\log(1/\varepsilon)$ *iterations will return classifier-rejector with* $\varepsilon$ *error and with probability at least* $1-\delta$.

Proof of the proposition can be found in Appendix B.6.

### 3.5.2 Disagreement on Disagreements

If we remove the uniqueness assumption for the rejector-disagreement algorithm in the previous subsection, we show in Appendix B.7 with an example that the algorithm no longer converges as $DIS(V)$ can remain constant. We expect that the uniqueness assumption may not hold in practice, so we now hope to design algorithms that do not require it. Instead, we now make a new assumption that we can learn the error boundary of the human expert via a function $f \in \mathcal{D}$, that is given any sample $(x,y,m)$ we have $f(x) = \mathbb{I}_{y\neq m}$. This assumption is identical to those made in active learning for cost-sensitive classification Krishnamurthy *et al.* (2017). This assumption is formalized as follows:

**Assumption 2.** *We assume that there exists* $f^* \in \mathcal{D}$ *such that* $Pr(\mathbb{I}_{M\neq Y} \neq f(X)) = 0$.

Our new active learning will now seek to directly take advantage of Assumption 2. The algorithm consists of two stages: the first stage runs a standard active learning algorithm, namely CAL, on the hypothesis space $\mathcal{D}$ to learn the expert disagreement with the label with error at most $\varepsilon$. In the second stage, we label our data with the predictor $\hat{f}$ that is the output of the first stage, and then learn a classifier-rejector pair from this pseudo-labeled data. Key to this two stage process, is to show that the error from the first stage is not too amplified by the second stage. The algorithm is named Disagreement on Disagreements (DoD) and is described in detail in Algorithm box 1.

**Algorithm 1:** Active Learning algorithm DoD (Disagreement on disagreements)

**Input**: parameter $n_u$, $T$, $k$, class $\mathcal{D}$, $\mathcal{H}$, and $\mathcal{R}$

1. $V \leftarrow \mathcal{D}$
2. **for** $i \in \{1,\ldots,T\}$ **do**
   - Sample from $\mu_X$ until you have $k$ samples $\{\mathbf{x}_i\}_{i=1}^k$ within $DIS_2(V)$
   - Query for $\{(y_i,m_i)\}_{i=1}^k$ for the samples $\{\mathbf{x}_i\}_{i=1}^k$
   - Update $V \leftarrow \{d \in V : \forall j\, d(\mathbf{x}_j) = \mathbb{I}_{m_j\neq y_j}\}$

   **end**
4. Collect $n_u$ samples $\{(\mathbf{x}'_i,y'_i)\}_{i=1}^{n_u}$ from $\mu_{XY}$

**Return:** $(h,r) \in \mathcal{H}\times\mathcal{R}$ such that $\sum_{(\mathbf{x}'_i,y'_i)} \mathbb{I}_{h(\mathbf{x}'_i)\neq y'_i}\big(1-r(\mathbf{x}'_i)\big) + r(\mathbf{x}'_i)d(\mathbf{x}'_i) = 0$, for some $d \in V$

In the following we prove a label complexity guarantee for Algorithm 1.

**Theorem 4.** *Let us define* $\Theta_2$ *as*

$$\Theta_2 = \sup_{t>0} \frac{Pr(X \in DIS_2(B_2(f^*,t)))}{r}, \tag{3.20}$$

*where* $B_2(f,t) = \{f' \in \mathcal{D} \mid Pr(f'(X) \neq f(X)) \leq t\}$, *and* $DIS_2(V) = \{\mathbf{x} \in \mathcal{X} \mid \exists f_1, f_2 \in V, f_1(\mathbf{x}) \neq f_2(\mathbf{x})\}$.

*Assume we have* $\mathcal{H}, \mathcal{R}, \mathcal{D}$ *that satisfy assumption 1 and 2.*

*Then for* $n_u = O(\frac{\log 1/\delta + \max\{d(\mathcal{H}), d(\mathcal{R})\} \log 1/\varepsilon}{\varepsilon^2})$, *and* $k = d(\mathcal{D})\Theta_2 \log(\frac{\Theta_2}{\delta}\log(\frac{1}{\varepsilon}))$, *then Algorithm 1 takes* $T = O(\log \frac{1}{\varepsilon})$ *iterations to output a solution with* $\varepsilon$*-upper-bound on deferral loss with probability at least* $1-\delta$. *As a result, the sample complexity of labeled data* $n_l$ *is* $O\big(d(\mathcal{D})\Theta_2 \log(\frac{\Theta_2}{\delta}\log(\frac{1}{\varepsilon}))\log(\frac{1}{\varepsilon})\big)$.

Proof of the proposition can be found in Appendix B.8. Recall that in Proposition 1, where the labeled data was chosen at random, the sample complexity $n_u$ is in order $O(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon})$. As we see in Theorem 4, the proposed active learning algorithm reduces sample complexity to $O(\log\frac{1}{\varepsilon})$, with the caveat that realizability is assumed for active learning. Further, note that for this algorithm, in contrast to previous subsection, the uniqueness of the consistent pair $(h,r)$ is not needed anymore. However, this algorithm ignores the classifier and rejector classes when querying for points, which makes the sample complexity $n_l$ dependent only on the complexity of $\mathcal{D}$ instead of $\mathcal{H}, \mathcal{R}$. In the next section, we try to understand how to use surrogate loss functions to practically optimize for our classifier-rejector pair.

## 3.6 Experimental Illustration

Code for our experiments is found in `https://github.com/clinicalml/active_learn_to_defer`.

**Dataset.** We use the CIFAR-10 image classification dataset Krizhevsky *et al.* (2009) consisting of $32 \times 32$ color images drawn from 10 classes. We consider the human expert models considered in Mozannar and Sontag (2020a): if the image is in the first 5 classes the human expert is perfect, otherwise the expert predicts randomly. Further experimental details are in Appendix B.9.

**Model and Optimization.** We parameterize the classifier and rejector by a convolutional neural network consisting of two convolutional layers followed by two feedforward layers. For staged learning, we train the classifer on the training data optimizing for performance on a validation set, and for the rejector we train a network to predict the expert error and defer at test time by comparing the confidence of the classifier and the expert as in Raghu *et al.* (2019). For joint learning, we use the loss $L^{\alpha}_{CE}$, a simple

extension of the loss (3.12) in Mozannar and Sontag (2020a), optimizing the parameter $\alpha$ on a validation set.

**Model Complexity Gap.** In Figure 3.2, we plot the difference of accuracy between joint learning and staged learning as we increase the complexity of the classifier class by increasing the filter size of the convolutional layers and the number of units in the feedforward layers. Model complexity is captured by the number of parameters in the classifier which serves only as a rough proxy of the VC dimension that varies in the same direction. The difference is decreasing as predicted by Theorem 1 as we increase the classifier class complexity as we fix the complexity of the rejector.

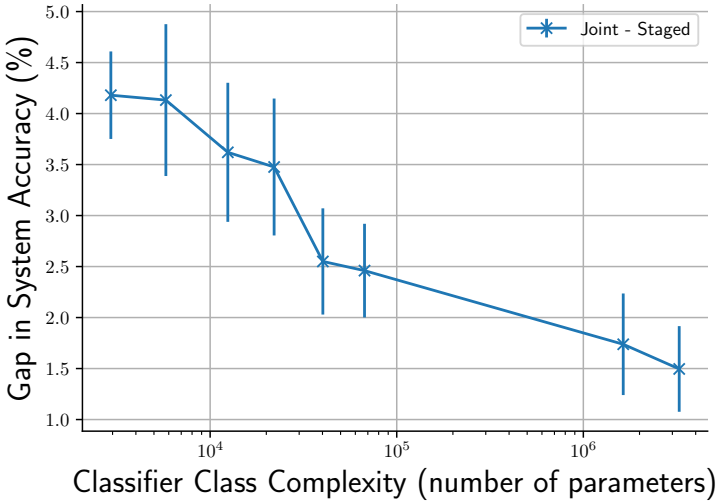



Figure 3.2: Difference of accuracy between joint learning and staged learning of the classifier-rejector pair (y-axis is log scale of number of parameters).

**Data Trade-Offs.** In Figure 3.3, we plot the of accuracy between joint learning and staged learning when only a subset of the data is labeled by the expert as in Section 3.3.2. We plot the average difference across 10 trials and error bars denote standard deviation. We only plot the performance of joint learning when initialized first on the unlabeled data to predict the target and then trained on the labeled expert data to defer, we denote this approach as 'Joint-SemiSupervised'. For staged learning, the classifier is trained on all of the data $S_l \cup S_u$, while for joint learning we only train on $S_l$. We can see that when there is more unlabeled data than labeled, staged learning outperforms joint learning in accordance with Proposition 1. The heuristic method 'Joint-SemiSupervised' improves on the sample complexity of 'Joint' but still lags behind the Staged approach in low data regimes.

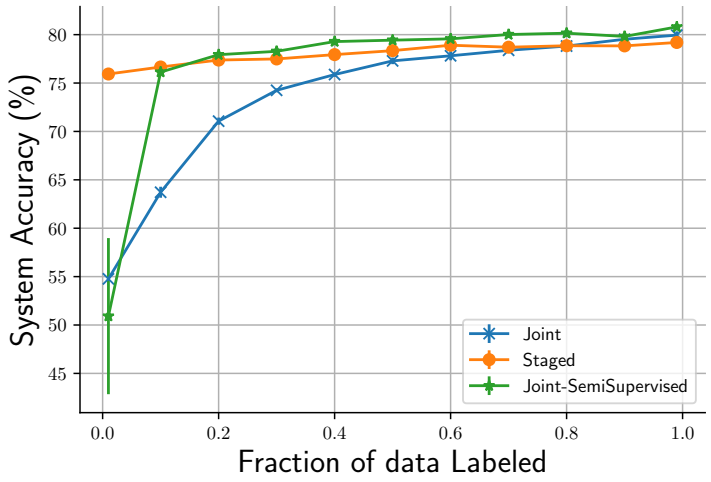



Figure 3.3: Performance of joint learning and staged learning as we increase the ratio of the data labeled by the expert $\frac{n_l}{n_u+n_l}$.

**DoD algorithm.** In Figure 3.4, we plot corresponding errors of the DoD algorithm and we compare them to the staged and joint learning. The features $\mathbf{x}$ of the synthetic data in here is generated from a uniform distribution on interval $[0,1]$, and the labels $y$ are equal to 1 where $\mathbf{x} > 0.3$ (full-information region) and are equal to random outcomes of a $Bernoulli(0.5)$ distribution otherwise (no-information region). The human's decision is inaccurate ($M \neq Y$) for $X > 0.3$ and accurate ($M = Y$) otherwise. We further assume each hypothesis class of rejectors and classifiers be 100 samples of half-spaces over the interval $[0,1]$. The error plotted in Figure 3.4 is an average of 1000 random generations of training data. The test set is formed by $N_{test} = 1000$ samples that are generated from the same distribution as training data. Here, we note that the number of unlabeled data in staged learning is set $N_u = 100$. The result of this experiment shows that in the DoD algorithm, one needs less number of samples that are labeled by human to reach a similar level of error.

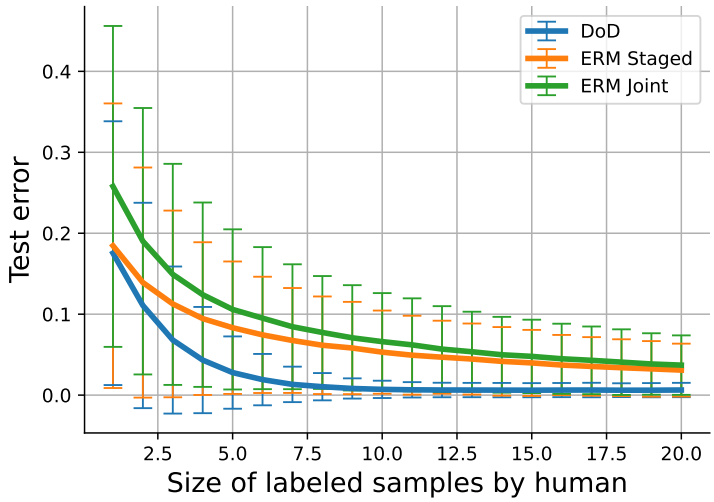



Figure 3.4: Error of the DoD algorithm compared to staged and joint learning for increasing number of training data that are labeled by human.

## 3.7 Discussion

In this work, we provided novel theoretical analysis of learning in the expert deferral setting. We first analyzed the gap in performance between jointly learning a classifier and rejector, and a staged learning approach. While our theorem on the gap is a worst-case statement, an experimental illustration on CIFAR-10 indicates a more general trend. Further analysis could explicitly compute the gap for certain hypothesis classes of interest. We further analyzed a popular approach to jointly learning to defer, namely consistent surrogate loss functions. To that end, we proposed a novel family of surrogates that generalize prior work and give a criteria, namely the surrogate excess-risk bound for evaluating surrogates. Future work will try to instantiate members of this family that minimize the excess-risk bound and provide improved empirical performances. Driven by the limited availability of human data, we sought to design active learning schemes that requires a minimal amount of labeled data for learning a classifier-rejector pair. While our results hold for the realizable setting, we believe it is feasible to generalize to the agnostic setting. Future work will also build and test practical active learning algorithms inspired by our theoretical analysis.

# Chapter 4

# A Post-Processing Framework for Multi-Objective Learn-to-Defer Problems

## 4.1 Introduction

Machine learning algorithms are increasingly used in diverse fields, including critical applications, such as medical diagnostics Vermeulen *et al.* (2023) and predicting optimal prognostics Sammut *et al.* (2022). To address the sensitivity of such tasks, existing approaches suggest keeping the human expert in the loop and using the machine learning prediction as advice Jiang *et al.* (2012), or playing a supportive role by taking over the tasks on which machine learning is uncertain Kompa *et al.* (2021); Raghu *et al.* (2019); Beede *et al.* (2020). The abstention of the classifier in making decisions, and letting the human expert do so, is where the paradigm of learn-to-defer (L2D) started to exist.

The development of L2D algorithms has mainly revolved around optimizing the accuracy of the final system under such paradigm Raghu *et al.* (2019); Mozannar and Sontag (2020a). Although they achieve better accuracy than either the machine learning algorithm or the human expert in isolation, these works provide inherently single-objective solutions to the L2D problem. In the critical tasks that are mentioned earlier, more often than not, we face a challenging multi-objective problem of ensuring the safety, algorithmic fairness, and practicality of the final solution. In such settings, we seek to limit the cost of incorrect decisions Metz (1978), algorithmic biases Chen *et al.* (2023a), or human expert intervention Okati *et al.* (2021b), while optimizing the accuracy of the system. Although the seminal paper that introduced the first L2D algorithm targeted an instance of such multi-objective problem Madras *et al.* (2018), a general solution to such class of problems, besides specific examples Donahue *et al.* (2022); Okati *et al.* (2021b); Narasimhan *et al.* (2022b, 2024), has remained unknown to date.

Multi-objective machine learning extends beyond the realm of L2D problems. A prime example that is extensively studied in various settings is ensuring algorithmic fairness Corbett-Davies *et al.* (2017) while optimizing accuracy. Recent advances in the algorithmic fairness literature have suggested the superiority of *post-processing* methodology for

tackling this multi-objective problem Xian *et al.* (2023); Chen *et al.* (2023b); Cruz and Hardt (2023); Zeng *et al.* (2022). Post-processing algorithms operate in two steps: first, they find a calibrated estimation of a set of probability scores for each input via learning algorithms, and then they obtain the optimal predictor as a function of these scores. Similarly, in a recent set of works, optimal algorithms to reject the decision-making under a variety of secondary objectives are determined via post-processing algorithms Narasimhan *et al.* (2022b, 2024), which is in line with classical results such as Chow's rule Chow (1970) that is the simplest form of a post-processing method, thresholding the likelihood.

Inspired by the above works, in this paper, we fully characterize the solution to multi-objective L2D problems using a post-processing framework. In particular, we consider a deferral system together with a set of conditional performance measures $\{\Psi_0, \ldots, \Psi_m\}$ that are functions of the system outcome $\hat{Y}$, the target label $Y$, and the input $X$. The goal is to optimize the average value of $\Psi_0$ over data distribution while keeping the average value of the rest of performance measures $\Psi_1, \ldots, \Psi_m$ for all inputs under control. As an example, in binary classification, $\Psi_0$ can be the $0-1$ deferral loss function, while $\Psi_1$ can be the difference between positive prediction rates of $\hat{Y}$ for all instances of $X$ that belong to demographic group $A = 0$ or $A = 1$. The solution for which we aim optimizes the accuracy while assuring that the demographic parity measure between the two groups is bounded by a tolerance value $\delta_1 \in [0, 1]$.

To provide the optimal solution, we move beyond staged learning Charusaie *et al.* (2022a) methodology, in which the classifier $h(x)$ is trained in the absence of human decision-makers, and then the optimal rejection function $r(x)$ is obtained for that classifier to decide when the human expert should intervene ($r(x) = 1$). Instead, we jointly obtain the classifier and rejection function. The reason that we avoid this methodology is that firstly, objectives such as algorithmic fairness are not compositional, i.e., even if the classifier and the human are fair, due to the emergence of Yule's effect Ruggieri *et al.* (2023) the obtained deferral system is not necessarily fair (see Appendix C.1), and in fact abstention systems can deter the algorithmic biases Jones *et al.* (2020). Secondly, the feasibility of constraints is not guaranteed under staged learning methodology Yin *et al.* (2023), e.g., there can be cases in which achieving a purely fair solution is impossible, while this occurs neither in vanilla classification Cruz and Hardt (2023) nor in our solution.

This paper shows that the joint learning of classifier and rejection function for finding the optimal multi-objective L2D solution boils down to a generalization of the fundamental Neyman-Pearson lemma Neyman and Pearson (1933). This lemma is initially introduced in studying hypothesis testing problems and characterizes the most powerful test (i.e., the test with the highest true positive rate) while keeping the significance level (true negative rate) under control. As a natural extension to this paradigm, we consider a multi-hypothesis setting where for each true positive prediction and false negative prediction, we receive a reward and loss, respectively. Then, we show that the extension of Neyman-Pearson lemma to this setting provides us with a solution for our multi-objective

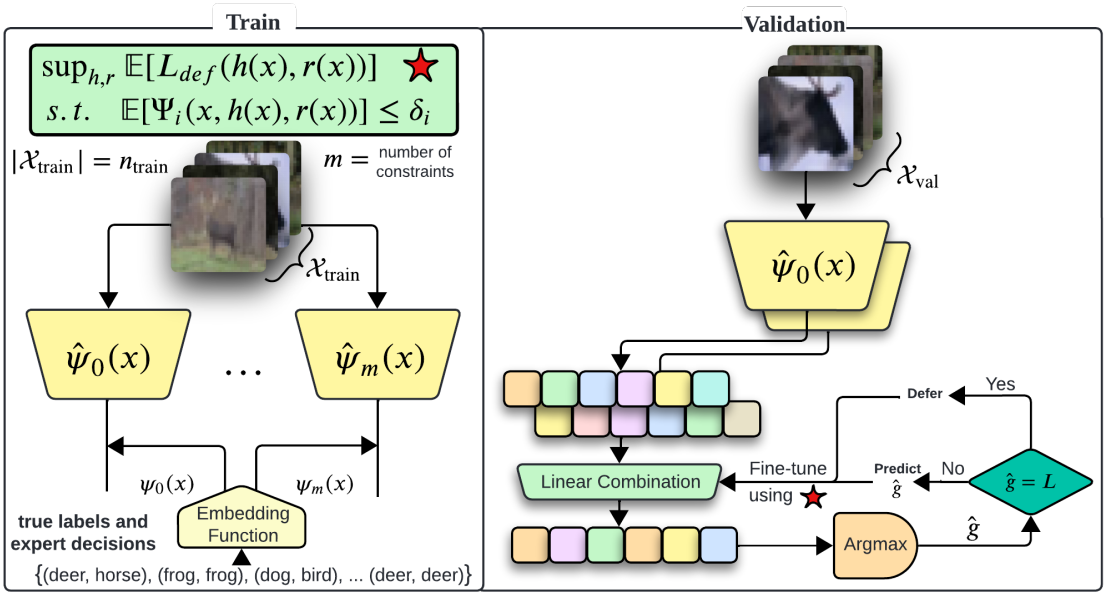



Figure 4.1: Diagram of applying $d$-GNP to solve multi-objective L2D problem via Algorithm 2. The role of randomness is neglected due to simplicity of presentation.

L2D problem.

In summary, the contribution of this paper is as below:

- In Section 4.3, we show that obtaining the optimal deterministic classifier and rejection function under a constraint is, in general, an NP-Hard problem, then
- by introducing randomness, we rephrase the multi-objective L2D problem into a functional linear programming.
- In Section 4.4, we show that such linear programming problem is an instance of $d$-dimensional generalized Neyman-Pearson ($d$-GNP) problem, then
- we characterize the solution to $d$-GNP problem, and we particularly derive the corresponding parameters of the solution when the optimization is restricted by a single constraint.
- In Section 4.5, we show that a post-processing algorithm that is based on $d$-GNP solution generalizes in constraints and objective with the rate $O(\sqrt{\log n/n}, \sqrt{\log(1/\varepsilon)/n}, \varepsilon')$ and $O((\log n/n)^{1/2\gamma}, (\log(1/\varepsilon)/n)^{1/2\gamma}, \varepsilon')$, respectively, with probability at least $1-\varepsilon$ where $n$ is the size of the set using which we fine-tune the algorithm, $\varepsilon'$ measures the accuracy of learned post-processing scores, and $\gamma$ is a parameter that measures the sensitivity of the constraint to the change of the predictor. Then,
- we show that the use of in-processing methods in L2D problem does not necessarily generalize to the unobserved data, and finally
- we experiment our post-processing algorithm on two tabular datasets, and observe its performance compared to the baselines for ensuring demographic parity and

equality of opportunity on final predictions.

Lastly, the $d$-GNP theorem has potential use cases beyond the L2D problem, particularly in vanilla classification problems under constraints. However, such applications are beyond the scope of this paper, and except for a brief explanation of the use of $d$-GNP in algorithmic fairness for multiclass classification, we leave them to future works.

## 4.2 Related Works

Human and ML's collaboration in decision-making has been demonstrated to enhance the accuracy of final decisions compared to predictions that are made solely by humans or ML Kamar *et al.* (2012); Tan *et al.* (2018). This overperformance is due to the ability to estimate the accuracy and confidence of each agent on different regions of data and subsequently allocate instances between human and ML to optimize the overall accuracy Bansal *et al.* (2021b). Since the introduction of the L2D problem, the implementation of its optimal rule has been the focus of interest in this field Cao *et al.* (2022); Mozannar and Sontag (2020a); Charusaie *et al.* (2022a); Narasimhan *et al.* (2022b); Cao *et al.* (2024); Liu *et al.* (2024); Mozannar *et al.* (2023a); Mao *et al.* (2024). The multi-objective classification with abstention problems is studied for specific objectives in Madras *et al.* (2018); Okati *et al.* (2021b); Mozannar *et al.* (2023a) via in-processing methods. The application of Neyman-Pearson lemma for learning problems with fairness criteria is recently introduced in Zeng *et al.* (2024).

We refer the reader to Appendix C.2 for further discussion on related works.

## 4.3 Problem Setting

Assume that we are given input features $x_i \in \mathcal{X}$ , corresponding labels $y_i \in \mathcal{Y} = \{1, \ldots, L\}$, and the human expert decision $m_i$ for such input, and assume that these are i.i.d. realizations of random variables $X, Y, M \sim \mu = \mu_{XYM}$. Since there exists randomness in the human decision-making process, for the sake of generality, we treat $M$ as a random variable similar to $Y$ and do not assume that $m_i = m(x_i)$ for some function $m$. Further, assume that for the true label $y$ and a certain feature vector $x$, the cost of incorrect predictions is measured by a loss function $\ell_{AI}(y, h(x))$ for the classifier prediction $h(x)$, and a loss function $\ell_H(y, m)$ for human's prediction $m$. The question that we tackle in this paper is the following: *What is an optimal classifier and otherwise an optimal way of deferring the decision to the human when there are constraints that limit the decision-making?* The constraints above can be algorithmic fairness constraints (e.g., demographic parity, equality of opportunity, equalized odds), expert intervention constraints (e.g., when the human expert can classify up to $b$ proportion of the data), or spatial constraints to enforce deferral on certain inputs, or any combination thereof.

Let us put the above question in a formal optimization form. To that end, let $r(x) \in$

$\{0,1\}$ be the rejection function[1], i.e., when $r(x) = 0$ the classifier makes the decision for input $x$ and otherwise $x$ is deferred to the expert. We obtain the deferral loss on $x$ and given a label $y$ and the expert decision $m$ as

$$\ell_{\text{def}}(y, m, h(x), r(x)) = r(x)\ell_H(y, m) + (1 - r(x))\ell_{AI}(y, h(x)).$$

Therefore, we can find the average deferral loss on distribution $\mu$ as

$$L^{\mu}_{\text{def}}(h, r) := \mathbb{E}_{X,Y,M\sim\mu}\left[\ell_{\text{def}}(Y, M, h(X), r(X))\right]. \tag{4.1}$$

We aim to find a randomized algorithm $\mathcal{A}$ that defines a probability distribution $\mu_{\mathcal{A}}$ on $\mathcal{H} \times \mathcal{R}$ that solves the optimization problem

$$\begin{aligned} \mu_{\mathcal{A}} \in \ & \underset{\mu_{\mathcal{A}}}{\operatorname{argmin}} \mathbb{E}_{(h,r)\sim\mathcal{A}}\left[L^{\mu}_{\text{def}}(h, r)\right], \\ s.t. \quad & \mathbb{E}_{X,Y,M\sim\mu}\mathbb{E}_{(h,r)\sim\mu_{\mathcal{A}}}\left[\Psi_i\big(X, Y, M, h(X), r(X)\big)\right] \leq \delta_i \end{aligned} \tag{4.2}$$

where $\Psi_i$ is a performance measure that induces the desired constraint in our optimization problem. We assume that $\Psi_i$, similar to $\ell_{\text{def}}$, is an *outcome-dependent* function, i.e., if the deferral occurs, the outcome of the classifier does not change $\Psi_i$, and otherwise, if deferral does not occur, the human decision does not change $\Psi_i$. In other words, the value of the constraints can only be a function of input feature $x$ and of the deferral system prediction $\hat{Y} = r(x)M + \big(1 - r(x)\big)h(x)$. Here, $\hat{Y}$ is the expert decision when deferral occurs, and is the classifier decision otherwise.

**Types of constraints.** Before we discuss our methodology to solve ((4.2)), it is beneficial to review the types of constraints with which we are concerned: **(1) expert intervention budget** that can be written in form of $\Pr\big(r(X) = 1\big) \leq \delta$, limits the rejection function to defer up to $\delta$ proportion of the instance, **(2) demographic parity** that is formulated as $\big|P(\hat{Y} = 1|A = 0) - P(\hat{Y} = 1|A = 0)\big| \leq \delta$, ensures that the proportion of positive predictions for the first demographic group ($A = 0$) is comparable to that for the second demographic group ($A = 1$). **(3) equality of opportunity** that is defined as $|Pr(\hat{Y} = 1|A = 1, Y = 1) - Pr(\hat{Y}|A = 0, Y = 1)| \leq \delta$ limits the differences between correct positive predictions among two demographic groups, **(4) equalized odds** that is similar to equality of opportunity but targets the differences of correct positive and negative predictions among two groups, i.e., $\max_{y=0,1}\big|Pr(\hat{Y} = 1|A = 1, Y = y) - Pr(\hat{Y} = 1|A = 0, Y = y)\big| \leq \delta$, **(5) out-of-distribution (OOD) detection** that is written as $\Pr_{\text{out}}(r(X) = 0) \leq \delta$ limits the prediction of the classifier on points that are outside its training distribution and incentivizes deferral in such cases, **(6) long-tail classification** deals with high class imbalances. This method aims to minimize a balanced error of classifier prediction on instances where deferral does not occur. Achiev-

[1]The rejection here differs from hypothesis rejection and indicates that the classifier rejects making a decision and defers the decision to the human expert.

Table 4.1: A list of embedding functions corresponding to the constraints that are discussed in Section 4.3. This list is a version of the results in Appendix C.4 when we assume that the input feature contains demographic group identifier $A$. To simplify the notations, we define $t(A,y) := \frac{\mathbb{I}_{A=1}}{Pr(Y=y,A=1)} - \frac{\mathbb{I}_{A=0}}{Pr(Y=y,A=0)}$.

| Name | Embedding Function $\psi_i(x)$ |
|---|---|
| Accuracy | $[\Pr(Y=0\|x),\ldots,\Pr(Y=n\|x),\Pr(Y=M\|x)]$ |
| Expert Intervention Budget | $[0,\ldots,0,1]$ |
| OOD Detection | $[0,\ldots,0,\frac{f_X^{\text{out}}(x)}{f_X^{\text{in}}(x)}]$ |
| Long-Tail Classification | $-\left[\sum_{i=1}^K \frac{\Pr(Y\neq 1,Y\in G_i\|X=x)}{\alpha_i \Pr(Y\in G_i)},\ldots,\sum_{i=1}^K \frac{\Pr(Y\neq l,Y\in G_i\|X=x)}{\alpha_i \Pr(Y\in G_i)},0\right]$ and $\frac{\Pr(Y\in G_i\|X=x)}{\Pr(Y\in G_i)}\left[1,\ldots,1,0\right]-\frac{\alpha_i}{K}$ |
| Bound on Type-$K$ Error | $\frac{\Pr(Y=k\|x)}{\Pr(Y=k)}[1,\ldots,\underbrace{0}_{k\text{-th}},\ldots,1,\Pr(M\neq k\|Y=k,x)]$ |
| Demographic Parity | $(\frac{\mathbb{I}_{A=1}}{Pr(A=1)} - \frac{\mathbb{I}_{A=0}}{Pr(A=0)})[0,1,\Pr(M=1\|x)]$ |
| Equality of Opportunity | $t(A,1)[0,\Pr(Y=1\|x),\Pr(M=1,Y=1\|x)]$ |
| Equalized Odds | $t(A,1)[0,\Pr(Y=1\|x),\Pr(M=1,Y=1\|x)]$ and $t(A,0)[\Pr(Y=0\|x),0,\Pr(M=0,Y=0\|x)]$ |

ing this objective as mentioned in Narasimhan *et al.* (2023) is equivalent to minimizing $\sum_{i=1}^{K} \frac{1}{\alpha_i} \Pr(Y \neq h(X), r(X) = 0 | Y \in G_i)$ when the feasible set is $\Pr(r(X) = 0, Y \in G_i) = \frac{\alpha_i}{K}$, and where $\{G_i\}_{i=1}^{K}$ is a partition of classes, and finally **(7) type-$k$ error bounds** that is a generalization of Type-I and Type-II errors, limits errors of a specific class $k$ using $\Pr(\hat{Y} \neq k | Y = k) \leq \delta$.

All above constraints are expected values of outcome-dependent functions (see Appendix C.4 for proof). To put it informally, if we change the classifier outcome after the rejection, such constraints do not vary.

**Linear Programming Equivalent to** ((4.2))**.** The outcome-dependence property helps us to show that (see Appendix C.3) obtaining the optimal classifier and rejection function is equivalent to obtaining the solution of

$$f^* = [f_1^*, \ldots, f_d^*] \in \underset{f \in \Delta_d^{\mathcal{X}}}{\operatorname{argmax}} \mathbb{E}\big[\langle f(X), \psi_0(X) \rangle\big], \quad \text{s.t. } \mathbb{E}\big[\langle f(x), \psi_i(x) \rangle\big] \leq \delta_i, i \in [1:m] \tag{4.3}$$

where $\Delta_d$ is a simplex of $d$ dimensions, $d = L + 1$, and $\psi_i : \mathcal{X} \to \mathbb{R}^d$ is defined as

$$\psi_i(x) := \mathbb{E}_{Y,M|X=x}\Big[\Big[\Psi_i(x,Y,M,1,0), \ldots, \Psi_i(x,Y,M,l,0), \Psi_i(x,Y,M,0,1)\Big]\Big] \tag{4.4}$$

that we name the *embedding function*[2] corresponding to the performance measure $\Psi_i$ for $i \in [0:m]$, where for simplifying the notation we define $\Psi_0 \equiv -\ell_{\text{def}}$. Furthermore, the optimal algorithm is obtained by predicting $h(x) = i$ with normalized probability of $f_i^*(x) / \sum_{j=1}^{d-1} f_j^*(x)$, where $\sum_{j=1}^{d-1} f_j^*(x) \neq 0$, and rejecting $r(x) = 1$ with probability $f_d^*(x)$. In case of $\sum_{j=1}^{d-1} f_j^*(x) = 0$ the classifier is defined arbitrarily. A list of embedding functions for the mentioned constraints and objectives is provided in Table 4.1 (See Appendix C.4 for derivations).

**Hardness.** We first derive the following negative result for the optimal deterministic predictor in ((4.3)). We use the similarity between ((4.3)) and $0-1$ Knapsack problem (see (Papadimitriou and Steiglitz, 1998, pp. 374)) to show that there are cases in which solving the former is equivalent to solving an NP-Hard problem. More particularly, if we assume that the distribution of $X$ contains finite atoms $x_1, \ldots, x_n$, each of which have probability of $\Pr(X = x_i) = p_i$, and if we set $\psi_1(x_i) = [0, \frac{w_i}{p_i}]$ and $\psi_0(x_i) = [0, \frac{v_i}{p_i}]$ for $v_i, w_i \in \mathbb{R}^+$, then ((4.3)) reduces in $\operatorname{argmax} \sum_i f^1(x_i) v_i$ subjected to $f^1 : \mathcal{X} \to \{0,1\}$ and $\sum_i f^1(x_i) w_i \leq \delta_1$, which is the main form of the Knapsack problem. In the following theorem, we show that a similar result can be obtained if we choose $\psi_0$ and $\psi_1$ to be embedding functions corresponding to accuracy and expert intervention budget. All proofs of theorems can be found in the appendix.

[2] We named this an embedding function because it embeds the constraint or loss of the optimization problem into a vector function.

**Theorem 5** (NP-Hardness of ((4.2)))**.** *Let the human expert and the classifier induce* $0-1$ *losses and assume* $\mathcal{X}$ *to be finite. Finding an optimal deterministic classifier and rejection function for a bounded expert intervention budget is an NP-Hard problem.*

Note that the above finding is different from the complexity results for deferral problems in (Mozannar *et al.*, 2023b, Theorem 1) and (De *et al.*, 2020, Theorem 1). NP-hardness results in these settings are consequences of restricting the search to a specific space of models, i.e., the intersection of half-spaces and linear models on a subset of the data. However, in our theorem, the hardness arises due to a possibly complex data distribution and not because of the complex model space.

The above hardness theorem for deterministic predictors justifies our choice of using randomized algorithms to solve multi-objective L2D. In the next section, by finding a closed-form solution for the randomized algorithm, we show that such relaxation indeed simplifies the problem.

## 4.4 $d$-dimensional Generalization of Neyman-Pearson Lemma

The idea behind minimizing an expected error while keeping another expected error bounded is naturally related to the problem that is designed by Neyman and Pearson Neyman and Pearson (1933). They consider two hypotheses $H_0, H_1$ as two distributions with density functions $g_0(x)$ and $g_1(x)$ for which a given point $x$ can be drawn. Then, they maximize the probability of correctly rejecting $H_0$, while bounding the probability of incorrectly rejecting $H_0$, i.e., for a test $T(x) \in [0,1]$ that rejects the null hypothesis when $T(x) = 1$, they solved the problem

$$\max_{T \in [0,1]^{\mathcal{X}}} \mathbb{E}_{X \sim g_1}\big[T(X)\big], \quad s.t.\ \ \mathbb{E}_{X \sim g_0}\big[T(X)\big] \leq \alpha. \tag{4.5}$$

They concluded that thresholding the likelihood ratio is a solution to the above problem. Formally, they show that all optimal hypothesis tests take the value $T(x) = 1$ when $g_1(x)/g_0(x) > k$ and take the value $T(x) = 0$ when $g_1(x)/g_0(x) < k$, where $k$ is a scalar and dependent on $\alpha$.

**Multi-hypothesis testing with rewards.** In this section, we aim to solve ((4.3)) as a generalization of Neyman-Pearson lemma for binary testing to the case of multi-hypothesis testing, in which correctly and incorrectly rejecting each hypothesis has a certain reward and loss. To clarify how the extension of this setting and the problem ((4.3)) are equivalent, assume the general case of $d$ hypotheses $H_0, \ldots, H_{d-1}$, each of which corresponding to $X$ being drawn from the density function $g_i(x)$ for $i \in \{0, \ldots, d-1\}$. Further, assume that for each hypothesis $H_i$, in case of true positive, we receive the reward $r_i(x)$, and in case of false negative, we receive the loss $\ell_i(x)$. Assume that we aim

to find a test $f : \mathcal{X} \to \Delta_d$ that for each input $x \in \mathcal{X}$ rejects $d-1$ hypotheses, each hypothesis $H_i$ with probability $1 - f^i(x)$ and maximizes a sum of true positive rewards, and that keeps the sum of false negative losses under control. Then, this is equivalent to $\operatorname{argmax}_{f \in \Delta_d^{\mathcal{X}}} \sum_{i=0}^{d-1} \mathbb{E}_{X \sim g_i}\Big[f^i(x) r_i(x)\Big]$ subjected to $\sum_{i=0}^{d-1} \mathbb{E}_{X \sim g_i}\Big[(1 - f^i(x))\ell_i(x)\Big] \leq \delta_1$ which in turns is equivalent to

$$\operatorname*{argmax}_{f \in \Delta_d^{\mathcal{X}}} \mathbb{E}_{X \sim g_0}\Big[\sum_{i=0}^{d-1} f^i(x) r_i(x) \frac{g_i(x)}{g_0(x)}\Big] \qquad \text{s.t. } \mathbb{E}_{X \sim g_0}\Big[\sum_{i=0}^{d-1} f^i(x) \sum_{j \neq i} \ell_j(x) \frac{g_j(x)}{g_0(x)}\Big] \leq \delta_1. \tag{4.6}$$

This problem can be seen as instance of ((4.3)), when we set

$$\psi_0(x) = [r_0(x), \ldots, r_{d-1}(x) \frac{g_{d-1}(x)}{g_0(x)}],$$

and

$$\psi_1(x) = \big[\sum_{j \neq 0} \ell_j(x) \tfrac{g_j(x)}{g_0(x)}, \ldots, \sum_{j \neq d-1} \ell_j(x) \tfrac{g_j(x)}{g_0(x)}\big].$$

Similarly, we can show that for all $\psi_0(x), \psi_1(x)$ in ((4.3)) there exists a set of densities $g_1(x), \ldots, g_{d-1}(x)$ and rewards and losses such that ((4.6)) and ((4.3)) are equivalent. This can be done by setting $g_i \equiv g_0$ and noting that the mapping from $\ell_i$s and $r_i$s into $\psi_0$ and $\psi_2$ is invertible.

The formulation of ((4.3)) can be seen as an extension of the setting in Tian and Feng (2021) when we move beyond type-$k$ error bounds to a general set of constraints. That work achieves the optimal test by applying strong duality on the Lagrangian form of the constrained optimization problem. However, we avoided using this approach in proving our solution, since finding $f^*$, and not the optimal objective, is possible via strong duality only when we know apriori that the Lagrangian has a single saddle point (for more details and fallacy of such approach, see Section C.5). As another improvement to the duality method, we not only find a solution to ((4.3)), but also show that there is no other solution that works as well as ours.

Before we express our solution in the following theorem, we define an import notation as an extension of the argmax function that helps us articulate the optimal predictor. In fact, we define

$$\mathcal{T}_d = \big\{\tau : \mathbb{R}^d \times \mathbb{R}^d \to \Delta_d \,\big|\, \sum_{i : x_i = \max\{x_1, \ldots, x_d\}} \big(\tau(\mathbf{x}_1^d, \cdot)\big)(i) = 1\big\} \tag{4.7}$$

that is a set of functions that result in one-hot encoded argmax when there is a clear maximum, and otherwise, based on its second argument, results in a probability distribution on all components that achieved the maximum value.

**Theorem 6** ($d$-GNP). *For a set of functions $\psi_i$ where $i \in [0,m]$, assume that $(\delta_1,\dots,\delta_m)$ is an interior point[3] of the set $\mathcal{F} = \left\{ \left(\mathbb{E}[\langle r(x), \psi_1(x)\rangle],\dots,\mathbb{E}[\langle r(x), \psi_m(x)\rangle]\right) : f \in \Delta_d^{\mathcal{X}} \right\}$. Then, there is a set of fixed values $k_1,\dots,k_m$ and $\tau \in \mathcal{T}_d$ such that the predictor*

$$f^*(x) = \tau\big(\psi_0(x) - \sum_{i=1}^{m} k_i \psi_i(x), x\big), \tag{4.8}$$

*obtains the optimal solution of $\sup_{f\in\Delta_d^{\mathcal{X}}} \mathbb{E}\big[\langle f(x), \psi_0(x)\rangle\big]$, subjected to the constraints being achieved tightly, i.e., when for $i \in [1:m]$ we have $\mathbb{E}\big[\langle f(x), \psi_i(x)\rangle\big] = \delta_i$. If $k_1,\dots,k_m$ are further non-negative, then $f^*(x)$ is the optimal solution to* ((4.3)). *Moreover, all optimal solutions of* ((4.3)) *that tightly achieve the constraints are in form of* ((4.8)) *almost everywhere on $\mathcal{X}$.*

**Example 1** (L2D with Demographic Parity). *In the setting that we have a deferral system and we aim for controlling demographic disparity under the tolerance $\delta$, we can set $\psi_0(x) = \big[Pr(Y=0|x), Pr(Y=1|x), Pr(Y=M|x)\big]$ and $\psi_1(x) = s(A)\big[0,1,Pr(M=1|x)\big]$, using Table 4.1, where $s(A) := \big(\frac{\mathbb{I}_{A=1}}{Pr(A=1)} - \frac{\mathbb{I}_{A=0}}{Pr(A=0)}\big)$. Therefore, d-GNP, together with the discussion after* ((4.4)) *shows that the optimal classifier and rejection function are obtained as*

$$h(x) = \begin{cases} 1 & Pr(Y=1|x) > \frac{1+ks(A)}{2} \\ 0 & Pr(Y=1|x) < \frac{1+ks(A)}{2} \end{cases},$$

*and*

$$r(x) = \begin{cases} 1 & Pr(Y=M|x) - ks(A)Pr(M=1|x) > \lambda(A,x) \\ 0 & Pr(Y=M|x) - ks(A)Pr(M=1|x) < \lambda(A,x) \end{cases},$$

*for a fixed value $k \in \mathbb{R}$, and where $\lambda(A,x) := \max\{Pr(Y=0|x), Pr(Y=1|x) - ks(A)\}$. The above identities imply that the optimal fair classifier for the deferral system thresholds the scores for different demographic groups using two thresholds $ks(0)$ and $ks(1)$. This is similar in form to the optimal fair classifier in vanilla classification problem Chen* et al. *(2023b); Cruz and Hardt (2023). However, the rejection function does not merely threshold the scores for different groups, but adds an input-dependent threshold $ks(A)Pr(M=1|x)$ to the unconstrained deferral system scores.*

*It is important to note that although we have a thresholding rule for the classifier, the thresholds are not necessarily the same as of isolated classifier under fairness criteria. Furthermore, the deferral rule is dependent on the thresholds that we use for the classifier. Therefore, we cannot train the classifier for a certain demographic parity and a rejection function in two independent stages. This further affirms the lack of compo-*

[3] A point is an interior point of a set, if the set contains an open neighborhood of that point.

*sitionality of algorithmic fairness that we discussed earlier in the introduction of this paper.*

**Example 2** (L2D with Equality of Opportunity)**.** *Here, similar to the previous example, we can obtain the embedding function for accuracy and equality of opportunity constraint as* $\psi_0(x) = \big[Pr(Y=0|x), Pr(Y=1|x), Pr(Y=M|x)\big]$ *and* $\psi_1(x) = t(A,1)\big[0, Pr(Y=1|x), Pr(M=1,Y=1|x)\big]$*, respectively. Therefore, the characterization of optimal classifier and rejection function using d-GNP results in*

$$h(x) = \begin{cases} 1 & \big(2-kt(A,1)\big)Pr(Y=1|x) > 1 \\ 0 & \big(2-kt(A,1)\big)Pr(Y=1|x) < 1 \end{cases},$$

*and*

$$r(x) = \begin{cases} 1 & Pr(Y=M|x)\big(1-kt(A,1)Pr(M=1|Y=M,x)\big) > \vartheta(A,x) \\ 0 & Pr(Y=M|x)\big(1-kt(A,1)Pr(M=1|Y=M,x)\big) < \vartheta(A,x) \end{cases},$$

*for* $k \in \mathbb{R}$ *and where* $\vartheta(A,x) := \max\{Pr(Y=0|x), \big(1-kt(A,1)\big)Pr(Y=1|x)\}$*. Assuming* $2-kt(A,1)$ *takes positive values for all choices of A, we conclude that the optimal classifier is to threshold positive scores differently for different demographic groups. However, the optimal deferral is a function of probability of positive prediction by human expert.*

**Example 3** (Algorithmic Fairness for Multiclass Classification)**.** *In addition to addressing the L2D problem, the formulation of d-GNP in Theorem 6 allows for finding the optimal solution in vanilla classification. In fact, for an L-class classifier, if we aim to set constraints on demographic parity* $\big|Pr(\hat{Y}=0|A=0) - Pr(\hat{Y}=0|A=1)\big| \leq \delta$ *or equality of opportunity* $\big|Pr(\hat{Y}=0|Y=0,A=0) - Pr(\hat{Y}=0|Y=0,A=1)\big| \leq \delta$ *on Class* 0*, then we can follow similar steps as in Appendix C.4 to find the embedding functions as*

$$\psi_{\text{DP}} = s(A)\big[1,0,\ldots,0\big],$$

*and*

$$\psi_{\text{EO}} = t(A,0)\big[Pr(Y=0|x),0,\ldots,0\big].$$

*As a result, since the accuracy embedding function is* $\psi_0(x) = \big[Pr(Y=0|x),\ldots,Pr(Y=L|x)\big]$*, then, by neglecting the effect of randomness, the optimal classifier under such constraints are as*

$$h_{\text{DP}}(x) = \ argmax\big\{Pr(Y=0|x) - ks(A), Pr(Y=1|x),\ldots,Pr(Y=L|x)\big\}, \tag{4.9}$$

*and*

$$h_{\text{EO}}(x) = \ argmax\big\{Pr(Y=0|x)\big(1-kt(A,0)\big), Pr(Y=1|x),\ldots,Pr(Y=L|x)\big\}. \tag{4.10}$$

*Equivalently, for demographic parity, the optimal classifier includes a shift on the score of Class* 0 *as a function of demographic group, and for equality of opportunity, the optimal classifier includes a multiplication of the score of Class* 0 *with a value that is a function of demographic group. It is easy to show that under condition of positivity of the multiplied value, these classifiers both reduce to thresholding rules in binary setting.*

Note that although Theorem 6 characterizes the optimal solution of ((4.3)), it leaves us uninformed regarding parameters $k_1, \dots, k_m$, and further does not give us the form of the optimal solution when $\psi_0(x) - \sum_{i=1}^{m} k_i \psi_i(x)$ has more than one maximizer. In the following theorem, we address these issues for the case that we have a single constraint.

**Theorem 7** ($d$-GNP with a single constraint)**.** *The optimal solution* ((4.8)) *of the optimization problem* ((4.3)) *with one constraint is equal to* $f^*_{k,p}(x) = \tau\big(\psi_0(x) - k\psi_1(x), x\big)$ *where* $\tau$ *is a member of* $\mathcal{T}_d$ *such that if there is a non-singleton set* $\mathcal{I}$ *of maximizers of a vector* $\mathbf{y} \in \mathbb{R}^d$*, then we have* $\big(\tau(\mathbf{y},x)\big)(i) = p$ *and* $\big(\tau(\mathbf{y},x)\big)(j) = 1-p$*, where i and j are the first indices in* $\mathcal{I}$ *that minimizes* $\psi_1(x)$*, and maximizes* $\psi_0(x)$*, respectively.*

*In this case, k is a member of the set* $\mathcal{K} = \Big\{t : \delta \in \big[\lim_{\tau \uparrow t} C(\tau), C(t)\big]\Big\}$ *where* $C(t) = \mathbb{E}\big[\langle f^*_{t,0}(x), \psi_0(x)\rangle\big]$ *is the expected constraint of the predictor* $f^*_{t,0}$*.*

*Moreover,* $p = \frac{C(k)-c}{C(k)-\lim_{\tau \uparrow t^-} C(\tau)}$*, if* $C(\cdot)$ *is lower-discontinuous at k, and otherwise* $p = 0$*.*

This theorem reduces the complexity of finding $k_i$s from the complexity of an exhaustive search to the complexity of finding the root of the monotone function $C(t) - \delta$ (see Lemma 16 for the proof of monotonicity), and further finds the randomized response for the cases that Theorem 6 leaves undetermined.

Before we proceed to the designed algorithm based on $d$-GNP, we should address two issues. Firstly, during the course of optimization, it can occur that the solution of Theorem 6 does not compute non-negative values $k_i$ for an $i \in [1:m]$. This means that the constraints are not achieved tightly in the final solution of ((4.3)). Therefore, we are able to achieve the optimal solution with the constraint $\delta'_i < \delta_i$. Now, if we can assure that the constraint tuples are still inner points of $\mathcal{F}$ when we substitute $\delta_i$ by $\delta'_i$, then Theorem 6 shows that ((4.8)) is still an optimal solution to ((4.3)).

Secondly, for tackling various objectives that are defined in Section 4.3, we usually need to upper- and lower-bound a performance measure by $\delta$ and $-\delta$. However, since both bounds cannot hold tightly and simultaneously unless the tolerance is $\delta = 0$, then we can use only one of the constraints in turn and apply the result of Theorem 7 and check whether the constraint is active in the final solution.

In the next section, we design an algorithm based on these results and show its generalization to the unseen data.

## 4.5 Empirical $d$-GNP and its Statistical Generalization

In previous sections, we obtained the optimal solution to the constrained optimization problem ((4.3)) using $d$-GNP. In this section, we propose a plug-in method in Algorithm 2 and tackle the generalization error of the objective and constraints based on this solution. The results in this section, which are extensions of the generalization results for Neyman-Pearson Audibert and Tsybakov (2007); Tong (2013), address single constraint setting, and the extension to the multiple constraint case is left for future research.

We start this section by the following theorem that shows if the solution to our plug-in method meets constraints of the optimization problem on training data, this generalizes to the unseen data.

**Theorem 8** (Generalization of the Constraints)**.** *For the approximation of the Neyman-Pearson solution $\hat{f}_{\hat{k},\hat{p}}(x)$ in Algorithm 2 such that $\mathbb{E}_{S^n}\left[\langle \hat{f}_{\hat{k},\hat{p}}(x), \hat{\psi}_1(x)\rangle\right] \leq \delta$, if we assume that embedding functions are bounded, then for $d_n(\varepsilon) \simeq O(\frac{\sqrt{\log n}+\sqrt{\log 1/\varepsilon}}{\sqrt{n}})$ and $S^n \sim \mu$ we have $\mathbb{E}_{\mu}\left[\langle \hat{f}_{\hat{k},\hat{p}}(x), \psi_1(x)\rangle\right] \leq \delta + d_n(\varepsilon)$ with probability at least $1-\varepsilon$.*

In the above theorem, we show that the optimal empirical solution for the constraint, probably and approximately satisfies the constraint on true distribution. Therefore, if we assume that we have an approximation $\hat{\psi}_1(x)$ in hand where $\|\hat{\psi}_1(x) - \psi_1(x)\|_\infty \leq \varepsilon'$ with high probability, this theorem together with Hölder's inequality shows that we need to assure $\mathbb{E}_{S^n}\left[\langle \hat{f}_{\hat{k},\hat{p}}(x), \hat{\psi}_1(x)\rangle\right] \leq \delta - d_n - \varepsilon'$ to achieve the corresponding generalization with high probability.

Next, we ask whether the objectives of the empirical optimal solution and the true optimal solution are close. We answer to this question positively in the following theorem. First, however, let us define the notions of ($\gamma$, $\Delta$)-sensitivity condition as the following. This is an extension to detection condition in Tong (2013) and assumes that changing the parameter in predictor leads to a detectable change in constraints.

**Definition 1.** *For an embedding function $\psi_1$, and a distribution $\mu_X$ on $\mathcal{X}$, we refer to a function $r_k(x)$ as a prediction with ($\gamma$, $\Delta$)-sensitivity around k, if there exists $C \in \mathbb{R}^+$ such that for all $\delta \in (0, \Delta]$ we have*

$$\left|\mathbb{E}_{\mu_X}\left[\langle r_k(x) - r_{k+\delta}(x), \psi_1(x)\rangle\right]\right| \geq C\delta^\gamma. \tag{4.11}$$

Now, we express the following generalization theorem for predictors that address the above conditions:

**Theorem 9** (Generalization of Objective)**.** *Assume that $(\delta - \varepsilon_l, \delta + \varepsilon_u)$ is a subset of of all achievable constraints $\mathbb{E}\left[\langle f(x), \psi_1(x)\rangle\right]$, and that $\|\psi_i(x)\|_\infty \leq 1$ for $i = 1, 2$. Further, let the size n of validation data be large enough such that $d_n(\delta/3) \leq \frac{\varepsilon_l}{2}$.*

*Now, if the optimal predictor* $f^*_{k,0}(x)$ *is* ($\gamma$, $\Delta$)*-sensitive around optimal* $k^*$ *for* $\Delta \geq \frac{\left(2\max\{d_n(\delta/3),\delta_1\}+\sqrt{2\gamma C(\delta_0+K\delta_1)}\right)^{1/\gamma}}{C}$ *and* $\gamma \leq 1$*, then for* $n \geq \frac{16}{\varepsilon_l^2}\log\frac{3}{\delta}$*, and with probability at least* $1-\delta$*, the optimal empirical classifier, as of Algorithm 2 has an objective that is close to the true optimal objective as*

$$\begin{aligned}\mathbb{E}\big[\langle f^*_{k^*,p^*}(x),\psi_0(x)\rangle\big]-\mathbb{E}\big[\langle f^*_{\hat{k},\hat{p}}(x),\psi_0(x)\rangle\big] \leq & 2\Big(\frac{2\max\{d_n(\delta/3),\delta_0\}}{C}\Big)^{1/\gamma} \\ & +4\sqrt{\frac{2(\delta_0+K\delta_1)}{\gamma C}} \\ & +2(\delta_0+K\delta_1)+2Kd_n(\delta/3), \qquad (4.12)\end{aligned}$$

*where K is an upper-bound to the absolute value of* $k^*$*.*

Now that we have proven generalization of our post-processing method, we should briefly compare this to other possible algorithms to learn an approximation of the optimal classifier and rejection function pair. A possible method is to find the appropriate 'defer' or 'no defer' value for each instance in the training dataset, and for a given set of constraints. Although these types of in-processing algorithms can perform computationally efficient (e.g., $O(n\log n)$ complexity for $\frac{1}{n}$-suboptimal solution for human intervention budget as shown in Theorem 16), they do not necessarily generalize to unseen data. In particular, we can show that for all algorithms that estimate *deferral labels* from empirical data, there exist two underlying distributions on the data on which the algorithm results in similar deferral labels, while the optimal rejection functions for these two distributions are not interchangeable. This argument is further formalized in the following proposition:

**Proposition 4** (Impossibility of generalization of deferral labels)**.** *For every deterministic deferral rule* $\hat{r}$ *for empirical distributions and based on the two losses* $\mathbb{1}_{m\neq y}$ *and* $\mathbb{1}_{h(x)\neq y}$*, there exist two probability measures* $\mu_1$ *and* $\mu_2$ *on* $\mathcal{X}\times\mathcal{Y}\times\mathcal{M}$ *such that the corresponding* $(\hat{r},X)$ *for both measures is distributed equally. However, the optimal deferral* $r^*_{\mu_1}$ *and* $r^*_{\mu_2}$ *for these measures are not interchangeable, that is* $L^{\mu_i}_{\text{def}}(h,r^*_{\mu_i}) \leq \frac{1}{3}$ *while* $L^{\mu_i}_{\text{def}}(h,r^*_{\mu_j}) = \frac{2}{3}$ *for* $i=1,2$ *and* $j\neq i$*.*

In a nutshell, this proposition implies that, every algorithm that reduces the two-bit data of human accuracy and AI accuracy for an input into a single-bit data of 'defer' or 'no defer' looses the information that is important for obtaining the optimal rejection function that generalizes to the unseen data. This is a drawback of in-processing algorithms that are used in multi-objective L2D problems. We refer the reader to Appendix C.12 for more details and proof of aforementioned proposition.

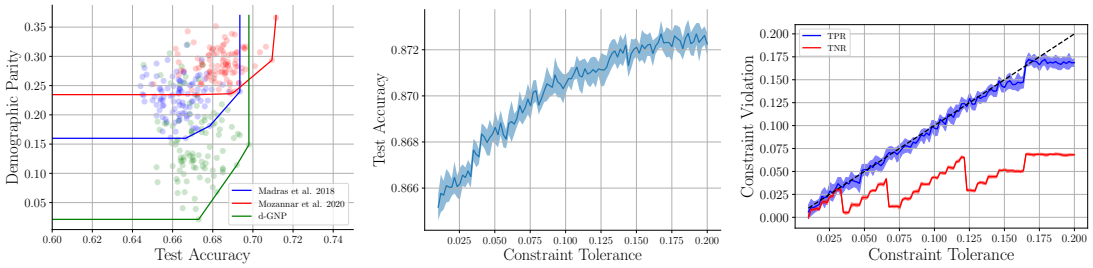



Figure 4.2: Performance of $d$-GNP on COMPAS dataset (left), and ACSIncome (center and right)

## 4.6 Experiments

We implemented [4] Algorithm 2, first for COMPAS dataset Dressel and Farid (2018) in which the recidivism rate of 7214 criminal defendants is predicted. The human assessment is done in this dataset on 1000 cases by giving humans a description of the case and asking them whether the defendant would recidivate within two years of their most recent crime.[5] The demographic parity is assessed for two racial groups of white and non-white defendants. Figure 4.2 shows the average performance of $d$-GNP over 10 random seeds compared to two baselines: (1) Madras et al. Madras *et al.* (2018) in which a demographic parity regularizer is added to the surrogate loss, and over a variation of 100 regularizer coefficient, and (2) Mozannar et al. Mozannar and Sontag (2020a) in which after training the classifier and rejector pair, we shift the corresponding scores to find a new thresholding rule. All scores, classifiers, and rejection functions are trained on a 1-layer feed-forward neural network. The figure shows that achieving better fairness criteria is possible using $d$-GNP, while this might not lead to better accuracy when the constraint violation is not of interest.

We further tested our method on `folktables` dataset Ding *et al.* (2021) that contains an income prediction task based on 1.6M rows of American Community Survey data. Since we had no access to human expert data, we simulated a human expert that has different accuracy on two racial groups of white and non-white individuals (85% and 60%, respectively). We considered the L2D problem with bounded equalized odds violation. Figure 4.2 shows our method's accuracy and constraint violation, coupled with a confidence bound that is obtained using ten iterations of bootstrapping. This figure shows that violation bounds are accurately met for the test data, and the performance increases when these bounds are loosened.

[4] The code is available in `https://github.com/AminChrs/PostProcess/`.

[5] This is as opposed to the experiment in Madras *et al.* (2018) where the human decision is simulated.

## 4.7 Conclusion

The $d$-GNP is a general framework that obtains the optimal solution to various constrained learning problems, including but not limited to multi-objective L2D problems. Using this post-processing framework, we can first estimate the scores related to our problem and then find a linear rule of these scores by fine-tuning for specific violation tolerances. This method reduces the computational complexity of in-processing methods while guaranteeing achieving a near-optimal solution in a large data regime.

**Algorithm 2:** Finding Optimal Classifier and Rejection Function

**Require:** The formulation of $\ell_{\text{def}}(\cdot,\cdot,\cdot)$ and $\{\Psi_i(\cdot,\cdot,\cdot)\}_{i=1}^{m}$, and the datasets $\mathcal{D}_{\text{train}} = \{(x^i,a^i,m^i,y^i)\}_{i=1}^{n_{\text{train}}}$, $\mathcal{D}_{\text{val}} = \{(x^i,a^i,m^i,y^i)\}_{i=n_{\text{train}}+1}^{n_{\text{train}}+n_{\text{val}}}$, and tolerances $\{\delta_i\}_{i=1}^{m}$

**Ensure:** Optimal deferral rule $r^*(x)$ and classifier $h^*(x)$

**Parameters:** $\varepsilon = 1e-8$

**procedure** CONSTRAINEDDEFER($\ell_{\text{def}}$, $\{\Psi_i\}_{i=1}^{m}$, $\mathcal{D}_{\text{train}}$, $\mathcal{D}_{\text{val}}$)

Obtain closed-form of $\{\psi_i(x)\}_{i=0}^{m}$ using $\ell_{\text{def}}$ and $\Psi_i$s via ((4.4)) and in terms of the scores as in Table 4.1

Estimate the scores in Table 4.1 using classification/regression methods on $\mathcal{D}_{\text{train}}$

Find estimate $\{\hat{\psi}_i\}_{i=0}^{m}$ using estimated scores in previous step and closed-form of Step 3 **if** $m = 2$ **then**

Define routine $\hat{f}_{k,p}(x) := \tau\big(\hat{\psi}_0(x) - k\hat{\psi}_1(x), x\big)$ for $\tau$ in Theorem 7

Define routine $\hat{C}(t) := \widehat{\mathbb{E}}_{\mathcal{D}_{\text{val}}}\Big[\langle \hat{f}_{k,0}(x_i), \hat{\psi}_1(x_i)\rangle\Big]$

Find $\hat{k} = \min k$ over the feasibility set $\hat{C}(t) \le \delta_1$ **if** $\hat{k} = \emptyset$ **then**

**Return** 'Not Feasible' **else**

If $\hat{C}(\hat{k}-\varepsilon) - \hat{C}(k^*) \le 1e-3$

$\hat{p} \leftarrow 0$ **else**

$\hat{p} \leftarrow \frac{\delta - \hat{C}(\hat{k})}{\hat{C}(\hat{k}-\varepsilon) - \hat{C}(\hat{k})}$

END

END

$s(x) := \hat{f}_{\hat{k},\hat{p}}(x)$ **else**

Optimize ((4.3)) for $\mathcal{D}_{\text{val}}$ and for $f(x) = \tau(\hat{\psi}_0(x) - \sum_{i=1}^{m}\hat{\psi}_i(x), x)$ for $\tau$ as in Theorem 6 and via exhaustive search over $\{k_1,\ldots,k_m\}$ and randomizations of $\tau$ and find $s(x) := \hat{f}(x)$

END

$h^*(x) := \underset{i\in[0:L-1]}{\operatorname{argmax}}\, s_i(x)$

$r^*(x) := \underset{i\in\{0,1\}}{\operatorname{argmax}}\, \big[s_{h^*(x)}(x), s_L(x)\big]$

**Return** $h^*(x), r^*(x)$

**end procedure**

# Chapter 5

# Defer-and-Fusion: Optimal Predictors that Incorporate Human Decisions

*This chapter is based on the joint work with Amirmehdi Jafari Fesharaki. I thank his contribution on implementation of the proposed method.*

## 5.1 Introduction

The advancement of machine learning (ML) has led to a proliferation of its deployment in various decision-making tasks, particularly in the medical domain Beede *et al.* (2020); Raghu *et al.* (2019). In recent years, numerous ML-powered products have received FDA clearance and are being utilized in clinical practice. A central example is in radiology, where for chest X-ray detection, there is a range of ML enabled clinical assistants van Leeuwen *et al.* (2021). These models have shown superior accuracy compared to human experts in some instances Rajpurkar *et al.* (2018); Killock (2020), raising the question of determining the optimal use of human and ML expertise in decision-making scenarios, and identifying when to defer to human expertise and when to rely on ML predictions.

As a strategy for utilizing both human and ML expertise in decision-making tasks, *learn-to-defer* methods have emerged in recent years El-Yaniv *et al.* (2010); Madras

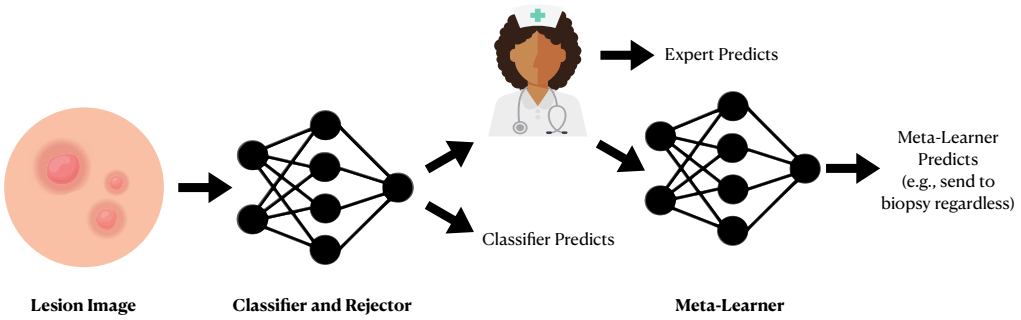



Figure 5.1: Description of a Defer-and-Fusion system. Given an instance, the rejector decides whether we should benefit from human feedback or not, and if the feedback is received should be adopted as the final prediction or taken as the input of a meta-learner.

*et al.* (2018). These methods design systems, commonly known as *deferral systems*, which distribute instances between human decision-makers and ML models, with the goal of maximizing accuracy by directing instances to the appropriate expertise. The underlying principle of learn-to-defer is that instances are delegated to human experts when they are deemed more likely to produce accurate outcomes.

In a deferral system, the final prediction on a given input is determined based on which of the human or ML has the lower prediction error. However, in sensitive domains such as medicine, it is recommended to use all available information for making a prediction. In such cases, the prediction should not be based solely on the feedback of either the human or the ML, but rather on a combination of both human and ML feedback Patel *et al.* (2019); Rajpurkar *et al.* (2020). For instance, in case of medical diagnosis, if either the human or the ML raises suspicion of the presence of a disease, it is prudent to subject the patient to additional testing, irrespective of the source of the prediction. Although combining human and ML decision is studied in previous works Kerrigan *et al.* (2021); Donahue *et al.* (2022), in these works it is mainly assumed that the human advice is always sought, which can increase the cost of human involvement.

In this paper, we close the gap between deferral systems and methods that seek human advice on all instances. We introduce *Defer-and-Fusion systems* (Figure 5.1) where on every instance, it decides whether the expert's feedback should be sought or not, and how to integrate the feedback into the final prediction, either adopting it as it is or fusing it with covariates using a meta-learner.

In this paper, we start by motivating the importance of using DaF systems instead of vanilla deferral systems. In fact, motivated by real-world applications, we give examples of imbalanced loss functions, where any deferral method, enforcing "either" ML or human to make the final prediction is strictly sub-optimal compared to DaF systems. Such utilization of imbalanced loss functions is prevalent in practical applications, particularly in the domain of medical diagnosis. In such scenarios, a false negative diagnosis, i.e., the misclassification of a patient as not having a disease when in fact they do, often incurs a larger amount of harm compared to the scenario in which a patient is falsely diagnosed as having the disease. This is due to the fact that false positive diagnoses (i.e., a patient without the disease being classified as having it) can be rectified through further investigations, while false negatives often lead to the premature discharge of patients, which can result in potentially hazardous outcomes.

Furthermore, we provide a lower-bound on Bayes probability of error for DaF systems and we show that such bound is smaller than the lower-bound on probability of error for deferral systems. Further, we discuss the tightness of this lower-bound and the cases under which we could conclude that even for a balanced $0-1$ loss, the deferral systems are strictly sub-optimal compared to DaF systems.

Next, we obtain the Bayes optimal classifier, rejector, and meta-learner for DaF systems by making use of the joint distribution on covariates, human decisions, and true labels. We further show the generalizability of our optimal solutions to scenarios where multiple experts are involved. Finally to show the superior performance of our proposed

Defer-and-Fusion system, we provide an extensive comparison of performance of our method compared to deferral systems suggested in the literature, on a variety of semi-synthetic and real-world datasets. We complement our empirical results by providing theoretical evidence of the scenarios where previously proposed methods, such as the one by Kerrigan *et al.* (2021), results in strictly suboptimal solutions compared to ours.

## 5.2 Related Works

Designing predictors that perceive whether they are reliable or not goes back to Chow's rule Chow (1970). This rule provides a classifier with a rejection option that sets off when a classifier's confidence is below a certain threshold.Cortes *et al.* (2016) posed a minimization problem based on rejection loss and showed that Chow's rule is Bayes optimal rejector for $0-1$ rejection loss. In that work, they assumed that the cost of rejection is only a function of covariates. However, they did not take the expert's error into account. Madras *et al.* (2018) rephrased rejection loss into a form in which the cost of rejection on each instance depends on the error of a human expert. This new loss is known as learn-to-defer loss in the literature.

To implement optimal solutions of learning with rejection, several works are introduced that use surrogate losses for binary Bartlett and Wegkamp (2008) and multi-class Ni *et al.* (2019) classification, and for learn-to-defer problem Mozannar and Sontag (2020b); Charusaie *et al.* (2022b); Verma and Nalisnick (2022b); Mozannar *et al.* (2023c). The problem of learn-to-defer is further extended to the case of multiple experts Verma *et al.* (2023) and the case of bounded human budget for classification and regression tasks Okati *et al.* (2021a); De *et al.* (2021, 2020).

The improvement observed in learn-to-defer methods revolves around the complementarity of human and machine that is formulated in Bansal *et al.* (2021a). In Charusaie *et al.* (2022b) the connection of such complementarity and hypothesis complexity of machine is studied. Although the complementarity in learn-to-defer is studied where human is either out of the loop or a final decision maker, a general combination of human and ML, as opposed to learn-to-defer Straitouri *et al.* (2021), is introduced in a consequential decision making setting Rastogi *et al.* (2022) and the possibility of complementarity for a linear combination is studied in Zhang and Bareinboim (2022); Donahue *et al.* (2022). Further, a Bayesian solution for such combination is derived in Steyvers *et al.* (2022) and Kerrigan *et al.* (2021). We generalize these results by providing an algorithm that certifiably provides complementarity, controls the human involvement, and fuses the human decision not only with ML prediction, but with the covariates.

The formulation of our paper, although novel, bears similarity with Wilder *et al.* (2020), in which instead of learn-to-defer loss, a loss that contains a general combination of human and ML is introduced. However, due to simplicity, they only attempted to minimize the loss by assuming the meta-learner being equal to the identity function. In contrast, we derive a general optimal system and a method to optimize the meta-learner.

Further, as opposed to what Wilder *et al.* (2020) claims on a similarity of DaF systems and standard deferral systems in terms of performance, we provide theoretical and practical evidence on the superiority of DaF systems. Finally, in our work, due to practical implications that we discuss in Appendix D.5, the option of adopting human as the final prediction is also preserved.

Furthermore, our work could be seen as an offline version of Active Feature Acquisition Shim *et al.* (2018); Natarajan *et al.* (2018). These systems train two classifiers given an incomplete and complete set of features, and in each iteration decide whether to predict based on an incomplete set of features or acquire more features and retrain their model. Although one could formulate our method similarly, our results show that since we obtain the optimal Bayes solution, the resulting performance in our cases remains higher.

Finally, although our work resembles ensemble learning methods in which the output of predictor models are combined to produce a prediction with higher accuracy Kittler *et al.* (1998); Dietterich (2000), the format of the predictors in our case (i.e., discrete for human expert), their learnability (i.e., the assumption of stationarity of human), and their cost of predictions (e.g., extra cost of human prediction) make this work different from existing literature on ensemble learning.

## 5.3 Problem Setting

A standard deferral system consists of a rejector $r(x)$ and a classifier $h(x)$. The rejector function decides for a datapoint $x$ in the feature space $\mathcal{X}$ whether to defer the decision to human, and the classifier predicts the target label $t \in \mathcal{Y}$ based on the features, if the decision is not deferred to human. After deferring to the human, then such systems adopt human decision $m \in \mathcal{Y}$ as a final prediction. In the analyses of this paper, we assume that features $X$, human decision $M$, and the true label $Y$ are random variables that are drawn from a joint stationary distribution $\mu_{X,Y,M}$. Therefore, the average loss of such methods can be obtained as

$$\begin{aligned} L_{\text{def}}(h,r) = \mathbb{E}_{\mu_{X,Y,M}}\big[&\mathbb{I}_{r(X)=0}\ell(h(X),Y) \\ &+ \mathbb{I}_{r(X)=1}\ell_{\text{def}}(M,Y)\big], \text{(Deferral loss)} \end{aligned} \tag{5.1}$$

where $\ell_{def}(m,y)$ and $\ell(h,y)$, respectively, represent the cost of human prediction $m$ and classification prediction $h$, where the true label is $y$.

The method that is proposed in this paper, however, is based on a loss function that gives the option to further fuse human and ML into a prediction. In fact, we extend rejector $r(x)$ to a ternary case, which as a third option, a meta-learner $g$ is responsible for such fusion. We refer to this method as a Defer-and-Fuse, from here on DaF, method,

which induces the loss function

$$\begin{aligned} &L_{\mathrm{DaF}}(h,r,g) \\ &\quad = \mathbb{E}_{\mu_{X,Y,M}}\big[\mathbb{I}_{r(X)=0}\ell(h(X),Y) + \mathbb{I}_{r(X)=1}\ell_{\mathrm{def}}(M,Y) \\ &\qquad + \mathbb{I}_{r(X)=2}\ell_{\mathrm{fus}}(g(M,X),Y)\big], \quad \text{(DaF loss)} \end{aligned} \tag{5.2}$$

where $\ell_{\mathrm{fus}}(g(x,m),y)$ represents the fusion cost induced by the meta-learner prediction $g(x,m)$ (see Figure 5.1) where the true label is $y$. In Appendix D.5, we will show that adoption of the human decision as final prediction is, theoretically and in terms of accuracy, inferior to the fusion option. However, the reason we have that option is to firstly reduce the sample complexity and not to train the meta-learner on all samples, and secondly to be able to control the autonomy of human in the end. Obviously, as long as the space of rejectors for the DaF method contains rejectors of learn-to-defer, the optimal average loss for the DaF method is smaller than that of the deferral system, i.e., if $\mathcal{R} \subseteq \mathcal{R}'$, we have

$$\min_{h,r,g \in \mathcal{H}\times\mathcal{R}'\times\mathcal{G}} L_{\mathrm{DaF}}(h,r,g) \leq \min_{h,r\in\mathcal{H}\times\mathcal{R}} L_{\mathrm{def}}(h,r). \tag{5.3}$$

Inequality (5.3) shows that the learn-to-defer method is sub-optimal compared to the DaF method. In the next section, we take this statement one step further and prove the strict sub-optimality of the DaF method compared to the standard deferral system in some settings in which the costs are imbalanced among labels.

## 5.4 Strict Sub-Optimality of Learn-to-Defer

The sub-optimality of learn-to-defer methods can be studied with respect to two measures, the loss function in case of imbalanced label costs, as well as the accuracy. In the former case, we show examples for which such sub-optimality occurs, and in the latter, we argue about the sub-optimality of the information that is used in such methods.

### 5.4.1 A Case of Cost-Sensitive Learning

The strict sub-optimality of learn-to-defer methods, as we will show in the following, is rooted in the fact that the use of meta-learner to reach optimality is inevitable. In fact, the following example shows that under mild assumptions on classifier and human prediction, there are cases in which some binary functions on the mentioned predictions obtain less expected loss than deferral to either of these predictors.

**Example 4.** *Assume that the predictions of human and the classifier are noisy versions of the true label $y \in \{0,1\}$, i.e., assume $h(\mathbf{X}) = Y \oplus n_1$, and $M = Y \oplus n_2$ where $n_1 \sim Bern(p_1)$, $n_2 \sim Bern(p_2)$, and $\oplus$ is the 'XOR' operation. Further, assume that the true*

*label is distributed as* $Y \sim Bern(1/2)$*, and* $n_1$*,* $n_2$*, and* $Y$ *being mutually independent. Then, consider the case that the cost of false positive prediction is* $c$*, while the cost of false negative prediction is* $1-c$*. We can show that the expected loss for 'AND' and 'OR' operations on these predictions, as well as the optimal deferral solution, is as Figure 5.2 (see Appendix D.1 for proof). As a result, the optimality of the deferral system under these conditions cannot occur unless the cost of false positive and false negative predictions are nearly equal. Therefore, for example, for small enough* $c$*, the optimal DaF loss is strictly upper-bounded as*

$$\begin{aligned} L_{DaF}(h, r^*, g^*) &\leq p_1 p_2 + c(1 - q_1 q_2 - p_1 p_2) \\ &< L_{def}(h, r^*) = \frac{\min\{p_1, p_2\}}{2}, \end{aligned} \tag{5.4}$$

*where* $q_i = 1 - p_i$ *for* $i = 1, 2$*, and the first inequality is followed by setting* $r(x) = 2$ *and* $g(x, m) = m \vee h(x)$*. By assuming that* $h$ *is the Bayes optimal classifier of* $Y$*, we show that the learn-to-defer method is strictly sub-optimal to the DaF method (to see empirical results on this example, see Appendix D.1).*

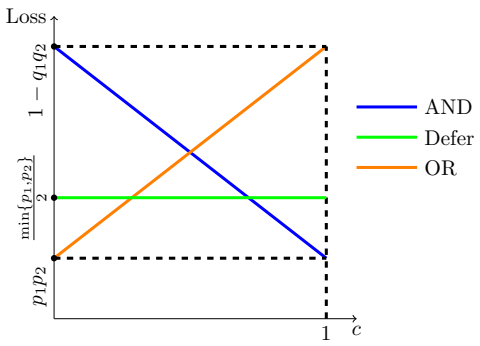



Figure 5.2: The loss of 'AND', and 'OR' operation on human and classifier prediction, as well as optimal learn-to-defer solution for the setting of Example 4 and for varying false positive cost $c$.

As discussed before, this example is central in the context of clinical diagnosis, since the cost-sensitive loss function can be applied as a means of mitigating the consequences of false negative diagnoses. As such, the cost associated with false negative errors is typically much higher than the cost associated with false positive errors. Given this, the optimal human-in-the-loop method in this scenario would involve categorizing a person for having a specific condition whenever either the classifier or the human expert makes such a prediction. This approach minimizes the risk of false negatives by ensuring that cases of such condition are not missed. In contrast, the deferral method, which takes the prediction with the smaller average loss as the final decision, increases the risk of false negative errors.

Although important in medical diagnosis, most scenarios in machine learning do not involve cost-sensitive settings. Therefore, it is important to show the superiority of our method in balanced cost settings too. In the following section, using a lower-bound on Bayes optimal error, we show that there are conditions under which learn-to-defer is strictly sub-optimal compared to the DaF method.

### 5.4.2 $0-1$ Loss and Fano's inequality

Converse bounds in the field of statistical estimation provide us with the accuracy of the optimal estimator of a parameter based on a set of observed variables. Such bounds derive the least achievable distance between the target and estimated parameter, based on the amount of information about the target parameter that is contained within the observed variables. As an instance, one could think of Cramér-Rao bound Rao (1992), and Fano's inequality Fano (1961), in which the distance is defined as the variance of error (in case of regression) and the probability of error (in case of classification), respectively. In particular, Fano's inequality bounds such error in terms of the conditional entropy of the estimated parameter given the observed variable $H(Y|X)$.

This choice, although as we discuss leads to an almost-tightness result, concludes a broader discussion on the use of such entropic notions. In fact, conditional entropy measures the uncertainty of the prediction of $Y$ given the knowledge of $X$ and plays a central role in the exact value of the optimal probability of error. There is an optimization principle for neural networks that has the conditional entropy as the objective Linsker (1987) and its optimization is known to roughly optimize the probability of error Zhao *et al.* (2013). The Fano's lower-bound on the probability of error is also shown to have a high correlation with that probability in practice Morishita *et al.* (2022).

Considering the importance of Fano's inequality, we obtain similar inequalities for the learn-to-defer and DaF methods as follows:

**Theorem 10.** *For predicting a target label $Y \in \mathcal{Y}$ based on a deferral system with a pair of rejector and classifier $(r,h)$, the prediction error $p_{e,\text{def}} = Pr(\hat{Y}_{\text{def}} \neq Y)$ is bounded as*

$$H_B(p_{e,\text{def}}) + p_{e,\text{def}} \log\big(|\mathcal{Y}|-1\big) \geq B(Y,M,r,h), \tag{5.5}$$

*where B is defined as*

$$\begin{aligned} B(Y,M,r,h) :=& H(Y|M,r=1)Pr(r=1) \\ &+ H(Y|h,r=0)Pr(r=0), \end{aligned}$$

*while that of the DaF system $p_{e,\text{DaF}} = Pr(\hat{Y}_{\text{DaF}} \neq Y)$ is bounded as*

$$H_B(p_{e,\text{DaF}}) + p_{e,\text{DaF}} \log\big(|\mathcal{Y}|-1\big) \geq H(Y|M,X), \tag{5.6}$$

*where $H_B(x) = -x\log x - (1-x)\log(1-x)$.*

Note that in the above theorem, the function $g(p) = H_B(p) + p\log(|\mathcal{Y}|-1)$ is a monotonically increasing, and therefore invertible, function. Hence, the above bounds impose direct lower-bounds on probabilities of error using the inverse function $g^{-1}(\cdot)$.

We show in Appendix D.2 that these two entropic lower-bounds have the following relationship with each other

**Remark 1.** $B(Y,M,r,h) \geq H(Y|M,X)$

Although such comparison between the two lower-bounds does not prove that one is necessarily sub-optimal than the other, these bounds are almost-tight, i.e., there are upper-bounds on the probability of error based on such entropic notions too. A simple one that is shown in Tebbe and Dwyer (1968) is $P_e \leq \frac{H(Y|X)}{2}$. As a result of this, we can show that if the gap between the two lower-bounds is as large to satisfy

$$B(Y,M,r,h) > g\big(H(Y|M,X)/2\big) \tag{5.7}$$

too, then the optimal error based on the learn-to-defer method is strictly higher than the optimal error in the DaF method. In Appendix D.1, we provide further theoretical and empirical evidence on such cases of strict sub-optimality.

Now that we provided evidence of the superiority of the DaF method in balanced and imbalanced cost settings, in the following section, we obtain the optimal DaF system that enables us to achieve such overperformance.

## 5.5 Optimal DaF System

Finding the optimal DaF system involves identifying a triple of classifier, rejector, and meta-learner that minimizes the DaF expected loss. According to the following theorem, the optimal classifier and meta-learner are simply equal to the Bayes optimal classifiers that have access to their corresponding inputs, while the optimal rejector sends the instances to the option with the highest value of an extended confidence measure.

**Theorem 11.** *A DaF system that achieves the optimal expected loss is formed by a classifier*

$$h^*(x) \in \underset{h\in\mathcal{Y}}{argmin}\, \mathbb{E}_{Y|X=x}\big[\ell(h,Y)\big], \tag{5.8}$$

*and a meta-learner*

$$g^*(x,m) \in \underset{g\in\mathcal{Y}}{argmin}\, \mathbb{E}_{Y|M=m,X=x}\big[\ell_{\text{fus}}(g,Y)\big], \tag{5.9}$$

*and a rejector*

$$r^*(x) = argmax\{\text{conf}_i\}, \tag{5.10}$$

*where the generalized confidence* $\text{conf}_i$ *for* $i = 0,1,2$ *is defined as following:*

$$\text{conf}_0 = 1 - \min_{h \in \mathcal{Y}} \mathbb{E}_{Y|X=x}\big[\ell(h,Y)\big], \tag{5.11}$$

$$\text{conf}_1 = 1 - \mathop{\mathbb{E}}_{Y,M|X=x}\big[\ell_{\text{def}}(M,Y)\big], \tag{5.12}$$

$$\text{conf}_2 = 1 - \mathop{\mathbb{E}}_{M|X=x}\big[\min_{g \in \mathcal{Y}} \mathop{\mathbb{E}}_{Y|M,X=x}\big[\ell_{\text{fus}}(g,Y)\big]\big]. \tag{5.13}$$

One can easily show that by choosing $0-1$ loss for $\ell, \ell_{\text{def}}$, and $\ell_{\text{fus}}$, the values of ((5.11))-((5.13)) are reduced to the actual confidence (probability of accuracy) of the classifier, expert, and meta-learner, respectively. Moreover, for such a choice of the loss function, Jensen's inequality Jensen (1906) assures that the complementarity occurs after the fusion, i.e., the confidence of the meta-learner is higher than the confidence of either human or classifier. For the theoretical proof and empirical evidence on this claim, we refer the reader to Appendix D.5.

### 5.5.1 Simulating Expert's Decision Model

An implication of ((5.13)) is that to find the meta-learner's confidence, we need information regarding the expert's decision given each instance. One way to provide such information is to approximate $\Pr(M|X=x)$, i.e., to simulate the expert's decision distribution. However, a natural question to ask is that if we can simulate the expert's decision distribution, why do we need to involve the expert in the system at all, and why cannot we just use the expert's simulation? The answer to this question goes back to the fact that in various works within the literature of learn-to-defer, the inherent assumption is that the expert has access to a hidden set of features $U$ as well as the covariate $X$. By this assumption, one can see that approximating $\Pr(M|X=x)$ does not give the full information on the human decision, and only averages out the decision for all instances that have similar covariate $X$. Therefore, the existence of the human in the loop is still essential.

Here, we should note that our work is not the first work that simulates the expert's decision. In fact, in learn-to-defer methods, we need to approximate the confidence $\text{conf}_1$ in ((5.12)). One way to do so is to directly learn such value based on true labels and expert's decision as done in Mozannar and Sontag (2020b). However, in case that we need a general system for which the label cost matrix $\ell_{\text{def}}$ is not pre-defined, and that we need to be able to choose such matrix after training, we have no choice but to simulate expert's decision $p_m(j) = \Pr(M=j|X=x)$ as well as Bayes optimal meta-learner scores $p_y(i,j) = \Pr(Y=i|M=j,X=x)$ and use it to find the confidence as

$$\text{conf}_1 = 1 - \sum_{i=1}^{|\mathcal{Y}|} \sum_{j=1}^{|\mathcal{Y}|} p_y(i,j)\, p_m(j)\, \ell_{\text{def}}(i,j). \tag{5.14}$$

### 5.5.2 DaF in Multiple Experts Setting

The optimal DaF system can further be obtained in the case that we have more than one expert. In medical diagnosis scenarios, this setting arises when we need feedback from a specific area of expertise out of many, or if we want to form a committee of expertise.

In the first setting, in which the feedback is requested from an expert out of many, the optimal DaF system first obtains the 'generalized confidence', defined as in Theorem 11, for the classifier, each expert, and their fusion with the covariate. Then, the rejector chooses the option with the highest confidence (See Theorem 20 in Appendix D.10). Further, the classifier and the meta-learner corresponding to each expert is further obtained using Bayes optimal rule.

To form a committee, however, we might need to exponentially grow the complexity of optimal solution. In fact, in this case, the optimal system needs a meta-learner with the decision of each combination of the experts as input (See Proposition 7), which is $2^N$ meta-learners where $N$ is the number of experts. To address this, we need to assume the following two conditions: (i) there is a pre-known aggregation method for experts decisions, and (ii) the decisions made by experts are independent of each other given the covariate $x$.

These two conditions are also mainly assumed in classical analysis of democratic solutions (see e.g. Jury's theorem Boland (1989) and Dietrich and Spiekermann (2021)). Here, we should note that the conditional independence of the experts does not mean that they cannot share similar decision distribution for each case, but it implies that they should not communicate with each other about cases.

Assuming these conditions, we can first design a general meta-learner that takes in the outcome of the aggregation of expert's votes and predicts the label. Then, we can simulate each expert's decision distribution, and use sampling methods to approximate the confidence of all combinations of options based on the mentioned distributions. For further details, we refer the reader to Appendix D.10.

### 5.5.3 Deferral, Fusion, and Combination

So far, we have obtained the optimal DaF system. Based on the discussion in Section 5.5, to optimize a $0-1$ loss, the system reduces to the fusion part, and as we mentioned fusion alone could theoretically overperform the learn-to-defer method. However, implementing such a fusion system requires high sample complexity due to the possibility of large dimensions of the covariates. To reduce the sample complexity, one might think of using classifier probabilities instead of covariates as the input of the meta-learner. This method, which from now on we refer to as the combination method, is already studied in Kerrigan *et al.* (2021), and its sub-optimality compared to fusion as a result of the reduction of its input information should be clear to the reader. However, finding a condition under which these methods are equivalent can be important to uncover the differences between these methods. In the following theorem, we provide a necessary and sufficient

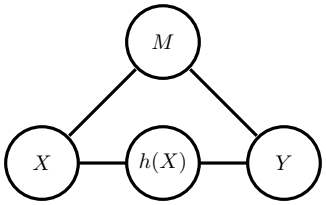



Figure 5.3: DaF and combination method are equivalent iff. such graphical model holds

condition on this equivalence:

**Theorem 12.** *In the case of* $0-1$ *loss, DaF and combination method are equivalent iff. the graphical model in Figure 5.3 is the underlying model of variables.*

As an instance, such a diagram holds if $X$ and $M$ are independent given the true label $Y$ (See Theorem 17 and Remark 2 in Appendix D.7). Although a similar but weaker condition is assumed for the underlying model in Kerrigan *et al.* (2021), but in the following, we argue such a diagram does not hold in some central examples of learn-to-defer, in which the algorithm should spot the neighborhoods of covariate space on which expert is accurate and defer accordingly.

**Example 5.** *Assume that there exist two regions in the feature space on which the expert behavior differs from each other. As an instance, assume that for* $x > 0$ *human is correct all the times, while for* $x \leq 0$ *they are correct* 70% *of the times. Further, assume that the classifier has the confidence of* 80% *everywhere. This means that in case there is no information of x in hand, the confidence of the classifier does not provide you with essential information from the feature, and the best combination of the classifier's confidence and human decision, in this case, is to accept human decision everywhere, since its average accuracy over the whole region is* 85%. *However, fusion and deferral methods give the human decision higher weight for* $x > 0$ *and less weight otherwise. See Appendix D.8 for further details on this example.*

Knowing the theoretical over-performance of DaF compared to other works in the literature, in the following section we implement DaF and evaluate it empirically.

## 5.6 Training DaF Components

As we discussed in Section 5.3, a DaF system comprises a classifier, a meta-learner, and a rejector. Following the literature on learn-to-defer methods on jointly training the components Mozannar and Sontag (2020b); Charusaie *et al.* (2022b); Verma and Nalisnick (2022b), we take similar approaches. In the following, we summarize the methods to train such components:

- **Simulation-based DaF (SDaF)**: In this method, we assume the label cost matrix is not given in training time. Here, we train three networks $f_h(x;\theta)$, $f_S(x;\theta)$, and $f_g(x,m;\theta)$, for classification, human simulation, and fusion, respectively, using a One-vs-All (OvA) surrogate loss

$$L_{SDaF} = \ell(f_h,\widetilde{y}) + \ell(f_S,m) + \ell(f_g,y), \tag{5.15}$$

  where $\ell(f,y) = \phi(f^y) + \sum_{y'\neq y}\phi(-f^{y'})$ and $\phi(\cdot)$ is a binary surrogate loss. Further, $\widetilde{y}$ is a one-hot coded version of $y$ with an extra dimension that is 1 when $y = m$. We use the obtained functions as proxies of the probabilities and use Theorem 11 to achieve a DaF system.

- **Learning-based DaF (LDaF)**: In this method, the label cost matrix is given in the beginning. Here, two networks $f_h(x;\theta)$ and $f_g(x,m;\theta)$ are trained as classifier/rejector, and meta-learner, respectively, by the following OvA loss

$$L_{LDaF} = \ell(f_h,\overline{y}) + \ell(f_g,y), \tag{5.16}$$

  where $\overline{y}$ is the one-hot coded version of $y$ with two extra dimensions which are the value of human decision cost and fusion decision cost.

- **Confidence-based DaF (CDaF)**: In this method, four networks $f_h(x;\theta)$, $f_g(x,m;\theta)$, $f_{def}(x;\theta)$, and $f_{fus}(x;\theta)$ are each trained on softmax loss and to learn true label $y$, cost of deferral, and cost of fusion, respectively. In each epoch, we take a step to minimize the corresponding losses one-by-one. This method helps to achieve a calibration that is not achievable in OvA methods (See Appendix D.12).

To guarantee the convergence of the above methods to the optimal solution, we provide the following theoretical result on the consistency of LDaF and SDaF surrogate losses. The consistency here means that for a large enough hypothesis class and enough data, the optimal solution of the true DaF loss ((5.2)) is also a global minimizer of the surrogates. The guarantee is as follows:

**Theorem 13.** *If $\phi(\cdot) : \mathbb{R} \to \mathbb{R}$ is a strictly proper binary surrogate function, then LDaF and SDaF surrogates are Fisher consistent.*

The strictly proper condition roughly means that the minimizer of the binary surrogate $\phi$ should recover the conditional probability of the labels given instances.

Next, we assess the above guarantee empirically and observe the results over a set of datasets.

## 5.7 Experiments

### 5.7.1 Settings

During our experiments, we compare DaF methods with the following methods: (i) confidence-based method (CC) of Raghu *et al.* (2019) that trains a classifier network as well as a network that captures human confidence, (ii) cross entropy method (LCE) Mozannar and Sontag (2020b) that trains the classifier and rejector using a joint cross entropy surrogate loss, (iii) One-vs-All method (OvA) Verma and Nalisnick (2022b) that proposes a surrogate that induces calibration on network outputs, (iv) reallizable surrogate (RS) Mozannar *et al.* (2023c) that is tailored for the models with low capacity, and (v) active feature elicitation (AFE) Shim *et al.* (2018) that similar to our work decides on either classifying the instance, or asking for an extra feature (here human decision).

Four datasets on which we mainly focus are: (i) CIFAR-10K Mozannar *et al.* (2023c) that is a semi-synthetic data based on CIFAR-10 dataset Krizhevsky *et al.* (2009) and the expert is accurate only on $K$ out of 10 classes, (ii) CIFAR-10H Peterson *et al.* (2019), a real-world dataset based on CIFAR-10 test set that has collected human label distributions on each instance, (iii) ImageNet-16H Steyvers *et al.* (2022) that collected human decisions on noisy versions on images in ImageNet Deng *et al.* (2009), and (iv) Hatespeech dataset Davidson *et al.* (2017) that collected crowd-sourced opinions on hateful language of a set of tweets.

### 5.7.2 Cost-Sensitive Risks

We assume a uniformly random label cost matrix with zero diagonals and train different methods to minimize the expected loss. Figure D.7 shows the resulting expected loss for different levels of human involvement. The human involvement is set by setting various thresholds in rejector networks. This figure shows the superiority of our method to other methods on Imagenet-16H dataset. The results on other datasets is deferred to Appendix D.13.

In the following, we list a series of experiments on the above datasets. For further experiments on sample complexity, accuracy, the role of deferral costs, and examples of sub-optimality of learn-to-defer methods, we refer the reader to Appendix D.13.

### 5.7.3 $0-1$ Risk

We train the aforementioned methods for CIFAR-10K and for varying $K$ value, and for the other three datasets. The results are shown in Figure 5.4 and in Appendix D.13. We observe that our method outperforms the other methods in a majority of cases.

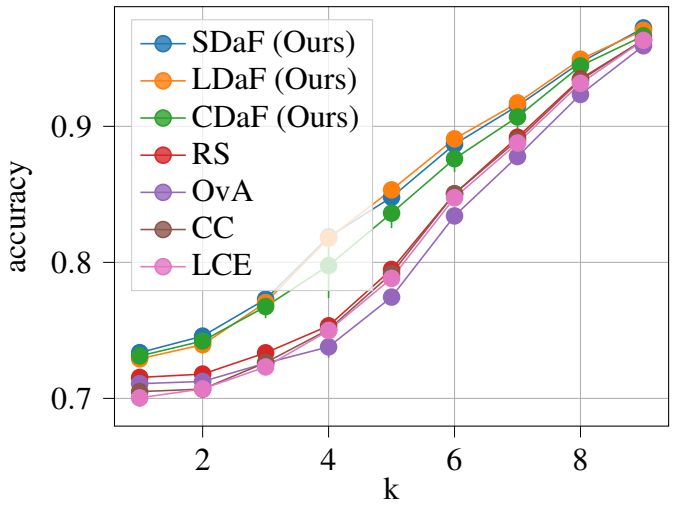



Figure 5.4: Accuracy of our method compared to deferral methods on CIFAR-10K

## 5.8 Conclusion

In this paper, a deferral-and-fusion (DaF) method for handling human feedback in machine-assisted processes is proposed. An example of strict-sub-optimality of learn-to-defer in imbalanced label cost cases, and the theoretical possibility of improvement further in 0-1 losses is provided. Further, the optimal DaF system in the case of one or multiple experts is theoretically obtained and empirically approximated. The provided evidence in this paper shows that DaF systems can be an alternative to fully automatized systems and to classifiers with rejection options. In summary, this paper shows that the correlation of human decision, covariates, and their interaction with the label cost matrix can improve the expected loss in predicting the true label and leave us with a more accurate prediction.

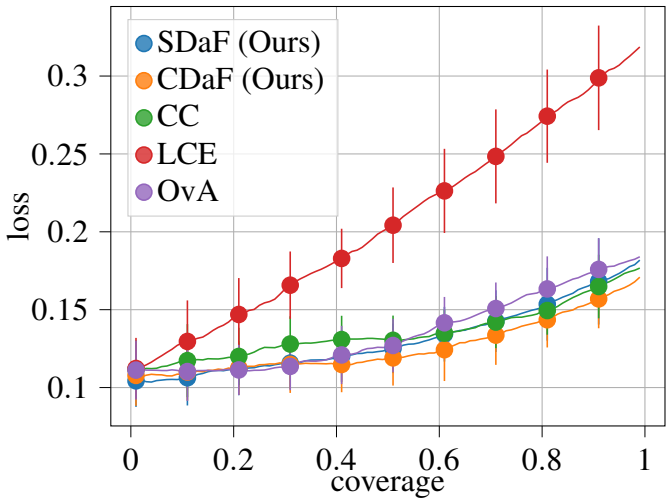



Figure 5.5: Average loss vs. coverage on Imagenet-16H datasets and for 10 random sets of prediction costs, with bars being standard deviation. Coverage is a number that represents the percentage of instances that are sent to ML. So the higher the coverage, there are less instances being sent to human.

# Appendix A

# Appendices I

## A.1 Effect of length scale on the kernel approximation

Fig. A.1 shows the effect of the kernel length scale on the kernel approximation for both HPs and RFs.

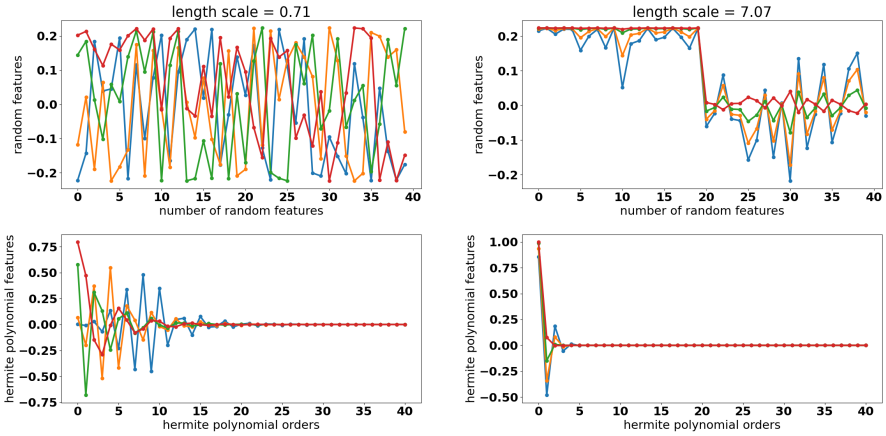



Figure A.1: Comparison between HP and random features at a different length scale value. Different color indicates a different datapoint, where four datapoints are drawn from $\mathcal{N}(0,1)$. **Left**: With length scale $l = 0.71$ (relatively small compared to 1), random features (top) at the four datapoints exhibit large variability while the Hermite polynomial features (bottom) at those datapoints decay at around order $\leq 20$. **Right**: With $l = 7.07$ (large compared to 1), random features (top) exhibit less variability, while it is not clear how many features are necessary to consider. On the other hand, the Hermite polynomial features (bottom) decay fast at around order $\leq 5$ and we can make a cut-off at that order without losing much information.

## A.2 Approximation error under HP and Random Fourier features

In the following proposition, we provide that provably our method converges with $O(\rho^{2C})$ where $\rho < 1$ is the constant in the Mehler's formula, while DP-MERF has the convergence $\Omega(1/C)$, where $C$ is the number of features in each case.

**Proposition A.2.1.** *Let $X$ and $Y$ be standard normal random variables. There exists a $C$-dimensional Hermite feature map $\phi_{HP}^{(C)}(\cdot)$ with the expected predictive error bounded as*

$$\mathbb{E}_{X,Y}\left[\left|k(X,Y) - \langle \hat{\phi}_{HP}^{(C)}(X), \hat{\phi}_{HP}^{(C)}(Y)\rangle\right|\right] \leq \frac{1}{3\sqrt{2}}(\frac{1}{3})^C. \tag{A.1}$$

*However, the expected predictive error of the random feature map $\hat{\phi}_{RF,\omega}(\cdot)$ with $C$ number of features (i.e., $\omega$ is a vector of length $C$) and the same approximating kernel is equal to*

$$\mathbb{E}_{\boldsymbol{\omega},X,Y}\left[\left|k(X,Y) - \langle \hat{\phi}_{RF,\omega}(X), \hat{\phi}_{RF,\omega}(Y)\rangle\right|\right] \geq \frac{1}{8C}. \tag{A.2}$$

*Proof.* We start by proving eq. (A.1). In this case, we write the squared error term as following:

$$A_{x,y} = \left|k(x,y) - \langle \hat{\phi}_{HP}^{(C)}(x), \hat{\phi}_{HP}^{(C)}(y)\rangle\right|^2 \overset{(a)}{=} \left|\sum_{C+1}^{\infty} \frac{\lambda_l}{\sqrt{N_l}} H_l(x) e^{-\frac{\rho}{1+\rho}x^2} \frac{1}{\sqrt{N_l}} H_l(y) e^{-\frac{\rho}{1+\rho}y^2}\right|^2 \tag{A.3}$$

$$= \sum_{l,l'=C+1}^{\infty} \frac{\lambda_l \lambda_{l'}}{N_l N_{l'}} H_l(x) H_{l'}(x) H_l(y) H_{l'}(y) e^{-\frac{2\rho}{1+\rho}x^2 - \frac{2\rho}{1+\rho}y^2}, \tag{A.4}$$

where $(a)$ is followed by the definition of $\hat{\phi}_{HP}^{(C)}$ in eq. (2.6) and its approximation property (i.e., Mehler's formula eq. (2.5)). Now, by setting $\rho = \frac{1}{3}$, we have

$$A_{x,y} = \sum_{l,l'=C+1}^{\infty} \frac{\lambda_l \lambda_{l'}}{N_l N_{l'}} H_l(x) H_{l'}(x) H_l(y) H_{l'}(y) e^{-\frac{1}{2}x^2 - \frac{1}{2}y^2}. \tag{A.5}$$

Next, we average out $A_{x,y}$ for $x$s and $y$s that are drawn from a standard normal distribution

as

$$\mathbb{E}_{X,Y\sim N(0,1)}\left[A_{X,Y}\right] = \int_{x,y=-\infty}^{\infty} \sum_{l,l'=C+1}^{\infty} \frac{\lambda_l \lambda_{l'}}{N_l N_{l'}} H_l(x) H_{l'}(x) H_l(y) H_{l'}(y) e^{-\frac{1}{2}x^2 - \frac{1}{2}y^2} \frac{e^{-\frac{1}{2}x^2 - \frac{1}{2}y^2}}{2\pi} \mathrm{d}x\mathrm{d}y \tag{A.6}$$

$$= \sum_{l,l'=C+1}^{\infty} \frac{\lambda_l \lambda_{l'}}{N_l N_{l'}} \frac{\int H_l(x) H_{l'}(x) e^{-x^2} \mathrm{d}x \int H_l(y) H_{l'}(y) e^{-y^2} \mathrm{d}y}{2\pi} \tag{A.7}$$

$$\overset{(a)}{=} \sum_{l=C+1}^{\infty} \frac{\lambda_l^2}{N_l^2} \frac{1}{2\pi} \sqrt{\pi} 2^l l! \sqrt{\pi} 2^l l! \overset{(b)}{=} \sum_{l=C+1}^{\infty} \frac{(2/3)^2 (1/3)^{2l}}{\frac{1}{2} 2^{2l} (l!)^2} \frac{2^{2l} (l!)^2}{2} \tag{A.8}$$

$$= \frac{4}{9} \sum_{l=C+1}^{\infty} (1/3)^{2l} \tag{A.9}$$

$$\overset{(c)}{=} \frac{1}{2} (1/3)^{2C+2}, \tag{A.10}$$

where $(a)$ is followed by orthogonality of Hermite polynomials, $(b)$ is followed by the definition of $\lambda_l$ and $N_l$ in Section 2.3.1, and $(c)$ is due to the infinite Geometric series.

As a result of eq. (A.10), the definition of $A_{x,y}$, and Jensen's inequality we have

$$\mathbb{E}_{X,Y}\left[\left|k(X,Y) - \langle \hat{\phi}_{HP}^{(C)}(X), \hat{\phi}_{HP}^{(C)}(X) \rangle\right|\right] \leq \mathbb{E}_{X,Y}^{1/2}\left[A_{X,Y}\right] \leq \frac{1}{3\sqrt{2}} \left(\frac{1}{3}\right)^C. \tag{A.11}$$

For bounding the expected error of random features, we expand the squared error using the definition given in eq. (2.4):

$$B_{x,y,\boldsymbol{\omega}} = \left|k(x,y) - \langle \phi_{RF,\boldsymbol{\omega}}(x), \phi_{RF,\boldsymbol{\omega}}(y) \rangle\right|^2 \tag{A.12}$$

$$= \left|e^{-\frac{\rho(x-y)^2}{1-\rho^2}} - \frac{2}{C} \sum_{i=1}^{C/2} \cos \omega_i x \cos \omega_i y - \frac{2}{C} \sum_{i=1}^{C/2} \sin \omega_i x \sin \omega_i y\right| \tag{A.13}$$

$$= \underbrace{\left|e^{-\frac{\rho(x-y)^2}{1-\rho^2}} - \frac{2}{C} \sum_{i=1}^{C/2} \cos \omega_i (x-y)\right|^2}_{B_{x,y,\boldsymbol{\omega}}}. \tag{A.14}$$

Next, by setting $\rho = \frac{1}{3}$, we have

$$B_{x,y,\boldsymbol{\omega}} = e^{-\frac{3}{4}(x-y)^2} - \frac{4}{C} e^{-\frac{3}{8}(x-y)^2} \underbrace{\sum_{i=1}^{C/2} \cos \omega_i (x-y)}_{E_{1,x,y,\boldsymbol{\omega}}} + \frac{4}{C^2} \underbrace{\left(\sum_{i=1}^{C/2} \cos \omega_i (x-y)\right)^2}_{E_{2},x,y,\boldsymbol{\omega}}. \tag{A.15}$$

Next, we calculate the average of terms $E_{1,x,y,\boldsymbol{\omega}}$ and $E_{2,x,y,\boldsymbol{\omega}}$ over $\boldsymbol{\omega}$.

Due to the Bochner's theorem (see Theorem 3.7 of Unser and Tafti (2014)) that shows a shift-invariant positive kernel could be written in the form of Fourier transform of a density function, we have

$$\mathbb{E}_{\boldsymbol{\omega}}\big[E_{1,x,y,\boldsymbol{\omega}}\big] = \mathbb{E}_{\boldsymbol{\omega}}\big[\sum_{i=1}^{C/2}\cos\omega_i(x-y)\big] \tag{A.16}$$

$$= \sum_{i=1}^{C/2}\mathbb{E}_{\omega_i}\big[e^{j\omega_i(x-y)}\big] = \frac{C}{2}e^{-\frac{3}{8}(x-y)^2}, \tag{A.17}$$

Next, we obtain the average of $E_{2,x,y,\boldsymbol{\omega}}$ as following:

$$\mathbb{E}_{\boldsymbol{\omega}}\big[E_{2,x,y,\boldsymbol{\omega}}\big] = \mathbb{E}_{\boldsymbol{\omega}}\big[\sum_{i,k=1}^{C/2}\cos\omega_i(x-y)\cos\omega_k(x-y)\big] \tag{A.18}$$

$$= \mathbb{E}_{\boldsymbol{\omega}}\big[\sum_{i,k=1}^{C/2}\frac{e^{j(\omega_i+\omega_k)(x-y)}+e^{j(\omega_i-\omega_k)(x-y)}+e^{j(-\omega_i+\omega_k)(x-y)}+e^{j(-\omega_i-\omega_k)(x-y)}}{4}\big] \tag{A.19}$$

$$\overset{(a)}{=} \sum_{i,k=1,i\neq k}^{C/2}\mathbb{E}_{\omega_i}\big[e^{j\omega_i(x-y)}\big]\mathbb{E}_{\omega_k}\big[e^{j\omega_k(x-y)}\big] + \frac{1}{2}\sum_{i=1}^{C/2}\big(\mathbb{E}_{\omega_i}\big[e^{j\omega_i(2x-2y)}\big]+1\big) \tag{A.20}$$

$$\overset{(b)}{=} \big(\frac{C^2}{4}-\frac{C}{2}\big)e^{-\frac{3}{4}(x-y)^2} + \frac{C}{4}\big(e^{-\frac{3}{4}(x-y)^2}+1\big) \tag{A.21}$$

$$= \frac{C^2}{4}e^{-\frac{3}{4}(x-y)^2} + \frac{C}{4}\big(e^{-\frac{3}{2}(x-y)^2} - 2e^{-\frac{3}{4}(x-y)^2}+1\big), \tag{A.22}$$

where $(a)$ is due to symmetry of the normal distribution of $\boldsymbol{\omega}$, and $(b)$ is followed by independence of $\omega_i$ and $\omega_k$ and their distribution symmetry.

Substituting eq. (A.17) and eq. (A.22) in eq. (A.15), and using Jensen's inequality, we have

$$\mathbb{E}_{X,Y\sim N(0,1)}\mathbb{E}_{\boldsymbol{\omega}}\big[B_{x,y,\boldsymbol{\omega}}\big] = \frac{1}{C}\mathbb{E}_{X,Y\sim N(0,1)}\Big[\big(e^{-\frac{3}{4}(X-Y)^2}-1\big)^2\Big] \geq \frac{1}{C}\mathbb{E}^2_{X,Y}\Big[e^{-\frac{3}{4}(x-y)^2}-1\Big] \tag{A.23}$$

$$= \frac{1}{C}\big(\underbrace{\mathbb{E}_{X,Y\sim N(0,1)}\big[e^{-\frac{3}{4}(X-Y)^2}\big]}_{G}-1\big)^2. \tag{A.24}$$

To calculate $G$, we have

$$G = \mathbb{E}_{X,Y\sim N(0,1)}\left[e^{-\frac{3}{4}(X-Y)^2}\right] = \int_{x,y} \frac{e^{-\frac{3}{4}(x^2+y^2-2xy)} e^{-\frac{x^2}{2}-\frac{y^2}{2}}}{2\pi} \mathrm{d}x\mathrm{d}y \tag{A.25}$$

$$= \int_{x,y} \frac{e^{-\frac{5}{4}(x^2+y^2)+\frac{3}{2}xy}}{2\pi} \mathrm{d}x\mathrm{d}y \tag{A.26}$$

$$= \int_{x,y} \frac{e^{-\frac{5}{4}(x^2-\frac{6}{5}xy+\frac{9}{25}y^2)+\frac{9}{25}\frac{5}{4}y^2-\frac{5}{4}y^2}}{2\pi} \mathrm{d}x\mathrm{d}y \tag{A.27}$$

$$\overset{(a)}{=} \int_y \frac{e^{-\frac{4}{5}y^2}}{\sqrt{2\pi\frac{5}{2}}} \int_x \frac{e^{-\frac{5}{4}(x-\frac{3}{5}y)^2}}{\sqrt{2\pi\frac{2}{5}}} \mathrm{d}x\mathrm{d}y \tag{A.28}$$

$$= \int_y \frac{e^{-\frac{4}{5}y^2}}{\sqrt{2\pi\frac{5}{2}}} \mathrm{d}y = \frac{1}{2}\int_y \frac{e^{-\frac{4}{5}y^2}}{\sqrt{2\pi\frac{5}{8}}} \tag{A.29}$$

$$\overset{(b)}{=} \frac{1}{2}, \tag{A.30}$$

where $(a)$ and $(b)$ hold since for a normal distribution $f_{a,b}(x) = \frac{e^{-\frac{(x-b)^2}{2a}}}{\sqrt{2\pi a}}$, then, we have $\int_x f_{a,b}(x)\mathrm{d}x = 1$. As a result of eq. (A.24) and eq. (A.30) we have

$$\mathbb{E}_{X,Y,\boldsymbol{\omega}}\left[B_{X,Y,\boldsymbol{\omega}}\right] \geq \frac{1}{4C}. \tag{A.31}$$

Finally, since $0 \leq B_{x,y,\boldsymbol{\omega}} \leq 4$, we have

$$\frac{1}{16C} \leq \mathbb{E}_{X,Y,\boldsymbol{\omega}}\left[\frac{B_{X,Y,\boldsymbol{\omega}}}{4}\right] \leq \mathbb{E}_{X,Y,\boldsymbol{\omega}}\left[\frac{|B_{X,Y,\boldsymbol{\omega}}|^{1/2}}{2}\right] \tag{A.32}$$

$$= \frac{1}{2}\mathbb{E}_{X,Y,\boldsymbol{\omega}}\left[\left|k(X,Y) - \langle \phi_{RF,\boldsymbol{\omega}}(X), \phi_{RF,\boldsymbol{\omega}}(Y)\rangle\right|\right], \tag{A.33}$$

which proves eq. (A.2). □

## A.3 Mercer's theorem and the generalized Hermite polynomials

We first review Mercer's theorem, which is a fundamental theorem on how can we find the approximation of a kernel via finite-dimensional feature maps.

**Theorem 14** (Smola and Schölkopf (1998) Theorem 2.10 and Proposition 2.11 )**.** *Suppose $k \in L_\infty(\mathcal{X}^2)$, is a symmetric real-valued function, for a non-empty set $\mathcal{X}$, such that*

*the integral operator* $T_k f(x) = \int_{\mathcal{X}} k(x,x')f(x')\partial\mu(x')$ *is positive definite. Let* $\psi_j \in L_2(\mathcal{X})$ *be the normalized orthogonal eigenfunctions of* $T_k$ *associated with the eigenvalues* $\lambda_j > 0$*, sorted in non-increasing order, then*

1. $(\lambda_j)_j \in \ell_1$,
2. $k(x,x') = \sum_{j=1}^{N_{\mathcal{H}}} \lambda_j \psi_j(x)\psi_j(x')$ *holds for almost all* $(x,x')$*. Either* $N_{\mathcal{H}} \in \mathbb{N}$*, or* $N_{\mathcal{H}} = \infty$*; in the latter case, the series converge absolutely and uniformly for almost all* $(x,x')$.

*Furthermore, for every* $\varepsilon > 0$*, there exists n such that*

$$|k(x,x') - \sum_{j=1}^{n} \lambda_j \psi_j(x)\psi_j(x')| < \varepsilon, \tag{A.34}$$

*for almost all* $x,x' \in \mathcal{X}$.

This theorem states that one can define a feature map

$$\Phi_n(x) = \left[\sqrt{\lambda_1}\psi_1(x), \ldots, \sqrt{\lambda_n}\psi_n(x)\right]^T \tag{A.35}$$

such that the Euclidean inner product $\langle \Phi(x), \Phi(x') \rangle$ approximates $k(x,x')$ up to an arbitrarily small factor $\varepsilon$.

By means of uniform convergence in Mercer's theorem, we can prove the convergence of the approximated MMD using the following lemma.

**Lemma 1.** *Let* $\mathcal{H}$ *be an RKHS that is generated by the kernel* $k(\cdot,\cdot)$*, and let* $\widehat{\mathcal{H}}_n$ *be an RKHS with a kernel* $k_n(\mathbf{x},\mathbf{y})$ *that can uniformly approximate* $k(\mathbf{x},\mathbf{y})$*. Then, for a positive real value* $\varepsilon$*, there exists n, such that for every pair of distributions* $P,Q$*, we have*

$$\left|\mathrm{MMD}^2_{\mathcal{H}}(P,Q) - \mathrm{MMD}^2_{\widehat{\mathcal{H}}_n}(P,Q)\right| < \varepsilon. \tag{A.36}$$

*Proof.* Firstly, using Theorem 14, we can find $n$ such that $\left|k(x,y) - \langle \Phi_n(x), \Phi_n(y)\rangle\right| < \frac{\varepsilon}{4}$. We define the RKHS $\widehat{H}_n$ as the space of functions spanned by $\Phi_n(\cdot)$. Next, we rewrite $\mathrm{MMD}^2_{\mathcal{H}}(P,Q) - \mathrm{MMD}^2_{\widehat{\mathcal{H}}_n}(P,Q)$, using the definition of MMD in Section 2.2.1, as

$$\begin{aligned}
&\mathrm{MMD}^2_{\mathcal{H}}(P,Q) - \mathrm{MMD}^2_{\widehat{\mathcal{H}}_n}(P,Q) \\
&= \mathbb{E}_{x,x'\sim P}\left[k(x,x')\right] + \mathbb{E}_{y,y'\sim Q}\left[k(y,y')\right] - 2\mathbb{E}_{x\sim P, y\sim Q}\left[k(x,y)\right] \\
&- \mathbb{E}_{x,x'\sim P}\left[\langle \Phi_n(x), \Phi_n(x')\rangle\right] + \mathbb{E}_{y,y'\sim Q}\left[\langle \Phi_n(y), \Phi_n(y')\rangle\right] - 2\mathbb{E}_{x\sim P, y\sim Q}\left[\langle \Phi_n(x), \Phi_n(y)\rangle\right]
\end{aligned} \tag{A.37}$$

Therefore, we can bound $\left|\mathrm{MMD}^2_{\mathcal{H}}(P,Q) - \mathrm{MMD}^2_{\widehat{\mathcal{H}}_n}(P,Q)\right|$ as

$$\left|\mathrm{MMD}^2_{\mathcal{H}}(P,Q) - \mathrm{MMD}^2_{\widehat{\mathcal{H}}_n}(P,Q)\right| \overset{(a)}{\leq} \left|\mathbb{E}_{x,x'\sim P}\left[k(x,x')\right] - \mathbb{E}_{x,x'\sim P}\left[\left\langle \Phi_n(x), \Phi_n(x')\right\rangle\right]\right|$$

$$+ \left|\mathbb{E}_{y,y'\sim Q}\left[k(y,y')\right] - \mathbb{E}_{y,y'\sim P}\left[\left\langle \Phi_n(y), \Phi_n(y')\right\rangle\right]\right|$$
$$+ 2\left|\mathbb{E}_{x,y\sim P,Q}\left[k(x,y)\right] - \mathbb{E}_{x,y\sim P,Q}\left[\left\langle \Phi_n(x), \Phi_n(y)\right\rangle\right]\right|$$

$$\overset{(b)}{\leq} \mathbb{E}_{x,x'\sim P}\left[\left|k(x,x') - \left\langle \Phi_n(x), \Phi_n(x')\right\rangle\right|\right] + \mathbb{E}_{y,y'\sim Q}\left[\left|k(y,y') - \left\langle \Phi_n(y), \Phi_n(y')\right\rangle\right|\right]$$
$$+ 2\mathbb{E}_{x,y\sim P,Q}\left[\left|k(x,y) - \left\langle \Phi_n(x), \Phi_n(y)\right\rangle\right|\right]$$
$$\overset{(c)}{\leq} \mathbb{E}_{x,x'\sim P}\left[\frac{\varepsilon}{4}\right] + \mathbb{E}_{y,y'\sim Q}\left[\frac{\varepsilon}{4}\right] + 2\mathbb{E}_{x,y\sim P,Q}\left[\frac{\varepsilon}{4}\right] = \varepsilon \tag{A.38}$$

where $(a)$ holds because of triangle inequality, $(b)$ is followed by Tonelli's theorem and Jensen's inequality for absolute value function, and $(c)$ is correct because of the choice of $n$ as mentioned earlier in the proof. □

As a result of the above theorems, we can approximate the MMD in RKHS $\mathcal{H}_k$ for a kernel $k(\cdot,\cdot)$ via MMD in RKHS $\widehat{\mathcal{H}}_n \subseteq \mathbb{R}^n$ that is spanned by the first $n$ eigenfunctions weighted by square roots of eigenvalues of the kernel $k(\cdot,\cdot)$. Therefore, in the following section, we focus on finding the eigenfunctions/eigenvalues of a multivariate Gaussian kernel.

### A.3.1 Generalized Mehler's approximation

As we have already seen in eq. (2.5), Mehler's theorem provides us with an approximation of a one-dimensional Gaussian kernel in terms of Hermite polynomials. To generalize Mehler's theorem to a uniform covergence regime (that enables us to approximate MMD via such feature maps as shown in Lemma 1), and for a multivariate Gaussian kernel, we make use of the following theorem.

**Theorem 15** (Slepian (1972), Section 6)**.** *Let the joint Gaussian density kernel* $k(\mathbf{x},\mathbf{y},C) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ *be*

$$k(\mathbf{x},\mathbf{y},C) = \frac{1}{(2\pi)^n |C|^{1/2}} \exp\Big(-\frac{1}{2}[\mathbf{x},-\mathbf{y}]C^{-1}[\mathbf{x},-\mathbf{y}]^T\Big) \tag{A.39}$$

*where C is a positive-definite matrix as*

$$C = \begin{bmatrix} C_{11} & C_{12} \\ C_{12}^T & C_{22} \end{bmatrix}, \tag{A.40}$$

*in which* $C_{ij} \in \mathbb{R}^{n\times n}$ *for* $i, j \in \{1,2\}$*, and* $C_{11} = C_{22}$*. Further, let the integral operator be defined with respect to a measure with density*

$$w(\mathbf{x}) = \frac{1}{\int k(\mathbf{x},\mathbf{y},C)\partial\mathbf{y}}. \tag{A.41}$$

*Then, the orthonormal eigenfunctions and eigenvalues for such kernel are*

$$\psi_{\mathbf{k}}(\mathbf{x}) = \sum_{\mathbf{l}:\|\mathbf{l}\|_1=\|\mathbf{k}\|_1} \big(\sigma_{\|\mathbf{k}\|_1}(P)^{-1}\big)_{\mathbf{kl}} \frac{\varphi_{\mathbf{l}}(\mathbf{x};C_{11})}{\sqrt{\prod_{i=1}^n l_i!}}, \tag{A.42}$$

*and*

$$\lambda_{\mathbf{k}} = \prod_{i=1}^n \mathbf{e}_i^{\mathbf{k}_i/2}. \tag{A.43}$$

*Here,* $\sigma_p(A)$ *is symmetrized Kronecker power of a matrix A, defined as*

$$\big(\sigma_{\|\mathbf{k}\|_1}(A)\big)_{\mathbf{kl}} = \sqrt{\prod_{i=1}^n k_i! l_i!} \sum_{M\in\mathbb{R}^{n\times n}: M\mathbb{1}_n=\mathbf{k}, \mathbb{1}_n^T M=\mathbf{l}} \frac{\prod_{ij} A_{ij}^{M_{ij}}}{\prod_{ij} M_{ij}!}, \tag{A.44}$$

*for two n-dimensional vectors* $\mathbf{k}$ *and* $\mathbf{l}$ *with* $\|\mathbf{k}\|_1 = \|\mathbf{l}\|_1$*, the vector* $\mathbf{e}$ *(the matrix P) is formed by eigenvalues (eigenvectors) of* $C_{11}^{-1}C_{12}$*, and* $\varphi_{\mathbf{l}}(\mathbf{x},A)$ *is generalized Hermite functions defined as*

$$\varphi_{\mathbf{l}}(\mathbf{x},A) = \frac{1}{(2\pi)^{n/2}|A|^{1/2}} \frac{\partial^{\|\mathbf{l}\|_1}}{\partial {x_1}^{l_1} \dots \partial {x_n}^{l_n}} \exp\big(-\frac{1}{2}\mathbf{x}^T A^{-1}\mathbf{x}\big). \tag{A.45}$$

The above theorem provides us with eigenfunctions/eigenvalues of a joint Gaussian density function. We utilize this theorem to approximate Mahalanobis kernels (i.e., a generalization of Gaussian radial basis kernels where $A = cI_n$) via Hermite polynomials as follow.

**Proposition A.3.1.** *A Mahalanobis kernel* $k(\mathbf{x},\mathbf{y},A) : \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$ *defined as*

$$k(\mathbf{x},\mathbf{y},A) = \exp\big(-(\mathbf{x}-\mathbf{y})A(\mathbf{x}-\mathbf{y})^T\big)$$

*can be uniformly approximated as*

$$k(\mathbf{x},\mathbf{y},A) \simeq \Big\langle \Phi_N\Big(\sqrt{\frac{\alpha^2-1}{\alpha}}\sqrt{A}\mathbf{x}\Big), \Phi_N\Big(\sqrt{\frac{\alpha^2-1}{\alpha}}\sqrt{A}\mathbf{y}\Big)\Big\rangle, \tag{A.46}$$

*where* $\Phi(\mathbf{x}) \in N^D$ *is defined as a tensor product*

$$\Phi_N(\mathbf{x}) = \bigotimes_{i=1}^{n} [\phi_{k_i}(x_i)]_{k_i=1}^{N}, \tag{A.47}$$

*where*

$$\phi_{k_i}(x_i) = \left(\frac{(\alpha^2-1)\alpha^{-k_i}}{\alpha^2 k_i!}\right)^{1/4} \exp\left(\frac{-x_i^2}{\alpha+1}\right) H_{k_i}(x_i) \tag{A.48}$$

**Remark 1.** *Using Proposition A.3.1 and Lemma 1, we can show that the MMD based on the tensor feature map in eq. (A.47) and between any two distributions approximates the real MMD based on Gaussian kernel with Mahalanobis norm.*

*Proof of Proposition A.3.1.* Let $C = \begin{bmatrix} \frac{1}{2}I_n & \frac{1}{2\alpha}I_n \\ \frac{1}{2\alpha}I_n & \frac{1}{2}I_n \end{bmatrix}$, or equivalently

$$C^{-1} = \begin{bmatrix} \frac{2\alpha^2}{\alpha^2-1}I_n & -\frac{2\alpha}{\alpha^2-1}I_n \\ -\frac{2\alpha}{\alpha^2-1}I_n & \frac{2\alpha^2}{\alpha^2-1}I_n \end{bmatrix},$$

for $\alpha \in [1,\infty)$.

Since $C$ is positive-definite, we can define a Gaussian density kernel as

$$k(\mathbf{x},\mathbf{y},C) = \frac{1}{(\frac{\pi\sqrt{\alpha^2-1}}{2\alpha})^n} \exp\big(-\frac{\alpha^2}{\alpha^2-1}\|\mathbf{x}\|^2 - \frac{\alpha^2}{\alpha^2-1}\|\mathbf{y}\|^2 + \frac{2\alpha}{\alpha^2-1}\mathbf{y}\cdot\mathbf{x}^T\big). \tag{A.49}$$

Moreover, we can calculate the integration over all values of $\mathbf{y}$ as

$$\int k(\mathbf{x},\mathbf{y},C)\partial\mathbf{y} = \int \frac{\exp\big(-\|\mathbf{x}\|^2\big)}{(\frac{\pi\sqrt{\alpha^2-1}}{2\alpha})^n} \exp\big(-\frac{\|\alpha\mathbf{y}-\mathbf{x}\|^2}{(\alpha^2-1)}\big)\partial\mathbf{y} = \frac{\exp\big(-\|\mathbf{x}\|^2\big)}{(\pi)^{n/2}}. \tag{A.50}$$

Next, by setting $w(\mathbf{x}) = \frac{1}{\int k(\mathbf{x},\mathbf{y},C)\partial\mathbf{y}}$ and using Theorem 15, we have

$$\int \frac{1}{(\pi\frac{\alpha^2-1}{\alpha^2})^{n/2}} \psi_{\mathbf{k}}(\mathbf{x}) \exp\big(-\frac{\|\alpha\mathbf{y}-\mathbf{x}\|^2}{\alpha^2-1}\big)\partial\mathbf{x} = \lambda_{\mathbf{k}}\psi_{\mathbf{k}}(\mathbf{y}). \tag{A.51}$$

Now to find the eigenfunctions of the Gaussian kernel $k'(\mathbf{x},\mathbf{y}) = \exp\big(-\frac{\alpha\|\mathbf{x}-\mathbf{y}\|^2}{(\alpha^2-1)}\big)$, we let $\psi'_{\mathbf{k}}(\mathbf{x}) = \psi_{\mathbf{k}}(\mathbf{x})\exp\big(\frac{\alpha}{\alpha+1}\|\mathbf{x}\|^2\big)$ and let the weight function be

$$w'(\mathbf{x}) = (\pi)^{n/2}\exp\big(-\frac{(\alpha-1)}{\alpha+1}\|\mathbf{x}\|^2\big).$$

As a result of such assumptions, we see that

$$\int \psi'_{\mathbf{k}}(\mathbf{x})k'(\mathbf{x},\mathbf{y})w'(\mathbf{x})\partial\mathbf{x}$$

$$= \int (\pi)^{n/2}\psi_{\mathbf{k}}(\mathbf{x})\exp\big(-\frac{1}{\alpha^2-1}\|\mathbf{x}\|^2 - \frac{\alpha}{\alpha^2-1}\|\mathbf{y}\|^2 + \frac{2\alpha}{\alpha^2-1}\mathbf{x}\cdot\mathbf{y}^T\big)\partial\mathbf{x} \tag{A.52}$$

$$= (\pi)^{n/2}\exp\big(\frac{\alpha}{\alpha+1}\|\mathbf{y}\|^2\big)\int \psi_{\mathbf{k}}(\mathbf{x})\exp\big(-\frac{\|\alpha\mathbf{y}-\mathbf{x}\|^2}{\alpha^2-1}\big)\partial\mathbf{x} \tag{A.53}$$

$$\overset{(a)}{=} (\pi)^{n/2}\exp\big(\frac{\alpha}{\alpha+1}\|\mathbf{y}\|^2\big)\sqrt{\lambda_{\mathbf{k}}}\psi_{\mathbf{k}}(\mathbf{y})\Big(\frac{\pi(\alpha^2-1)}{\alpha^2}\Big)^{n/2} \tag{A.54}$$

$$\overset{(b)}{=} (\pi)^n\Big(\frac{\alpha^2-1}{\alpha^2}\Big)^{n/2}\lambda_{\mathbf{k}}\psi'_{\mathbf{k}}(\mathbf{y}), \tag{A.55}$$

where $(a)$ holds because of eq. (A.51), and $(b)$ is followed by the definition of $\psi'_{\mathbf{k}}(\mathbf{y})$. As a result, $\psi'_{\mathbf{k}}(\mathbf{x})$ is an eigenfunction of the integral operator with kernel $k'(\mathbf{x},\mathbf{y})$ and with weight function $w'(\mathbf{x})$.

Equation eq. (A.55) shows that the eigenvalue of $k'(\mathbf{x},\mathbf{y})$ corresponding to $\psi_{\mathbf{k}}(\mathbf{x})$ is as

$$\lambda'_{\mathbf{k}} = (\pi)^n\Big(\frac{\alpha^2-1}{\alpha^2}\Big)^{n/2}\lambda_{\mathbf{k}} \tag{A.56}$$

Now we show that such eigenfunctions are orthonormal. Deploying the idea in eq. (A.55), for two eigenfunctions $\psi'_{\mathbf{k}}(\cdot)$ and $\psi'_{\mathbf{l}}(\cdot)$ for fixed vectors $\mathbf{k},\mathbf{l}\in\mathbb{N}^n$, we have

$$\int \psi'_{\mathbf{k}}(\mathbf{y})\psi'_{\mathbf{l}}(\mathbf{y})w'(\mathbf{y})\partial\mathbf{y} \overset{(a)}{=} \int \psi_{\mathbf{k}}(\mathbf{y})\psi_{\mathbf{l}}(\mathbf{y})\frac{(\pi)^{n/2}}{\exp\big(-\|\mathbf{x}\|^2\big)}\partial\mathbf{y} \overset{(b)}{=} \int \psi_{\mathbf{k}}(\mathbf{y})\psi_{\mathbf{l}}(\mathbf{y})w(\mathbf{y}) \overset{(c)}{=} \delta[\mathbf{l}-\mathbf{k}], \tag{A.57}$$

where $(a)$ is followed by the definition of eigenfunctions $\psi'_{\mathbf{k}}(\cdot)$, $\psi'_{\mathbf{l}}(\cdot)$ and the definition of weight function $w'(\mathbf{x})$, $(b)$ is due to the definition of $w(\mathbf{x})$ and eq. (A.50), and $(c)$ holds because of orthonormality of $\psi_{\mathbf{k}}$s as a result of Theorem 15.

Further, in this case we have $C_{11}^{-1}C_{12} = \frac{1}{\alpha}I_n$, or equivalently $P = I_n$ and $\mathbf{e} = \frac{1}{\alpha}\mathbb{1}_n$. Hence, firstly using eq. (A.43), one can see that

$$\lambda_{\mathbf{k}} = \alpha^{-\|\mathbf{k}\|/2}. \tag{A.58}$$

Secondly, in finding symmetrized Kronecker power $\sigma_{\|k\|_1}(P)$ in eq. (A.44), for non-diagonal matrices $M$, the term $\prod_{ij}P_{ij}^{M_{ij}} = 0$. Further, for a diagonal matrix $M$, we have $M\mathbb{1}_n = \mathbb{1}_n M$. This induces the fact that

$$\sigma_{\|k\|_1}(P) = \begin{cases} 0 & \mathbf{k}\neq\mathbf{l}, \\ 1 & \mathbf{k}=\mathbf{l} \end{cases}. \tag{A.59}$$

This shows that

$$\psi_{\mathbf{l}}(\mathbf{x}) = \frac{\varphi_{\mathbf{l}}(\mathbf{x})}{\sqrt{\prod_{i=1}^{n} l_i!}}. \tag{A.60}$$

To find the formulation of eigenfunction $\psi_k(\mathbf{x})$, we can rewrite the term $\varphi_{\mathbf{l}}(\mathbf{x}, C_{11})$ in eq. (A.42) for $C_{11} = \frac{1}{2} I_n$ as

$$\varphi_{\mathbf{l}}(\mathbf{x}, I) = \frac{1}{(\pi)^{n/2}} \frac{\partial^{\|\mathbf{l}\|_1}}{\partial {x_1}^{l_1} \dots \partial {x_n}^{l_n}} \exp\Big( - \sum_{i=1}^{n} x_i^2 \Big). \tag{A.61}$$

We note that the exponential function can be written as the product of functions that are only dependent on one variable $x_i$ for $i \in [n]$. Hence, we can rephrase eq. (A.61) as a product of the derivative of each function as

$$\varphi_{\mathbf{l}}(\mathbf{x}, I) = \prod_{i=1}^{n} \frac{1}{\sqrt{\pi}} \frac{\partial^{l_i}}{\partial^{l_i} x_i} \exp\big( - x_i^2 \big). \tag{A.62}$$

As a result of this equation and the definition of Hermite functions in one dimension, we have

$$\varphi_{\mathbf{l}}(\mathbf{x}, I) = \frac{\exp(-\|\mathbf{x}\|^2)}{(\pi)^{n/2}} \prod_{i=1}^{n} H_{l_i}(x_i) \tag{A.63}$$

Hence, we can calculate $\psi'_{\mathbf{k}}(\mathbf{x})$ as

$$\psi'_{\mathbf{k}}(\mathbf{x}) = \frac{1}{\sqrt{(\pi)^n \prod_{i=1}^{n} k_i!}} \exp(\frac{-\|\mathbf{x}\|^2}{\alpha + 1}) \prod_{i=1}^{n} H_{k_i}(x_i). \tag{A.64}$$

Using above discussion, we see that $\mathbf{k}$-th element $[\Phi_N(\mathbf{x})]_{\mathbf{k}}$ of the tensor $\Phi_N(x)$, which is defined in the proposition statement, is equal to

$$[\Phi_N(\mathbf{x})]_{\mathbf{k}} = \sqrt{\lambda'_{\mathbf{k}}} \psi'_{\mathbf{k}}(\mathbf{x}). \tag{A.65}$$

This fact and Theorem 14 concludes that we can uniformly approximate $k'(\mathbf{x}, \mathbf{y})$ as

$$k'(\mathbf{x}, \mathbf{y}) = \langle \Phi_N(\mathbf{x}), \Phi_N(\mathbf{y}) \rangle. \tag{A.66}$$

Further, for any positive-definite matrix $A$, since the singular values of $\sqrt{\frac{\alpha^2-1}{\alpha}}\sqrt{A}$ are bounded, one can uniformly approximate $k''(\mathbf{x}, \mathbf{y}) := \exp\big( - (\mathbf{x} - \mathbf{y}) A (\mathbf{x} - \mathbf{y})^T \big) = k'\Big( \sqrt{\frac{\alpha^2-1}{\alpha}}\sqrt{A}\mathbf{x}, \sqrt{\frac{\alpha^2-1}{\alpha}}\sqrt{A}\mathbf{y} \Big)$ as

$$k''(\mathbf{x}, \mathbf{y}) \simeq \Big\langle \Phi_N\Big( \sqrt{\frac{\alpha^2 - 1}{\alpha}} \sqrt{A}\mathbf{x} \Big), \Phi_N\Big( \sqrt{\frac{\alpha^2 - 1}{\alpha}} \sqrt{A}\mathbf{y} \Big) \Big\rangle \tag{A.67}$$

□

## A.4 Sum-kernel upper-bound

Instead of using Generalized Hermite mean embedding which takes a huge amount of memory, one could use an upper bound to the joint Gaussian kernel. We use the inequality of arithmetic and geometric means to prove that.

$$k(\mathbf{x},\mathbf{y}) = \exp\Big(-\frac{1}{2l^2}(\mathbf{x}-\mathbf{y})^T(\mathbf{x}-\mathbf{y})\Big) = \exp(-\frac{1}{2l^2}\sum_{d=1}^{D}(x_d-y_d)^2\Big) \tag{A.68}$$

$$= \prod_{d=1}^{D}\exp\Big(-\frac{1}{2l^2}(x_d-y_d)^2\Big) \tag{A.69}$$

$$\overset{(a)}{\leq} \frac{1}{D}\sum_{d=1}^{D}\exp\Big(-\frac{D}{2l^2}(x_d-y_d)^2\Big) \tag{A.70}$$

$$= \frac{1}{D}\sum_{d=1}^{D}k_{X_d}(x_d,y_d), \tag{A.71}$$

where $(a)$ holds due to inequality of arithmetic and geometric means (AM-GM), and $k_{X_d}(\cdot,\cdot)$ is defined as

$$k_{X_d}(x_d,y_d) := \exp\Big(-\frac{D}{2l^2}(x_d-y_d)^2\Big). \tag{A.72}$$

Next, we approximate such kernel via an inner-product of the feature maps

$$\boldsymbol{\phi}_C(\mathbf{x}) = \begin{bmatrix} \boldsymbol{\phi}^{(C)}_{HP,1}(x_1)/\sqrt{D} \\ \boldsymbol{\phi}^{(C)}_{HP,2}(x_2)/\sqrt{D} \\ \vdots \\ \boldsymbol{\phi}^{(C)}_{HP,D}(x_D)/\sqrt{D} \end{bmatrix} \in \mathbb{R}^{((C+1)\cdot D)\times 1}. \tag{A.73}$$

Although such feature maps are not designed to catch correlation among dimensions, they provide us with a guarantee on marginal distributions as follows.

**Lemma 2.** *Define $k_{X_i}(\cdot,\cdot)$ as in eq. (A.72) and define $\phi_C(\mathbf{x})$ as in eq. (A.73). For $\varepsilon \in \mathbb{R}^+$, there exists N such that for $C \geq N$ we have*

- $\big\|\mathbb{E}_{\mathbf{x}\sim P}\big[\phi_C(\mathbf{x})\big] - \mathbb{E}_{\mathbf{y}\sim Q}\big[\phi_C(\mathbf{y})\big]\big\|_2 \leq \varepsilon \Rightarrow \mathrm{MMD}_{k_{X_i}}(P_i,Q_i) \leq \sqrt{D+1}\varepsilon$ *for every* $i \in \{1,\ldots,D\}$, *and*

- $\mathrm{MMD}_{k_{X_i}}(P_i, Q_i) \leq \varepsilon$ *for every* $i \in \{1,\dots,D\} \Rightarrow$
  $\left\|\mathbb{E}_{\mathbf{x}\sim P}\left[\phi_C(\mathbf{x})\right] - \mathbb{E}_{\mathbf{y}\sim Q}\left[\phi_C(\mathbf{y})\right]\right\| \leq \sqrt{2}\varepsilon,$

*where $P_i$ and $Q_i$ are marginal probability distributions corresponding to $P$ and $Q$, respectively.*

*Proof.* Since $\boldsymbol{\phi}^{(C)}_{HPi}(x_i)$ has the certain form as in Theorem 14, then Lemma 1 shows that we can use such feature maps to uniformly approximate the MMD in an RKHS based on the kernel $k_i(x_i, y_i) = \exp\left(-\frac{1}{2l^2}(x_i - y_i)^2\right)$. As a result, there exists $N$ such that for $C \geq N$, we have

$$\left|\left\|\mathbb{E}_{x_i\sim P_i}\left[\phi^{(C)}_{HP,i}(x_i)\right] - \mathbb{E}_{y_i\sim Q_i}\left[\phi^{(C)}_{HP,i}(y_i)\right]\right\|_2^2 - \mathrm{MMD}^2_{k_{X_i}}(P_i, Q_i)\right| \leq D\varepsilon^2. \tag{A.74}$$

Now we prove the first part. Knowing

$$\left\|\mathbb{E}_{\mathbf{x}\sim P}\left[\phi_C(\mathbf{x})\right] - \mathbb{E}_{\mathbf{y}\sim Q}\left[\phi_C(\mathbf{y})\right]\right\|_2 \leq \varepsilon, \tag{A.75}$$

and by the definition of $\phi_C(\cdot)$, we deduce that

$$\left\|\mathbb{E}_{x_i\sim P_i}\left[\phi^{(C)}_{HP,i}(x_i)\right] - \mathbb{E}_{y_i\sim Q_i}\left[\phi^{(C)}_{HP,i}(y_i)\right]\right\|_2^2 \leq \varepsilon^2. \tag{A.76}$$

Using this and eq. (A.74) we can prove the first part.

Inversely, by setting $\mathrm{MMD}_{k_{X_i}}(P_i, Q_i) \leq \varepsilon$ and eq. (A.74), one sees that

$$\left\|\mathbb{E}_{x_i\sim P_i}\left[\phi^{(C)}_{HP,i}(x_i)\right] - \mathbb{E}_{y_i\sim Q_i}\left[\phi^{(C)}_{HP,i}(y_i)\right]\right\|_2 \leq \sqrt{2}\varepsilon. \tag{A.77}$$

This coupled with the definition of $\Phi_C$ completes the second part of lemma. □

## A.5 $\phi$ Recursion

$$\begin{aligned}
\phi_{k+1}(x) &= ((1+\rho)(1-\rho))^{\frac{1}{4}} \frac{\rho^{\frac{k+1}{2}}}{\sqrt{2^{k+1}(k+1)!}} H_{k+1}(x) \exp\left(-\frac{\rho}{\rho+1}x^2\right), \quad \text{by definition} \\
&= ((1+\rho)(1-\rho))^{\frac{1}{4}} \frac{\rho^{\frac{k+1}{2}}}{\sqrt{2^{k+1}(k+1)!}} \left[2xH_k(x) - 2kH_{k-1}(x)\right] \exp\left(-\frac{\rho}{\rho+1}x^2\right), \\
&= \frac{\sqrt{\rho}}{\sqrt{2(k+1)}} 2x\phi_k(x) - \frac{\rho}{\sqrt{k(k+1)}} k\phi_{k-1}(x).
\end{aligned} \tag{A.78}$$

# A.6 Sensitivity of mean embeddings (MEs)

## A.6.1 Sensitivity of ME under the sum kernel

Here we derive the sensitivity of the mean embedding corresponding to the sum kernel.

$$S_{\widehat{\boldsymbol{\mu}}^s_P} = \max_{\mathcal{D},\mathcal{D}'} \|\widehat{\boldsymbol{\mu}}^s_P(\mathcal{D}) - \widehat{\boldsymbol{\mu}}^s_P(\mathcal{D}')\|_F = \max_{\mathcal{D},\mathcal{D}'} \|\tfrac{1}{m}\sum_{i=1}^{m} \widehat{v}_s(\mathbf{x}_i)\mathbf{f}(\mathbf{y}_i)^T - \tfrac{1}{m}\sum_{i=1}^{m} \widehat{v}_s(\mathbf{x}'_i)\mathbf{f}(\mathbf{y}'_i)^T\|_F$$

where $\|\cdot\|_F$ represents the Frobenius norm. Since $\mathcal{D}$ and $\mathcal{D}'$ are neighbouring, then $m-1$ of the summands on each side cancel and we are left with the only distinct datapoints, which we denote as $(\mathbf{x},\mathbf{y})$ and $(\mathbf{x}',\mathbf{y}')$. We then apply the triangle inequality and the definition of $\mathbf{f}$. As $\mathbf{y}$ is a one-hot vector, all but one column of $\widehat{v}_s(\mathbf{x})\mathbf{f}(\mathbf{y})^\top$ are 0, so we omit them in the next step:

$$\begin{aligned} S_{\boldsymbol{\mu}^s_P} &= \max_{(\mathbf{x},\mathbf{y}),(\mathbf{x}',\mathbf{y}')} \|\tfrac{1}{m}\widehat{v}_s(\mathbf{x})\mathbf{f}(\mathbf{y})^T - \tfrac{1}{m}\widehat{v}_s(\mathbf{x}')\mathbf{f}(\mathbf{y}')^T\|_F \\ &\leq \max_{(\mathbf{x},\mathbf{y})} \tfrac{2}{m}\|\widehat{v}_s(\mathbf{x})\mathbf{f}(\mathbf{y})^T\|_F = \max_{\mathbf{x}} \tfrac{2}{m}\|\widehat{v}_s(\mathbf{x})\|_2. \end{aligned} \tag{A.79}$$

We recall the definition of the feature map given in eq. (2.10),

$$\|\widehat{v}_s(\mathbf{x})\|_2 = \frac{1}{\sqrt{D}}\left(\sum_{d=1}^{D} \|\boldsymbol{\phi}^{(C)}_{HP,d}(x_d)\|_2^2\right)^{\frac{1}{2}}. \tag{A.80}$$

To bound $\|\widehat{v}_s(\mathbf{x})\|_2$, we first prove that $\|\boldsymbol{\phi}^{(C)}_{HP,d}(x_d)\|_2^2 \leq 1$. Using Mehler's formula (see eq. (2.5)), and by plugging in $y = x_d$, one can show that

$$1 = \exp\Big(-\frac{\rho}{1-\rho^2}(x_d - x_d)^2\Big) = \sum_{c=0}^{\infty} \lambda_c f_c(x_d)^2. \tag{A.81}$$

Using this, we rewrite the infinite sum in terms of the $C$th-order approximation and the rest of summands to show that

$$1 = \sum_{c=0}^{\infty} \lambda_c f_c^2(x_d) \overset{(a)}{=} \|\boldsymbol{\phi}^{(C)}_{HP,d}(x_d)\|_2^2 + \sum_{c=C+1}^{\infty} \lambda_c f_c^2(x) \overset{(b)}{\geq} \|\boldsymbol{\phi}^{(C)}_{HP,d}(x_d)\|_2^2, \tag{A.82}$$

where $(a)$ holds because of the definition of $\boldsymbol{\phi}^{(C)}_{HP,d}(x_d)$ in eq. (2.6): $\|\boldsymbol{\phi}^{(C)}_{HP,d}(x_d)\|_2^2 = \sum_{c=0}^{C} \lambda_c f_c^2(x_d)$, and $(b)$ holds, because $\lambda_c$ and $f_c^2(x)$ are non-negative scalars.

Finally, deploying eq. (A.79), eq. (A.80), and eq. (A.82), we bound the sensitivity as

$$S_{\boldsymbol{\mu}_P} \leq \max_{\mathbf{x}} \tfrac{2}{m}\|\widehat{v}_s(\mathbf{x})\|_2 \leq \tfrac{2}{m\sqrt{D}}\sqrt{D} = \tfrac{2}{m}. \tag{A.83}$$

### A.6.2 Sensitivity of ME under the product kernel

Similarly, we derive the sensitivity of the mean embedding corresponding to the product kernel.

$$S_{\widehat{\boldsymbol{\mu}}_P^p} = \max_{\mathcal{D},\mathcal{D}'} \|\widehat{\boldsymbol{\mu}}_P^p(\mathcal{D}) - \widehat{\boldsymbol{\mu}}_P^p(\mathcal{D}')\|_F \leq \max_{\mathbf{x}} \tfrac{2}{m} \|\widehat{v}_p(\mathbf{x}^{D_{prod}})\|_2$$

Given the definition in eq. (2.8), the L2 norm is given by

$$\tfrac{2}{m}\|\widehat{v}_p(\mathbf{x}^{D_{prod}})\|_2 = \tfrac{2}{m} \prod_{d=1}^{D_{prod}} \|\boldsymbol{\phi}_{HP}^{(C)}(x_d)\|_2, \tag{A.84}$$

$$\leq \tfrac{2}{m} \tag{A.85}$$

where the last line is due to eq. (A.82).

## A.7 Descriptions on the tabular datasets

In this section we give more detailed information about the tabular datasets used in our experiments. Unless otherwise stated, the datasets were obtained from the UCI machine learning repository Dua and Graff (2017).

**Adult**

Adult dataset contains personal attributes like age, gender, education, marital status or race from the different dataset participants and their respective income as the label (binarized by a threshold set to 50K). The dataset is publicly available at the UCI machine learning repository at the following link: `https://archive.ics.uci.edu/ml/datasets/adult`.

**Census**

The Census dataset is also a public dataset that can be downloaded via the SDGym package [1]. This is a clear example of an imbalaned dataset since only 12382 of the samples are considered positive out of a total of 199523 samples.

**Cervical**

The Cervical cancer dataset comprises demographic information, habits, and historic medical records of 858 patients and was created with the goal to identify the cervical cancer risk factors. The original dataset can be found at the following link: `https://archive.ics.uci.edu/ml/datasets/Cervical+cancer+%28Risk+Factors%29`.

**Covtype**

This dataset contains cartographic variables from four wilderness areas located in the Roosevelt National Forest of northern Colorado and the goal is to predict forest cover

[1] SDGym package website: `https://pypi.org/project/sdgym/`

type from the 7 possible classes. The data is publicly available at https://archive.ics.uci.edu/ml/datasets/covertype.

**Credit**

The Credit Card Fraud Detection dataset contains the categorized information of credit card transactions made by European cardholders during September 2013 and the corresponding label indicating if the transaction was fraudulent or not. The dataset can be found at: https://www.kaggle.com/mlg-ulb/creditcardfraud. The original dataset has a total number of 284807 samples where only 492 of them are frauds. In our experiments, we descarded the feature related to the time the transaction was done. The data is released under a Database Contents License (DbCL) v1.0.

**Epileptic**

The Epileptic Seizure Recognition dataset contains the brain activity measured in terms of the EEG across time. The dataset can be found at https://archive.ics.uci.edu/ml/datasets/Epileptic+Seizure+Recognition. The original dataset contains 5 different labels that we binarized into two: seizure (2300 samples) or not seizure (9200 samples).

**Intrusion**

The dataset was used for The Third International Knowledge Discovery and Data Mining Tools Competition held at the Conference on Knowledge Discovery and Data Mining, 1999, and can be found at http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. We used the file named "kddcup.data10percent.gz" that contains the 10% of the orginal dataset. The goal is to distinguish between intrusion/attack and normal connections categorized in 5 different labels.

**Isolet**

The Isolet dataset contains the features sounds as spectral coefficients, contour features, sonorant features, pre-sonorant features, and post-sonorant features of the different letters on the alphabet as inputs and the corresponding letter as the label. The original dataset can be found at https://archive.ics.uci.edu/ml/datasets/isolet. However, in our experiments we considered this dataset as a binary classification task as we only considered the labels as constants or vowels.

Table A.1 summarizes the number of samples, labeled classes and type of different inputs (numerical, ordinal or categorical) for each tabular dataset used in our experiments. The content of the table reflects the results after pre-processing or binarizing the corresponding datasets.

### A.7.1 Hyperparameters for discrete tabular datasets

Here we include the hyperparameters used in obtaining the results obtained in Table 2.1. In the main text we describe the choices of the Hermitian hyperparameters. In the separate section A.7.2 we present a broader view over the gamma hyperparameter.

Table A.1: Tabular datasets. Size, number of classes and feature types descriptions.

| dataset | # samps | # classes | # features |
|---|---|---|---|
| isolet | 4366 | 2 | 617 num |
| covtype | 406698 | 7 | 10 num, 44 cat |
| epileptic | 11500 | 2 | 178 num |
| credit | 284807 | 2 | 29 num |
| cervical | 753 | 2 | 11 num, 24 cat |
| census | 199523 | 2 | 7 num, 33 cat |
| adult | 48842 | 2 | 6 num, 8 cat |
| intrusion | 394021 | 5 | 8 cat, 6 ord, 26 num |

Table A.2: Hyperparameters for discrete tabular datasets

| | privacy | batch rate | order hermite prod | prod dimension | gamma | order hermite |
|---|---|---|---|---|---|---|
| Adult | $\varepsilon = 0.3$ | 0.1 | 10 | 5 | 1 | 100 |
| | $\varepsilon = 0.1$ | 0.1 | 5 | 7 | 1 | 100 |
| Census | $\varepsilon = 0.3$ | 0.01 | 5 | 7 | 0.1 | 100 |
| | $\varepsilon = 0.1$ | 0.01 | 5 | 7 | 0.1 | 100 |

### A.7.2 Gamma hyperparameter ablation study

Here we study the impact of gamma $\gamma$ hyperparameter on the quality of the generated samples. Gamma describes the weight that is given to the product kernel in relation to the sum kernel. We elaborate on the results from the Table 2.1 which describe $\alpha$-way marginals evaluated on generated samples with discretized Census dataset. We fix all the hyperparameters and vary gamma. The Table A.3 shows the impact of gamma. The $k-$way results remain indifferent for $\gamma \leq 1$ but deterioriate for $\gamma > 1$. In this experiment, we set $\varepsilon_1 = \varepsilon_2 = \varepsilon/2$. Here, "order hermite prod " means the HP order for the outer product kernel, "prod dimension" means the number of subsampled input dimensions, and "order hermite" means the HP order for the sum kernel.

Table A.3: The impact of gamma hyperparamer.

| epsilon | batch rate | order hermite prod | prod dimension | gamma | epochs | 3-way | 4-way |
|---|---|---|---|---|---|---|---|
| 0.3 | 0.1 | 10 | 5 | 0.001 | 8 | 0.474 | 0.570 |
| 0.3 | 0.1 | 10 | 5 | 0.01 | 8 | 0.473 | 0.570 |
| 0.3 | 0.1 | 10 | 5 | 0.1 | 8 | 0.499 | 0.597 |
| 0.3 | 0.1 | 10 | 5 | 1 | 8 | 0.474 | 0.570 |
| 0.3 | 0.1 | 10 | 5 | 10 | 8 | 0.585 | 0.671 |
| 0.3 | 0.1 | 10 | 5 | 100 | 8 | 0.674 | 0.757 |
| 0.3 | 0.1 | 10 | 5 | 1000 | 8 | 0.676 | 0.761 |

### A.7.3 Training DP-HP generator

Here we provide the details of the DP-HP training procedure we used on the tabular data experiments. Table A.4 shows the Hermite polynomial order, the fraction of dataset used in a batch, the number of epochs and the undersampling rate we used during training for each tabular dataset.

Table A.4: Tabular datasets. Hyperparameter settings for private constraints $\varepsilon = 1$ and $\delta = 10^{-5}$.

| data name | batch rate | order hermite prod | prod dimension | order hermite | gamma |
|---|---|---|---|---|---|
| adult | 0.1 | 5 | 5 | 100 | 0.1 |
| census | 0.5 | 5 | 5 | 100 | 0.1 |
| cervical | 0.5 | 13 | 5 | 20 | 1 |
| credit | 0.5 | 7 | 5 | 20 | 1 |
| epileptic | 0.1 | 5 | 7 | 20 | 0.1 |
| isolet | 0.5 | 13 | 5 | 150 | 1 |
| covtype | 0.01 | 7 | 2 | 10 | 1 |
| intrusion | 0.01 | 5 | 5 | 7 | 1 |

### A.7.4 Non-private results

We also show the non-private MERF and HP results in Table A.5.

Table A.5: Performance comparison on Tabular datasets. The average over five independent runs.

| | Real | | DP-MERF (non-priv) | | DP-HP (non-priv) | | DP-CGAN $(1, 10^{-5})$-DP | | DP-GAN $(1, 10^{-5})$-DP | | DP-MERF $(1, 10^{-5})$-DP | | **DP-HP** $(1, 10^{-5})$-DP | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **adult** | 0.786 | 0.683 | 0.642 | 0.525 | **0,673** | **0,621** | 0.509 | 0.444 | 0.511 | 0.445 | 0.642 | 0.524 | **0,688** | **0,632** |
| **census** | 0.776 | 0.433 | 0.696 | 0.244 | **0,707** | **0,32** | 0.655 | 0.216 | 0.529 | 0.166 | 0.685 | 0.236 | **0,699** | **0,328** |
| **cervical** | 0.959 | 0.858 | **0.863** | **0.607** | 0,823 | 0,574 | 0.519 | 0.200 | 0.485 | 0.183 | 0.531 | 0.176 | **0,616** | **0,312** |
| **credit** | 0.924 | 0.864 | **0.902** | 0.828 | 0.89 | **0,863** | 0.664 | 0.356 | 0.435 | 0.150 | 0.751 | 0.622 | **0,786** | **0,744** |
| **epileptic** | 0.808 | 0.636 | 0.564 | 0.236 | **0,602** | **0,546** | 0.578 | 0.241 | 0.505 | 0.196 | 0.605 | 0.316 | **0,609** | **0,554** |
| **isolet** | 0.895 | 0.741 | 0.755 | 0.461 | **0,789** | **0,668** | 0.511 | 0.198 | 0.540 | 0.205 | 0.557 | 0.228 | **0,572** | **0,498** |
| | F1 | | F1 | | F1 | | F1 | | F1 | | F1 | | F1 | |
| **covtype** | 0.820 | | **0.601** | | 0.580 | | 0.285 | | 0.492 | | 0.467 | | **0.537** | |
| **intrusion** | 0.971 | | 0.884 | | **0.888** | | 0.302 | | 0.251 | | **0.892** | | 0.890 | |

### A.7.5 The effect of subsampled input dimensions for the product kernel on Adult dataset

Table A.6 shows the 3-way (Left) and 4-way (Right) marginals evaluated at different number of dimensions for the product kernel ($D_{prod}$) where the rest of hyperparameters are fixed. The results show that increasing the number of dimensions in the product kernel improved the result.

Table A.6: Trade-off for subsampling dimensions in the product kernel for Adult dataset.

| | $D_{prod}$ | | | $D_{prod}$ | | |
|---|---|---|---|---|---|---|
| $\varepsilon$ | 2 | 5 | 7 | 2 | 5 | 7 |
| 1 | 0.367 | 0.34 | **0.332** | 0.466 | 0.434 | **0.422** |

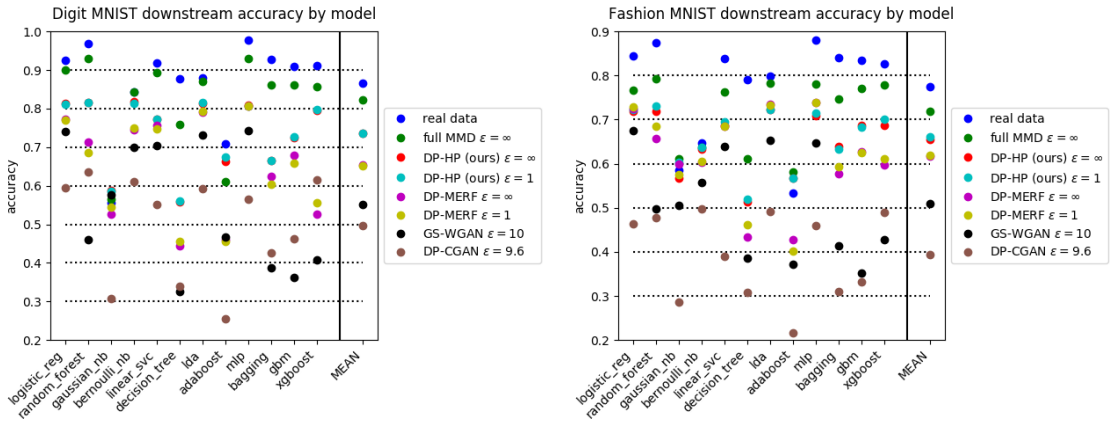
## A.8 Image data

### A.8.1 Results by model

In the following we provide a more detailed description of the downstreams models accuracy over the different methods considered in the image datasets.

### A.8.2 MNIST and fashionMNIST hyper-parameter settings

Here we give a detailed hyper-parameter setup and the architectures used for generating synthetic samples via DP-HP for MNIST and FashionMNIST datasets in Table A.7. The




(a) MNIST (b) FashionMNIST

Figure A.2: We compare the real data test accuracy of models trained on synthetic data for various models: DP-HP, DP-MERF, GS-WGAN and DP-CGAN. As baselines we also include results for real training data and a generator, which is non-privately trained with MMD, listed as "full MMD". We show accuracy sorted by downstream classifier and the mean accuracy across classifiers on the right. Each score is the average of 5 independent runs.

non-private version of our method does not exhibit a significant accuracy difference between 2, 3 and 4 subsampled dimensions for the product kernel, so we considered product dimension to be 2 for memory savings. Table A.8 summarizes the 12 predictive models hyper-parameters setup for the image datasets trained on the generated samples via DP-HP. In this experiment, we optimize this loss $\min_{\boldsymbol{\theta}} ||\widehat{\boldsymbol{\mu}}^p_P - \widehat{\boldsymbol{\mu}}^p_{Q_{\boldsymbol{\theta}}}||_2^2 + \gamma||\widehat{\boldsymbol{\mu}}^s_P - \widehat{\boldsymbol{\mu}}^s_{Q_{\boldsymbol{\theta}}}||_2^2$, where $\gamma$ is multiplied by the sum kernel's loss.

Table A.7: Hyperparameter settings for image data experiments. All parameters not listed here are used with their default values.

| | MNIST | |
|---|---|---|
| | (non-priv) | $(1, 10^{-5})$-DP |
| Hermite order (sum kernel) | 100 | 100 |
| Hermite order (product kernel) | 20 | 20 |
| kernel length (sum kernel) | 0.005 | 0.005 |
| kernel length (product kernel) | 0.005 | 0.005 |
| product dimension | 2 | 2 |
| subsample product dimension | beginning of each epoch | beginning of each epoch |
| gamma | 5 | 20 |
| mini-batch size | 200 | 200 |
| epochs | 10 | 10 |
| learning rate | 0.01 | 0.01 |
| architecture | fully connected | fully connected |
| | FashionMNIST | |
| | (non-priv) | $(1, 10^{-5})$-DP |
| Hermite order (sum kernel) | 100 | 100 |
| Hermite order (product kernel) | 20 | 20 |
| kernel length (sum kernel) | 0.15 | 0.15 |
| kernel length (product kernel) | 0.15 | 0.15 |
| product dimension | 2 | 2 |
| subsample product dimension | beginning of each epoch | beginning of each epoch |
| gamma | 20 | 10 |
| mini-batch size | 200 | 200 |
| epochs | 10 | 10 |
| learning rate | 0.01 | 0.01 |
| architecture | CNN + bilinear upsampling | CNN + bilinear upsampling |

Table A.8: Hyperparameter settings for downstream models used in image data experiments. Models are taken from the scikit-learn 0.24.2 and xgboost 0.90 python packages and hyperparameters have been set to achieve reasonable accuracies while limiting runtimes. Paramters not listed are kept at their default values.

| Model | Parameters |
|---|---|
| Logistic Regression | solver: lbfgs, max_iter: 5000, multi_class: auto |
| Gaussian Naive Bayes | - |
| Bernoulli Naive Bayes | binarize: 0.5 |
| LinearSVC | max_iter: 10000, tol: 1e-8, loss: hinge |
| Decision Tree | class_weight: balanced |
| LDA | solver: eigen, n_components: 9, tol: 1e-8, shrinkage: 0.5 |
| Adaboost | n_estimators: 1000, learning_rate: 0.7, algorithm: SAMME.R |
| Bagging | max_samples: 0.1, n_estimators: 20 |
| Random Forest | n_estimators: 100, class_weight: balanced |
| Gradient Boosting | subsample: 0.1, n_estimators: 50 |
| MLP | - |
| XGB | colsample_bytree: 0.1, objective: multi:softprob, n_estimators: 50 |

# Appendix B

# Appendix II

## Notations

We employ the notations $L_{\text{def}}^{\mu_X}$, $L_{\text{def}}^{\mu_X\mu_{Y|X}}$, $L_{\text{def}}^{\mu_{XYM}}$ to indicate $L_{\text{def}}^{0-1}$ and stress on marginal, conditional, and joint probability measures on $X, Y$, and $M$. We further use $L_{0-1}^{\mu_X\mu_{Y|X}}$ to indicate zero-one loss $L_{0-1}$ and to represent the underlying probability measures on $X$ and $Y$. The cardinality of a set $\mathcal{A}$ is indicated by $|\mathcal{A}|$. The notation for the set of numbers from 1 to $K$ is: $[K] = \{1, \cdots, K\}$.

## B.1 Proof of Theorem 1

We first introduce some useful lemmas as below. In Lemma 3, we show that there exists a pair of hypothesis classes $(\mathcal{H}, \mathcal{R})$ such that for all non-atomic measures on $\mathcal{X}$ the deferral loss takes a fixed value. In Lemma 4, we use the aforementioned lemma to show that the difference of deferral losses for all two pairs of classifier/rejector $(h_1, r_1)$ and $(h_2, r_2)$ is bounded by the difference of two deferral losses with atomic measures on $\mathcal{X}$. In Lemma 5, we upper-bound the difference of two deferral losses for pairs of classifier/rejector that are obtained by staged and joint learning and on hypothesis classes that are defined in Lemma 3. Such upper-bound is in terms of expected loss of an optimal classifier on a certain hypothesis class. In Lemma 6, we further calculate the optimal expected loss on such classes. In Lemma 7, we show that on a set of events with size $n$, we could find a subset with size $a$ and probability at most $\frac{a}{n}$. Next, we uses these lemmas in the main proof of theorem.

**Lemma 3.** *For a probability measure $\mu_X$ with no atomic component on $\mathcal{X}$, hypothesis class $\mathcal{H}$ such that for every $h \in \mathcal{H}$, we have $|\{\mathbf{x} : h(\mathbf{x}) = 1\}| \leq d(\mathcal{H})$, and hypothesis class $\mathcal{R}$ such that for every $r \in \mathcal{R}$, we have $|\{\mathbf{x} : r(\mathbf{x}) = 1\}| \leq d(\mathcal{R})$, for every choice of $(h, r) \in \mathcal{H} \times \mathcal{R}$, the loss*

$$L_{\text{def}}^{0-1}(h, r) = \mathbb{E}_{X,Y,M}[\mathbb{I}_{h(X)\neq Y}\mathbb{I}_{r(X)=0} + \mathbb{I}_{M\neq Y}\mathbb{I}_{r(X)=1}],$$

*takes a constant value.*

*Proof.* Firstly, we know that

$$L_{\text{def}}^{0-1}(h,r) = \mathbb{E}_{X,Y,M}[\mathbb{I}_{h(X)\neq Y}\mathbb{I}_{r(X)=0}] + \mathbb{E}_{X,Y,M}[\mathbb{I}_{M\neq Y}\mathbb{I}_{r(X)=1}]. \tag{B.1}$$

Since probability measure of the set $\{x : r(x) = 1\}$ is zero in the absence of atomic components in $\mu_X$, one can show that $\Pr(r(X) = 1) = 0$ (, and equivalently $\Pr(r(X) = 0) = 1$). This fact together with ((B.1)) concludes that

$$L_{\text{def}}^{0-1}(h,r) = \mathbb{E}_{X,Y}[\mathbb{I}_{h(X)\neq Y}]. \tag{B.2}$$

Further, we have

$$\begin{aligned}\mathbb{E}_{X,Y}[\mathbb{I}_{h(X)\neq Y}] &= \mathbb{E}_{X,Y}[\mathbb{I}_{h(X)\neq Y}|h(X)=0]\Pr(h(X)=0) \\ &\quad + \mathbb{E}_{X,Y}[\mathbb{I}_{h(X)\neq Y}|h(X)=1]\Pr(h(X)=1) &\text{(B.3)}\\ &\overset{(a)}{=} \mathbb{E}_{X,Y}[\mathbb{I}_{Y=1}], &\text{(B.4)}\end{aligned}$$

where $(a)$ holds because probability measure of $\{\mathbf{x} : h(\mathbf{x}) = 1\}$ is zero in the absence of atomic components in the measure, that concludes $\Pr(h(X) = 0) = 1 - \Pr(h(X) = 1) = 1$. The proof is complete by ((B.2)) and ((B.4)). □

**Lemma 4.** *Let $\mu_X$ be a probability measure on $\mathcal{X}$, and let $\mathcal{H}$ and $\mathcal{R}$ be hypothesis classes as in Lemma 3. Further, let $h_1, h_2 \in \mathcal{H}$ and $r_1, r_2 \in \mathcal{R}$. Then, we have*

$$\left|L_{\text{def}}^{\mu_X}(h_1,r_1) - L_{\text{def}}^{\mu_X}(h_2,r_2)\right| \leq \left|L_{\text{def}}^{\mu_d}(h_1,r_1) - L_{\text{def}}^{\mu_d}(h_2,r_2)\right|, \tag{B.5}$$

*where $\mu_d$ is pure atomic (discrete) probability measure on $\mathcal{X}$.*

*Proof.* We know that for probability measure $\mu_X$, there exists $p \in [0,1]$ and probability measures $\mu_d$ and $\mu_{cs}$, such that

$$\mu_X = p\mu_d + (1-p)\mu_{cs}, \tag{B.6}$$

where $\mu_d$ is pure atomic and $\mu_{cs}$ has no atomic components. As a result, for every function $f(\cdot) : \mathcal{X} \to \mathbb{R}$, we have

$$\mathbb{E}_{X\sim\mu_X}\big[f(X)\big] = p\mathbb{E}_{x\sim\mu_d}\big[f(X)\big] + (1-p)\mathbb{E}_{x\sim\mu_d}\big[f(X)\big]. \tag{B.7}$$

With the same reasoning, we have

$$L_{\text{def}}^{\mu_X}(h,r) = pL_{\text{def}}^{\mu_d}(h,r) + (1-p)L_{\text{def}}^{\mu_{cs}}(h,r). \tag{B.8}$$

Next, we have

$$L_{\text{def}}^{\mu_X}(h_1,r_1) - L_{\text{def}}^{\mu_X}(h_2,r_2) = p\big[L_{\text{def}}^{\mu_d}(h_1,r_1) - L_{\text{def}}^{\mu_d}(h_2,r_2)\big] + (1-p)\big[L_{\text{def}}^{\mu_{cs}}(h_1,r_1) - L_{\text{def}}^{\mu_{cs}}(h_2,r_2)\big] \tag{B.9}$$

$$\overset{(a)}{=} p\big[L_{\text{def}}^{\mu_d}(h_1,r_1) - L_{\text{def}}^{\mu_d}(h_2,r_2)\big], \tag{B.10}$$

where $(a)$ holds because of Lemma 3 that proves $L_{\text{def}}^{\mu_{cs}}(h,r)$ is constant for every $(h,r) \in \mathcal{R}\times\mathcal{H}$.

Finally, using ((B.10)), and since $p \in [0,1]$, the proof is complete. □

**Lemma 5.** *Let* $\text{supp}(h) = \max_{\mathcal{S}:\forall x\in\mathcal{S}, h(x)=1} |\mathcal{S}|$ *and* $\mathcal{H}_d = \{h : \mathcal{X} \to \{0,1\} \,|\, \text{supp}(h) \le d\}$. *Further, let* $\mu_X$ *be an atomic measure on* $\mathcal{X}$*, and define*

$$\hat{h} := \underset{h\in\mathcal{H}_{d(\mathcal{H})}}{\text{argmin}}\, L_{0-1}^{\mu_X\mu_{Y|X}}(h), \tag{B.11}$$

*where*

$$L_{0-1}^{\mu_X\mu_{Y|X}}(h) = \mathbb{E}_{\mu_X\mu_{Y|X}}\big[\mathbb{I}_{h(X)\neq Y}\big], \tag{B.12}$$

*and*

$$\hat{r} := \underset{r\in\mathcal{H}_{d(\mathcal{R})}}{\text{argmin}}\, L_{\text{def}}^{\mu_X}(\hat{h},r). \tag{B.13}$$

*Further, define the pair* $(h^*,r^*)$ *be the optimal classifier*

$$(h^*,r^*) = \underset{(h,r)\in\mathcal{H}_{d(\mathcal{H})}\times\mathcal{H}_{d(\mathcal{R})}}{\text{argmin}}\, L_{\text{def}}^{\mu_X\mu_{Y|X}}(h,r). \tag{B.14}$$

*Then, if* $d(\mathcal{H}) \ge d(\mathcal{R})$*, we have*

$$L_{\text{def}}^{\mu_X\mu_{Y|X}}(\hat{h},\hat{r}) - L_{\text{def}}^{\mu_X\mu_{Y|X}}(h^*,r^*) \le \min_{h\in\mathcal{H}_{d(\mathcal{H})-d(\mathcal{R})}} L_{0-1}^{\mu'_X\mu_{Y|X}}(h) - \min_{h\in\mathcal{H}_{d(\mathcal{H})}} L_{0-1}^{\mu'_X\mu_{Y|X}}(h), \tag{B.15}$$

*where* $\mu'_X$ *is a purely atomic measure on* $\mathcal{X}$.

*Proof.* Firstly, using ((B.13)), we know that

$$L_{\text{def}}^{\mu_X\mu_{Y|X}}(\hat{h},\hat{r}) \le L_{\text{def}}^{\mu_X\mu_{Y|X}}(\hat{h},r^*). \tag{B.16}$$

Hence, we have

$$\underbrace{L_{\text{def}}^{\mu_X\mu_{Y|X}}(\hat{h},\hat{r}) - L_{\text{def}}^{\mu_X\mu_{Y|X}}(h^*,r^*)}_{D} \leq L_{\text{def}}^{\mu_X\mu_{Y|X}}(\hat{h},r^*) - L_{\text{def}}^{\mu_X\mu_{Y|X}}(h^*,r^*) \tag{B.17}$$

$$= \mathbb{E}\big[\mathbb{I}_{r^*(X)=0}\mathbb{I}_{\hat{h}(X)\neq Y}\big] - \mathbb{E}\big[\mathbb{I}_{r^*(X)=0}\mathbb{I}_{h^*(X)\neq Y}\big]. \tag{B.18}$$

Next, we form the conditional probability measure $\mu_X' = \mu_{X|r^*(X)=0}$. Therefore, using ((B.18)) we have

$$D = \Pr\big(r^*(X)=0\big)\big[L_{0-1}^{\mu_X'\mu_{Y|X}}(\hat{h}) - L_{0-1}^{\mu_X'\mu_{Y|X}}(h^*)\big]. \tag{B.19}$$

Next, since $h^* \in \mathcal{H}_{d(\mathcal{H})}$, we know that

$$L_{0-1}^{\mu_X'\mu_{Y|X}}(h^*) \geq \min_{h\in\mathcal{H}_{d(\mathcal{H})}} L_{0-1}^{\mu_X'\mu_{Y|X}}(h). \tag{B.20}$$

Moreover, we prove that

$$L_{0-1}^{\mu_X'\mu_{Y|X}}(\hat{h}) \leq \min_{h\in\mathcal{H}_{d(\mathcal{H})-d(\mathcal{R})}} L_{0-1}^{\mu_X'\mu_{Y|X}}(h). \tag{B.21}$$

We prove this inequality by contradiction. If ((B.21)) is not correct, then there exists $h' \in \mathcal{H}_{d(\mathcal{H})-d(\mathcal{R})}$, such that

$$L_{0-1}^{\mu_X'\mu_{Y|X}}(h') < L_{0-1}^{\mu_X'\mu_{Y|X}}(\hat{h}). \tag{B.22}$$

Then, we define a function $h'' : \mathcal{X} \to \{0,1\}$ as below

$$h''(\mathbf{x}) = \begin{cases} h'(\mathbf{x}) & r^*(\mathbf{x}) = 0 \\ \hat{h}(\mathbf{x}) & r^*(\mathbf{x}) = 1 \end{cases}. \tag{B.23}$$

Using the definition of $\mathcal{H}_d$ and since $\text{supp}(r^*) \leq d(\mathcal{R})$, one could show that $h'' \in \mathcal{H}_{d(\mathcal{H})}$. Furthermore, we have

$$L_{0-1}^{\mu_X\mu_{Y|X}}(h'') = \Pr\big(r^*(X)=0\big)L_{0-1}^{\mu_X'\mu_{Y|X}}(h') + \Pr\big(r^*(X)=1\big)L_{0-1}^{\mu_{X|r^*(X)=1}\mu_{Y|X}}(\hat{h}) \tag{B.24}$$

$$\overset{(a)}{<} \Pr\big(r^*(X)=0\big)L_{0-1}^{\mu_X'\mu_{Y|X}}(\hat{h}) + \Pr\big(r^*(X)=1\big)L_{0-1}^{\mu_{X|r^*(X)=1}\mu_{Y|X}}(\hat{h}) \tag{B.25}$$

$$= L_{0-1}^{\mu_X\mu_{Y|X}}(\hat{h}), \tag{B.26}$$

where $(a)$ holds using ((B.22)), and ((B.26)) and $\hat{h} \in \mathcal{H}_{d(\mathcal{H})}$ is a contradiction of ((B.11)).

Using ((B.19)), ((B.20)), ((B.21)), and since $\Pr(r^*(X)=0) \leq 1$, the proof is complete. □

**Lemma 6.** *Let $\mu_X$ be a purely atomic measure on $\mathcal{X}$. Further, let $\{\mathbf{x}_{i,1}\}_i$ be the points in $\mathcal{X}$ for which we have*

$$Pr(Y=1|X=\mathbf{x}_{i,1}) \leq Pr(Y=0|X=\mathbf{x}_{i,1}), \tag{B.27}$$

*and without loss of generality, assume that $\{\mathbf{x}_{i,2}\}$ are the points for which we have*

$$Pr(Y=1|X=\mathbf{x}_{i,2}) > Pr(Y=0|X=\mathbf{x}_{i,2}), \tag{B.28}$$

*and if $i<j$, then we have*

$$\begin{aligned} &Pr(X=\mathbf{x}_{i,2})\big[Pr(Y=1|X=\mathbf{x}_{i,2}) - Pr(Y=0|X=\mathbf{x}_{i,2})\big] \\ &\geq Pr(X=\mathbf{x}_{j,2})\big[Pr(Y=1|X=\mathbf{x}_{j,2}) - Pr(Y=0|X=\mathbf{x}_{j,2})\big]. \end{aligned} \tag{B.29}$$

*Then, we have*

$$\begin{aligned} \min_{h\in\mathcal{H}_d} L_{0-1}^{\mu_X\mu_{Y|X}}(h) = &\sum_i Pr(\mathbf{x}_{i,1})Pr(Y=1|X=\mathbf{x}_{i,1}) + \sum_{i=1}^{d} Pr(\mathbf{x}_{i,2})Pr(Y=0|X=\mathbf{x}_{i,2}) \\ &+ \sum_{i=d+1}^{\infty} Pr(\mathbf{x}_{i,2})Pr(Y=1|X=\mathbf{x}_{i,2}), \end{aligned} \tag{B.30}$$

*where $\mathcal{H}_d$ is defined as in Lemma 5.*

*Proof.* Let $h^*$ be the optimal classifier

$$h^* = \operatorname*{argmin}_{h\in\mathcal{H}_d} L_{0-1}^{\mu_X\mu_{Y|X}}(h). \tag{B.31}$$

Then, firstly, either $h(\mathbf{x}_{i,1})=0$, or $\Pr(Y=1|X=\mathbf{x}_{i,1}) = \Pr(Y=0|X=\mathbf{x}_{i,1})$ for all $i$s. We prove this claim by contradiction. If for some $i$, we have $h(\mathbf{x}_{i,1})=1$, and $\Pr(Y=1|X=\mathbf{x}_{i,1}) \neq \Pr(Y=0|X=\mathbf{x}_{i,1})$, then we could define $h'$ as

$$h'(\mathbf{x}) = \begin{cases} h^*(\mathbf{x}) & \mathbf{x}\neq\mathbf{x}_{i,1} \\ 0 & \mathbf{x}=\mathbf{x}_{i,1} \end{cases}. \tag{B.32}$$

One could see that $h'\in\mathcal{H}_d$, and that

$$L_{0-1}^{\mu_X\mu_{Y|X}}(h') - L_{0-1}^{\mu_X\mu_{Y|X}}(h^*) = \Pr(\mathbf{x}_{i,1})\big[\Pr(Y=1|X=\mathbf{x}_{i,1}) - \Pr(Y=0|X=\mathbf{x}_{i,1})\big] \overset{(a)}{<} 0, \tag{B.33}$$

where $(a)$ holds by ((B.27)) and since $\big[\Pr(Y=1|X=\mathbf{x}_{i,1}) \neq \Pr(Y=0|X=\mathbf{x}_{i,1})\big]$. The inequality ((B.33)) has contradiction with ((B.31)).

As a result, by forming a set $\mathcal{S}$ of indices $i$ for which $h^*(\mathbf{x}_{i,1}) = 1$, we have

$$\begin{aligned}
&\min_{h\in\mathcal{H}_d} L_{0-1}^{\mu_X\mu_{Y|X}}(h) \\
&\quad= \sum_{i\notin\mathcal{S}} \Pr(\mathbf{x}_{i,1})\Pr(Y=1|X=\mathbf{x}_{i,1}) + \sum_{i\in\mathcal{S}} \Pr(\mathbf{x}_{i,1})\Pr(Y=0|X=\mathbf{x}_{i,1}) \\
&\qquad+ \sum_i \Pr(\mathbf{x}_{i,2})\Pr(Y=1|X=\mathbf{x}_{i,2}) \\
&\qquad+ \min_{|S|} \min_{h\in\mathcal{H}_{d-|\mathcal{S}|}} \sum_i \mathbb{I}_{h(\mathbf{x}_{i,2})=1}\Pr(\mathbf{x}_{i,2})\big[\Pr(Y=0|X=\mathbf{x}_{i,2}) - \Pr(Y=1|X=\mathbf{x}_{i,2})\big] \qquad \text{(B.34)} \\
&\quad\overset{(a)}{=} \sum_i \Pr(\mathbf{x}_{i,1})\Pr(Y=1|X=\mathbf{x}_{i,1}) + \sum_i \Pr(\mathbf{x}_{i,2})\Pr(Y=1|X=\mathbf{x}_{i,2}) \\
&\qquad+ \min_{h\in\mathcal{H}_d} \sum_i \mathbb{I}_{h(\mathbf{x}_{i,2})=1}\Pr(\mathbf{x}_{i,2})\big[\Pr(Y=0|X=\mathbf{x}_{i,2}) - \Pr(Y=1|X=\mathbf{x}_{i,2})\big] \qquad \text{(B.35)} \\
&\quad\overset{(b)}{=} \sum_i \Pr(\mathbf{x}_{i,1})\Pr(Y=1|X=\mathbf{x}_{i,1}) + \sum_i \Pr(\mathbf{x}_{i,2})\Pr(Y=1|X=\mathbf{x}_{i,2}) \\
&\qquad- \max_{\mathcal{P}:|\mathcal{P}|\le d} \sum_{i\in\mathcal{P}} \Pr(\mathbf{x}_{i,2})\big[\Pr(Y=0|X=\mathbf{x}_{i,2}) - \Pr(Y=1|X=\mathbf{x}_{i,2})\big] \qquad \text{(B.36)} \\
&\quad\overset{(c)}{=} \sum_i \Pr(\mathbf{x}_{i,1})\Pr(Y=1|X=\mathbf{x}_{i,1}) + \sum_i \Pr(\mathbf{x}_{i,2})\Pr(Y=1|X=\mathbf{x}_{i,2}) \\
&\qquad+ \sum_{i=1}^{d} \Pr(\mathbf{x}_{i,2})\big[\Pr(Y=0|X=\mathbf{x}_{i,2}) - \Pr(Y=1|X=\mathbf{x}_{i,2})\big], \qquad \text{(B.37)}
\end{aligned}$$

where $(a)$ holds using that for $i\in\mathcal{S}$ we have $\big[\Pr(Y=1|X=\mathbf{x}_{i,1}) = \Pr(Y=0|X=\mathbf{x}_{i,1})\big]$, and since $\mathcal{H}_{d-|\mathcal{S}|}\subseteq\mathcal{H}_d$. Further, $(b)$ holds by the definition of $\mathcal{H}_d$ in which $\text{supp}(h)$ is assumed to be bounded by $d$, and, $(c)$ holds using the assumption ((B.29)). Finally, one could see that ((B.37)) is equal to ((B.30)). □

**Lemma 7.** *For an ordered probability mass function*

$$p_1 \le p_2 \le \ldots \le p_n, \qquad \text{(B.38)}$$

*on a finite set, and for $a\in\{1,\ldots,n\}$, we have*

$$\sum_{i=1}^{a} p_i \le \frac{a}{n}. \qquad \text{(B.39)}$$

*Proof.* We prove this lemma by contradiction. Assume that

$$\sum_{i=1}^{a} p_i > a. \qquad \text{(B.40)}$$

Since $p_i$s are ordered, one could see that for all sets $\mathcal{S}_t \subseteq \{1,\dots,n\}$ with $|\mathcal{S}_t| = a$, we have

$$\sum_{i\in\mathcal{S}_t} p_i > \frac{a}{n}. \tag{B.41}$$

We know that $\binom{n}{a}$ number of such distinct sets exist. Hence, we have

$$\sum_{t=1}^{\binom{a}{n}} \sum_{i\in\mathcal{S}_t} p_i > \frac{a}{n}\binom{n}{a}. \tag{B.42}$$

Moreover, one could see that for each $i$, $p_i$ is repeated in LHS of ((B.42)) for $\binom{n-1}{a-1}$ times. Consequently, we see that

$$\binom{n-1}{a-1} = \sum_{j=1}^{\binom{n-1}{a-1}} \sum_{i=1}^{n} p_i > \frac{a}{n}\binom{n}{a} = \binom{n-1}{a-1}, \tag{B.43}$$

that is a contradiction. □

*Proof of Theorem 1.* We derive the lower- and upper-bound in two steps as follows.
• **Lower-bound**: To prove the lower-bound, for every hypothesis class $\mathcal{H}$ and $\mathcal{R}$, we design a distribution of $(x,y,m)$ such that

$$L^{0-1}_{\text{def}}(h^*, r^*) = 0, \tag{B.44}$$

while

$$L^{0-1}_{\text{def}}(\widetilde{h}, \widetilde{r}) = \frac{1}{d(\mathcal{H})+1}. \tag{B.45}$$

For every $d(\mathcal{H})+1$ samples $\mathbf{X} = (\mathbf{x}_1,\dots,\mathbf{x}_{d(\mathcal{H})+1})$, using the definition of VC dimension, we can find labels $\mathbf{y} = (y_1,\dots,y_{d(\mathcal{H})+1})$ such that no classifier $h \in \mathcal{H}$ can obtain them (i.e., there is no $h$ such that $h(\mathbf{x}_i) = y_i$, for $i \in [1 : d(\mathcal{H})+1]$). We set

$$p(\mathbf{x}_i) = \begin{cases} \frac{1+\varepsilon}{d(\mathcal{H})+1} & i = 1 \\ \frac{1}{d(\mathcal{H})+1} & i \in [2 : d(\mathcal{H})] \\ \frac{1-\varepsilon}{d(\mathcal{H})+1} & i = d(\mathcal{H})+1 \end{cases}, \tag{B.46}$$

$$p(y|\mathbf{x}_i) = \begin{cases} 1 & y = y_i, \\ 0 & o.w. \end{cases}, \tag{B.47}$$

and

$$p(m|\mathbf{x}_i, y) = \begin{cases} 1 & m = y_i, i = 1, \\ 1 & m = 1 - y_i, i = d(\mathcal{H}) + 1, \\ 0 & o.w. \end{cases} . \tag{B.48}$$

If we train $\hat{h}$ and $\hat{r}$ separately, it means

$$\hat{h} = \operatorname*{argmin}_{h \in \mathcal{H}} \mathbb{E}_{(\mathbf{x},y,m)\sim p}[\mathbb{I}_{h(\mathbf{x})\neq y}] \tag{B.49}$$

$$= \operatorname*{argmin}_{h \in \mathcal{H}} \frac{1+\varepsilon}{d(\mathcal{H})+1} \mathbb{I}_{h(\mathbf{x}_1)\neq y_1} + \sum_{i=2}^{d(\mathcal{H})} \frac{1}{d(\mathcal{H})+1} \mathbb{I}_{h(\mathbf{x}_i)\neq y_i} + \frac{1-\varepsilon}{d(\mathcal{H})+1} \mathbb{I}_{h(\mathbf{x}_{d(\mathcal{H})+1})\neq y_{d(\mathcal{H})+1}}. \tag{B.50}$$

By the definition of $\mathbf{y}$, we know that at least one of the terms in the RHS of ((B.50)) is non-zero. In such case, for every subset $T$ of $[1 : d(\mathcal{H}) + 1]$ of size $d(\mathcal{H})$, one can find $h \in \mathcal{H}$, such that $h(\mathbf{x}_i) = y_i$ for $i \in T$. Hence, to minimize RHS of ((B.50)), we should have $\hat{h}(\mathbf{x}_i) \neq y_i$ only for $i = d(\mathcal{H}) + 1$.

Further, $\hat{r}$ is obtained as

$$\hat{r} = \operatorname*{argmin}_{r \in \mathcal{R}} \mathbb{E}_{(\mathbf{x},y,m)\sim p}[\mathbb{I}_{\hat{h}(\mathbf{x})\neq y} \mathbb{I}_{r(\mathbf{x})=0} + \mathbb{I}_{m\neq y} \mathbb{I}_{r(\mathbf{x})=1}]. \tag{B.51}$$

By the definition of $p(m|y, \mathbf{x})$ and $\hat{h}$, we can rewrite ((B.51)) as

$$\hat{r} = \operatorname*{argmin}_{r \in \mathcal{R}} \frac{1-\varepsilon}{d(\mathcal{H})+1} \mathbb{I}_{r(\mathbf{x}_{d+1})=1} + \frac{1-\varepsilon}{d(\mathcal{H})+1} \mathbb{I}_{r(\mathbf{x}_{d+1})=0}. \tag{B.52}$$

One can see that by any choice of $\hat{r}(\cdot)$, we have

$$L_{\text{def}}^{0-1}(\hat{h}, \hat{r}) = \frac{1-\varepsilon}{d(\mathcal{H})+1}. \tag{B.53}$$

Finally, by the arbitrariness of $\varepsilon$, we have

$$L_{\text{def}}^{0-1}(\hat{h}, \hat{r}) = \frac{1}{d(\mathcal{H})+1}. \tag{B.54}$$

Further, we prove that $L_{\text{def}}^{0-1}(h^*, r^*) = 0$ by constructing $h^*$ and $r^*$. Since $d(\mathcal{R}) \geq 2$, we can shatter $\{\mathbf{x}_1, \mathbf{x}_{d(\mathcal{H})+1}\}$ by $\mathcal{R}$, which means that there exists $r^* \in \mathcal{R}$ such that $r(\mathbf{x}_1) = 1$,

and $r(\mathbf{x}_{d(\mathcal{H})+1}) = 0$. As a result, we have

$$L_{\text{def}}^{0-1}(h^*, r^*) = \sum_{i=2}^{d(\mathcal{H})} \frac{1}{d(\mathcal{H})+1} \mathbb{I}_{r^*(\mathbf{x}_i)=0} \mathbb{I}_{h^*(\mathbf{x}_i)\neq y_i} + \frac{1-\varepsilon}{d(\mathcal{H})+1} \mathbb{I}_{h^*(\mathbf{x}_{d(\mathcal{H})+1})\neq y_i}. \tag{B.55}$$

Since VC dimension of $\mathcal{H}$ is $d(\mathcal{H})$, we can find $h^*$ such that all terms in the RHS of ((B.55)) is zero. Hence, we have $L_{\text{def}}^{0-1}(h^*, r^*) = 0$, that completes the proof.

• **Upper-bound**: For $d(\mathcal{H}) \leq d(\mathcal{R})$ the upper-bound is trivial. Then, we asssume $d(\mathcal{H}) > d(\mathcal{R})$. Let $D_{\mu_{XYM}}$ be

$$D_{\mu_{XYM}}^{\mathcal{H},\mathcal{R}} = L_{\text{def}}^{\mu_X \mu_{Y|X}}(\hat{h}, \hat{r}) - L_{\text{def}}^{\mu_X \mu_{Y|X}}(h^*, r^*). \tag{B.56}$$

To upper-bound $\inf_{\mathcal{H},\mathcal{R}} \sup_{\mu_{XYM}} D_{\mu_{XYM}}^{\mathcal{H},\mathcal{R}}$, we find a pair of hypothesis classes $\mathcal{H}$ and $\mathcal{R}$, such that for all joint probability measures $\mu_{XYM}$, we have $D_{\mu_{XYM}}^{\mathcal{H},\mathcal{R}} \leq \frac{d(\mathcal{R})}{d(\mathcal{H})}$.

We choose $\mathcal{H} = \mathcal{H}_{d(\mathcal{H})}$, and $\mathcal{R} = \mathcal{H}_{d(\mathcal{R})}$, where $\mathcal{H}_d$ is defined in Lemma 5. One could check that $VC(\mathcal{H}) = d(\mathcal{H})$, and $VC(\mathcal{R}) = d(\mathcal{R})$. Further, using Lemma 4, we know that $D_{\mu_{XYM}}^{\mathcal{H},\mathcal{R}}$ is bounded by $D_{\mu'_{XYM}}^{\mathcal{H},\mathcal{R}}$, in which $\mu'_X$ is purely atomic. For such measures, Lemma 5 proves that

$$D_{\mu_{XYM}}^{\mathcal{H},\mathcal{R}} \leq \min_{h\in\mathcal{H}_{d(\mathcal{H})-d(\mathcal{R})}} L_{0-1}^{\mu'_X \mu_{Y|X}}(h) - \min_{h\in\mathcal{H}_{d(\mathcal{H})}} L_{0-1}^{\mu'_X \mu_{Y|X}}(h). \tag{B.57}$$

As a result, we have

$$\sup_{\mu_{XYM}} D_{\mu_{XYM}}^{\mathcal{H},\mathcal{R}} \leq \sup_{\mu_{XYM}:\, \mu_X \text{ atomic}} D_{\mu_{XYM}}^{\mathcal{H},\mathcal{R}} \tag{B.58}$$

$$\leq \sup_{\mu_{XY}:\, \mu_X \text{ atomic}} \min_{h\in\mathcal{H}_{d(\mathcal{H})-d(\mathcal{R})}} L_{0-1}^{\mu_X \mu_{Y|X}}(h) - \min_{h\in\mathcal{H}_{d(\mathcal{H})}} L_{0-1}^{\mu_X \mu_{Y|X}}(h). \tag{B.59}$$

Next, by applying Lemma 6, we have

$$\sup_{\mu_{XYM}} D_{\mu_{XYM}}^{\mathcal{H},\mathcal{R}} \leq \sup_{\mu_{XY}:\, \mu_X \text{ atomic}} \sum_{i=d(\mathcal{H})-d(\mathcal{R})+2}^{d(\mathcal{H})} \Pr(x_{i,2}) \big[\Pr(Y=1|X=x_{i,2}) - \Pr(Y=0|X=x_{i,2})\big], \tag{B.60}$$

where $\{x_{i,2}\}_i$ are defined in Lemma 6.

Since $\Pr(Y=1|X=x_{i,2}) > \Pr(Y=0|X=x_{i,2})$ we could define

$$q_i = \frac{\Pr(x_{i,2}) \big[\Pr(Y=1|X=x_{i,2}) - \Pr(Y=0|X=x_{i,2})\big]}{\sum_{j=1}^{d(\mathcal{H})} \Pr(x_{j,2}) \big[\Pr(Y=1|X=x_{j,2}) - \Pr(Y=0|X=x_{j,2})\big]}. \tag{B.61}$$

Then, by the definition of $x_{i,2}$ we know that

$$q_1 \geq q_2 \geq \ldots \geq q_{d(\mathcal{H})}, \tag{B.62}$$

and $\sum_{i=1}^{d(\mathcal{H})} q_i = 1$. Hence, using Lemma 7 we have

$$\frac{\sum_{j=d(\mathcal{H})-d(\mathcal{R})+1}^{d(\mathcal{H})} \Pr(x_{j,2})\big[\Pr(Y=1|X=x_{j,2}) - \Pr(Y=0|X=x_{j,2})\big]}{\sum_{j=1}^{d(\mathcal{H})} \Pr(x_{j,2})\big[\Pr(Y=1|X=x_{j,2}) - \Pr(Y=0|X=x_{j,2})\big]} \tag{B.63}$$

$$= \sum_{j=d(\mathcal{H})-d(\mathcal{R})+1}^{d(\mathcal{H})} q_i \leq \frac{d(\mathcal{R})}{d(\mathcal{H})}, \tag{B.64}$$

which concludes that

$$\sum_{j=d(\mathcal{H})-d(\mathcal{R})+1}^{d(\mathcal{H})} \Pr(x_{j,2})\big[\Pr(Y=1|X=x_{j,2}) - \Pr(Y=0|X=x_{j,2})\big]$$

$$\leq \frac{d(\mathcal{R})}{d(\mathcal{H})} \sum_{j=1}^{d(\mathcal{H})} \Pr(x_{j,2})\big[\Pr(Y=1|X=x_{j,2}) - \Pr(Y=0|X=x_{j,2})\big] \leq \frac{d(\mathcal{R})}{d(\mathcal{H})}. \tag{B.65}$$

This, together with ((B.60)) completes the proof. □

## B.2 Proof of Proposition 1

We will prove the following proposition from which Proposition 1 can be obtained from by re-arranging the terms.

Let $\mathcal{S}_l = \{(\mathbf{x}_i, y_i, m_i)\}_{i=1}^{n_l}$ and $\mathcal{S}_u = \{(\mathbf{x}_{i+n_l}, y_{i+n_l})\}_{i=1}^{n_u}$ be two iid sample sets that are drawn from the joint distribution $P_{X,Y,M}$ and are labeled and not labeled by human, respectively. Assume that the optimal classifier $\bar{h} = \operatorname{argmin}_h \mathbb{E}_{X,Y\sim\mu_{XY}}[\mathbb{I}_{h(X)\neq Y}]$ is a member of $\mathcal{H}$ (i.e., realizability). Then, with probability at least $1-\delta$ we have

$$L_{\text{def}}^{0-1}(\hat{r},\hat{h}) \leq L_{0-1}(h^*,r^*) + \mathfrak{R}_{n_u}(\mathcal{H}) + 2\mathfrak{R}_{n_l}(\mathcal{R})$$
$$+ 2\min\big\{\Pr(M\neq Y), \mathcal{R}_{n_l\Pr(M\neq Y)/2}(\mathcal{R})\big\} + C\sqrt{\frac{\log 1/\delta}{n_l}} + e^{-n_l\Pr(M\neq Y)^2/2}$$
$$+ C'\sqrt{\frac{\log 1/\delta}{n_u}} \tag{B.66}$$

where $h^*, r^* = \arg\min_{(h,r)\in\mathcal{H}\times\mathcal{R}} L_{0-1}(h,r)$.

Compare this to using only $S_l$ to learn jointly $\widetilde{h}, \widetilde{r}$ we get Mozannar and Sontag (2020a)[1]:

$$
\begin{aligned}
&L_{\text{def}}^{0-1}(\widetilde{r},\widetilde{h}) \leq L_{0-1}(h^*,r^*) + \mathfrak{R}_{n_l}(\mathcal{H}) + 2\mathfrak{R}_{n_l}(\mathcal{R}) \\
&+ 2\mathcal{R}_{n_l \Pr(M\neq Y)/2}(\mathcal{R}) + C'\sqrt{\frac{\log 1/\delta}{n_l}} \\
&+ \tfrac{\Pr(M\neq Y)}{2} e^{-n\Pr(M\neq Y)/2}
\end{aligned} \tag{B.67}
$$

We start by introducing some useful lemmas, and then we continue with the proof of proposition.

**Lemma 8.** *Let $h^*(x) = \underset{h\in\mathcal{F}}{\operatorname{argmin}} L_{0-1}(h)$, where $\mathcal{F}$ is the class of all functions $h(\cdot) : \mathcal{X} \to \mathcal{Y}$. Then, for every function $r(\cdot) : \mathcal{X} \to \{0,1\}$, we have*

$$
\mathbb{E}_{X,Y}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{h^*(x)\neq y}\right] \leq \mathbb{E}_{X,Y}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{h(x)\neq y}\right], \tag{B.68}
$$

*for all function $h(\cdot) \in \mathcal{F}$.*

*Proof.* Since $h^*(\cdot)$ could be any function, it is easy to show that for $x \in \mathcal{D}$, where $\mathcal{D} = \{x : f_X(x) \neq 0\}$, we have

$$
h^*(x) = \underset{v}{\operatorname{argmin}}\, \mathbb{E}_{Y|X=x}[\mathbb{I}_{v\neq Y}], \tag{B.69}
$$

which concludes that

$$
\mathbb{E}_{Y|X=x}[\mathbb{I}_{h^*(x)\neq Y}] \leq \mathbb{E}_{Y|X=x}[\mathbb{I}_{h(x)\neq Y}], \tag{B.70}
$$

for all $h(\cdot) \in \mathcal{F}$. Hence, we have

$$
\begin{aligned}
\mathbb{E}_{X,Y}[\mathbb{I}_{r(X)=0}\mathbb{I}_{h^*(X)\neq Y}] &= \mathbb{E}_X\left[\mathbb{I}_{r(X)=0}\mathbb{E}_{Y|X=x}[\mathbb{I}_{h^*(x)\neq Y}]\right] && \text{(B.71)}\\
&\leq \mathbb{E}_X\left[\mathbb{I}_{r(X)=0}\mathbb{E}_{Y|X=x}[\mathbb{I}_{h(x)\neq Y}]\right] && \text{(B.72)}\\
&= \mathbb{E}_{X,Y}[\mathbb{I}_{r(X)=0}\mathbb{I}_{h(X)\neq Y}], && \text{(B.73)}
\end{aligned}
$$

which completes the proof.

□

**Lemma 9.** *Let $h^*(x) = \underset{h\in\mathcal{F}}{\operatorname{argmin}} L_{0-1}(h)$, where $\mathcal{F}$ is the class of all functions $h(\cdot) : \mathcal{X} \to \mathcal{Y}$. If we have $h^*(\cdot) \in \mathcal{H}$, then there exists $r \in \mathcal{R}$ such that the pair $(h^*, r)$ is a minimizer*

[1] Note that in Mozannar and Sontag (2020a), they set the notation in a manner that $r \in \{-1,1\}$. Hence, $\mathfrak{R}_n(\mathcal{R})$ under such notation is twice as much as the case in this paper (i.e., $r \in \{0,1\}$). Here, we express their results with our choice of notation.

*of the optimization problem*

$$\underset{(h,r)\in\mathcal{H}\times\mathcal{R}}{\textit{argmin}}\ \mathbb{E}_{X,Y,M}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{h(X)\neq Y}+\mathbb{I}_{r(X)=1}\mathbb{I}_{M\neq Y}\right]. \tag{B.74}$$

*Proof.* We prove this lemma by showing

$$\begin{aligned}\min_{(h,r)\in\mathcal{H}\times\mathcal{R}}&\mathbb{E}_{X,Y,M}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{h(X)\neq Y}+\mathbb{I}_{r(X)=1}\mathbb{I}_{M\neq Y}\right]\\&=\min_{r\in\mathcal{R}}\mathbb{E}_{X,Y}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{h^*(X)\neq Y}+\mathbb{I}_{r(X)=1}\mathbb{I}_{M\neq Y}\right].\end{aligned} \tag{B.75}$$

To show ((B.75)), using Lemma 8, we know that

$$\begin{aligned}&\min_{(h,r)\in\mathcal{H}\times\mathcal{R}}\mathbb{E}_{X,Y,M}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{h(X)\neq Y}+\mathbb{I}_{r(X)=1}\mathbb{I}_{M\neq Y}\right]\\&\geq\min_{(h,r)\in\mathcal{H}\times\mathcal{R}}\mathbb{E}_{X,Y,M}[\mathbb{I}_{r(X)=0}\mathbb{I}_{h^*(X)\neq Y}]+\mathbb{E}_{X,Y,M}\left[\mathbb{I}_{r(X)=1}\mathbb{I}_{M\neq Y}\right] &\text{(B.76)}\\&=\min_{r\in\mathcal{R}}\mathbb{E}_{X,Y,M}[\mathbb{I}_{r(X)=0}\mathbb{I}_{h^*(X)\neq Y}]+\mathbb{E}_{X,Y,M}\left[\mathbb{I}_{r(X)=1}\mathbb{I}_{M\neq Y}\right]. &\text{(B.77)}\end{aligned}$$

On the other hand, using the minimum property, one could show that

$$\begin{aligned}&\min_{h\in\mathcal{H}}\min_{r\in\mathcal{R}}\mathbb{E}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{h(X)\neq Y}+\mathbb{I}_{r(X)=1}\mathbb{I}_{M\neq Y}\right]\\&\leq\min_{r\in\mathcal{R}}\mathbb{E}_{X,Y,M}[\mathbb{I}_{r(X)=0}\mathbb{I}_{h^*(X)\neq Y}]+\mathbb{E}_{X,Y,M}\left[\mathbb{I}_{r(X)=1}\mathbb{I}_{M\neq Y}\right].\end{aligned} \tag{B.78}$$

Hence, using the lower- and upper-bound in ((B.77)) and ((B.78)), one could show ((B.75)) and complete the proof. □

*Proof of Proposition 1.* We prove ((B.66)) in three steps: (i) we bound the expected $0-1$ loss of the classifier $\hat{h}$ when deferral does not happen by a function of the optimal expected $0-1$ loss in such cases, (ii) we bound the joint loss $L_{\text{def}}$ by a function of the optimal joint loss and the Rademacher complexity of a hypothesis class, and (iii) we bound the Rademacher complexity of the aforementioned class by the Rademacher complexity of the deferral hypothesis class $\mathcal{R}$.

• **Step (i)**: Using Rademacher inequality (Theorem 3.3 of Mohri *et al.* (2018)), with probability $1-\delta/4$, we have

$$L_{0-1}(\hat{h})\leq\hat{L}_{0-1}(\hat{h})+2\mathfrak{R}_{n_u}(\mathcal{G})+\sqrt{\frac{\log 4/\delta}{2n_u}}, \tag{B.79}$$

where $\mathcal{G}=\{\mathbf{x},y\to\mathbb{I}_{h(\mathbf{x})\neq y}:h\in\mathcal{H}\}$.

Furthermore, using ((B.79)), since $\hat{h}$ is an optimizer of the empirical loss in $\mathcal{H}$ and

since $h^* \in \mathcal{H}$, with probability $1-\delta/2$ we have

$$L_{0-1}(\hat{h}) \leq \hat{L}_{0-1}(h^*) + 2\mathfrak{R}_{n_u}(\mathcal{G}) + \sqrt{\frac{\log 4/\delta}{2n_u}} \tag{B.80}$$

$$\overset{(a)}{\leq} L_{0-1}(h^*) + 2\mathfrak{R}_{n_u}(\mathcal{G}) + \frac{3\sqrt{2}}{2}\sqrt{\frac{\log 4/\delta}{n_u}}, \tag{B.81}$$

where $(a)$ holds using McDiarmid's inequality, union bound, and by that the empirical loss is $\frac{2}{n}$-bounded difference.

Next, using Lemma 3.4 of Mohri *et al.* (2018) we know that $\mathfrak{R}_n(\mathcal{G}) = \frac{1}{2}\mathfrak{R}_n(\mathcal{H})$. By means of such identity and ((B.81)), with probability $1-\delta/2$ we have

$$L_{0-1}(\hat{h}) \leq L_{0-1}(h^*) + \mathfrak{R}_n(\mathcal{H}) + \frac{3\sqrt{2}}{2}\sqrt{\frac{\log 4/\delta}{n_u}}. \tag{B.82}$$

It remains to show that for each function $r(\cdot) : \mathcal{X} \to \{0,1\}$, we could bound $\mathbb{E}_{X,Y}[\mathbb{I}_{r(X)=0}\mathbb{I}_{\hat{h}(X)\neq Y}]$ by sum of $\mathbb{E}_{X,Y}[\mathbb{I}_{r(X)=0}\mathbb{I}_{h^*(X)\neq Y}]$ and a term that is corresponded to the concentration of measure for large sample size. For proving such inequality, first we know that

$$L_{0-1}(h^*) = \mathbb{E}_{X,Y}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{h^*(X)\neq Y}\right] + \mathbb{E}_X\left[\mathbb{I}_{r(X)=1}\mathbb{E}_{Y|X=\mathbf{x}}[\mathbb{I}_{h^*(X)\neq Y}]\right] \tag{B.83}$$

$$\overset{(a)}{\leq} \mathbb{E}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{h^*(X)\neq Y}\right] + \mathbb{E}_X\left[\mathbb{I}_{r(X)=1}\mathbb{E}_{Y|X=\mathbf{x}}[\mathbb{I}_{h(X)\neq Y}]\right], \tag{B.84}$$

for all $h \in \mathcal{F}$, where $(a)$ is followed by Lemma 8.

Using ((B.82)) and ((B.84)), we have

$$\begin{aligned}
&\mathbb{E}_{X,Y}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{\hat{h}(X)\neq Y}\right] + \mathbb{E}_X\left[\mathbb{I}_{r(X)=1}\mathbb{E}_{Y|X=\mathbf{x}}[\mathbb{I}_{\hat{h}(X)\neq Y}]\right] \\
\leq &\mathbb{E}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{h^*(X)\neq Y}\right] + \mathbb{E}_X\left[\mathbb{I}_{r(X)=1}\mathbb{E}_{Y|X=\mathbf{x}}[\mathbb{I}_{\hat{h}(X)\neq Y}]\right] + \mathfrak{R}_{n_u}(\mathcal{H}) + \frac{3\sqrt{2}}{2}\sqrt{\frac{\log 4/\delta}{n_u}},
\end{aligned} \tag{B.85}$$

which concludes

$$\mathbb{E}_{X,Y}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{\hat{h}(X)\neq Y}\right] \leq \mathbb{E}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{h^*(X)\neq Y}\right] + \mathfrak{R}_{n_u}(\mathcal{H}) + \frac{3\sqrt{2}}{2}\sqrt{\frac{\log 4/\delta}{n_u}}. \tag{B.86}$$

• **Step (ii)**: We know that $\hat{r}(\cdot)$ is obtained as

$$\hat{r}(x) = \underset{r\in\mathcal{R}}{\operatorname{argmin}} \frac{1}{n_l}\sum_{i=1}^{n_l}\left[\mathbb{I}_{r(\mathbf{x}_i)=0}\mathbb{I}_{h(\mathbf{x}_i)\neq y_i} + \mathbb{I}_{r(\mathbf{x}_i)=1}\mathbb{I}_{m_i\neq y_i}\right], \tag{B.87}$$

or equivalently,

$$\hat{r}(x) = \underset{r\in\mathcal{R}}{\operatorname{argmin}} \frac{1}{n_l}\sum_{i=1}^{n_l}\left[\mathbb{I}_{r(\mathbf{x}_i)=0}[\mathbb{I}_{h(\mathbf{x}_i)\neq y_i} - \mathbb{I}_{m_i\neq y_i}]\right]. \tag{B.88}$$

Hence, using Rademacher inequality (Theorem 3.3 of Mohri *et al.* (2018)), with probability $1-\delta/4$, we have

$$\begin{aligned}\mathbb{E}_{X,Y,M}\left[\mathbb{I}_{\hat{r}(X)=0}[\mathbb{I}_{\hat{h}(X)\neq Y} - \mathbb{I}_{M\neq Y}]\right] \leq & \frac{1}{n_l}\sum_{i=1}^{n_l}\mathbb{I}_{\hat{r}(\mathbf{x}_i)=0}[\mathbb{I}_{\hat{h}(\mathbf{x}_i)\neq y_i} - \mathbb{I}_{m_i\neq y_i}] \\ & + 2\mathfrak{R}_{n_l}(\mathcal{J}) + \sqrt{\frac{\log 4/\delta}{n_l}},\end{aligned} \tag{B.89}$$

where

$$\mathcal{J} = \left\{\mathbf{x}, y, m \to \mathbb{I}_{r(\mathbf{x})=0}[\mathbb{I}_{\hat{h}(\mathbf{x})\neq y} - \mathbb{I}_{m\neq y}] : r\in\mathcal{R}\right\}. \tag{B.90}$$

Using Lemma 9, we know that there exists $r^*\in\mathcal{R}$ such that $(r^*,h^*)$ are the minimizers of the joint loss $L_{\text{def}}^{0-1}(h,r)$ in $\mathcal{H}\times\mathcal{R}$. Next, since $\hat{r}$ is the minimizer of the empirical joint loss given the classifier be $\hat{h}$, and using ((B.89)), we have

$$\begin{aligned}\mathbb{E}_{X,Y,M}\left[\mathbb{I}_{\hat{r}(X)=0}[\mathbb{I}_{\hat{h}(X)\neq Y} - \mathbb{I}_{M\neq Y}]\right] \leq & \frac{1}{n_l}\sum_{i=1}^{n_l}\mathbb{I}_{r^*(\mathbf{x}_i)=0}[\mathbb{I}_{\hat{h}(\mathbf{x}_i)\neq y_i} - \mathbb{I}_{m_i\neq y_i}] \\ & + 2\mathfrak{R}_{n_l}(\mathcal{J}) + \sqrt{\frac{\log 4/\delta}{n_l}},\end{aligned} \tag{B.91}$$

for $r^*(\cdot)$ defined as above.

Next, using McDiarmid's inequality, union bound, and since the empirical loss in RHS of ((B.91)) is $\frac{2}{n}$-bounded difference, then with probability at least $1-\delta/2$ we have

$$\begin{aligned}\mathbb{E}_{X,Y,M}\left[\mathbb{I}_{\hat{r}(X)=0}[\mathbb{I}_{\hat{h}(X)\neq Y} - \mathbb{I}_{M\neq Y}]\right] \leq & \mathbb{E}_{X,Y,M}\left[\mathbb{I}_{r^*(X)}[\mathbb{I}_{\hat{h}(X)\neq Y} - \mathbb{I}_{M\neq Y}]\right] + 2\mathfrak{R}_n(\mathcal{J}) \\ & + (\sqrt{2}+1)\sqrt{\frac{\log 4/\delta}{n_l}}.\end{aligned} \tag{B.92}$$

Therefore, using step (i), and by means of union bound, one could prove that with

probability at least $1-\delta$ we have

$$\mathbb{E}_{X,Y,M}\big[\mathbb{I}_{\hat{r}(X)}[\mathbb{I}_{\hat{h}(X)\neq Y}-\mathbb{I}_{M\neq Y}]\big] \leq \mathbb{E}_{X,Y,M}\big[\mathbb{I}_{r^*(X)}[\mathbb{I}_{h^*(X)\neq Y}-\mathbb{I}_{M\neq Y}]\big]+\mathfrak{R}_{n_u}(\mathcal{H})+2\mathfrak{R}_{n_l}(\mathcal{J}) + \frac{3\sqrt{2}}{2}\sqrt{\frac{\log 4/\delta}{n_u}}+(\sqrt{2}+1)\sqrt{\frac{\log 4/\delta}{n_l}}, \tag{B.93}$$

or equivalently

$$L^{0-1}_{\text{def}}(\hat{r},\hat{h}) \leq L^{0-1}_{\text{def}}(h^*,r^*)+\mathfrak{R}_{n_u}(\mathcal{H})+2\mathfrak{R}_{n_l}(\mathcal{J})+\frac{3\sqrt{2}}{2}\sqrt{\frac{\log 4/\delta}{n_u}}+(\sqrt{2}+1)\sqrt{\frac{\log 4/\delta}{n_l}}, \tag{B.94}$$

• **Step (iii)**: In this step, we bound $\mathfrak{R}_n(\mathcal{G})$ to complete the proof. By recalling the definition of $\mathcal{J}$ in ((B.90)), we bound $\mathfrak{R}_n(\mathcal{J})$ as

$$\begin{aligned}
\mathfrak{R}_n(\mathcal{J}) &= \mathbb{E}_{\{(\mathbf{x}_i,y_i,m_i)\}_{i=1}^n}\mathbb{E}_{\boldsymbol{\sigma}}\big[\frac{1}{n}\sup_{g\in\mathcal{J}}\sum_{i=1}^n \sigma_i g(\mathbf{x}_i,y_i,m_i)\big] && \text{(B.95)}\\
&= \mathbb{E}_{\{(\mathbf{x}_i,y_i,m_i)\}_{i=1}^n}\mathbb{E}_{\boldsymbol{\sigma}}\big[\frac{1}{n}\sup_{r\in\mathcal{R}}\sum_{i=1}^n \sigma_i\big[\mathbb{I}_{r(\mathbf{x}_i)=0}(\mathbb{I}_{\hat{h}(\mathbf{x}_i)\neq y_i}-\mathbb{I}_{m_i\neq y_i})\big]\big] && \text{(B.96)}\\
&\overset{(a)}{\leq} \mathbb{E}_{\{(\mathbf{x}_i,y_i,m_i)\}_{i=1}^n}\mathbb{E}_{\boldsymbol{\sigma}}\big[\frac{1}{n}\sup_{r\in\mathcal{R}}\sum_{i=1}^n \sigma_i\big[r(\mathbf{x}_i)\mathbb{I}_{\hat{h}(\mathbf{x}_i)\neq y_i}\big]\big]\\
&\quad+\mathbb{E}_{\{(\mathbf{x}_i,y_i,m_i)\}_{i=1}^n}\mathbb{E}_{\boldsymbol{\sigma}}\big[\frac{1}{n}\sup_{r\in\mathcal{R}}\sum_{i=1}^n \sigma_i\big[r(X)\mathbb{I}_{m_i\neq y_i}\big]\big] && \text{(B.97)}\\
&\overset{(b)}{\leq} \mathfrak{R}_n(\mathcal{R})+\mathbb{E}_{\{(\mathbf{x}_i,y_i,m_i)\}_{i=1}^n}\mathbb{E}_{\boldsymbol{\sigma}}\big[\frac{1}{n}\sum_{i=1}^n \sigma_i\mathbb{I}_{\hat{h}(\mathbf{x}_i)\neq y_i}\big]\\
&\quad+\underbrace{\mathbb{E}_{\{(\mathbf{x}_i,y_i,m_i)\}_{i=1}^n}\mathbb{E}_{\boldsymbol{\sigma}}\big[\frac{\sum_{i=1}^n \mathbb{I}_{m_i\neq y_i}}{n}\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{R})\big]}_{A} && \text{(B.98)}\\
&\overset{(c)}{=} \mathfrak{R}_n(\mathcal{R})+A, && \text{(B.99)}
\end{aligned}$$

where $\mathcal{S}=\{(\mathbf{x}_i,y_i,m_i): m_i\neq y_i\}$ and $(a)$ holds because of sub-linearity of supremum, $(b)$ holds by sub-linearity of supremum, since for two events $E_1$ and $E_2$ we have $\mathbb{I}_{E_1}\cdot\mathbb{I}_{E_2}\leq\mathbb{I}_{E_1}+\mathbb{I}_{E_2}$, and using Lemma 3.4 of Mohri *et al.* (2018), and $(c)$ is followed by $\sigma_i$ being zero-mean.

Now, we should bound $A$. Since $u=\sum_{i=1}^n \mathbb{I}_{m_i\neq y_i}$ is a random variable with distribution Binomial$(n,\Pr(M\neq Y))$ and using Hoeffding's inequality, we have

$$\Pr\big(\tfrac{u}{n}<\Pr(M\neq Y)-t\big)\leq e^{-2nt^2}. \tag{B.100}$$

Next, by decomposing $A$, we have

$$A = \Pr\big(\tfrac{u}{n} < \Pr(M \neq Y) - t\big)\mathbb{E}_{\{(\mathbf{x}_i, y_i, m_i)\}_{i=1}^n}\big[\tfrac{u}{n}\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{R}) \,|\, \tfrac{u}{n} < \Pr(M \neq Y) - t\big] + \Pr\big(\tfrac{u}{n} \geq \Pr(M \neq Y) - t\big)\mathbb{E}_{\{(\mathbf{x}_i, y_i, m_i)\}_{i=1}^n}\big[\tfrac{u}{n}\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{R}) \,|\, \tfrac{u}{n} \geq \Pr(M \neq Y) - t\big] \tag{B.101}$$

$$\leq |\Pr(M \neq Y) - t| e^{-2nt^2} + \min\{\Pr(M \neq Y), \mathcal{R}_{n(\Pr(M \neq Y) - t)}(\mathcal{R})\}, \tag{B.102}$$

where the inequality holds since Rademacher complexity is bounded by 1 and is non-increasing in terms of sample-space size, followed by $\frac{u}{n} \leq 1$, and by means of Lemma 3.4 of Mohri *et al.* (2018). As a result, by setting $t = \frac{\Pr(M \neq Y)}{2}$, we have

$$A \leq \frac{\Pr(M \neq Y)}{2} e^{-\frac{n\Pr^2(M \neq Y)}{2}} + \min\{\Pr(M \neq Y), \mathcal{R}_{n\Pr(M \neq Y)/2}(\mathcal{R})\}. \tag{B.103}$$

Finally using ((B.94)), ((B.99)), and ((B.103)) we complete the proof. □

## B.3 Proof of Proposition 2

To prove the consistency of the deferral surrogate, we know that since $\ell_\phi$ is consistent, for every $\{p_1, \ldots, p_{k+1}\}$, such that $\sum_{i=1}^{k+1} p_i = 1$, we have

$$\operatorname*{argmax}_{i \in [k+1]} \operatorname*{argmin}_{h \in \mathcal{D}} \sum_{i=1}^{k+1} p_i \widetilde{\ell}_\phi(i, h) = \operatorname*{argmax}_{i \in [k+1]} p_i. \tag{B.104}$$

(One could prove this by setting $\Pr(X = x) = \delta[x]$, and $\Pr(Y = y | X = x) = p_y$. )

Next, we find the minimizer of the loss $\widetilde{\ell}_\phi$ as

$$\operatorname*{argmin}_{h \in \mathcal{F}} \mathbb{E}_{X,Y,M}\big[\widetilde{\ell}_\phi(\mathbf{c}, h)\big] = \operatorname*{argmin}_{h(x)} \mathbb{E}_{Y,M|X=x}\big[\widetilde{\ell}_\phi(\mathbf{c}, h(x))\big] \tag{B.105}$$

$$= \operatorname*{argmin}_{h(x)} \sum_{i=1}^{k+1} \mathbb{E}[\max_{j \in [k+1]} c(j) - c(i) | X = x] \widetilde{\ell}_\phi(i, h(x)). \tag{B.106}$$

Next, we form the probability mass function $\{q_1, \ldots, q_{k+1}\}$ as

$$q_i = \frac{\mathbb{E}\big[\max_{j \in [k+1]} c(j) - c(i)\big]}{\sum_{t=1}^{k+1} \mathbb{E}\big[\max_{j \in [k+1]} c(j) - c(t)\big]}. \tag{B.107}$$

One could see that the optimizer in ((B.106)) is equivalent to

$$\operatorname*{argmin}_{h(x)} \sum_{i=1}^{k+1} q_i \ell_\phi\big(i, h(x)\big). \tag{B.108}$$

Now, using ((B.104)) and ((B.108)), we can show that

$$\operatorname*{argmax}_{i\in[k+1]} \operatorname*{argmin}_{h\in\mathcal{F}} \mathbb{E}_{X,Y,M}\big[\widetilde{\ell}_\phi(\mathbf{c},h)\big] = \operatorname*{argmax}_{i\in[k+1]} q_i = \operatorname*{argmin}_{i\in[k+1]} \mathbb{E}[c(i)|X=x]. \tag{B.109}$$

The above identity means that $h_{k+1}(x) \geq \max_{i\in[k]} h_i(x)$ (i.e., $r(x)=1$) iff. we have $\mathbb{E}[c(k+1)|X=x] \leq \min_{i\in[k]} \mathbb{E}[c(i)|X=x]$. Further, we have

$$h(x) = \operatorname*{argmax}_{i\in[k]} h_i(x) = \operatorname*{argmin}_{i\in[k]} \mathbb{E}[c(i)|X=x]. \tag{B.110}$$

Recalling Proposition 1 in Mozannar and Sontag (2020a), one sees that $r(x)$ and $h(x)$ are that of Bayesian optimal classifier, which proves that $\widetilde{\ell}_\phi$ is Fisher consistent.

# B.4 Proof of Theorem 2

To show the result for the calibration function, by setting $\Pr(X=x)=\delta[x']$, and $\Pr(Y=y|X=x')=p_y$ for $y\in[k+1]$, we see that

$$L^{0-1}(h) - \min_{h\in\mathcal{F}} L^{0-1}(h) = \sum_{i\neq h(x')} p_i - \sum_{i\neq \operatorname{argmax} p_i} p_i \tag{B.111}$$

$$= \max_{i\in[k+1]} p_i - p_{h(x')}. \tag{B.112}$$

Furthermore, we have

$$\widetilde{L}_\phi(h) - \min_{h\in\mathcal{F}} \widetilde{L}_\phi(h) = \sum_{i=1}^{k+1} p_i\big[\widetilde{\ell}_\phi(i,h(x')) - \widetilde{\ell}_\phi(i,h^*)\big]. \tag{B.113}$$

Hence, $\psi$ being a calibration function proves that

$$\psi(\max p_i - p_{h(x')}) \leq \sum_{i=1}^{k+1} p_i\big[\widetilde{\ell}_\phi(i,h(x')) - \widetilde{\ell}_\phi(i,h^*)\big], \tag{B.114}$$

for every choice of $h(x')$.

On the other hand, one could calculate the conditional cost-sensitive loss as

$$L_{\mathbf{c},x}(h) = \mathbb{E}_{Y|X=x}\big[c(h(X))\big] = \sum_{i\neq h(x)} \mathbb{E}_{Y|X=x}\big[c(i)|X=x\big]. \tag{B.115}$$

Hence, we have

$$L_{c,x}(h) - L_{c,x}(h^*) = \mathbb{E}\big[c(h(X))|X=x\big] - \min_{i\in[k+1]} \mathbb{E}[c(i)|X=x\big], \tag{B.116}$$

where $h^* = \operatorname*{argmin}_{h\in\mathcal{F}} L_c(h)$.

By defining $q_i$s as ((B.107)), one can prove that

$$L_{c,x}(h) - L_{c,x}(h^*) = \sum_{i=1}^{k+1} \big[\max_{j\in[k+1]} c(j) - c(i)|X=x\big]\big(\max q_i - q_{h(x)}\big). \tag{B.117}$$

For the new surrogate, we further know that

$$\widetilde{L}_{\mathbf{c},x}(h) = \mathbb{E}_{Y|X=x}\big[\widetilde{\ell}(\mathbf{c},h(x))\big] \tag{B.118}$$

$$= \sum_{i=1}^{k+1} \mathbb{E}\big[\max_{j\in[k+1]} c(j) - c(i)|X=x\big] \sum_{i=1}^{k+1} q_i \widetilde{\ell}_\phi(i,h(x)). \tag{B.119}$$

Furthermore, one could show that

$$\widetilde{h}_1^{k+1} = \operatorname*{argmin}_{h\in\mathcal{F}} \widetilde{L}_{\mathbf{c},x}(h) \tag{B.120}$$

$$= \operatorname*{argmin}_{h\in\mathcal{F}} \sum_{i=1}^{k+1} q_i \widetilde{\ell}_\phi(i,h), \tag{B.121}$$

and consequently,

$$\widetilde{L}_{\mathbf{c},x}(h) - \widetilde{L}_{\mathbf{c},x}(\widetilde{h}_1^{k+1}) = \sum_{i=1}^{k+1} \mathbb{E}\big[\max_{j\in[k+1]} c(j) - c(i)|X=x\big] \cdot \sum_{i=1}^{k+1} q_i(\widetilde{\ell}_\phi(i,h) - \widetilde{\ell}_\phi(i,\widetilde{h}_1^{k+1}). \tag{B.122}$$

Hence, using ((B.114)) and ((B.121)), we have

$$\mathbb{E}\big[\max_{j\in[k+1]} c(j) - c(i)|X=x\big]\,\psi(\max_{i\in[k+1]} q_i - q_{h(x)}) \le \widetilde{L}_{\mathbf{c},x}(h) - \widetilde{L}_{\mathbf{c},x}(\widetilde{h}_1^{k+1}). \tag{B.123}$$

Hence, since $\psi(x) = C|x|^\varepsilon$, we have

$$\psi(L_{c,x}(h) - L_{c,x}(h^*)) = \psi(\mathbb{E}\big[\max_{j\in[k+1]} c(j) - c(i)|X=x\big](\max_{i\in[k+1]} q_i - q_{h(x)}) \tag{B.124}$$

$$\le \mathbb{E}^{\varepsilon-1}\big[\max_{j\in[k+1]} c(j) - c(i)|X=x\big](\widetilde{L}_{\mathbf{c},x}(h) - \widetilde{L}_{\mathbf{c},x}(\widetilde{h}_1^{k+1})) \tag{B.125}$$

$$\overset{(a)}{\le} M^{\varepsilon-1}(\widetilde{L}_{\mathbf{c},x}(h) - \widetilde{L}_{\mathbf{c},x}(\widetilde{h}_1^{k+1})), \tag{B.126}$$

where $(a)$ holds using the assumption of the theorem.

Finally, using convexity of $\psi$ and by Jensen's inequality, we have

$$\psi(L_c(h) - L_c(h^*)) = \psi(\mathbb{E}_X[L_{c,x}(h) - L_{c,x}(h^*)] \tag{B.127}$$
$$\leq \mathbb{E}_X \psi(L_{c,x}(h) - L_{c,x}(h^*)) \tag{B.128}$$
$$\overset{(a)}{\leq} M^{\varepsilon-1}\mathbb{E}_X\left[\widetilde{L}_{\mathbf{c},x}(h) - \widetilde{L}_{\mathbf{c},x}(\widetilde{h}_1^{k+1})\right] \tag{B.129}$$
$$= M^{\varepsilon-1}(\widetilde{L}_{\mathbf{c},x}(h) - \widetilde{L}_{\mathbf{c},x}(\widetilde{h}_1^{k+1}), \tag{B.130}$$

in which $(a)$ is followed by ((B.126)). This completes the proof of the first part of theorem.

To obtain the calibration function of the cross-entropy error, we first introduce the following lemma.

**Lemma 10.** *For every two distributions P and G, we have*

$$\left|\max_i P_i - \max_i G_i\right| \leq \sqrt{2D_{KL}(P\|G)}. \tag{B.131}$$

*Proof.* We define $\operatorname{argmax}_i G_i = i_G^*$, and $\operatorname{argmax}_i P_i = i_P^*$. If we have $\max_i G_i = G_{i_G^*} \geq G_{i_P^*} = \max_i P_i$, then

$$0 \leq G_{i_G^*} - P_{i_P^*} \overset{(a)}{\leq} G_{i_G^*} - P_{i_G^*} \overset{(b)}{\leq} \sqrt{2D_{KL}(P\|G)}, \tag{B.132}$$

where $(a)$ is correct due to the fact that $\max_i P_i \geq P_{i_G^*}$, and $(b)$ holds due to Pinsker's inequality. Further, if we have $\max_i G_i = P_{i_G^*} \leq P_{i_P^*} = \max_i P_i$, using a similar argument, we have

$$0 \leq P_{i_P^*} - G_{i_G^*} \leq P_{i_P^*} - G_{i_P^*} \leq \sqrt{2D_{KL}(P\|G)}. \tag{B.133}$$

□

Next, we note that the conditional surrogate risk can be rewritten as

$$\widetilde{L}_{CE,x}(g_1,\ldots,g_{K+1}) = -\sum_{i=1}^{K+1}\mathbb{E}\big[\max_{j\in[K+1]} c(j) - c(i)|X=x\big]\log\frac{\exp(g_i(x))}{\sum_k \exp(g_k(x))} \tag{B.134}$$
$$= N_x H_{\mathcal{P}_x}(\mathcal{G}_x), \tag{B.135}$$

where $N_x = \sum_{i=1}^{K+1}\mathbb{E}\big[\max_{j\in[K+1]} c(j) - c(i)|X=x\big]$, $H_{\mathcal{P}_x}(\mathcal{G}_x)$ refers to the relative entropy of the distribution $\mathcal{G}_x$ w.r.t $\mathcal{P}_x$ which are defined as

$$\mathcal{P}_{x,i} = \frac{\mathbb{E}\big[\max_{j\in[K+1]} c(j) - c(i)|X=x\big]}{N_x}, \tag{B.136}$$

and

$$\mathcal{G}_{x,i} = \frac{\exp(g_i(x))}{\sum_k \exp(g_k(x))}. \tag{B.137}$$

Secondly, one note that since in the minimizer of surrogate risk

$$\operatorname*{argmin}_{\mathbf{g}\in\mathcal{F}} \widetilde{L}_{CE}(\mathbf{g}),$$

$\mathcal{F}$ contains every function, hence there is no dependency between different point $x$s, and as a result, the minimization is equivalent to finding minimize every conditional surrogate risk. More formally, if $g_1^*, \ldots, g_{K+1}^*$ are such pair of minimizers, we have

$$\begin{aligned}
\big(g_1^*(x), \ldots, g_{K+1}^*(x)\big) &= \operatorname*{argmin}_{g_1(x),\ldots,g_{K+1}(x)} \widetilde{L}_{CE,x}\big(g_1(x), \ldots, g_{K+1}(x)\big) && \text{(B.138)}\\
&\overset{(a)}{=} \operatorname*{argmin}_{g_1(x),\ldots,g_{K+1}(x)} N_x H_{\mathcal{P}_x}(\mathcal{G}_x) && \text{(B.139)}\\
&= \operatorname*{argmin}_{g_1(x),\ldots,g_{K+1}(x)} H_{\mathcal{P}_x}(\mathcal{G}_x) && \text{(B.140)}\\
&\overset{(b)}{=} \big(\mathcal{P}_{x,1}, \ldots, \mathcal{P}_{x,K+1}\big), && \text{(B.141)}
\end{aligned}$$

where $(a)$ holds because of ((B.135)), and $(b)$ is a property of relative entropy.

As a result, the conditional excess surrogate risk can be rewritten as

$$\widetilde{L}_{CE,x}(g_1, \ldots, g_{K+1}) - \widetilde{L}^*_{CE,x} = N_x H_{\mathcal{P}_x}(\mathcal{G}_x) - N_x H_{\mathcal{P}_x}(\mathcal{P}_x) = N_x D_{KL}(\mathcal{P}_x, \mathcal{G}_x). \tag{B.142}$$

Further, we can write the conditional excess risk as

$$\begin{aligned}
L_x^{0-1}(g_1, \ldots, g_{K+1}) - L_x^{0-1}(g_1^*, \ldots, g_{K+1}^*) & \\
= \mathbb{E}\Big[c\big(\operatorname*{argmax}_{i(x)} g_{i(x)}(x) | X = x\big] - \min_{i(x)} \mathbb{E}\Big[c\big(i(x)\big) | X = x\Big], & \quad \text{(B.143)}
\end{aligned}$$

where $L_x^{0-1}$ is defined as

$$L_x^{0-1}(g_1, \ldots, g_{K+1}) = \mathbb{E}\big[\mathbb{I}_{Y \neq \operatorname*{argmax}_{i\in[K+1] g_i(X)}} | X = x\big] \tag{B.144}$$

Next, we can rewrite this conditional excess risk in terms of $\mathcal{P}_{x,i}$s as

$$L_x^{0-1}(g_1,\dots,g_{K+1}) - L_x^{0-1}(g_1^*,\dots,g_{K+1}^*) = N_x\big(\max_{i(x)} \mathcal{P}_{x,i(x)} - \mathcal{P}_{x,\,\operatorname{argmax}_{i(x)} g_{i(x)}(x)}\big) \quad \text{(B.145)}$$

$$= N_x\big(\max_{i(x)} \mathcal{P}_{x,i(x)} - \mathcal{P}_{x,\,\operatorname{argmax}_{i(x)} \mathcal{G}_{x,i(x)}}\big). \quad \text{(B.146)}$$

To bound such a value, we use Pinsker's inequality which states that for every two distributions $P$ and $G$ supported on $\mathbb{N}$, we have

$$TV(P,Q) = \frac{1}{2}\sum_i |P_i - Q_i| \leq \sqrt{\frac{D_{KL}(P\|Q)}{2}}. \quad \text{(B.147)}$$

To make use of that inequality, by defining $i_{\mathcal{P}_x} := \operatorname{argmax}_{i(x)} \mathcal{P}_{x,i(x)}$ and $i_{\mathcal{G}_x}; = \operatorname{argmax}_{i(x)} \mathcal{G}_{x,i(x)}$ and using triangle inequality, we know that

$$N_x\big|\max_{i(x)} \mathcal{P}_{x,i(x)} - \mathcal{P}_{x,\,\operatorname{argmax}_{i(x)} \mathcal{G}_{x,i(x)}}\big| \leq N_x\big|\mathcal{P}_{x,i_{\mathcal{P}_x}} - \mathcal{G}_{x,i_{\mathcal{G}_x}}\big| + N_x\big|\mathcal{G}_{x,i_{\mathcal{G}_x}} - \mathcal{P}_{x,i_{\mathcal{G}_x}}\big|. \quad \text{(B.148)}$$

Next, we bound each of these terms separately. Firstly, we know that

$$N_x\big|\mathcal{G}_{x,i_{\mathcal{G}_x}} - \mathcal{P}_{x,i_{\mathcal{G}_x}}\big| \leq N_x \sum_i |\mathcal{P}_{x,i} - \mathcal{G}_{x,i}| = N_x TV(\mathcal{P}_x\|\mathcal{G}_x) \quad \text{(B.149)}$$

$$\leq N_x\sqrt{2D_{KL}(\mathcal{P}_x\|\mathcal{G}_x)}. \quad \text{(B.150)}$$

Further, using Lemma 10, one can show that

$$N_x\big|\mathcal{P}_{x,i_{\mathcal{P}_x}} - \mathcal{G}_{x,i_{\mathcal{G}_x}}\big| \leq N_x\sqrt{2D_{KL}(\mathcal{P}_x\|\mathcal{G}_x)}. \quad \text{(B.151)}$$

As a result, we have

$$L_x^{0-1}(g_1,\dots,g_{K+1}) - L_x^{0-1}(g_1^*,\dots,g_{K+1}^*) \leq N_x\sqrt{8D_{KL}(\mathcal{P}_x\|\mathcal{G}_x)} \quad \text{(B.152)}$$

$$= \sqrt{8N_x}\sqrt{\widetilde{L}_{CE,x}(g_1,\dots,g_{K+1}) - \widetilde{L}^*_{CE,x}}, \quad \text{(B.153)}$$

where the last equality is followed by ((B.142)). Next, using the upper-bound on $c(i)$s, we have $N_x \leq 2KM$. As a result, we have

$$\frac{\big(L_x^{0-1}(g_1,\dots,g_{K+1}) - L_x^{0-1}(g_1^*,\dots,g_{K+1}^*)\big)^2}{16MK} \leq \widetilde{L}_{CE,x}(g_1,\dots,g_{K+1}) - \widetilde{L}^*_{CE,x}. \quad \text{(B.154)}$$

Finally, using Jensen's inequality, we have

$$\frac{\big(L^{0-1}(g_1,\dots,g_{K+1})-L^{0-1}(g_1^*,\dots,g_{K+1}^*)\big)^2}{16MK}\leq \widetilde{L}_{CE}(g_1,\dots,g_{K+1})-\widetilde{L}_{CE}^*, \tag{B.155}$$

which yields the statement of theorem.

## B.5 Proof of Theorem 3

We first introduce some useful lemmas, then we get back to the proof of theorem.

**Lemma 11.** *Let $\mathcal{F}_1,\dots,\mathcal{F}_k$ be hypothesis classes with Rademacher complexity $\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{F}_1),\dots,\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{F}_k)$ on set $\mathcal{S}$. The Rademacher complexity of the hypothesis class $\mathcal{G}=\{\log\sum_{i=1}^k e^{f_i(x)}:\ f_i(\cdot)\in\mathcal{F}_i\}$ on set $\mathcal{S}$ is bounded as*

$$\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{G})\leq\sum_{i=1}^k\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{F}_i). \tag{B.156}$$

*Proof.* We prove this lemma for $k=2$. By following similar steps, one could generalize this proof for every $k\in\mathbb{N}$.

We write the Rademacher complexity of $\mathcal{G}$ as

$$\begin{aligned}
\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{G}) &= \frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f_1\in\mathcal{F}_1,f_2\in\mathcal{F}_2}\sum_{i=1}^m\sigma_i\log(e^{f_1(x)}+e^{f_2(x)})\big] && \text{(B.157)}\\
&= \frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f_1\in\mathcal{F}_1,f_2\in\mathcal{F}_2}\sum_{i=1}^m\frac{\sigma_i}{2}f_1+\sum_{i=1}^m\frac{\sigma_i}{2}f_2\\
&\qquad +\sum_{i=1}^m\sigma_i\log(e^{f_1(x)/2-f_2(x)/2}+e^{f_2(x)/2-f_1(x)/2})\big] && \text{(B.158)}\\
&\overset{(a)}{\leq} \frac{1}{2m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f_1\in\mathcal{F}_1}\sum_{i=1}^m\sigma_i f_1\big]+\frac{1}{2m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f_2\in\mathcal{F}_2}\sum_{i=1}^m\sigma_i f_2\big]\\
&\qquad +\frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sum_{i=1}^m\sigma_i\log(e^{f_1(x)/2-f_2(x)/2}+e^{f_2(x)/2-f_1(x)/2})\big] && \text{(B.159)}\\
&= \frac{1}{2}\hat{\mathcal{R}}_{\mathcal{S}}(\mathcal{F}_1)+\frac{1}{2}\hat{\mathcal{R}}_{\mathcal{S}}(\mathcal{F}_2)+\hat{\mathcal{R}}_{\mathcal{S}}\big(\Phi o(\mathcal{F}_1-\mathcal{F}_2)\big), && \text{(B.160)}
\end{aligned}$$

where $(a)$ is followed by the sublinearity of supremum, and $\Phi(\cdot)$ is defined as $\Phi(x)=\log(e^{x/2}+e^{-x/2})$.

One could see that $\frac{\partial\Phi(x)}{\partial x}=\frac{1}{2}\frac{e^{x/2}-e^{-x/2}}{e^{x/2}+e^{-x/2}}\leq\frac{1}{2}$, that leads to $\frac{1}{2}$-Lipschitzness of $\Phi(\cdot)$.

Using this, and by Ledoux-Talagrand theorem Ledoux and Talagrand (1991), we have

$$\hat{\mathcal{R}}_{\mathcal{S}}\big(\Phi o(\mathcal{F}_1-\mathcal{F}_2)\big) \leq \frac{1}{2}\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{F}_1-\mathcal{F}_2) \tag{B.161}$$

$$= \frac{1}{2m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f_1\in\mathcal{F},f_2\in\mathcal{F}_2}\sum_{i=1}^{m}\sigma_i\big(f_1(x)-f_2(x)\big)\big] \tag{B.162}$$

$$\overset{a}{\leq} \frac{1}{2m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f_1\in\mathcal{F}_1}\sum_{i=1}^{m}\sigma_i f_1(x)\big] + \frac{1}{2m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f_2\in\mathcal{F}_2}\sum_{i=1}^{m}\sigma_i f_2(x)\big] \tag{B.163}$$

$$= \frac{1}{2}\big(\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{F}_1)+\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{F}_2)\big), \tag{B.164}$$

where $(a)$ is again followed by sublinearity of supremum.

Finally, using ((B.160)) and ((B.164)), we complete the proof. □

**Lemma 12.** *Let $\mathcal{F}$ be a hypothesis class of functions $f(x,y):\mathcal{X}\times[k+1]\to\mathbb{R}$, and $\Pi_1(\mathcal{F})=\{x\to f(x,y): f(\cdot,\cdot)\in\mathcal{F}, y\in[k+1]\}$. Then,*

- *for $\mathcal{G}=\{x,y\to f(x,y)-\log\sum_{j=1}^{k+1}f(x,y): f(\cdot,\cdot)\in\mathcal{F}\}$ and given the assumption that for every label inside sets of pairs $(x_i,y_i)\in\mathcal{S}$ is within the range $\{1,\dots,k\}$, we have*

$$\hat{\mathcal{R}}_{\mathcal{S}}(\mathcal{G}) \leq (2k+1)\hat{\mathcal{R}}_{\mathcal{S}_x}\big(\Pi_1(\mathcal{F})\big), \tag{B.165}$$

- *and for $\mathcal{H}_i=\{x\to f(x,i)-\log\sum_{y=1}^{k+1}f(x,y): f(\cdot,\cdot)\in\mathcal{F}\}$, we have*

$$\hat{\mathcal{R}}_{\mathcal{S}}(\mathcal{H}_i) \leq (k+2)\hat{\mathcal{R}}_{\mathcal{S}_x}\big(\Pi_1(\mathcal{F})\big). \tag{B.166}$$

*Proof.* 1. We write Rademacher complexity of $\mathcal{G}$ as

$$\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{G}) = \frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f\in\mathcal{F}}\sum_{i=1}^{m}\sigma_i f(x_i,y_i) - \sigma_i\log\sum_{y=1}^{k+1}e^{f(x_i,y)}\big] \tag{B.167}$$

$$\overset{(a)}{\leq} \underbrace{\frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f\in\mathcal{F}}\sigma_i f(x_i,y_i)\big]}_{A} + \underbrace{\frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f\in\mathcal{F}}\sum_{i=1}^{m}\sigma_i\log\sum_{y=1}^{k+1}e^{f(x_i,y)}\big]}_{B}, \tag{B.168}$$

where $(a)$ holds because of sublinearity of supremum. Next, we bound $A$ and $B$ as follows.

First, we know that

$$A = \frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f\in\mathcal{F}}\sum_{y=1}^{k}\sum_{i=1}^{m}\sigma_i f(x_i,y)\mathbb{I}_{y_i=y}\big] \tag{B.169}$$

$$\leq \sum_{y=1}^{k}\frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f\in\mathcal{F}}\sum_{i=1}^{m}\sigma_i f(x_i,y)(\varepsilon_i/2+1/2)\big], \tag{B.170}$$

where $\varepsilon_i = 2\mathbb{I}_{y_i=y} - 1$. Hence, again, applying sublinearity of supremum, we have

$$A \leq \sum_{y=1}^{k}\frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f\in\mathcal{F}}\sum_{i=1}^{m}\frac{\sigma_i\varepsilon_i}{2}f(x_i,y)\big] + \frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f\in\mathcal{F}}\sum_{i=1}^{m}\frac{\sigma_i}{2}f(x_i,y)\big]. \tag{B.171}$$

Since $\boldsymbol{\varepsilon}\in\{-1,1\}$, then $\sigma_i\varepsilon_i$ take Rademacher distribution as well. Hence, using ((B.171)), we have

$$A \leq \sum_{y=1}^{k}\frac{1}{2}\hat{\mathcal{R}}_{\mathcal{S}_x}\big(\Pi_1(\mathcal{F})\big) + \frac{1}{2}\hat{\mathcal{R}}_{\mathcal{S}_x}\big(\Pi_1(\mathcal{F})\big) \tag{B.172}$$

$$= k\hat{\mathcal{R}}_{\mathcal{S}_x}\big(\Pi_1(\mathcal{F})\big). \tag{B.173}$$

Next, to bound $B$, using Lemma 11, we have

$$B \leq \sum_{y=1}^{k+1}\frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f\in\mathcal{F}}\sum_{i=1}^{m}\sigma_i f(x_i,y)\big] \leq \hat{\mathcal{R}}_{\mathcal{S}_x}\big(\Pi_1(\mathcal{F})\big). \tag{B.174}$$

Finally, using ((B.168)), ((B.173)), and ((B.174)), we complete the proof.

2. We bound Rademacher complexity of $\mathcal{H}_i$ as

$$\hat{\mathfrak{R}}_{\mathcal{S}_x}(\mathcal{H}_i) = \frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f\in\mathcal{F}}\sum_{j=1}^{m}\sigma_j f(x_j,i) - \sigma_j\log\sum_{y=1}^{k+1}e^{f(x_j,i)}\big] \tag{B.175}$$

$$\overset{(a)}{\leq} \frac{1}{m}\mathbb{E}\big[\sup_{f\in\mathcal{F}}\sum_{j=1}^{m}\sigma_j f(x_j,i)\big] + \frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f\in\mathcal{F}}\sum_{j=1}^{m}\sigma_j\log\sum_{y=1}^{k+1}e^{f(x_j,y)}\big] \tag{B.176}$$

$$\overset{(b)}{\leq} \hat{\mathcal{R}}_{\mathcal{S}_x}\big(\Pi_1(\mathcal{F})\big) + \sum_{y=1}^{k+1}\frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{f\in\mathcal{F}}\sum_{j=1}^{m}f(x_j,y)\big] \tag{B.177}$$

$$\leq (k+2)\hat{\mathcal{R}}_{\mathcal{S}_x}\big(\Pi_1(\mathcal{F})\big), \tag{B.178}$$

where $(a)$ is followed by sublinearity of supermum, and $(b)$ because of Lemma 11 and using definition of $\Pi_1(\mathcal{F})$.

□

**Lemma 13.** *For $i \in \{1,\dots,k+1\}$ let $\mathcal{H}_i$ be hypothesis class of functions $h_i(x) : \mathcal{X} \to \mathbb{R}$ with bounded norm $\|h_i\|_\infty < C$. Further, let $\Pi_1(\mathcal{H}) = \{x \to h_i(x) : h_i \in \mathcal{H}_i, i \in [k+1]\}$. The Rademacher complexity of the class $\mathcal{L}$ of loss functions*

$$\ell(x,y,m) = -\log \frac{e^{-h_y(x)}}{\sum_{i=1}^{k+1} e^{-h_i(x)}} - \mathbb{I}_{m\neq y} \log \frac{e^{-h_{k+1}(x)}}{\sum_{i=1}^{k+1} e^{-h_i(x)}}, \tag{B.179}$$

*for $m, y \in [k]$ is bounded as*

$$\begin{aligned}\mathfrak{R}_n(\mathcal{L}) \leq (k+1)\mathcal{R}_n\big(\Pi_1(\mathcal{H})\big) + (k+2)\min\{Pr(M\neq Y), \mathcal{R}_{nPr(M\neq Y)/2}\big(\Pi_1(\mathcal{H})\big)\} \\ + \frac{C}{2} Pr(M\neq Y)(k+2)e^{-nPr^2(M\neq Y)/2}.\end{aligned} \tag{B.180}$$

*Proof.* We write empirical Rademacher complexity of $\mathcal{L}$ as

$$\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{L}) = \frac{1}{n}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{\ell\in\mathcal{L}} \sum_{i=1}^{n} \sigma_i \ell(x_i,y_i,m_i)\big] \tag{B.181}$$

$$= \frac{1}{n}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{h_j\in\mathcal{H}_j} \sum_{i=1}^{n} \sigma_i(-\log \frac{e^{-h_y(x_i)}}{\sum_{j=1}^{k+1} e^{-h_j(x_i)}} - \mathbb{I}_{m\neq y}\log \frac{e^{-h_{k+1}(x_i)}}{\sum_{j=1}^{k+1} e^{-h_j(x_i)}})\big] \tag{B.182}$$

$$\leq \frac{1}{n}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{h_j\in\mathcal{H}_j} \sum_{i=1}^{n} \sigma_i \log \frac{e^{-h_y(x_i)}}{\sum_{j=1}^{k+1} e^{-h_j(x_i)}}\big] + \frac{1}{n}\mathbb{E}_{\boldsymbol{\sigma}}\big[\sup_{h_j\in\mathcal{H}_j} \sum_{i=1,y_i\neq m_i}^{n} \sigma_i \log \frac{e^{-h_{k+1}(x_i)}}{\sum_{j=1}^{k+1} e^{-h_j(x_i)}}\big] \tag{B.183}$$

$$\overset{(a)}{\leq} (k+1)\hat{\mathcal{R}}_{\mathcal{S}_x}\big(\Pi_1(\mathcal{H})\big) + (k+2)\frac{\sum_{i=1}^{n}\mathbb{I}_{m_i\neq y_i}}{n}\hat{R}_{\mathcal{S}_x|m_i\neq y_i}\big(\Pi_(\mathcal{H})\big), \tag{B.184}$$

where $(a)$ holds by applying Lemma 12.

Using (B.184), and by calculating the expectation over $\{(x_i,y_i,m_i)\}_{i=1}^n$, we have

$$\mathcal{R}_n(\mathcal{L}) \leq (k+1)\mathcal{R}_n\big(\Pi_1(\mathcal{H})\big) + (k+2)\underbrace{\mathbb{E}_{\{(x_i,y_i,m_i)\}_{i=1}^n}\Big[\frac{\sum_{i=1}^{n}\mathbb{I}_{m_i\neq y_i}}{n}\hat{\mathcal{R}}_{\mathcal{S}_x|m_i\neq y_i}\big(\Pi_1(\mathcal{H})\big)\Big]}_{A}. \tag{B.185}$$

It is remained to bound $A$. For this task, we first notice that $u = \sum_{i=1}^{n}\mathbb{I}_{m_i\neq y_i}$ is a random variable with distribution $\text{Binomial}\big(n, \Pr(M\neq Y)\big)$. Further, by Hoeffding's inequality we know that for $t > 0$, we have

$$\Pr\Big(\frac{u}{n} < \Pr(M\neq y) - t\Big) \leq e^{-2nt^2}. \tag{B.186}$$

Hence, by decomposing $A$, we have

$$A = \Pr(\frac{u}{n} < \Pr(M \neq Y) - t)\mathbb{E}\big[\frac{u}{n}\hat{\mathfrak{R}}_{\mathcal{S}_x|y_i \neq m_i}|\frac{u}{n} < \Pr(M \neq Y) - t\big]$$
$$+ \Pr(\frac{u}{n} \geq \Pr(M \neq Y) - t)\mathbb{E}\big[\frac{u}{n}\hat{\mathfrak{R}}_{\mathcal{S}_x|y_i \neq m_i}|\frac{u}{n} \geq \Pr(M \neq Y) - t\big] \tag{B.187}$$
$$\leq C|\Pr(M \neq Y) - t|e^{-2nt^2} + \min\{\Pr(M \neq Y), \mathcal{R}_{n(\Pr(M \neq Y)-t)}\big(\Pi_1(\mathcal{H})\big)\}, \tag{B.188}$$

where the last inequality holds because (1) every function in $\Pi_1(\mathcal{H})$ is bound by $C$, and so is the Rademacher complexity of $\Pi_1(\mathcal{H})$, and (2) the Rademacher complexity is non-increasing with the sample space size.

Hence, using (B.185), (B.188), and by setting $t = \frac{\Pr(M \neq Y)}{2}$ the proof is complete. □

*Proof of Theorem 3.* Using Rademacher inequality on generalization error (e.g., Theorem 3.3 of Mohri *et al.* (2018)), we know that with probability at least $1 - \delta/2$, we have

$$\widetilde{L}_{CE}(\mathbf{f}_1^{k+1}) \leq \hat{L}_{CE}(\mathbf{f}_1^{k+1}) + 2\mathfrak{R}_n(\mathcal{L}) + \sqrt{\frac{D\log 2/\delta}{2n}}, \tag{B.189}$$

where $D$ is an a upper-bound on $\|\ell\|_\infty$ for $\ell \in \mathcal{L}$, and where $\hat{L}_{CE}$ is the empirical loss corresponding to $\ell_{CE}$, and $f_i \in \mathcal{F}_i$.

We follow the proof in three steps, (i) we find $D$, (ii) we find a lower-bound on $\widetilde{L}_{CE}(\mathbf{f}_1^{k+1})$ in terms of $L_{\text{def}}^{0-1}(\mathbf{f}_1^{k+1})$, and (iii) we complete the proof by bounding the difference $|\min_{\mathbf{f} \in \mathcal{F}_1^{k+1}} \hat{L}_{CE}(\mathbf{f}_1^{k+1}) - \min_{\mathbf{f}_1^{k+1} \in \mathcal{F}} \widetilde{L}_{CE}(\mathbf{f}_1^{k+1})|$.

• **Step (i)**: For calculating a bound on $\|\ell\|_\infty$ for $\ell \in \mathcal{L}$, we use boundedness of $\|f_i\|_\infty$ for $i \in [k+1]$ and $\mathbf{f}_1^{k+1} \in \mathcal{F}_1^{k+1}$. Indeed, we know the function

$$b_D(x) = -\log \frac{e^{-x}}{e^{-x} + D}, \tag{B.190}$$

for $D > 0$ is a monotonically non-increasing function of $x$. Hence, over a closed interval, it takes the minimum and maximum on the limit points. As a result, for $|x| \leq C$, we have

$$0 \leq b_D(x) \leq -\log \frac{e^{-C}}{e^{-C} + D}. \tag{B.191}$$

Hence, for the loss function $\ell(x,y,m)$, in which $\|\mathbf{f}_1^{k+1}\| \leq C$, we have

$$0 < \ell(x,y,m) = b_{\sum_{i=1,i\neq y}^{k+1} e^{-f_i(x)}}\big(f_y(x)\big) + \mathbb{I}_{m\neq y} b_{\sum_{i=1}^{k} e^{-f_i(x)}}\big(f_{k+1}(x)\big) \tag{B.192}$$

$$\leq -\log \frac{e^{-C}}{e^{-C}+\sum_{i=1,i\neq y}^{k+1} e^{-f_i(x)}} - \mathbb{I}_{m\neq y} \log \frac{e^{-C}}{e^{-C}+\sum_{i=1}^{k} e^{-f_i(x)}} \tag{B.193}$$

$$-2\log \frac{e^{-C}}{e^{-C}+ke^{C}} \leq -2\log \frac{e^{-2C}}{k+1} = 4C - 2\log(k+1). \tag{B.194}$$

• **Step (ii)**: Using excessive surrogate risk bound, we see that

$$\psi\big(L_{\text{def}}^{0-1}(\mathbf{f}_1^{k+1}) - \min_{\mathbf{h}_1^{k+1}} L_{\text{def}}^{0-1}(\mathbf{h}_1^{k+1})\big) + \min_{\mathbf{h}_1^{k+1}} \widetilde{L}_{CE}(\mathbf{h}_1^{k+1}) \leq \widetilde{L}_{CE}(\mathbf{f}_1^{k+1}). \tag{B.195}$$

• **Step (iii)**: In this step, we find a bound on $\hat{L}_{CE}(\mathbf{f}_1^{k+1}) - \min_{\mathbf{h}} \widetilde{L}_{CE}(\mathbf{h}_1^{k+1})$. Indeed, we know that

$$\begin{aligned}
\hat{L}_{CE}(\mathbf{f}_1^{k+1}) - \min_{\mathbf{h}_1^{k+1}} \widetilde{L}_{CE}(\mathbf{h}_1^{k+1}) &= \underbrace{\hat{L}_{CE}(\mathbf{f}_1^{k+1}) - \min_{\mathbf{h}_1^{k+1}\in\mathcal{F}_1^{k+1}} \hat{L}_{CE}(\mathbf{h}_1^{k+1})}_{e_{min}} \\
&\quad + \min_{\mathbf{h}_1^{k+1}\in\mathcal{F}_1^{k+1}} \hat{L}_{CE}(\mathbf{h}_1^{k+1}) \quad - \min_{\mathbf{h}_1^{k+1}\in\mathcal{F}_1^{k+1}} \widetilde{L}_{CE}(\mathbf{h}_1^{k+1}) \\
&\quad + \underbrace{\min_{\mathbf{h}_1^{k+1}\in\mathcal{F}_1^{k+1}} \widetilde{L}_{CE}(\mathbf{h}_1^{k+1}) - \min_{\mathbf{h}_1^{k+1}} \widetilde{L}_{CE}(\mathbf{h}_1^{k+1})}_{e_{\phi-\text{appr}}} \qquad \text{(B.196)} \\
&\leq \hat{L}_{CE}(\widetilde{\mathbf{h}}_1^{k+1}) - \widetilde{L}_{CE}(\widetilde{\mathbf{h}}_1^{k+1}) + e_{\min} + e_{\phi-\text{appr}}, \qquad \text{(B.197)}
\end{aligned}$$

where $\widetilde{\mathbf{h}}_1^{k+1} = \operatorname*{argmin}_{\mathbf{h}_1^{k+1}\in\mathcal{F}_1^{k+1}} \widetilde{L}_{CE}(\mathbf{h}_1^{k+1})$. Hence, using Hoeffding's inequality, with probability at least $1-\delta/2$, we have

$$\hat{L}_{CE}(\mathbf{f}_1^{k+1}) - \min_{\mathbf{h}_1^{k+1}} \widetilde{L}_{CE}(\mathbf{h}_1^{k+1}) \leq \sqrt{\frac{D}{2n}\log 2/\delta} + e_{\min} + e_{\phi-\text{appr}}. \tag{B.198}$$

Finally, using Lemma 13, (B.189), (B.194), (B.195), (B.198), and by union bound, we complete the proof. □

## B.6 Proof of Proposition 3

We prove this proposition in four steps: (i) we first prove that in each iteration, the deferral loss $L_{\text{def}}^{0-1}(h,r)$ is bounded, (ii) using Step (i), we show that $\Pr(X \in DIS(V_i))$

halves in each iteration with high probability, (iii) using Step (ii) we conclude that $\Pr(X \in DIS(V_{\lceil \log \frac{1}{\varepsilon} \rceil})) \leq \varepsilon$ with high probability, and finally (iv) we provide a bound on $L_{\text{def}}^{0-1}(h,r)$ using the result in Step (iii).

• **Step (i)**: We use Theorem 2 of Mozannar and Sontag (2020a) that making use of realizability of $(h,r)$ on empirical distribution shows that with probability at least $1-\delta'$ we have

$$\mathbb{E}\left[\mathbb{I}_{r(X)=0}\mathbb{I}_{h(X)\neq Y} + \mathbb{I}_{r(X)=1}\mathbb{I}_{M\neq Y} | X \in DIS(V_i)\right] \leq \sqrt{\frac{2\log 2/\delta'}{m_i}} + \sqrt{\frac{2d(\mathcal{H})\log\frac{em_i}{d(\mathcal{H})}}{m_i}} + \sqrt{\frac{32d(\mathcal{R})\log\frac{em_i}{d(\mathcal{R})}}{m_i}}, \tag{B.199}$$

where $m_i$ is the size of the set on which human provides the prediction in each iteration. Note that we draw only samples from $DIS(V_i)$, and that is the reason that we condition the loss on $X$ being in $DIS(V_i)$ .

To analyze the sample complexity that corresponds to ((B.199)), we let $\delta' = \frac{\delta}{(2+\lceil \log \frac{1}{\varepsilon} \rceil - i)^2}$ and we assume that

$$m_i \geq \max\{108\Theta^2 \log \frac{(2+\lceil \log \frac{1}{\varepsilon} \rceil - i)^2}{\delta}, 360\Theta^2 d(\mathcal{H})\log\Theta, 2,276\Theta^2 d(\mathcal{H})\log\Theta\}. \tag{B.200}$$

Using the first term in RHS of ((B.200)), we bound the first term in the upper-bound ((B.199)) as

$$\sqrt{\frac{2\log\frac{2}{\delta'}}{m_i}} \leq \sqrt{\frac{2\log\frac{2(2+\lceil \log \frac{1}{\varepsilon} \rceil - i)^2}{\delta}}{108\Theta^2 \log\frac{(2+\lceil \log \frac{1}{\varepsilon} \rceil - i)^2}{\delta}}} = \frac{1}{6\Theta}\sqrt{\frac{2}{3} + \frac{2}{3\log\frac{(2+\lceil \log \frac{1}{\varepsilon} \rceil - i)^2}{\delta}}}. \tag{B.201}$$

Then, for $i \leq \lceil \log \frac{1}{\varepsilon} \rceil$ and since $\delta \leq 1$, we know that $\log \frac{4}{\delta} \geq 2$, which concludes that

$$\sqrt{\frac{2\log\frac{2}{\delta'}}{m_i}} \leq \frac{1}{6\Theta}. \tag{B.202}$$

Further, using the second and third term in RHS of ((B.200)) and since $\sqrt{\frac{2d(\mathcal{H})\log em_i}{m_i}}$ is monotonically decreasing for $m_i \geq 2$ (note that $\frac{\partial}{\partial x}\left(\frac{\log x}{x}\right) = \frac{1}{x^2} - \frac{\log x}{x^2} \leq 0$ for $x \geq 2$) we

have

$$\sqrt{\frac{2d(\mathcal{H})\log em_i}{m_i}} \leq \sqrt{\frac{2d(\mathcal{H})\log\frac{e\Theta^2 d(\mathcal{H})\log\Theta}{d(\mathcal{H})}}{360\Theta^2 d(\mathcal{H})\log\Theta}} = \sqrt{\frac{2\log\left(e\Theta^2\cdot\log\Theta\right)}{360\Theta^2\log\Theta}} \tag{B.203}$$

$$= \frac{1}{\Theta}\sqrt{\frac{\log e + 2\log\Theta + \log\log\Theta}{180\log\Theta}}. \tag{B.204}$$

If we set $\Theta \geq e$, we have $\log\Theta \geq \log e$, and since $\log\log\Theta \leq \log\Theta$ for $\Theta \geq 1$, we have

$$\sqrt{\frac{2d(\mathcal{H})\log\frac{em_i}{d(\mathcal{H})}}{m_i}} \leq \frac{1}{\Theta}\sqrt{\frac{5\log\Theta}{180\log\Theta}} = \frac{1}{6\Theta}. \tag{B.205}$$

Similarly, we could show that

$$\sqrt{\frac{32d(\mathcal{R})\log\frac{em_i}{d(\mathcal{R})}}{m_i}} \leq \frac{1}{6\Theta}, \tag{B.206}$$

which together with ((B.199)), ((B.202)), and ((B.205)) proves that for $m_i = O\big(\Theta^2(d(\mathcal{H})\log\Theta + d(\mathcal{R})\log\Theta + \log\frac{(2+\lceil\log\frac{1}{\varepsilon}\rceil - i)^2}{\delta})\big)$ we have

$$\mathbb{E}\big[\mathbb{I}_{r(X)=0}\mathbb{I}_{h(X)\neq Y} + \mathbb{I}_{r(X)=1}\mathbb{I}_{M\neq Y}|X \in DIS(V_i)\big] \leq \frac{1}{2\Theta}, \tag{B.207}$$

with probability at least $1 - \frac{\delta}{(2+\lceil\log\frac{1}{\varepsilon}\rceil - i)^2}$.

Since $X \in DIS(V_i)$ is a necessary condition for $\mathbb{I}_{r(X)=0}\mathbb{I}_{h(X)\neq Y} + \mathbb{I}_{r(X)=1}\mathbb{I}_{M\neq Y} = 1$, we conclude that

$$L^{0-1}_{\text{def}}(h,r) = \Delta(V_i)\mathbb{E}\big[\mathbb{I}_{r(X)=0}\mathbb{I}_{h(X)\neq Y} + \mathbb{I}_{r(X)=1}\mathbb{I}_{M\neq Y}|X \in DIS(V_i)\big] \leq \frac{\Delta(V_i)}{2\Theta}, \tag{B.208}$$

with probability at least $1 - \frac{\delta}{(2+\lceil\log\frac{1}{\varepsilon}\rceil - i)^2}$, where $\Delta(V_i)$ is defined as

$$\Delta(V_i) := \Pr\big(X \in DIS(V_i)\big) \tag{B.209}$$

• **Step (ii)**: Since $L^{0-1}_{\text{def}}(h,r) = \Pr\big(r(X)M + (1-r(X))h(X) \neq Y\big)$, and because $L^{0-1}_{\text{def}}(h^*,r^*) = 0$, and using Step (i), we have

$$\Pr\big(r(X)M + (1-r(X))h(X) \neq r^*(X)M + (1-r^*(X))h^*(X)\big) \leq \frac{\Delta(V_i)}{2\Theta}, \tag{B.210}$$

with probability at least $1-\frac{\delta}{(2+\lceil\log\frac{1}{\varepsilon}\rceil-i)^2}$. As a result, for all $(h,r)\in V_{i+1}$, we have $(h,r)\in B\big((h^*,r^*),\frac{\Delta(V_i)}{2\Theta}\big)$ with such probability.

Hence, we have

$$\Delta(V_{i+1})\leq\Delta\Big(B\big((h^*,r^*),\tfrac{\Delta(V_i)}{2\Theta}\big)\Big)\leq\Theta\cdot\frac{\Delta(V_i)}{2\Theta}=\frac{\Delta(V_i)}{2}, \tag{B.211}$$

where the last inequality is followed by the definition of $\Theta$.

• **Step (iii)**: Using union bound, and since

$$\sum_{i=1}^{\lceil\log\frac{1}{\varepsilon}\rceil}\frac{\delta}{(2+\lceil\log\frac{1}{\varepsilon}\rceil-i)^2}=\sum_{i=2}^{\lceil\log\frac{1}{\varepsilon}\rceil+1}\frac{\delta}{i^2}\leq\sum_{i=2}^{\infty}\frac{\delta}{i^2}=\frac{\pi^2-6}{6}\cdot\delta\leq\delta, \tag{B.212}$$

and using Step (iii), we have that

$$\Delta(V_{\lceil\log\frac{1}{\varepsilon}\rceil})\leq\frac{1}{2^{\lceil\log\frac{1}{\varepsilon}\rceil}}\Delta(V_0)\leq\varepsilon, \tag{B.213}$$

with probability at least $1-\delta$.

• **Step (iv)**: Since we know that $L_{\text{def}}^{0-1}(h^*,r^*)=0$, we conclude that

$$\Pr\big(M\neq Y,r^*(X)=1\big)=0. \tag{B.214}$$

Next, since for all $h\in V_{\lceil\log\frac{1}{\varepsilon}\rceil}$ we have $\Pr\big(h(X)\neq Y,r(X)=0\big)=0$, we can show that

$$\begin{aligned}
L_{\text{def}}^{0-1}(h,r)&=\Pr\big(h(X)\neq Y,r(X)=0\big)+\Pr\big(M\neq Y,r(X)=1\big) && \text{(B.215)}\\
&=\Pr\big(M\neq Y,r(X)=1\big) && \text{(B.216)}\\
&=\Pr(M\neq Y,r(X)=1,r^*(X)=0)+\Pr(M\neq Y,r(X)=1,r^*(X)=1) && \text{(B.217)}\\
&\overset{(a)}{=}\Pr(M\neq Y,r(X)=1,r^*(X)=0) && \text{(B.218)}\\
&=\Pr(M\neq Y,r(X)\neq r^*(X),r^*(X)=0)\leq\Pr\big(r(X)\neq r^*(X)\big), && \text{(B.219)}
\end{aligned}$$

where $(a)$ is followed by ((B.214)).

Next, since $(h^*,r^*)$ is not removed in any iteration because of its consistency, we have $(h^*,r^*)\in V_{\lceil\log\frac{1}{\varepsilon}\rceil}$. Hence using Step (iii), for all $(h,r)\in V_{\lceil\log\frac{1}{\varepsilon}\rceil}$ we have

$$\Pr\big(r(X)\neq r^*(X)\big)\leq\Pr\big(X\in DIS(V_{\lceil\log\frac{1}{\varepsilon}\rceil})\big)\leq\varepsilon, \tag{B.220}$$

with probability at least $1-\delta$.

Using ((B.219)) and ((B.220)) the proof is complete.

## B.7 An example on which CAL algorithm fails

Here, we provide the reader with an example on which vanilla CAL algorithm in Section 3.5.1 does not converge. Let $\mathcal{X} = \{0,1\}$ and let $X \sim \text{Uniform}\{\mathcal{X}\}$ and $\mu_{XYM} = \mu_X \mathbb{I}_{Y=X} \mathbb{I}_{M=0}$, which means for all instances on $\mathcal{X}$, $Y = X$ and $M = 0$. Further, let $\mathcal{H} = \{h_1, h_2\}$, and $\mathcal{R} = \{r_1, r_2\}$, where

$$h_1(\mathbf{x}) = r_1(\mathbf{x}) = \mathbf{x},\ h_2(\mathbf{x}) = r_2(\mathbf{x}) = 0. \tag{B.221}$$

One could see that in this case three pairs $(h_1, r_1), (h_1, r_2), (h_2, r_1)$ as deferral systems provide zero loss.

To run CAL, we draw a sample from $\mu_X$. Assume that we observe $\mathbf{x} = 1$. We see that since $\mathbf{x} \in DIS(V_0) = \{1\}$, then we need to query human's prediction and true label on such instance. Hence, we collect the corresponding values $y = m = 1$ for that instance. Next, we update the version space

$$V_1 = \{(h_1, r_1), (h_1, r_2), (h_2, r_1)\}, \tag{B.222}$$

to induce consistency. However, we note that $DIS(V_1)$ does not change comparing to $DIS(V_0)$. Hence, $\Pr(X \in DIS(V_0)) = \Pr(X \in DIS(V_1)) = \ldots = \frac{1}{2}$. As a result, CAL algorithm does not converge, and in each iteration queries human prediction for $\mathbf{x} = 1$.

## B.8 Proof of Theorem 4

Using Theorem 5.1 of Hanneke (2014), we know that if $n_l = C\Theta d(\mathcal{D}) \log\left(\frac{4\Theta}{\delta} \log \frac{4}{\varepsilon}\right) \log \frac{4}{\varepsilon}$, then with probability at least $1 - \frac{\delta}{4}$ we have

$$\Pr\left[f(X) \neq \mathbb{I}_{M \neq Y}\right] \leq \frac{\varepsilon}{4}. \tag{B.223}$$

Next, we bound the empirical joint loss on unlabeled samples. We know that

$$\hat{L}^{0-1}_{\text{def}}(h,r) = \frac{1}{n_u}\sum_i \mathbb{I}_{h(x_i)\neq y_i}\mathbb{I}_{r(x_i)=0} + \mathbb{I}_{m_i\neq y_i}\mathbb{I}_{r(x_i)=1} \tag{B.224}$$

$$= \frac{1}{n_u}\sum_i \left[\mathbb{I}_{h(x_i)\neq y_i}\mathbb{I}_{r(x_i)=0} + f(x_i)\mathbb{I}_{r(x_i)=1}\right] + \frac{1}{n_u}\sum_i \left(\mathbb{I}_{m_i\neq y_i} - f(x_i)\right)\mathbb{I}_{r(x_i)=1} \tag{B.225}$$

$$\overset{(a)}{=} \frac{1}{n_u}\sum_i \left(\mathbb{I}_{m_i\neq y_i} - f(x_i)\right)\mathbb{I}_{r(x_i)=1} \tag{B.226}$$

$$\leq \frac{1}{n_u}\sum_i |\mathbb{I}_{m_i\neq y_i} - f(x_i)| \tag{B.227}$$

$$= \frac{1}{n_u}\sum_i \mathbb{I}_{f(x_i)\neq \mathbb{I}_{m_i\neq y_i}} \tag{B.228}$$

where $(a)$ holds because of Line 5 in Algorithm 1.

As a result, we use Hoeffding's inequality coupled with ((B.223)) to show that

$$\hat{L}^{0-1}_{\text{def}}(h,r) \leq \frac{\varepsilon}{4} + \sqrt{\frac{\log 2/\delta}{2n_u}}, \tag{B.229}$$

with probability at least $1-\frac{3\delta}{4}$. Further, by generalization bound in Theorem 2 of Mozannar and Sontag (2020a), with probability at least $1-\frac{\delta}{4}$ we have

$$L^{0-1}_{\text{def}}(h,r) \leq \hat{L}^{0-1}_{\text{def}}(h,r) + \sqrt{\frac{2\log 8/\delta}{n_u}} + \mathfrak{R}_{n_u}(\mathcal{H}) + 4\mathfrak{R}_{n_u}(\mathcal{R}) + \Pr(M\neq Y)e^{\frac{-n_u\Pr(M\neq Y)}{8}}, \tag{B.230}$$

where following ((B.229)) we conclude that with probability at least $1-\delta$ we have

$$L^{0-1}_{\text{def}}(h,r) \leq \frac{\varepsilon}{4} + \sqrt{\frac{\log 2/\delta}{2n_u}} + \sqrt{\frac{2\log 8/\delta}{n_u}} + \mathfrak{R}_{n_u}(\mathcal{H}) + 8\mathfrak{R}_{n_u}(\mathcal{R}) + \Pr(M\neq Y)e^{\frac{-n\Pr(M\neq Y)}{8}}. \tag{B.231}$$

One can further calculate an upper-bound on $\mathfrak{R}_{n_u}(\mathcal{H})$ and $\mathfrak{R}_{n_u}(\mathcal{R})$ using Corollary 3.8 and 3.18 of Mohri *et al.* (2018) as

$$\mathfrak{R}_{n_u}(\mathcal{H}) \leq \sqrt{\frac{2d(\mathcal{H})\log\frac{en_u}{d(\mathcal{H})}}{n_u}}, \tag{B.232}$$

and

$$\mathfrak{R}_{n_u}(\mathcal{R}) \leq \sqrt{\frac{2d(\mathcal{R})\log\frac{en_u}{d(\mathcal{R})}}{n_u}}, \tag{B.233}$$

which by substituting in ((B.231)) we conclude that

$$\begin{aligned} L_{\text{def}}^{0-1}(h,r) \leq & \frac{\varepsilon}{4} + \sqrt{\frac{\log\frac{2}{\delta}}{2n_u}} + \sqrt{\frac{2\log\frac{8}{\delta}}{n_u}} + \sqrt{\frac{2d(\mathcal{H})\log\frac{en_u}{d(\mathcal{H})}}{n_u}} + \sqrt{\frac{32d(\mathcal{R})\log\frac{en_u}{d(\mathcal{R})}}{n_u}} \\ & + \Pr(M \neq Y)e^{\frac{-n\Pr(M\neq Y)}{8}}. \end{aligned} \tag{B.234}$$

Finally, using ((B.234)) and by letting $n_u \geq \max\{\frac{8\log\frac{2}{\delta}}{\varepsilon^2}, \frac{288\log 8/\delta}{\varepsilon^2}, \frac{C'\max\{d(\mathcal{H}),d(\mathcal{R})\}\log\frac{1}{\varepsilon}}{\varepsilon^2}\}$ in which $C' = 2^{10}$ and for $\varepsilon \leq \frac{1}{2^{18}e^4}$, we have

$$L_{\text{def}}^{0-1}(h,r) \leq \varepsilon, \tag{B.235}$$

with probability at least $1-\delta$, which completes the proof.

## B.9 Experimental Details

**Data.** We use the CIFAR validation set of 10k images as the test set and split the CIFAR training set 90/10 for training and validation.

**Optimization.** We use the AdamW optimizer Loshchilov and Hutter (2017) with learning rate 0.001 and default parameters on PyTorch. We also use a cosine annealing learning rate scheduler and train for 100 epochs and saving the best performing model on the validation set. For the surrogate $L_{CE}^{\alpha}$ Mozannar and Sontag (2020a), we perform a search for $\alpha$ over a grid $[0, 0.1, 0.5, 1]$.

**Model Complexity.** For the model complexity gap figure, we use a convolutional neural network consisting of two convolutional layers with a max pooling layer in between followed by three fully connected layers with ReLU activations. We modify respectively: the number of channels produced by the convolution of the first layer and of the second layer, and the number of units in the first and second fully connected layers. We use this set of parameters to produce the plot for the classifier model:

$$[[1,1,50,25],[3,3,50,25],[4,4,80,40],[6,6,100,50],[12,12,100,50],$$
$$[20,20,100,50],[100,100,500,250],[100,100,1000,500]]$$

For the rejector model, and for the expert confidence model used for Staged we use the parameters $[100, 100, 1000, 500]$. The error bars in the plot are produced by repeating the training process 10 times and obtaining standard deviations to average over the randomness in training. We used a rather simple network architecture so that we can more easily illustrate the model complexity gap, as more complex architectures can easily obtain $\sim 100\%$ accuracy on CIFAR and would not allow us to have a more fine-grained analysis of the gap.

**Data Trade-Offs.** We use the model parameters $[100, 100, 1000, 500]$ for all networks in this plot. For each fraction of data labeled, we sample randomly from the training set the corresponding number of points. The error bars are obtained by repeating the training process 10 times for different random samplings of the training set.

# Appendix C

# Appendix III

## C.1 Lack of Compositionality of Fairness Criteria

Here, we show an example of lack of compositionality of fairness criteria for learn-to-defer problems. This falls in line with Dwork and Ilvento (2018), where the authors studied the effect of the operators such as 'OR' or 'AND'. Here, we show that a similar non-compositionality holds for the operator 'DEFER'. The following example is found based on the insight that a fair predictor is fair over all the space $\mathcal{X}$, and if it could take a decision over only a subset of $\mathcal{X}$ it will not necessarily be a fair predictor. This can be seen as a particular application of Yule's effect Ruggieri *et al.* (2023) which explains that vanishing correlation in a mixture of distributions does not necessarily concludes vanishing correlation on each of such distributions.

Let us assume that the space $\mathcal{X}$ contains only four points $x_1$, $x_2$, $x_3$, and $x_4$, and that the input takes these values with probability $\Pr(X = x_1) = \Pr(X = x_2) = \Pr(X = x_3) = \Pr(X = x_4) = \frac{1}{4}$. The first two points $x_1, x_2$ are corresponded to the demographic group $A = 0$ and the last two points are corresponded to the demographic group $A = 1$. Further, assume that the conditional target probability is $\Pr(Y = 1|x_1) = \Pr(Y = 1|x_2) = \Pr(Y = 1|x_3) = \Pr(Y = 1|x_4) = 1$. Moreover, we consider the equality of opportunity as the measure of fairness. Now, assume that the classifier $h(\cdot) : \mathcal{X} \to \{0,1\}$ is taking values $h(x_1) = 1$, $h(x_2) = 0$, $h(x_3) = 1$, and $h(x_4) = 0$ and the human decision maker predicts $M = 0$ conditioned on $x_1$, $M = 1$ conditioned on $x_2$, and $M = 1$ conditioned on $x_3$, and $M = 0$ conditioned on $x_4$. Therefore, both classifier and human expert have accuracy of $\frac{1}{2}$ on the data.

Following the above assumptions, we can find the fairness measure for classifier as

$$\begin{aligned}
&\Pr(h(X)=1|Y=1,A=0)-\Pr(h(X)=1|Y=1,A=1)\\
&=\Pr(h(X)=1|Y=1,A=0,X=x_1)\Pr(X=x_1|Y=1,A=0)\\
&\quad+\Pr(h(X)=1|Y=1,A=0,X=x_2)\Pr(X=x_2|Y=1,A=0)\\
&\quad-\Pr(h(X)=1|Y=1,A=1,X=x_3)\Pr(X=x_3|Y=1,A=1)\\
&\quad-\Pr(h(X)=1|Y=1,A=1,X=x_4)\Pr(X=x_4|Y=1,A=1)=\frac{1}{2}+0-\frac{1}{2}-0=0,
\end{aligned} \tag{C.1}$$

which means that the classifier is fully fair. We can derive a similar result for the human expert, i.e.,

$$\Pr(M=1|Y=1,A=0)-\Pr(M=1|Y=1,A=1)=0. \tag{C.2}$$

Now that we established a fair classifier and a fair expert, we take the step to find an optimal deferral solution, i.e., a deferral system that minimizes the overall loss. We can observe that for $x_1$ the classifier is accurate, while for $x_2$ the human expert is accurate. Furthermore, for $x_3$ and $x_4$ they both are equally inaccurate. Therefore, an optimal solution is not to defer for $x_1$, and defer for $x_2$, and take an arbitrary decision for $x_3$ and $x_4$. Now, if we find the fairness measure of the resulting deferral predictor, we have

$$\begin{aligned}
&\Pr(\hat{Y}=1|Y=1,A=0)-\Pr(\hat{Y}=1|Y=1,A=1)\\
&=\Pr(h(X)=1|Y=1,A=0,X=x_1)\Pr(X=x_1|Y=1,A=0)\\
&\quad+\Pr(M=1|Y=1,A=0,X=x_2)\Pr(X=x_2|Y=1,A=0)\\
&\quad-\Pr(h(X)=1|Y=1,A=1,X=x_3)\Pr(X=x_3|Y=1,A=1)\\
&\quad-\Pr(h(X)=1|Y=1,A=1,X=x_4)\Pr(X=x_4|Y=1,A=1)=\frac{1}{2}+\frac{1}{2}-\frac{1}{2}-0=\frac{1}{2},
\end{aligned} \tag{C.3}$$

or equivalently the resulting predictor is unfair for the demographic group $A=1$. This means the 'DEFER' composition of the predictors does not preserve fairness. One can further easily show that no deferral system from the above classifier and human expert that has the accuracy better than $\frac{1}{2}$ is fair.

## C.2 Extended Related Works

The deferral problem has been studied under a variety of conditions. Rejection learning Cortes *et al.* (2016); Bartlett and Wegkamp (2008); Charoenphakdee *et al.* (2021); Cheng *et al.* (2024) or selective classification El-Yaniv *et al.* (2010); Geifman and El-Yaniv (2017); Gangrade *et al.* (2021), assumes that a fixed cost is incurred to the overall

loss, when ML decides not to make a prediction on an input. The first Bayes optimal rule for rejection learning was derived in Chow (1970). Assuming that the accuracy of human, and consequently the cost of deferring to the human, can vary for different inputs, Mozannar and Sontag (2020a) obtained the Bayes optimal deferral rule. The deferral problem is further studied assuming that the number of available instances for deferral are bounded and a near-optimal classifier and deferral rule is required as a solution of empirical risk minimization De *et al.* (2020, 2021). Most recently, the implementation of deferral rules using neural networks and surrogate losses is studied for binary and multi-class classification Cao *et al.* (2022); Mozannar and Sontag (2020a); Charusaie *et al.* (2022a); Narasimhan *et al.* (2022b); Cao *et al.* (2024); Liu *et al.* (2024); Mozannar *et al.* (2023a); Mao *et al.* (2024). A possible shift in human expert for L2D methods recently studied in Tailor *et al.* (2024). The problem multi-objective L2D and rejection learning is mainly studied in an in-processing approach. A few instances of tackling such problems can be found in Okati *et al.* (2021b); Narasimhan *et al.* (2024, 2023) and Yin *et al.* (2023); Lee *et al.* (2021) for L2D and rejection learning, respectively.

Neyman-Pearson's fundamental lemma is introduced in Neyman and Pearson (1933) originally for binary hypothesis testing and later was generalized in Neyman and Pearson (1936) to give a close-form formulation for a variety of binary constrained optimization problems. Later, Dantzig and Wald (1951) found conditions for which Neyman and Pearson solution exists and is unique. The generalization of the empirical solution to Neyman-Pearson problem is studied in two lines of works: (i) the generalization of direct (in-processing) solutions to the optimization problem Scott and Nowak (2005); Scott (2007); Rigollet and Tong (2011), and (ii) the generalization of plug-in methods Tong (2013) that first approximate the score functions and then use Neyman-Pearson lemma to approximate the predictor. The generalization of Neyman-Pearson lemma to multiclass setting is first empirically studied in Landgrebe and Duin (2005) and under strong duality assumption is proved in Tian and Feng (2021). Our lemma $d$-GNP extends these works in order to (i) be able to control a general set of constraints instead of Type-$K$ errors, and (ii) be valid in absence of strong duality assumption. Further, the idea of using Neyman-Pearson lemma for controlling fairness criteria originally dates back to Zeng *et al.* (2022) (later as Zeng *et al.* (2024)). More recently, a similar post-processing method is introduced in Chen *et al.* (2023b) using cost-sensitive learning and strong duality technique. Although these works cover binary classification problem, in this paper we focus on solving multi-class classification problem, and particularly in a deferral system.

Lastly, this work differs from multi-class classification with complex performance metrics Narasimhan *et al.* (2022a) in the sense that they consider constraints that are non-linear functions of confusion matrix, while ignoring the dependence on input $x$. In our setting, the constraints are linear in terms of confusion matrix when conditioned on the input, but the linear coefficients vary with the input.

## C.3 Rephrasing ((4.2)) into Linear Functional Programming

Here, we first characterize functions that are outcome-dependent. To that end, we define $\iota(x)$ as

$$\iota = \big[\mathbb{I}_{r(x)=0}\mathbb{I}_{h(x)=1}, \ldots, \mathbb{I}_{r(x)=0}\mathbb{I}_{h(x)=L}, \mathbb{I}_{r(x)=1}\big]. \tag{C.4}$$

This function can retrieve the value of $r(x)$ and can retrieve the value of $h(x)$ only if $r(x) = 0$. In fact, we can obtain $r(x) = \big(\iota(x)\big)(L+1)$ and $h(x) = i$ if $r(x) = 0$ and $\big(\iota(x)\big)(i) = 1$. Therefore, for a function $\overline{\Psi}(x, h(x), r(x)) = \mathbb{E}_{Y,M|X=x}\big[\Psi(x, Y, M, h(x), r(x))\big]$ and $\overline{\ell}_{\text{def}}(x, h(x), r(x)) = \mathbb{E}_{Y,M|X=x}\big[\ell_{\text{def}}(x, Y, M, h(x), r(x))\big]$ to be outcome dependent, it must only be a function of $x$ and $\iota(x)$. In fact, we must have

$$\overline{\Psi}_i(x, h(x), r(x)) = \Psi'_i(x, \iota(x)), \tag{C.5}$$

and

$$\overline{\ell}_{\text{def}}(x, h(x), r(x)) = \ell'_{\text{def}}(x, \iota(x)), \tag{C.6}$$

for a choice of $\Psi'$ and $\ell'_{\text{def}}$, where $\overline{\ell}_{\text{def}}(x, h(x), r(x)) = \mathbb{E}_{Y,M|X=x}\big[\ell_{\text{def}}(x, Y, M, h(x), r(x))\big]$.

Now, we can check that $\iota(x)$ can take $L+1$ different values, in each of which one of its components takes the value 1 and others take the value 0. Therefore, by conditioning on each of these $L+1$ values we have

$$\Psi'_i(x, \iota(x)) = \sum_{i=1}^{L+1} \Psi'(x, [0, \ldots, \underbrace{1}_{i}, \ldots, 0])\Big(\big(\iota(x)\big)(i)\Big) = \langle \iota(x), \psi_i(x) \rangle, \tag{C.7}$$

where $\psi_i(x)$ is defined as

$$\begin{aligned} \psi_i(x) &= \Big[\Psi'_i(x, [1, 0, \ldots, 0]), \ldots, \Psi'(x, [0, 0, \ldots, 1])\Big] \\ &= \big[\overline{\Psi}_i(x, 1, 0), \ldots, \overline{\Psi}_i(x, L, 0), \overline{\Psi}_i(x, 0, 1)\big]. \end{aligned} \tag{C.8}$$

Similarly, we can show that

$$\ell'_{\text{def}}(x, \iota(x)) = \langle \iota(x), \vec{\ell}_{\text{def}}(x) \rangle, \tag{C.9}$$

where $\vec{\ell}_{\text{def}}(x)$ is defined as

$$\vec{\ell}_{\text{def}}(x) = \big[\overline{\ell}_{\text{def}}(x, 1, 0), \ldots, \overline{\ell}_{\text{def}}(x, L, 0), \overline{\ell}_{\text{def}}(x, 0, 1)\big]. \tag{C.10}$$

Next, we know that due to the randomization of $\mathcal{A}$, the vector $\iota(x)$ can take various

values on each instance $x$. This, however, is not the case for $\psi_i(x)$ and $\vec{\ell}_{\text{def}}(x)$, since they are defined independent of $r(x)$ and $h(x)$. Therefore, the average of constraints and loss can be rewritten as

$$\mathbb{E}_{(r,h)\sim\mathcal{A}}\big[\overline{\Psi}_i(x,h(x),r(x))\big] = \mathbb{E}_{(r,h)\sim\mathcal{A}}\big[\langle\psi_i(x),\iota(x)\rangle\big] = \langle f(x),\psi_i(x)\rangle, \tag{C.11}$$

and

$$\mathbb{E}_{(r,h)\sim\mathcal{A}}\big[\ell_{\text{def}}(x,h(x),r(x))\big] = \mathbb{E}_{(r,h)\sim\mathcal{A}}\big[\langle\vec{\ell}_{\text{def}}(x),\iota(x)\rangle\big] = \langle f(x),\vec{\ell}_{\text{def}}(x)\rangle, \tag{C.12}$$

where $f(x)$ is defined as

$$f(x) = \mathbb{E}[\iota(x)] = \big[\Pr(r(x)=0,h(x)=1),\ldots,\Pr(r(x)=0,h(x)=L),\Pr(r(x)=1)\big]. \tag{C.13}$$

Therefore, the optimization problem in ((4.2)) is effectively reduced to the linear programming problem in ((4.3)). Moreover, if $f^*(x)$ is the solution to that linear program, then the corresponding $r(x)$ should be distributed as $\Pr(r(x)=1) = \big(f^*(x)\big)(L+1)$, where $h(x)$ should be distributed as $\Pr(h(x)=i) = \Pr(h(x)=i|r(x)=0) = \frac{\big(f(x)\big)(i)}{\sum_{i=1}^{L}\big(f(x)\big)(j)}$. Note that the assumption of independence of $h(x)$ and $r(x)$ comes with no loss of generality, since the value of $h(x)$ does not variate the loss or constraints in the system when we have $r(x)=1$.

## C.4 Derivation of Embedding Functions

In this appendix we derive the embedding functions in Table 4.1 that are corresponded to the constraints of choice, as named in Section 4.3. The trick that we use for all these constraints is that we first rewrite the constraint in terms of the expected value of a function over the randomness of the algorithm $\mathcal{A}$ and the input variable $X$, and then we use ((C.8)) to transform that function into the embedding function.

- **Overall Loss**: To find the embedding function that is corresponded to the overall loss of the system, we should first note that by loss we mean the probability of incorrectness of $\hat{Y}$. Therefore, the corresponding $\ell_{def}(x,h(x),r(x))$ in this case, as defined in ((4.1)) is obtained as

$$\begin{aligned}\overline{\ell}_{def}(x,h(x),r(x)) =& \mathbb{E}_{Y,M|X=x}\big[\mathbb{I}_{r(x)=1}\mathbb{I}_{M\neq Y} + \mathbb{I}_{r(x)=0}\mathbb{I}_{h(x)\neq Y}\big] \\ =& \mathbb{I}_{r(x)=1}\Pr(M=Y|X=x) + \mathbb{I}_{r(x)=0}\Pr(Y\neq h(x)|X=x).\end{aligned}$$

Therefore, using ((C.10)) we find $\vec{\ell}_{\text{def}}$ as

$$\vec{\ell}_{\text{def}} = \big[\Pr(Y \neq 1|X = x), \ldots, \Pr(Y \neq n|X = x), \Pr(Y \neq M|X = x)\big].$$

- **Expert intervention budget**: In this case, similar to the case before, we first derive $\overline{\Psi}(x, h(x), r(x))$. To that end, we first note that the expert intervention constraint in Section 4.3 is equivalent with

$$\Pr(r(X) = 1) = \mathbb{E}_{x\sim\mu_X,(r,h)\sim\mathcal{A}}\big[\mathbb{I}_{r(x)=1}\big] \leq \delta,$$

which in turn suggests that

$$\overline{\Psi}(x, h(x), r(x)) = \mathbb{I}_{r(x)=1}.$$

Next, we find $\psi(x)$ using ((C.8)), as

$$\psi(x) = [0, \ldots, 0, 1].$$

- **OOD Detection**: To obtain the corresponding embedding function to the OOD detection constraint in Section 4.3, we can rewrite $\Pr_{\text{out}}\big(r(X) = 1\big)$ as

$$\Pr_{\text{out}}\big(r(X) = 1\big) = \mathbb{E}_{X\sim f_X^{\text{out}},(r,h)\sim\mathcal{A}}\big[\mathbb{I}_{r(X)=1}\big] = \mathbb{E}_{X\sim\mu_{X^{\text{in}}},(r,h)\sim\mathcal{A}}\big[\tfrac{\mathbb{I}_{r(X)=1} f_X^{\text{out}}(X)}{f_X^{\text{in}}(X)}\big],$$

where the last equation holds when $X$ and $X_{\text{out}}$ are absolutely continuous distributions, and therefore have probability density functions. A similar assumption is made by Narasimhan *et al.* (2023). This results in $\overline{\Psi}(x, h(x), r(x))$ being obtained as

$$\overline{\Psi}(x, h(x), r(x)) = \frac{\mathbb{I}_{r(x)=1} f_X^{\text{out}}(X)}{f_X^{\text{in}}(X)}.$$

Therefore, we conclude that the embedding function can be calculated using ((C.8)) as

$$\psi(x) = \big[0, \ldots, 0, \tfrac{f_X^{\text{out}}(X)}{f_X^{\text{in}}(X)}\big].$$

In the simple case that $f_X^{\text{out}}(x) = \frac{f_X^{\text{in}}(x)\mathbb{I}_{f_X^{\text{in}}(x)\leq\varepsilon}}{\int f_X^{\text{in}}(x)\mathbb{I}_{f_X^{\text{in}}(x)\leq\varepsilon}\mathrm{d}x}$, the embedding function is equal to

$$\psi(x) = \big[0, \ldots, 0, \frac{\mathbb{I}_{f_X^{\text{in}}(x)\leq\varepsilon}}{\Pr_{\text{in}}(f_X^{\text{in}}(X) \leq \varepsilon)}\big].$$

- **Long-Tail Classification**: This methodology aims to minimize the balanced loss

$$\frac{1}{K}\sum_{i=1}^{K}\Pr(Y \neq h(X)|r(X)=0, Y \in G_i).$$

However, as mentioned in Narasimhan *et al.* (2024), this optimization problem can be rewritten as

$$\sum_{i=1}^{K}\frac{\Pr(Y \neq h(X), r(X)=0|Y \in G_i)}{\alpha_i}, \quad \text{s.t.} \quad \Pr(r(X)=0|Y \in G_i) = \frac{\alpha_i}{K}.$$

Therefore, the objective can rewritten as

$$\sum_{i=1}^{K}\frac{\mathbb{E}_{(r,h)\sim\mathcal{A},X'\sim\mu_X}\big[\Pr(Y \neq h(X), r(X)=0, Y \in G_i|X=X')\big]}{\alpha_i \Pr(Y \in G_i)},$$

which together with ((C.8)) shows that

$$\psi_0(x) = -\Big[\sum_{i=1}^{K}\frac{\Pr(Y \neq 1, Y \in G_i|X=x)}{\alpha_i \Pr(Y \in G_i)}, \ldots, \sum_{i=1}^{K}\frac{\Pr(Y \neq L, Y \in G_i|X=x)}{\alpha_i \Pr(Y \in G_i)}, 0\Big].$$

The reason that we use negative sign is because in the definition of ((4.3)) we aim to maximize the objective.

Similarly, we can rewrite the objectives as

$$\frac{\mathbb{E}_{(r,h)\sim\mathcal{A},X'\sim\mu_X}\big[\Pr(r(X)=0, Y \in G_i|X=X') - \frac{\alpha_i}{K}\Pr(Y \in G_i)\big]}{\Pr(Y \in G_i)}.$$

Therefore, using ((C.8)) we can obtain $\psi_i(x)$ as

$$\psi_i(x) = \frac{\Pr(Y \in G_i|X=x)}{\Pr(Y \in G_i)}\Big[1, \ldots, 1, 0\Big] - \frac{\alpha_i}{K}. \tag{C.14}$$

- **Type-$k$ Error Bound**: We first rewrite Type-$k$ constraint in 4.3 as

$$\begin{aligned}\Pr(\hat{Y} \neq k|Y=k) &= \frac{\Pr(\hat{Y} \neq k, Y=k)}{\Pr(Y=k)} \\ &\overset{(a)}{=} \frac{\mathbb{E}_{X\sim\mu_X}\big[\Pr(\hat{Y} \neq k, Y=k|X=x)\big]}{\Pr(Y=k)} \\ &= \frac{\mathbb{E}_{X\sim\mu_X}\big[\Pr(\hat{Y} \neq k|Y=k, X=x)\Pr(Y=k|X=x)\big]}{\Pr(Y=k)},\end{aligned} \tag{C.15}$$

where $(a)$ is followed by chain rule.

Next, we condition $\Pr(\hat{Y} \neq k|Y = k, X = x)$ on $r(X)$ being 1 and 0, which concludes that

$$\begin{aligned}
\Pr(\hat{Y} \neq k|Y = k, X = x) &= \Pr(\hat{Y} \neq k, r(x) = 1|Y = k, X = x) \\
&\quad + \Pr(\hat{Y} \neq k, r(x) = 0|Y = k, X = x) \\
&= \Pr(M \neq k, r(x) = 1|Y = k, X = x) \\
&\quad + \Pr(h(x) \neq k, r(x) = 0|Y = k, X = x) \\
&= \mathbb{E}_{(r,h)\sim\mathcal{A},M|X=x,Y=k}\left[\mathbb{I}_{M\neq k}\mathbb{I}_{r(x)=1} + \mathbb{I}_{h(x)\neq k}\mathbb{I}_{r(x)=0}\right] \\
&= \mathbb{E}_{(r,h)\sim\mathcal{A}|X=x,Y=k}\Big[\Pr(M \neq k|X = x, Y = k)\mathbb{I}_{r(x)=1} \\
&\qquad\qquad + \mathbb{I}_{h(x)\neq k}\mathbb{I}_{r(x)=0}\Big].
\end{aligned}$$

Therefore, using ((C.15)) we conclude that

$$\begin{aligned}
\Pr(\hat{Y} \neq k|Y = k) = &\frac{\mathbb{E}_{X'\sim\mu_X,(r,h)\sim\mathcal{A}}\left[\mathbb{I}_{r(X)=1}\Pr(M \neq k, Y = k|X = X')\right]}{\Pr(Y = k)} \\
&+ \frac{\mathbb{E}_{X'\sim\mu_X,(r,h)\sim\mathcal{A}}\left[\mathbb{I}_{h(X')\neq k}\mathbb{I}_{r(X')=0}\Pr(Y = k|X = X')\right]}{\Pr(Y = k)},
\end{aligned}$$

which together with ((C.8)) shows that the embedding function is obtained as

$$\psi(x) = \frac{\Pr(Y = k|X = x)}{\Pr(Y = k)}\Big[1, \ldots, 1, \underbrace{0}_{k}, 1, \ldots, 1, \Pr(M \neq k|X = x, Y = k)\Big].$$

Note that here we used the assumption that $(Y, M)$ and $\mathcal{A}$ are independent for each choice of $X$, i.e., the value noise that is introduced in $\mathcal{A}$ for each $X = x$ is generated independent of the value of $Y$ and $M$, which is the true assumption, since the algorithm only has access to $X$ and not true label or the human label.

- **Demographic Parity**: We know that the demographic parity constraint in Section 4.3 can be written as

$$-\delta \leq \Pr(\hat{Y} = 1|A = 0) - \Pr(\hat{Y} = 1|A = 1) \leq \delta. \tag{C.16}$$

Here, we find the corresponding embedding function $\psi(x)$ for the upper-bound in the above inequality. For the lower-bound, we can use $-\psi(x)$ and follow the steps that are proposed in the main text of the manuscript.

To find the embedding function that corresponds to the upper-bound of ((C.16)),

we first rewrite $\Pr(\hat{Y}=1|A=0)-\Pr(\hat{Y}=1|A=1)$ as

$$\Pr(\hat{Y}=1|A=0)-\Pr(\hat{Y}=1|A=1)=\frac{\Pr(\hat{Y}=1,A=0)}{\Pr(A=0)}-\frac{\Pr(\hat{Y}=1,A=1)}{\Pr(A=1)}. \tag{C.17}$$

Now, similar to what we did in previous section, we condition $\Pr(\hat{Y}=1,A=a)$ for $a\in\{0,1\}$ on the value of $h(x)$ and $r(x)$, and we conclude

$$\begin{aligned}\Pr(\hat{Y}=1,A=a)&=\Pr(\hat{Y}=1,A=a,r(X)=1)+\Pr(\hat{Y}=1,A=a,r(X)=0)\\&=\Pr(M=1,A=a,r(X)=1)+\Pr(h(X)=1,A=a,r(X)=0)\\&=\mathbb{E}_{X,A,M,\mathcal{A}}\big[\mathbb{I}_{M=1}\mathbb{I}_{A=a}\mathbb{I}_{r(X)=1}+\mathbb{I}_{h(X)=1}\mathbb{I}_{A=a}\mathbb{I}_{r(X)=0}\big]\\&=\mathbb{E}_{X,\mathcal{A}}\big[\Pr(M=1,A=a|X=x)\mathbb{I}_{r(X)=1}\\&\qquad+\Pr(A=a|X=x)\mathbb{I}_{h(X)=1}\mathbb{I}_{r(X)=0}\big].\end{aligned} \tag{C.18}$$

Here, we used the assumption of independence of $X$ and $(M,Y)$ given a choice of $X$.

As a result of ((C.17)), ((C.18)), and ((C.8)) we can find the embedding function as

$$\begin{aligned}\psi(x)=\Big[&0,\frac{\Pr(A=1|X=x)}{\Pr(A=1)}-\frac{\Pr(A=0|X=x)}{\Pr(A=0)},\\&\frac{\Pr(M=1,A=1|X=x)}{\Pr(A=1)}-\frac{\Pr(M=1,A=0|X=x)}{\Pr(A=0)}\Big].\end{aligned}$$

- **(In-)Equality of Opportunity**: Similar to the previous items, we rewrite equality of opportunity constraint in Section 4.3 as

$$-\delta\leq\Pr(\hat{Y}=1|Y=1,A=1)-\Pr(\hat{Y}=1|Y=1,A=0)\leq\delta.$$

Again, we only consider the upper-bound and rewrite $\Pr(\hat{Y}=1|Y=1,A=1)-\Pr(\hat{Y}=1|Y=1,A=0)$ as

$$\begin{aligned}&\Pr(\hat{Y}=1|Y=1,A=1)-\Pr(\hat{Y}=1|Y=1,A=0)\\&\qquad=\frac{\Pr(\hat{Y}=1,Y=1,A=1)}{\Pr(Y=1,A=1)}-\frac{\Pr(\hat{Y}=1,Y=1,A=0)}{\Pr(Y=1,A=0)}.\end{aligned} \tag{C.19}$$

Next, by conditioning on $r(X)=1$ and $r(X)=0$, we rewrite $\Pr(\hat{Y}=1,Y=1,A=$

$a$) for $a \in \{0,1\}$ as

$$\begin{aligned}
\Pr(\hat{Y}=1,Y=1,A=a) &= \Pr(\hat{Y}=1,Y=1,A=a,r(X)=1) \\
&\quad + \Pr(\hat{Y}=1,Y=1,A=a,r(X)=0) \\
&= \Pr(M=1,Y=1,A=a,r(X)=1) \\
&\quad + \Pr(h(X)=1,Y=1,A=a,r(X)=0) \\
&= \mathbb{E}_{X,Y,M,A,\mathcal{A}}\big[\mathbb{I}_{M=1}\mathbb{I}_{Y=1}\mathbb{I}_{A=a}\mathbb{I}_{r(X)=1} \\
&\qquad + \mathbb{I}_{h(X)=1}\mathbb{I}_{Y=1}\mathbb{I}_{A=a}\mathbb{I}_{r(X)=0}\big] \\
&= \mathbb{E}_{X,\mathcal{A}}\big[\mathbb{I}_{r(X)=1}\Pr(M=1,Y=1,A=a|X=x) \\
&\qquad + \mathbb{I}_{h(X)=1}\mathbb{I}_{r(X)=0}\Pr(Y=1,A=a|X=x)\big],
\end{aligned} \tag{C.20}$$

where the last identity is followed by the assumption of independence of $\mathcal{A}$ and $(Y,M,A)$ given an instance $X=x$.

As a result of ((C.19)), ((C.20)), and ((C.8)) we can obtain the embedding function as

$$\begin{aligned}
\psi(x) = \Big[0, &\frac{\Pr(Y=1,A=1|X=x)}{\Pr(Y=1,A=1)} - \frac{\Pr(Y=1,A=0|X=x)}{\Pr(Y=1,A=0)} \\
&\frac{\Pr(M=1,Y=1,A=1|X=x)}{\Pr(Y=1,A=1)} - \frac{\Pr(M=1,Y=1,A=0|X=x)}{\Pr(Y=1,A=0)}\Big].
\end{aligned}$$

- **(In-)Equality of Odds**: This induces the same constraint as that of equality of opportunity, and further induces an extra constraint that is in nature similar to equality of opportunity with the difference that it uses $Y=0$ instead of $Y=1$. Therefore, we have two embedding functions, one is similar to that of equality of opportunity as

$$\begin{aligned}
\psi_1(x) = \Big[0, &\frac{\Pr(Y=1,A=1|X=x)}{\Pr(Y=1,A=1)} - \frac{\Pr(Y=1,A=0|X=x)}{\Pr(Y=1,A=0)} \\
&\frac{\Pr(M=1,Y=1,A=1|X=x)}{\Pr(Y=1,A=1)} - \frac{\Pr(M=1,Y=1,A=0|X=x)}{\Pr(Y=1,A=0)}\Big],
\end{aligned}$$

and another similar to that with changing $Y=1$ into $Y=0$, and therefore as

$$\begin{aligned}
\psi_2(x) = \Big[&\frac{\Pr(Y=0,A=1|X=x)}{\Pr(Y=0,A=1)} - \frac{\Pr(Y=0,A=0|X=x)}{\Pr(Y=0,A=0)}, 0 \\
&\frac{\Pr(M=1,Y=0,A=1|X=x)}{\Pr(Y=0,A=1)} - \frac{\Pr(M=1,Y=0,A=0|X=x)}{\Pr(Y=0,A=0)}\Big].
\end{aligned}$$

## C.5 Limitations of Cost-Sentitive Methods

A variety of works have tackled constrained classification problems using cost-sensitive modeling Lehmann *et al.* (1986); Chzhen *et al.* (2019); Okati *et al.* (2021b). In other words, they use the expected loss that is penalized with the constraints and solve that for certain coefficients for those constraints (a.k.a., they form Lagrangian from that problem). In the next step, they optimize the coefficients and obtain the optimal predictor. The issue that we discuss further in the following we concern is that during this process, the optimal predictor is achieved only when the corresponding cost-sensitive Lagrangian has a single saddle point in terms of coefficients and predictors. Such assumption, unless by analyzing the Lagrangian closely, is hard to be validated. However, our results in this paper have no such assumption, and instead use statistical hypothesis testing methods to show their optimality.

To further clarify the issue with such methodology, we bring an example of L2D problem when human intervention budget is controlled. Suppose that the features in $\mathcal{X}$ are distributed with an atomless probability measure $\mu_X$ (e.g., normal or uniform distribution).[1] Further, assume that the human has perfect information of the label, i.e. $Y = M$, while the input features have no information of the label, i.e., $\Pr(Y = 1|X = x) = 1/2$ for all $x \in \mathcal{X}$. Moreover, let the classifier and the human induce the $0-1$ loss function. In this case, we can see that the optimal classifier is the maximizer of the scores (see the early discussion of Section C.7), which in this case, since there is no clear maximizer, without loss of generality can be set to $h(x) \equiv 1$.

For such assumptions, if we write the Lagrangian in form of

$$L(\lambda, r) = L^{\mu}_{\text{def}}(h, r) + \lambda(\mathbb{E}[r(X)] - b) = \frac{1}{2} - \frac{1}{2}\mathbb{E}\big[r(X)\big] + \lambda(\mathbb{E}[r(X)] - b),$$

then strong duality shows that

$$\min_{r\in[0,1]^{\mathcal{X}}} \max_{\lambda\geq 0} L(\lambda, r) = \max_{\lambda\geq 0} \min_{r\in[0,1]^{\mathcal{X}}} L(\lambda, r), \tag{C.21}$$

or to put it informally, the objective is invariant under the interchange of minimum and maximum over Lagrange multipliers and the variable of interest. However, this does not prove the interchangeability of the saddle points in these settings, i.e., we cannot guarantee $\operatorname{argmin}_{r\in[0,1]} L(\lambda^*, r) = f^*$, where $\lambda^* \in \operatorname{argmax}_{\lambda} \min_{r\in[0,1]} L(\lambda, r)$, and $f^* \in \operatorname{argmin}_{r\in[0,1]} \max_{\lambda} L(\lambda, r)$. In fact, this guarantee holds only in particular examples, e.g., when $L(\lambda^*_r, r)$ is strictly convex (Boyd *et al.*, 2011, page 8).

In fact, if we optimize $r$ for all $\lambda$ as in RHS of ((C.21)), we can show that $r_\lambda(x) = \begin{cases} 1 & \lambda < \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$. Therefore, $\lambda^*$ can be obtained as $\lambda^* = \operatorname*{argmax}_{\lambda\geq 0} (\lambda - 1/2)^- - \lambda b$

[1] If we have a probability measure that contains atoms, one can follow the same steps for the first counterexample.

where $(x)^- := \min\{x, 0\}$. This can be rewritten as

$$\lambda^* = \underset{\lambda \geq 0}{\operatorname{argmax}} \begin{cases} -\frac{1}{2} - \lambda(b-1) & 0 \leq \lambda \leq \frac{1}{2} \\ -\lambda b & \lambda > \frac{1}{2} \end{cases} = \frac{1}{2}.$$

Hence, the condition $\lambda < 1/2$ is never satisfied and we have $r_{\lambda^*}(x) = 0$, i.e., we should never defer. For the deferral rule $r_{\lambda^*}$, the deferral loss ((4.1)) is

$$L^{\mu}_{\text{def}}(h, \hat{f}) = \mathbb{E}_{X,Y,M}\big[\ell_{AI}(Y, h(X), X)\big] = \frac{1}{2}.$$

To show that $r_{\lambda^*}$ is not optimal, we provide random and deterministic deferral rules $f^*$ and $r^{**}$ that satisfy the constraint in ((4.2)), while having a smaller deferral loss:

- ◇ Let $f^*(x) = b$, that is a random deferral rule that defers with probability $b$ everywhere on $\mathcal{X}$. Therefore, on average $b$ proportion of inquiries are deferred and thus it satisfies the constraint in ((4.2)). The deferral loss for $f^*(x)$ is equal to

$$\begin{aligned} L^{\mu}_{\text{def}}(h, f^*) &= \underbrace{\mathbb{E}[r(X)]}_{b} \cdot \underbrace{\mathbb{E}[\ell_H(Y, M)]}_{0} \\ &\quad + \underbrace{\mathbb{E}[1 - r(X)]}_{1-b} \cdot \underbrace{\mathbb{E}[\ell_{AI}(Y, h(X))]}_{\frac{1}{2}} \\ &= \frac{1-b}{2} < \frac{1}{2}. \end{aligned}$$

- ◇ The second example is a deterministic deferral rule. Since the probability measure on $\mathcal{X}$ is atomless, for all $b \in [0, 1]$ there exists a set $\mathcal{A}$ such that $\Pr(X \in \mathcal{A}) = b$ (Fremlin, 2000, Proposition 215D). Hence, defining $r^{**}(x) = \mathbb{1}_{x \in \mathcal{A}}$ the constraint in ((4.2)) is met. Similar to the last example $L^{\mu}_{\text{def}}(h, r^{**}) = \frac{1-b}{2} < \frac{1}{2}$.

The above two examples show that the deferral rule $r_{\lambda^*}$ is sub-optimal. The reason is that, for optimality of $r_{\lambda^*}$ we should make sure that $L(\lambda^*_r, r)$ has a single minimizer of $r$. However, in our example we had $L(\frac{1}{2}, r) = -\lambda b$ has infinite number of minimizers in terms of $r(x)$. Therefore, the equality of the solutions to minimax problem and maximin problem is not guaranteed.

## C.6 On Failure of In-Processing Methods

One might think that the need of using post-processing methods does not necessarily appear in some examples of multi-objective L2D problem. As an instance, for the expert intervention budget we can rank samples based on the difference between machine and human loss and defer the top $b$-proportion of samples for which the machine loss is

higher than the human one. This method is illustrated in Algorithm 3. Indeed, in the following we show that the sub-optimality of such deterministic deferral rule, compared to the optimal deferral rule diminishes as the size of training set increases.

**Algorithm 3:** Deterministic Algorithm for Deferring Tasks to Human or AI for the Empirical Distribution and Expert Intervention Budget

**Input**: The dataset $\mathcal{D}$, he human and classifier loss $\ell_H$ and $\ell_{AI}$ and available proportion $b$ of instances to defer

**Output**: Labels of "defer" or "no defer" for each instance in $\mathcal{D}$

**procedure** DEFERTASKS($\mathcal{D}, \ell_H, \ell_{AI}, b$)
  Make the set $A = \{(x,y,m) \in \mathcal{D} : \ell_H(y,m) - \ell_{AI}(y,h(x)) \leq 0\}$ **if** $|A| \geq b|\mathcal{D}|$ **then**
    Defer all tasks in $A$ to human **else**
    Defer the $b|\mathcal{D}|$ tasks with the lowest $\ell_H(x,y,m) - \ell_{AI}(x,y)$ to human
  END
**end procedure**

**Theorem 16** (Optimal Deferral for Empirical Distribution)**.** *For a classifier $h(x)$ and dataset $\mathcal{D} = \{(x_i, y_i, m_i)\}_{i=1}^n$, where we assume $x_i \neq x_j$, $i \neq j$, the deterministic deferral rule as in Algorithm 3 is (i) the optimal deterministic deferral rule for the empirical distribution on $\mathcal{D}$ and bounded expert intervention budget, and (ii) at most $\frac{1}{n}$-suboptimal (in terms of deferral loss) compared to the optimal random deferral rule for the empirical distribution on $\mathcal{D}$.*

Next in the following, we show that such policy does not provide sufficient information to determine the optimal deferral rule for the true distribution. To that end, we first recall that in classification tasks, the optimal classifier typically thresholds an estimation of conditional probability of the label $Y$ given $X$ that is obtained using the available dataset. As a result, if we observe enough pairs of $(x_i, y_i)$, then we improve upon such estimation of conditional probability and increase the accuracy of the obtained classifier. However, we argue that this paradigm is inapplicable in the case of deferral rule as follows.

Although the output $\hat{r}$ of Algorithm 3 for each feature $x$ is a deterministic 0 or 1 label, it varies with the choice of the dataset $\mathcal{D}$. Hence, if we draw datasets from a distribution $\mu$, the outcome of $\hat{r}$ becomes probabilistic. In the following, we introduce two probability distributions $\mu_1$ and $\mu_2$ over $(X,Y,M)$ such that for random draws of the dataset from $\mu_i$, the conditional probability of such optimal deferral label $\hat{r}$ given $X$ is equal, yet the optimal deferral rule for the true distribution is different.

Although the following discussion bears some resemblance with the No-Free-Lunch theorem (e.g. Shalev-Shwartz and Ben-David, 2014), there exists the following difference between the two. The No-Free-Lunch theorem states that for each learning algorithm, there exists a data distribution on which the algorithm does not generalize well.

However, in the following discussion, we assume that we can observe infinite number of datasets and indeed, we can find the underlying probability of the deferral labels. In fact, we show that the limiting factor for finding the optimal deferral for the true distribution is that we only use deferral labels and if we use values of both losses, we can accordingly find the optimal deferral rule as suggested in Theorem 6.

Assume that we have a dataset $\mathcal{D} = \{(x_i, y_i, m_i)\}_{i=1}^n$ that contains i.i.d. samples from the distribution $\mu_{XYM}$. Further, assume that we have no budget constraint, that is $b = 1$ in Algorithm 3. Therefore, the optimal randomized deferral rule over the empirical distribution is the solution of the problem

$$\min_{\hat{r}_i \in [0,1]} \sum_{i=1}^{n} \mathbb{1}_{m_i \neq y_i} \hat{r}_i + \mathbb{1}_{h(x_i) \neq y_i} \big(1 - \hat{r}_i\big).$$

It is easy to see that the solution to this problem is given by $\hat{r}_i = 0$ if $\mathbb{1}_{h(x_i)\neq y_i} < \mathbb{1}_{m_i \neq y_i}$ and $\hat{r}_i = 1$ if $\mathbb{1}_{h(x_i)\neq y_i} > \mathbb{1}_{m_i \neq y_i}$. As a result, the optimal deferral is obtained as

$$\hat{r}_i = \begin{cases} 1 & m_i = y_i \,, h(x_i) \neq y_i \\ 0 & m_i \neq y_i, h(x_i) = y_i \\ \text{any value in } [0,1] & o.w. \end{cases} . \tag{C.22}$$

Among all the possible policies, we can choose

$$\hat{r}_i = \begin{cases} 1 & m_i = y_i \,\&\, h(x_i) \neq y_i \\ 0 & o.w. \end{cases} .$$

Next, we assume that we have a classifier $h$ and two probability distributions $\mu_1$ and $\mu_2$ over $(X, Y, M)$. For both distributions $X$ is uniformly distributed over $[0,1]$, and we have $\mu_1(Y = M, h(X) = Y) = \frac{2}{3}, \mu_1(Y = M, h(X) \neq Y) = \frac{1}{3}$ and $\mu_2(Y \neq M, h(X) = Y) = \frac{2}{3}, \mu_2(Y = M, h(X) \neq Y) = \frac{1}{3}$. We can see that although the observed $\hat{r}$s are fixed for a given choice of $\mathcal{D}$, since $\mathcal{D}$ is randomly drawn, $\hat{r}$ values are randomly distributed. Furthermore, the distribution of $\Pr(\hat{r}|X)$ is according to $Bern(\frac{1}{3})$, since in both cases we have $\mu_i(Y = M, h(X) \neq Y) = \frac{1}{3}$. However, the optimal deferral rule for the first distribution is $r_1^*(x) = 1$ for all $x \in \mathcal{X}$, since we have $L_{\text{def}}^{\mu_1}(h, r_1^*) = 0$, while for the second case the optimal deferral rule is $r_2^*(x) = 0$ for all $x \in \mathcal{X}$ because we have $L_{\text{def}}^{\mu_2}(h, r_2^*) = \frac{1}{3}$. Furthermore, such deferral rules are not interchangeable, because we have $L_{\text{def}}^{\mu_1}(h, r_2^*) = L_{\text{def}}^{\mu_2}(h, r_1^*) = \frac{2}{3}$. As a result, $\Pr(\hat{r}|X)$ does not provide sufficient information for obtaining optimal deferral rule on true distribution.

For an arbitrary choice of deterministic deferral rule for empirical distribution, we state the following proposition as a proof of insufficiency of deferral labels for obtaining optimal deferral rule over the true distribution.

**Proposition 5** (Impossibility of generalization of deferral labels)**.** *For every deterministic deferral rule $\hat{r}$ for empirical distributions and based on the two losses $\mathbb{1}_{m\neq y}$ and $\mathbb{1}_{h(x)\neq y}$, there exist two probability measures $\mu_1$ and $\mu_2$ on $\mathcal{X}\times\mathcal{Y}\times\mathcal{M}$ such that the corresponding $(\hat{r},X)$ for both measures is distributed equally. However, the optimal deferral $r^*_{\mu_1}$ and $r^*_{\mu_2}$ for these measures are not interchangeable, that is $L^{\mu_i}_{\text{def}}(h,r^*_{\mu_i})\leq\frac{1}{3}$ while $L^{\mu_i}_{\text{def}}(h,r^*_{\mu_j})=\frac{2}{3}$ for $i=1,2$ and $j\neq i$.*

*Proof.* As mentioned in ((C.22)), there are four possibilities of a deterministic deferral rule for empirical distribution based on the events $h(X)\neq Y$ and $M\neq Y$. The reason is that

$$\hat{r}=\begin{cases}1 & h(x)\neq y\,,m=y\\ 0 & h(x)=y\,,m\neq y\\ a & h(x)\neq y\,,m\neq y\\ b & h(x)=y\,,m=y\end{cases},$$

the parameters $a$ and $b$ can take binary values. One of the cases in which $a=b=0$ is analyzed previously in this section. We study the other cases as follows:

1. $\mathbf{a=1,b=0}$: In this case we have

$$\hat{r}=\begin{cases}1 & h(x)\neq y\\ 0 & o.w.\end{cases}.$$

If we define a measure $\mu_1$ such that

$$\mu_1\big(h(X)\neq Y,M=Y\big)=\frac{1}{3},\;\;\mu_1\big(h(X)=Y,M\neq Y\big)=\frac{2}{3},$$

and a measure $\mu_2$ such that

$$\mu_2\big(h(X)\neq Y,M=Y\big)=\frac{1}{3},\;\;\mu_2\big(h(X)=Y,M=Y\big)=\frac{2}{3},$$

then on one hand one can see that $\hat{r}$ is according to $Bern(\frac{1}{3})$ in both cases. On the other hand, because the probability of classifier accuracy is larger than human accuracy in $\mu_1$ and is smaller than human accuracy in $\mu_2$, we have $r^*_{\mu_1}(x)=0$ while $r^*_{\mu_2}(x)=1$. Therefore, we conclude that

$$L^{\mu_1}_{\text{def}}(r^*_{\mu_1},h)=\frac{1}{3},$$

and

$$L_{\text{def}}^{\mu_2}(r_{\mu_2}^*, h) = 0,$$

while the losses with interchanging deferral policies are equal to

$$L_{\text{def}}^{\mu_1}(r_{\mu_2}^*, h) = L_{\text{def}}^{\mu_2}(r_{\mu_1}^*, h) = \frac{2}{3}.$$

2. $\mathbf{a} = \mathbf{0}, \mathbf{b} = \mathbf{1}$: In this case, the deferral rule is as

$$\hat{r} = \begin{cases} 0 & m \neq y \\ 1 & o.w. \end{cases}.$$

Next, if we set two probability measures $\mu_1$ and $\mu_2$ such that

$$\mu_1\big(M \neq Y, h(X) = Y\big) = \frac{1}{3}, \;\; \mu_1\big(M = Y, h(X) \neq Y\big) = \frac{2}{3},$$

and

$$\mu_2\big(M \neq Y, h(X) = Y\big) = \frac{1}{3}, \;\; \mu_2\big(M = Y, h(X) = Y\big) = \frac{2}{3},$$

then $\hat{r}$ is according to $Bern(\frac{2}{3})$ in both cases. However, $r_{\mu_1}^* = 1$ while $r_{\mu_2}^* = 0$. Furthermore, the expected deferral losses are equal to

$$L_{\text{def}}^{\mu_1}(r_{\mu_1}^*, h) = \frac{1}{3}, \;\; L_{\text{def}}^{\mu_2}(r_{\mu_2}^*, h) = 0,$$

while after interchanging the deferral policies we have

$$L_{\text{def}}^{\mu_1}(r_{\mu_2}^*, h) = L_{\text{def}}^{\mu_2}(r_{\mu_1}^*, h) = \frac{2}{3}.$$

3. $\mathbf{a} = \mathbf{1}, \mathbf{b} = \mathbf{1}$: In this case, the deferral rule is as

$$\hat{r} = \begin{cases} 0 & h(x) = y, m \neq y \\ 1 & o.w. \end{cases}.$$

Next, if we set two probability measures $\mu_1$ and $\mu_2$ such that

$$\mu_1\big(M \neq Y, h(X) = Y\big) = \frac{1}{3}, \;\; \mu_1\big(M = Y, h(X) \neq Y\big) = \frac{2}{3},$$

and

$$\mu_2\big(M \neq Y, h(X) = Y\big) = \mu_2\big(M \neq Y, h(X) \neq Y\big) = \mu_2(M = Y, h(X) = Y) = \frac{1}{3},$$

then we can see that $\hat{r}$ has the distribution $Bern(\frac{2}{3})$. However, one can find the optimal deferral policies for the true distributions are $r^*_{\mu_1} = 1$ and $r^*_{\mu_2} = 0$. Furthermore, we have

$$L^{\mu_1}_{\text{def}}(r^*_{\mu_1}, h) = \frac{1}{3},$$

and

$$L^{\mu_2}_{\text{def}}(r^*_{\mu_2}, h) = \frac{2}{3},$$

while

$$L^{\mu_1}_{\text{def}}(r^*_{\mu_1}, h) = \frac{1}{3}, \;\; L^{\mu_2}_{\text{def}}(r^*_{\mu_2}, h) = \frac{1}{3}.$$

□

## C.7 Proof of Theorem 5

Let $\mathcal{X} = \{x_1, \ldots, x_n\}$ and $\mathcal{Y} = \{1, \ldots, n\}$. We first show that obtaining the optimal classifier is of $O(n)$ complexity, since in this case is equivalent to obtaining the Bayes optimal classifier in isolation. The reason is that, the unconstrained Bayes optimal classifier is a deterministic classifier that minimizes

$$h^*(x) \in \operatorname*{argmin}_{\hat{y}} \mathbb{E}_{\mu_{Y|X}}\big[\ell_{AI}(Y, \hat{y}, X) | X = x\big],$$

for all $x \in \mathcal{X}$. This is regardless of whether the deferral occurs or not. Therefore, this solution is further the solution to

$$\begin{aligned} h^*(x) &\in \operatorname*{argmin}_{\hat{y}} \mathbb{E}_{\mu_{Y|X}}\big[(1 - r(X))\ell_{AI}(Y, \hat{y}, X) | X = x\big] \\ &= \operatorname*{argmin}_{\hat{y}} \mathbb{E}_{\mu_{Y,M|X}}\big[(1 - r(X))\ell_{AI}(Y, \hat{y}, X) + r(X)\ell_H(Y, M, X) | X = x\big], \end{aligned}$$

for every rejection function $r$, including the optimal rejection function of the constrained optimization problem. In the particular case of expert intervention budget, the constraint is further independent of $h$ and is only a function of $r$. Therefore, the unconstrained Bayes classifier is an optimal classifier for the constrained L2D problem with human

intervention budget.

Next, we consider a specific case in which $\mathbb{E}_{\mu_{Y|X}}\big[\ell_{AI}(Y,1,X)|X=x\big] > \mathbb{E}_{\mu_{Y|X}}\big[\ell_{AI}(Y,0,X)|X=x\big]$ for all $x \in \mathcal{X}$, and therefore $h(x)=1$ over all input space. Further, we assume the data distribution has the property $\mu_{XYM} = \mu_{XY}\delta(M=Y)$, i.e. $M=Y$ on all the data. In this case, we know that

$$\mathbb{E}\big[\ell_H(Y,M,X)|X=x_i\big] = \mathbb{E}\big[\mathbb{1}_{M\neq Y}|X=x_i\big] = 0,$$

and we define

$$\mathbb{E}\big[\ell_{AI}(Y,h(X),X)|X=x_i\big] = \mathbb{E}\big[\mathbb{1}_{Y\neq 1}|X=x_i\big] = \ell_i.$$

Now, if we set $\Pr(X=x_i)=p_i$, and $r(x_i)=r_i$, then the optimization problem

$$f^* = \operatorname*{argmin}_{r(\cdot)\in\{0,1\}^{\mathcal{X}}} L^{\mu}_{\text{def}}(h,r),$$

is equivalent to

$$\operatorname*{argmin}_{r_i\in\{0,1\}} \sum_{i=1}^{n} p_i \times 0 \times r_i + p_i \times (1-r_i) \times \ell_i, \quad s.t. \ \sum_{i=1}^{n} p_i r_i \leq b,$$

that is equivalent to

$$\operatorname*{argmax}_{r_i\in\{0,1\}} \sum_{i=1}^{n} p_i r_i \ell_i, \quad s.t. \ \sum_{i=1}^{n} p_i r_i \leq b. \tag{C.23}$$

Next, we show that the above problem spans all instances of the $0-1$ knapsack problem, which is known to be NP-hard (Theorem 15.8 of Papadimitriou and Steiglitz (1998)). Let

$$\operatorname*{argmax}_{r_i\in\{0,1\}} \sum_{i=1}^{n} r_i c_i, \quad s.t. \ \sum_{i=1}^{n} w_i r_i \leq K, \tag{C.24}$$

be an instance of the $0-1$ knapsack problem [2] with $w_i, c_i > 0$, $i \in [n]$, and $K > 0$. With $\ell_i = \frac{c_i/w_i}{\sum_{i=1}^n c_i/w_i}$, $p_i = \frac{w_i}{\sum_{i=1}^n w_i}$ and $b = \frac{K}{\sum_{i=1}^n w_i}$, problem ((C.24)) can be written in the form of ((C.23)). Because of $\sum_{i=1}^n l_i = \sum_{i=1}^n p_i = 1$ this yields indeed a valid problem.

[2] Note that in case that $w_i = 0$ the Knapsack problem has a degenerate solution of $r_i = 1$. Hence, we could drop that point and without loss of generality assume that $w_i$ is non-zero.

## C.8 Proof of Theorem 6

We start this proof by introducing a few useful lemmas:

**Lemma 14.** *The set $\mathcal{F} = \Delta_n^{\mathcal{X}}$ of all functions that map $\mathcal{X}$ to an $n-$dimensional probability is weakly compact, i.e., for each sequence $\{f_n\}_{n=1}^{\infty}$, there is a sub-sequence $\{f_{n_i}\}$ and a function $f^* \in \mathcal{F}$ such that for all measurable embedding functions $\psi$, we have*

$$\lim_{k\to\infty} \mathbb{E}\big[\langle f_{n_k}, \psi\rangle\big] = \mathbb{E}\big[\langle f^*, \psi\rangle\big].$$

*Proof.* We know that all components of each element of the function sequence is bounded by 1. We define $\{f_m^i\}_{m=1}^{\infty}$ as the sequence of the $i$th component of the function sequence. Therefore, using (Lehmann *et al.*, 1986, Theorem A.5.1) we know that there is a sub-sequence $\{f^1_{m_k^1}\}_{k=1}^{\infty}$ and a non-negative 1-bounded function $f_1^*$, such that for each $\mu$-integrable function $\psi_1(x)$ we have

$$\lim_{k\to\infty} \mathbb{E}_\mu\big[f^1_{m_k}(x)\psi_1(x)\big] = \mathbb{E}_\mu\big[f_1^*(x)\psi_1(x)\big].$$

Next, we can repeat the same process for $\{f^i_{m_k^1}\}_{k=1}^{\infty}$ where $i \in [2:n]$, and we can find a sub-sequence $m_k^{i+1}$ of $m_k^i$ and a non-negative $1-$bounded function $f_{i+1}^*$ for which

$$\lim_{k\to\infty} \mathbb{E}_\mu\big[f^{i+1}_{m_k^{i+1}}(x)\psi_{i+1}(x)\big] = \mathbb{E}_\mu\big[f_{i+1}^*(x)\psi_{i+1}(x)\big].$$

Now, since all sub-sequences of a converging sequence converges to the same limit, we can use $m_k^n$ that is the intersection of all sequences and show that

$$\lim_{k\to\infty} \mathbb{E}_\mu\big[f^i_{m_k^n}(x)\psi_i(x)\big] = \mathbb{E}_\mu\big[f_i^*(x)\psi_i(x)\big],$$

for all $i \in [1:n]$ and integrable functions $\psi_i$. As a result, due to interchangeability of limit and summation, when the sum is over a finite set of elements, it is easy to show that

$$\begin{aligned}\lim_{k\to\infty} \mathbb{E}\big[\langle f_{m_k^n}, \psi\rangle\big] &= \lim_{k\to\infty} \mathbb{E}\big[\sum_{i=1}^{n} f^n_{m_k^n}(x)\psi_i(x)\big] = \sum_{i=1}^{n} \lim_{k\to\infty} \mathbb{E}\big[f^n_{m_k^n}(x)\psi_i(x)\big] \\ &= \sum_{i=1}^{n} \mathbb{E}_\mu\big[f_i^*(x)\psi_i(x)\big] = \mathbb{E}_\mu\big[\langle f^*(x), \psi(x)\rangle\big].\end{aligned}$$

Next, we need to show that $f^* \in \mathcal{F}$. We already know that all components of $f^*$ is 1-bounded and non-negative. Therefore, we only need to prove that all elements of $f^*$ sum up to 1 almost everywhere. If not, then assume that there is a non-zero set $A$ where $\mu(A) > 0$ and there exists $l > 0$ such that $|\sum_i f_i^* - 1| \geq l$ for all $x \in A$. We know that there is either a subset $B \subseteq A$ with $\mu(B) > 0$ such that for all $x \in B$ we have $\sum_i f_i^*(x) \geq 1 + l$,

or similarly a subset for which $\sum_i f_i^*(x) \leq 1-l$. The reason is that otherwise a non-zero measure set $A$ is a union of two zero-measure set, which is a contradiction. Without loss of generality we assume the first, which means $\sum_i f_i^*(x) \geq 1+l$ for $x \in B$. Now, if we define $\hat{\psi}(x) = [1,\ldots,1]$ for $x \in B$ and otherwise $\hat{\psi}(x) = [0,\ldots,0]$, then we have

$$\mathbb{E}_\mu\big[\langle f^*(x), \hat{\psi}(x)\rangle\big] \geq (1+l)\mu(B),$$

while

$$\mathbb{E}_\mu\big[\langle f_{m_k^n}(x), \psi\rangle\big] = 1,$$

for all $k \in \mathbb{N}$. This is a contradction, because the limit of a constant sequence is not different from that constant value. Hence, $f^*$ sums up to 1 almost everywhere, and that completes the proof.

**Proof of Theorem 6**: We prove the theorem using the following steps: (i) for the class $\mathcal{C}$ of prediction functions for which $\mathbb{E}\big[\langle f(x), \psi_i(x)\rangle\big] = \delta_i$ for $i \in [1:m]$, we show that the supremum of the objective function $\mathbb{E}\big[\langle f(x), \psi_0(x)\rangle\big]$ is a maximum, (ii) we show that it is sufficient for a predictor $f \in \mathcal{C}$ to be in form of ((4.8)) to achieve the maximum objective $\mathbb{E}\big[\langle f(x), \psi_0(x)\rangle\big]$ in $\mathcal{C}$ and also for all predictors with $\mathbb{E}\big[\langle f(x), \psi_i(x)\rangle\big] \leq \delta_i$, (iii) we show that the space of all possible constraints for any prediction function in $\Delta_d^{\mathcal{X}}$ is convex and compact, and (iv) we show that if the tuple of constraints is an interior point of all possible tuples of constraints, then a point in $\mathcal{C}$ achieves its maximum if and only if it follows the thresholding rule ((4.8)) almost everywhere.

- **Step (i)**: Due to the definition of supremum, we know that for each $\varepsilon > 0$, there exists a function $f_\varepsilon$ in $\mathcal{C}$ such that $\mathbb{E}\big[\langle f_\varepsilon, \psi_0(x)\rangle\big] \geq \sup_{f\in\mathcal{C}} \mathbb{E}\big[\langle f, \psi_0(x)\rangle\big] - \varepsilon$. Equivalently, there is a sequence of functions $f_n$ for which $\lim_{n\to\infty} \mathbb{E}\big[\langle f_n, \psi_0(x)\rangle\big] = \sup_{f\in\mathcal{C}} \mathbb{E}\big[\langle f, \psi_0(x)\rangle\big]$. Using weakly-compactness of the function class $\Delta_{n+1}^{\mathcal{X}}$ as in Lemma 14, we know that for the sequence $f_n$, there exists a subsequence $f_{n_k}$ and a function $f^* \in \Delta_{n+1}^{\mathcal{X}}$ such that

  $$\lim_{k\to\infty} \mathbb{E}\big[\langle f_{n_k}, \psi_{m+1}(x)\rangle\big] = \mathbb{E}\big[\langle f^*(x), \psi_{m+1}(x)\rangle\big].$$

  Furthermore, we know that each subsequence $a_{n_k}$ of a converging sequence $a_n$ has the same limit as the limit of the mother sequence $a_n$ (Pugh and Pugh, 2002, Chapter 2, Theorem 1). Therefore, we have

  $$\mathbb{E}\big[\langle f^*(x), \psi_{m+1}(x)\rangle\big] = \sup_{f\in\mathcal{C}} \mathbb{E}\big[\langle f, \psi_{m+1}(x)\rangle\big],$$

  which means that the supremum of the objective is achievable by $f^*$.

  Moreover, for $\psi_i(x)$ where $i \in [1:m]$, we have $\mathbb{E}\big[\langle f_n, \psi_i(x)\rangle\big] = \delta_i$ for all $n$, which

concludes

$$\delta_i = \lim_{k\to\infty} \mathbb{E}\big[\langle f_{n_k}, \psi_i(x)\rangle\big] = \mathbb{E}\big[\langle f^*(x), \psi_i(x)\rangle\big].$$

This means that the equality constraints holds for $f^*$, i.e., $f^* \in \mathcal{C}$, if it holds for each predictor $f_n$.

- **Step (ii)**: Assume that there is a predictor $\hat{f}$ such that $\mathbb{E}\big[\langle \hat{f}, \psi_i\rangle\big] \leq \delta_i$. In this step, we show that if exists a predictor $f$ in form of ((4.8)) and in $\mathcal{C}$, then $\hat{f}$ always has smaller objective than $\hat{f}$. To that end, consider the following expression:

$$A = \mathbb{E}\big[\langle f(x) - \hat{f}(x), \psi_0(x) - \sum_{i=1}^{m} k_i \psi_i(x)\rangle\big].$$

Now, we know that

$$\begin{aligned}\mathbb{E}\big[\langle f(x) - \hat{f}(x), \sum_{i=1}^{m} k_i \psi_i(x)\rangle\big] &= \sum_{i=1}^{m} k_i \Big(\mathbb{E}\big[\langle f(x), \psi_i(x)\rangle\big] - \mathbb{E}\big[\langle \hat{f}(x), \psi_i(x)\rangle\big]\Big) \\ &\overset{(a)}{=} \sum_{i=1}^{m} k_i \Big(\delta_i - \mathbb{E}\big[\langle \hat{f}(x), \psi_i(x)\rangle\big]\Big) \geq 0,\end{aligned}$$

where $(a)$ holds because $f \in \mathcal{C}$. As a result, if $A \geq 0$, then we could show that

$$\mathbb{E}\big[\langle f(x) - \hat{f}(x), \psi_0(x)\rangle\big] \geq 0, \tag{C.25}$$

and complete the proof.

To that end, first note that both $f$ and $\hat{f}$ are in $\Delta_d^{\mathcal{X}}$, and therefore

$$\langle f(x), [1, \ldots, 1]\rangle = \langle \hat{f}(x), [1, \ldots, 1]\rangle = 1.$$

As a result, for any fixed scalar $c$, we have

$$\langle f(x) - \hat{f}(x), \psi_0(x) - \sum_{i=1}^{m} k_i \psi_i(x)\rangle = \langle f(x) - \hat{f}(x), \psi_0(x) - \sum_{i=1}^{m} k_i \psi_i(x) - c\rangle. \tag{C.26}$$

Next, we fix $c$ to be the maximum component of the vector $\psi_0(x) - \sum_{i=1}^{m} k_i \psi_i(x)$, i.e.,

$$c := \max_{i\in[1:d]} \{\psi_0^i(x) - \sum_{j=1}^{m} k_j \psi_j^i(x)\}.$$

Now, we rewrite $A$ using ((C.26)) as

$$A = \mathbb{E}\Big[\langle f(x) - \hat{f}(x), \psi_0(x) - \sum_{i=1}^{m} k_i \psi_i(x) - c\rangle\Big]$$
$$= \sum_{i=1}^{d} \mathbb{E}\Big[(f_i(x) - \hat{f}_i(x))(\psi_0^i(x) - \sum_{j=1}^{m} k_j \psi_j^i(x) - c)\rangle\Big]$$

Now, we consider two cases for which $E_1^i(x) : f_i(x) > \hat{f}_i(x)$, and $E_2^i(x) : f_i(x) \leq \hat{f}_i(x)$. If $f_i(x) > \hat{f}_i(x)$, then we have $f_i(x) > 0$, because $1 \geq \hat{f}_i(x) \geq 0$ for all $i \in [1 : d]$. Therefore, using the definition of $\mathcal{S}_d$ and because $f \in \mathcal{S}_d$ we have

$$\psi_0^i(x) - \sum_{j=1}^{m} k_j \psi_j^i(x) = \max_{i \in [1:d]} \big\{ \psi_0^i(x) - \sum_{j=1}^{m} k_j \psi_j^i(x) \big\} = c. \tag{C.27}$$

Consequently, we have

$$A = \sum_{i=1}^{d} \mathbb{E}\Big[(f_i(x) - \hat{f}_i(x))(\psi_0^i(x) - \sum_{j=1}^{m} k_j \psi_j^i(x) - c)\rangle\Big]$$
$$= \sum_{i=1}^{d} \mathbb{E}\Big[(f_i(x) - \hat{f}_i(x))(\psi_0^i(x) - \sum_{j=1}^{m} k_j \psi_j^i(x) - c)\rangle | E_1^i(x)\Big] \Pr\big(E_1^i(x)\big)$$
$$+ \sum_{i=1}^{d} \mathbb{E}\Big[(f_i(x) - \hat{f}_i(x))(\psi_0^i(x) - \sum_{j=1}^{m} k_j \psi_j^i(x) - c)\rangle | E_2^i(x)\Big] \Pr\big(E_2^i(x)\big)$$
$$\overset{(a)}{=} \sum_{i=1}^{d} \mathbb{E}\Big[(f_i(x) - \hat{f}_i(x))(\psi_0^i(x) - \sum_{j=1}^{m} k_j \psi_j^i(x) - c)\rangle | E_2^i(x)\Big] \Pr\big(E_2^i(x)\big)$$
$$\overset{(b)}{\geq} 0,$$

where $(a)$ holds due to ((C.27)) and $(b)$ holds because $f_i(x) \leq \hat{f}_i(x)$ and $\psi_{m+1}^i(x) - \sum_{j=1}^{m} k_j \psi_j^i(x) \leq c = \max_{i \in [1:n+1]} \{\psi_{m+1}^i(x) - \sum_{j=1}^{m} k_j \psi_j^i(x)\}$. As a result, we have $A \geq 0$ that concludes ((C.25)) and completes the proof of this step.

- **Step (iii)**: In this step, we show that the space of joint set of expected inner-products

$$\mathcal{G} = \Big\{ \big(\mathbb{E}[\langle f(x), \psi_1(x)\rangle], \ldots, \mathbb{E}[f(x), \psi_m(x)]\big) : f \in \Delta_d^{\mathcal{X}} \},$$

is compact under Euclidean metric, and convex.

To show the compactness of that space, assume that there is a sequence $\{g_n\}_{n=1}^{\infty}$ such that $\lim_{n\to\infty} g_n = g$, or accordingly there is a sequence of $f_n \in \Delta_d^{\mathcal{X}}$ for which

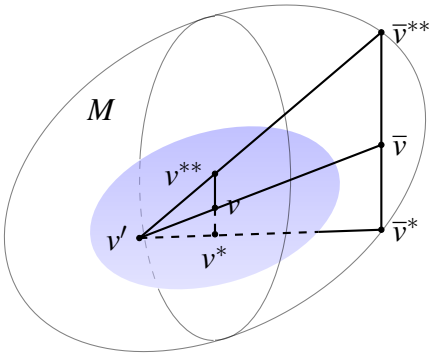



Figure C.1: If an interior point of $\mathcal{N}$ has one corresponding point at $M$, then so are all interior points of $N$

$\lim_{n\to\infty}\big(\mathbb{E}[\langle f_n(x),\psi_1(x)\rangle],\ldots,\mathbb{E}[f_n(x),\psi_m(x)]\big)=(g_1,\ldots,g_m)$. Since the metric is Euclidean, this is equivalent to $\lim_{n\to\infty}\mathbb{E}[\langle f_n(x),\psi_i(x)\rangle]=g_i$ for all $i\in[1:m]$. The weak compactness of $\Delta_d^{\mathcal{X}}$, as proved in Lemma 14, shows that there exists $f^*$ and a sub-sequence $f_{n_k}$ such that $\lim_{k\to\infty}\mathbb{E}[\langle f_{n_k}(x),\psi_i(x)\rangle]=\mathbb{E}[\langle f^*,\psi_i(x)\rangle]$ for all $i\in[1:d]$. Therefore, because of the choice of Euclidean metric, we have

$$\lim_{k\to\infty}\Big(\mathbb{E}[\langle f_{n_k}(x),\psi_1(x)\rangle],\ldots,\mathbb{E}[\langle f_{n_k}(x),\psi_m(x)\rangle]\Big)$$
$$=\Big(\mathbb{E}[\langle f^*,\psi_1(x)\rangle],\ldots,\mathbb{E}[\langle f^*,\psi_m(x)\rangle]\Big),$$

which is equivalent to compactness of $\mathcal{G}$.

To show the convexity of $\mathcal{G}$, it is enough to prove the convexity of $\Delta_d^{\mathcal{X}}$. The reason is that $g(f)=\big(\mathbb{E}[\langle f(x),\psi_1(x)\rangle],\ldots,\mathbb{E}[f(x),\psi_m(x)]\big)$ is a linear functional of $f$, and a linear functional images a convex set to another convex set.

To prove the convexity of $\Delta_d^{\mathcal{X}}$, let $f,f'\in\Delta_d^{\mathcal{X}}$. This means that $f_i(x),f_i'(x)\in[0,1]$ for all $i\in[1:d]$ and $\sum_{i=1}^d f_i(x)=\sum_{i=1}^d f_i'(x)=1$. Consequently, $af_i(x)+(1-a)f_i'(x)\geq 0$, since $a,1-a\geq 0$. Moreover, $\sum_{i=1}^d af_i(x)+(1-a)f_i'(x)=a\sum_{i=1}^d f_i(x)+(1-a)\sum_{i=1}^d f_i'(x)=a+1-a=1$. As a result of these two facts, $af+(1-a)f'\in\Delta_d^{\mathcal{X}}$, and the proof of this step is completed.

- **Step (iv)**: In this step we show that if the tuple of constraints is an interior points of all possible achievable tuples of constraints using different prediction functions, then a point in $\mathcal{C}$ achieves its supremum in terms of objective $\mathbb{E}\big[\langle f(x),\psi_0(x)\big]$ if and only if it is in form of ((4.8)) almost everywhere. This is an extension to (Dantzig and Wald, 1951, Theorem 3.1) and its proof resembles to the proof that is provided there. The sufficiency is already shown in Step (ii). Therefore, we only need to show that if a prediction function in $\mathcal{C}$ maximizes the objective, then it is

in form of ((4.8)).

Firstly, using Step (iii), we know that the space $\mathcal{N}$ of all points $\big(\mathbb{E}[\langle f(x),\psi_1(x)\rangle],\ldots,\mathbb{E}[f(x),\psi_m(x)]\big)$ and the space $\mathcal{M}$ of all points $\big(\mathbb{E}[\langle f(x),\psi_1(x)\rangle],\ldots,\mathbb{E}[f(x),\psi_0(x)]\big)$ are compact and convex. Now, assume that $v=(\delta_1,\ldots,\delta_m)$ is an interior point of $\mathcal{N}$. Then, the corresponding set in $\mathcal{M}$, i.e., $B_v=\{(\delta_0,\ldots,\delta_m)\in\mathcal{M}:\delta_0\in\mathbb{R}\}$ has a supremum and an infimum of the first component that we name $\delta^{**}$ and $\delta^{*}$. Now, since $\mathcal{M}$ is compact, then $v^{**}=(\delta^{**},\delta_1,\ldots,\delta_m)$ and $v^{*}=(\delta^{*},\delta_1,\ldots,\delta_m)$ are contained in $\mathcal{M}$. Next, assume the following two cases:

1. $\delta^{**}=\delta^{*}$: In this case for all other points $\overline{v}=(\overline{\delta}_1,\ldots,\overline{\delta}_m)$ of $\mathcal{N}$, the corresponding set $B_{v'}$ in $\mathcal{M}$ is a single point. The reason is that, if it is not so, then we have two points $\overline{v}^{**}=(\overline{\delta}^{**},\overline{\delta}_1,\ldots,\overline{\delta}_m)$ and $\overline{v}^{*}=(\overline{\delta}^{*},\overline{\delta}_1,\ldots,\overline{\delta}_m)$ where $\overline{\delta}^{**}>\overline{\delta}^{*}$. Now, since $v$ is an interior point of $\mathcal{N}$, then on any direction in a small neighborhood around $v$ there exists a point $v'$ within $\mathcal{N}$. Let that direction be opposite the connecting line of $v$ and $\overline{v}$, i.e., let $v$ be on a connecting line of $v'$ and $\overline{v}^{*}$. Now, make a convex hull using the three points $v'$, $\overline{v}^{**}$, and $\overline{v}^{*}$, which are all in $\mathcal{M}$. Because of the convexity of $\mathcal{M}$, the convex hull is also a subset of $\mathcal{M}$. Since $v$ is an interior point of the convex hull, this means that a neighborhood of $v$ along any direction is inside $\mathcal{M}$. Now, if we set $(m+1)$th axis to be that direction, we contradict with the fact that $\delta^{*}=\delta^{**}$. (See Figure C.1)

   Now, we know that in such case all points within $\mathcal{N}$ have one corresponding point in $\mathcal{M}$. Because of the convexity of $\mathcal{M}$ this is equivalent to $\mathcal{M}$ being a subset of a hyperplane with the generating formula $x_0=\sum_{i=1}^{m}k_ix_i+k_0$. Therefore, we have $\mathbb{E}\big[\langle f,\psi_0\rangle\big]=\mathbb{E}\big[\langle f,\sum_{i=1}^{m}k_i\psi_i\rangle\big]+k_0$ for all $f\in\Delta_d^{\mathcal{X}}$. Therefore, for $d\geq 2$, if we set $f_1=(\frac{p(x)}{d-2},\ldots,\frac{p(x)}{d-2},\underbrace{1-p(x)}_{i},\frac{p(x)}{d-2},\ldots,\underbrace{0}_{j},\frac{p(x)}{d-2},\ldots,\frac{p(x)}{d-2})$ and $f_2=(\frac{p(x)}{d-2},\ldots,\frac{p(x)}{d-2},\underbrace{0}_{i},\frac{p(x)}{d-2},\ldots,\underbrace{1-p(x)}_{j},\frac{p(x)}{d-2},\ldots,\frac{p(x)}{d-2})$ for $p(x)\in[0,1]^{\mathcal{X}}$, then we have

$$\mathbb{E}\big[\langle f_1,\psi_0\rangle\big]-\mathbb{E}\big[\langle f_1,\sum_{i=1}^{m}k_i\psi_i\rangle\big]=\mathbb{E}\big[\langle f_2,\psi_0\rangle\big]-\mathbb{E}\big[\langle f_2,\sum_{i=1}^{m}k_i\psi_i\rangle\big],$$

   or equivalently

$$\mathbb{E}\big[(1-p(x))(\psi_0^i(x)-\sum_{t=1}^{m}k_t\psi_t^i(x)-\psi_{m+1}^j(x)+\sum_{t=1}^{m}k_t\psi_t^j(x))\big]=0,$$

   for all function $p(x)\in\Delta_d^{\mathcal{X}}$. A similar result can be achieved for $d=2$ and by

setting $f_1 = (p(x), 1-p(x))$ and $f_2 = (1-p(x), p(x))$. As a result, we have

$$\psi_0^i(x) - \sum_{t=1}^{m} k_t \psi_t^i(x) = \psi_0^j(x) - \sum_{t=1}^{m} k_t \psi_t^j(x),$$

for all $i \neq j \in [1:d]$, and consequently

$$\psi_0^i(x) - \sum_{t=1}^{m} k_t \psi_t^i(x) = \max_{j \in [1:d]} \{\psi_0^j(x) - \sum_{t=1}^{m} k_t \psi_t^j(x)\},$$

for all $i \in [1:n+1]$. As a result, there is a set of $k_1, \ldots, k_m$ such that $\psi_0(x) - \sum_{i=1}^{m} k_i \psi_i(x)$ has equal components almost everywhere. As a result, due to the freedom of choice for $\tau(\psi_0(x) - \sum_{i=1}^{m} k_i \psi_i(x), x)$ where $\tau \in \mathcal{S}_d$ and when we have more than one maximizer component, then, without loss of generality we can assume that every prediction function $f$ almost everywhere is in form of $\tau(\psi_{m+1}(x) - \sum_{i=1}^{m} k_i \psi_i(x), x)$.

2. $\delta^{**} > \delta^*$: In such case, for all $\delta_0 \in [\delta^*, \delta^{**}]$, we can show that $v = (\delta_0, \ldots, \delta_m)$ is an interior point of $\mathcal{M}$. To show that, we first find $m$ points $v'_1, \ldots, v'_m \in \mathcal{N}$ that are linearly independent and such that their convex hull include $(\delta_1, \ldots, \delta_m)$. Using the definition of $\mathcal{M}$, for each of these points $v'_i = ({\delta'_1}^i, \ldots, {\delta'_m}^i)$, there exists $h'_i \in \mathbb{R}$ such that $v''_i = (h'_i, {\delta'_1}^i, \ldots, {\delta'_m}^i)$ is within $\mathcal{M}$. Now, we add the two points $v^{**}$ and $v^*$ to these sets of points. It is easy to see that $v''_i$s are linearly independent. Furthermore, we know that $(\delta_1, \ldots, \delta_m)$ is a convex combination of $v'_i$s, i.e., $\sum_i a_i v'_i = (\delta_1, \ldots, \delta_m)$. As a result, if $\sum_i b_i v''_i - v^{**} = (0, \ldots, 0)$, then we have $b_i = a_i$ for $i \in [1:m]$. Furthermore, we have $\sum a_i h'_i = \sum b_i h'_i = \delta^{**}$. Similarly, if $\sum_i c_i v''_i - v^* = (0, \ldots, 0)$ we have $c_i = a_i$ and $\sum a_i h'_i = \sum c_i h'_i = \delta^*$. As a result, since $\delta^* \neq \delta^{**}$ at least one of these cases would not occur, or equivalently, the dimension of the convex hull of $v''_1, \ldots v''_m, v^{**}, v^*$ is of dimension $m+1$. As a result, $v$ is an interior point of this convex hull, and because the convex hull is $(m+1)$-dimensional, it is an interior point of $\mathcal{M}$.

Now, since $v^{**}$ is a border point in $\mathcal{M}$ and due to the convexity of $\mathcal{M}$ there is a hyperplane $\mathcal{P}$ such that it passes $v^{**}$ and it lays above all points of $\mathcal{M}$. Since $v$ is an interior point of $\mathcal{M}$, a neighborhood of $v$ is laid under the hyperplane $\mathcal{P}$, hence $v$ cannot be laid on the hyperplane. Therefore, if the generating formula of such hyperplane is $\sum_{i=0}^{m} k_i x_i = \sum_{i=1}^{m} k_i \delta_i + k_0 \delta^{**}$, since $v$ is not laid on the hyperplane we assure that $\sum_{i=1}^{m} k_i \delta_i + k_0 \delta_0 \neq \sum_{i=1}^{m} k_i \delta_i + k_0 \delta^{**}$, or equivalently $k_0 \neq 0$. Hence, without loss of generality assume that $k_0 = -1$.

This shows that for all points in $(u_0,\dots,u_m)\in\mathcal{M}$ we have

$$u_0-\sum_{i=1}^{m}k_iu_i\leq\delta^{**}-\sum_{i=1}^{m}k_i\delta_i,$$

or equivalently, by the definition of $\mathcal{M}$, for all prediction function $f$, we have

$$\mathbb{E}\big[\langle f(x),\psi_0(x)-\sum_{i=1}^{m}k_i\psi_i(x)\rangle\big]\leq\delta^{**}-\sum_{i=1}^{m}k_i\delta_i.$$

Assuming that $\hat{f}\in\mathcal{C}$ maximizes the objective, we have

$$\mathbb{E}\big[\langle f(x),\psi_0(x)-\sum_{i=1}^{m}k_i\psi_i(x)\rangle\big]\leq\mathbb{E}\big[\langle \hat{f}(x),\psi_0(x)-\sum_{i=1}^{m}k_i\psi_i(x)\rangle\big]. \quad \text{(C.28)}$$

This shows that almost everywhere when there is a unique maximizing component $j$ in $\psi_0(x)-\sum_{i=1}^{m}k_i\psi_i(x)$, then $\hat{f}_j(x)=1$. The reason is that otherwise and if there is a set $A$ such that $\mu(A)>0$ and for $x\in A$ and a choice of $l\in[0,1)$, $\varepsilon\in\mathbb{R}$, and all $t\neq j$ we have $\psi^j_{m+1}(x)-\sum_{i=1}^{m}k_i\psi^j_i(x)\geq\varepsilon+\psi^t_{m+1}(x)-\sum_{i=1}^{m}k_i\psi^t_i(x)$ while $f_j\leq 1-l$, then we can make a function $f(x)$ that is $f(x)=\hat{f}(x)$ for $x\in\mathcal{X}\setminus A$ and $f(x)=[0,\dots,\underbrace{1}_{j},\dots,0]$ for $x\in A$. Such function leads to

$$\mathbb{E}\big[\langle f(x),\psi_0(x)-\sum_{i=1}^{m}k_i\psi_i(x)\rangle\big]\geq\varepsilon l\mu(A)+\mathbb{E}\big[\langle \hat{f}(x),\psi_0(x)-\sum_{i=1}^{m}k_i\psi_i(x)\rangle\big],$$

that is in contradiction with ((C.28)). This completes the proof of this step.

## C.9 Proof of Theorem 7

In the following, we introduce a few lemmas that are useful in our proofs.

**Lemma 15.** *For every random variable $X$ on $\mathbb{R}$ we have*

$$\lim_{\tau\to t^-}Pr(\tau\leq X<t)=\lim_{\tau\to t^+}Pr(t<X<\tau)=0$$

*Proof.* For each increasing sequence $\{\tau_i\}_{i=1}^{\infty}$ we show that the first limit is zero, which proves the claim that the function of $\tau$ has a zero limit.

We define

$$\mathcal{S}_i=[\tau_i,t),$$

and notice that

$$\mathcal{S}_1 \supseteq \mathcal{S}_2 \supseteq \dots.$$

Further, we note that

$$\bigcap_{i=1}^{\infty} \mathcal{S}_i = \emptyset.$$

As a result

$$\mathcal{S}_1^c \subseteq \mathcal{S}_2^c \subseteq \dots,$$

and

$$\bigcup_{i=1}^{\infty} \mathcal{S}_i^c = \mathbb{R}.$$

Next, because probability measure is $\sigma$-additive, we conclude its lower-semicontinuity (Klenke, 2013, Theorem 13.6), and therefore we have

$$\lim_{i\to\infty} \Pr(X \in \mathcal{S}_i^c) = \Pr(X \in \cup_{i=1}^{\infty} \mathcal{S}_i^c) = 1,$$

which proves $\lim_{i\to\infty} \Pr(X \in \mathcal{S}_i) = 0$.

We could take similar steps to show that since $\bigcap_{i=1}^{\infty}(t, \tau_i') = \emptyset$ for decreasing $\tau_i'$ we have

$$\lim_{i\to\infty} \Pr(X \in (t, \tau_i')) = 0.$$

□

**Lemma 16.** *Let $\psi_1 : \mathcal{X} \to \mathbb{R}^d$ be a bounded function. Further, we define two functions $C(k) = \mathbb{E}\big[\langle f_{k,0}^*(x), \psi_1(x)\rangle\big]$, $D(k) = \mathbb{E}\big[\langle f_{k,1}^*(x), \psi_1(x)\rangle\big]$, and $F(k) = \mathbb{E}\big[\langle f_{k,1}^*(x), \psi_0(x)\rangle\big]$, where $f_{k,p}^*$ is defined in Theorem 7. Then,*

1. *$C(k)$ is monotonically non-increasing,*
2. *$C(k)$ is upper semi-continuous,*
3. *$F(k)$ is monotonically non-decreasing,*
4. *$D(k)$ is lower semi-continuous, and we have*
5. *$\lim_{k'\uparrow k} C(k) = \lim_{k'\uparrow k} D(k)$*

*Proof.* 1. Firstly, let us define $\ell_k(x) = \psi_0(x) - k\psi_1(x)$. For the setting where $p = 0$, the prediction function $f^*_{k,p}(x)$ is defined as

$$f^*_{k,0}(x,p) = \begin{cases} 1 & i = \min\{ \underset{j \in \operatorname{argmax} \ell_k(x)}{\operatorname{argmin}} \big(\psi_1(x)\big)(j)\} \\ 0 & \text{otherwise} \end{cases}. \tag{C.29}$$

Further, for $k_1, k_2$ such that $k_1 \leq k_2$, let us define $j_1$ and $j_2$ as the only non-zero index of $f^*_{k_1,0}(x,p)$ and $f^*_{k_2,0}(x,p)$, respectively. To show that $C(k)$ is monotonically non-increasing we only need to show that $\big(\psi_1(x)\big)(j_1) = \langle f^*_{k_1,0}(x), \psi_1(x)\rangle \geq \langle f^*_{k_2,0}(x), \psi_1(x)\rangle = \big(\psi_1(x)\big)(j_2)$. Assume that this does not occur, or equivalently $\big(\psi_1(x)\big)(j_1) < \big(\psi_1(x)\big)(j_2)$. In such case we have

$$\begin{aligned} \max \ell_{k_2}(x) &\overset{(a)}{=} \big(\ell_{k_2}(x)\big)(j_2) \\ &= \big(\ell_{k_1}(x) - (k_2 - k_1)\psi_1(x)\big)(j_2) \\ &\leq (k_1 - k_2)\big(\psi_1(x)\big)(j_2) + \max_j \big(\ell_{k_1}(x)\big)(j) \\ &\overset{(b)}{=} (k_1 - k_2)\big(\psi_1(x)\big)(j_2) + \big(\ell_{k_1}(x)\big)(j_1) \\ &\overset{(c)}{<} (k_1 - k_2)\big(\psi_1(x)\big)(j_1) + \big(\ell_{k_1}(x)\big)(j_1) \\ &= \big(\ell_{k_2}(x)\big)(j_2), \end{aligned} \tag{C.30}$$

where $(a)$ and $(b)$ holds due to the definition of $j_1$ and $j_2$, and $(c)$ holds due to the assumption $\big(\psi_1(x)\big)(j_1) < \big(\psi_1(x)\big)(j_2)$. The last inequality is clearly a contradiction, and shows that $\langle f^*_{k_1,0}(x), \psi_1(x)\rangle \geq \langle f^*_{k_2,0}(x), \psi_1(x)\rangle$, and therefore $C(k_1) \geq C(k_2)$.

2. Let us divide the space $\mathcal{X}$ into two subsets

$$A_k = \Big\{ x \in \mathcal{X} : \Big| \operatorname*{argmax}_i (\ell_k(x))(i) \Big| = d \Big\},$$

$$B_k = \Big\{ x \in \mathcal{X} : \Big| \operatorname*{argmax}_i (\ell_k(x))(i) \Big| \in [1 : d-1] \Big\}.$$

For each $x \in A_k$ we know

$$\big(f^*_{k,0}(x)\big)(i) = \begin{cases} 1 & i = \min\{ \operatorname*{argmin}_j \big(\psi_1(x)\big)(j)\} \\ 0 & \text{otherwise} \end{cases}$$

Using previous part, we know that by increasing $k$ we have no increase in $\langle f^*_{k,0}(x), \psi_1(x)\rangle$, and in this case since $\langle f^*_{k,0}(x), \psi_1(x)\rangle = \min_j \big(\psi_1(x)\big)(j)$, then this value cannot reduce with the change of $k$. Therefore, $\langle f^*_{k,0}(x), \psi_1(x)\rangle$ is a constant function for all $k' \geq k$, and consequently $\mathbb{E}\big[\langle f^*_{k',0}(x), \psi_1(x)\rangle | x \in A_k\big]\Pr(x \in A_k)$ is a constant function for $k' \geq k$.

If $x \in B_k$, then for $j \notin \operatorname{argmax}_i \big(\ell_k(x)\big)(i)$ and $l \in \operatorname{argmax}_i \big(\ell_k(x)\big)(i)$, we have $\big(\ell_k(x)\big)(j) < \big(\ell_k(x)\big)(l)$. Define the set $C_\delta$ for $\delta \geq 0$ as

$$C_\delta = \{x \in B_k : \big(\ell_k(x)\big)(j) \leq \big(\ell_k(x)\big)(l) - \delta\}.$$

Using Lemma 15 we know that

$$\lim_{\delta \to 0} \Pr(B_k \setminus C_\delta) = 0,$$

or equivalently for all $\varepsilon \geq 0$, there exists $\delta$ such that

$$\Pr(B_k \setminus C_\delta) \leq \varepsilon'.$$

Therefore, if without loss of generality, we assume that $\psi_1(x)$ is bounded by 1, then there exists $\delta \geq 0$ such that we have

$$\begin{aligned}\mathbb{E}\big[\langle f^*_{k',0}(x), \psi_1(x)\rangle | x \in B_k \setminus C_\delta\big]\Pr(x \in & B_k \setminus C_\delta) \\ &\overset{(a)}{\leq} \|\psi_1(x)\|_\infty \Pr(x \in B_k \setminus C_\delta) \leq \varepsilon/2,\end{aligned}$$

where $(a)$ holds due to Hölder's inequality.

If $x \in C_\delta$, and because we assumed $\|\psi_1(x)\|_\infty \leq 1$, then we know that by increasing $k$ to $k' \in [k - \delta/2, k + \delta/2)$, we have

$$\mathcal{I} = \operatorname{argmax} \ell_{k'}(x) \subseteq \operatorname{argmax} \ell_k(x) = \mathcal{J}. \tag{C.31}$$

This means that

$$\langle f^*_{k,0}(x), \psi_1(x)\rangle = \min_{j \in \mathcal{J}} \big(\psi_1(x)\big)(j) \leq \min_{j \in \mathcal{I}} \big(\psi_1(x)\big)(j) = \langle f^*_{k',0}(x), \psi_1(x)\rangle.$$

This, together with the previous part in which we showed $\langle f^*_{k,0}(x), \psi_0(x)\rangle \geq \langle f^*_{k',0}(x), \psi_0(x)\rangle$, concludes that $\langle f^*_{k,0}(x), \psi_0(x)\rangle = \langle f^*_{k',0}(x), \psi_0(x)\rangle$. This means that $\mathbb{E}\big[\langle f^*_{k',0}(x), \psi_0(x)\rangle | x \in C_\delta\big]\Pr(x \in C_\delta)$ is a constant function for all $k' \geq k$ .

Finally, since we have

$$\begin{aligned} C(k') = \mathbb{E}\big[\langle f^*_{k',0}(x), \psi_1(x)\rangle\big] =& \mathbb{E}\big[\langle f^*_{k',0}(x), \psi_1(x)\rangle | x \in A_k\big]\Pr(x \in A_k) \\ &+ \mathbb{E}\big[\langle f^*_{k',0}(x), \psi_1(x)\rangle | x \in B_k \setminus C_\delta\big]\Pr(x \in B_k \setminus C_\delta) \\ &+ \mathbb{E}\big[\langle f^*_{k',0}(x), \psi_1(x)\rangle | x \in C_\delta\big]\Pr(x \in C_\delta), \end{aligned}$$

and because the first term and the third term in RHS are constant in terms of $k'$ and for $k' \geq k$, and the second term is diminishing, then we have

$$\begin{aligned} \big|C(k') - C(k)\big| = \big|&\mathbb{E}\big[\langle f^*_{k',0}(x), \psi_1(x)\rangle | x \in B_k \setminus C_\delta\big]\Pr(x \in B_k \setminus C_\delta) \\ &- \mathbb{E}\big[\langle f^*_{k,0}(x), \psi_1(x)\rangle | x \in B_k \setminus C_\delta\big]\Pr(x \in B_k \setminus C_\delta)\big| \leq \varepsilon/2 + \varepsilon/2, \end{aligned}$$

which is equivalent to say that $\lim_{k' \uparrow k} C(k') = C(k)$.

3. For $p = 1$, we know that the prediction function $f^*_{k,p}(x)$ is obtained as

$$f^*_{k,1}(x) = \begin{cases} 1 & i = \min\{ \underset{j \in \operatorname{argmax} \ell_k(x)}{\operatorname{argmax}} \big(\psi_0(x)\big)(j)\} \\ 0 & \text{otherwise} \end{cases}.$$

If we define $\psi'_1(x) := -\psi_0(x)$, then we have

$$f^*_{k,1}(x) = \begin{cases} 1 & i = \min\{ \underset{j \in \operatorname{argmax} \ell_k(x)}{\operatorname{argmin}} \big(\psi'_1(x)\big)(j)\} \\ 0 & \text{otherwise} \end{cases}.$$

Since the above is equal to ((C.29)), then using the first part of this lemma, we know that $\mathbb{E}\big[\langle f^*_{k,1}(x), \psi'_1(x)\rangle\big] = -\mathbb{E}\big[\langle f^*_{k,1}(x), \psi_0(x)\rangle\big]$ is monotonically non-increasing, which is equivalent to $F(k) = \mathbb{E}\big[\langle f^*_{k,1}(x), \psi_0(x)\rangle\big]$ being monotonically non-decreasing.

4. This part is similar to the second part of the proof. In fact, if $x \in A_k$, then we have

$$\big(f^*_{k,1}(x)\big)(i) = \begin{cases} 1 & i = \min\{ \underset{j}{\operatorname{argmax}} \big(\psi_0(x)\big)(j)\} \\ 0 & \text{otherwise} \end{cases}. \tag{C.32}$$

For $k' \leq k$ and because of the third part of this lemma, we know that $\langle f^*_{k',1}(x), \psi_0(x)\rangle \geq \langle f^*_{k,1}(x), \psi_0(x)\rangle$. Furthermore, because of ((C.32)) we know that $\langle f^*_{k,1}(x), \psi_0(x)\rangle = \max \psi_0(x)$, and therefore by reducing $k'$, the prediction function $f^*_{k',1}(x)$ stays constant. As a result, $\mathbb{E}\big[\langle f^*_{k',1}(x), \psi_1(x)\rangle | x \in A_k\big]\Pr(x \in A_k)$ is a constant function for $k' \leq k$.

Furthermore, similar to the second part of this lemma, we can show that for each

$\varepsilon > 0$, there exists $\delta' \geq 0$ such that for all $0 \leq \delta \leq \delta'$ we have

$$\begin{aligned}\mathbb{E}\big[\langle f^*_{k',1}(x), \psi_1(x)\rangle | x \in B_k \setminus C_\delta\big]\Pr(x \in & B_k \setminus C_\delta) \\ &\overset{(a)}{\leq} \|\psi_1(x)\|_\infty \Pr(x \in B_k \setminus C_\delta) \leq \varepsilon/4,\end{aligned} \tag{C.33}$$

Moreover, for the case of $x \in C_\delta$, since in this case $\mathcal{J} \subseteq \mathcal{I}$, then we know that

$$\langle f^*_{k,1}(x), \psi_0(x)\rangle = \max_{j\in\mathcal{J}} \big(\psi_0(x)\big)(j) \leq \max_{j\in\mathcal{I}} \big(\psi_0(x)\big)(j) = \langle f^*_{k',1}(x), \psi_0(x)\rangle. \tag{C.34}$$

Next, using the third part of this lemma, we know that for $k' \leq k$ we have $\langle f^*_{k',1}(x), \psi_0(x)\rangle \leq \langle f^*_{k,1}(x), \psi_0(x)\rangle$, which together with ((C.34)) concludes that $\langle f^*_{k,1}(x), \psi_0(x)\rangle = \langle f^*_{k',1}(x), \psi_0(x)\rangle$. Next, because $\big(\psi_0(x) - k\psi_1(x)\big)(i) = \big(\psi_0(x) - k\psi_1(x)\big)(j)$ for $i, j \in \mathcal{J}$, then we know that $\Big|\big(\ell_{k'}(x)\big)(i) - \big(\ell_{k'}(x)\big)(j)\Big| = \Big|(k-k')\Big(\big(\psi_1(x)\big)(i) - \big(\psi_1(x)\big)(j)\Big) \leq 2|k-k'|$. Therefore, if for $i, j \in \mathcal{J}$ we know that $\big(\psi_0(x)\big)(i) = \big(\psi_0(x)\big)(j)$, then the difference between $\psi_1$ for those indices is bounded as

$$\begin{aligned}\Big|\big(\psi_1(x)\big)(i) - \big(\psi_1(x)\big)(j)\Big| &\leq \frac{1}{k}\Big|\big(\psi_0(x)\big)(i) - \big(\psi_0(x)\big)(j)\Big| \\ &\quad + \Big|\big(\ell_k(x)\big)(i) - \big(\ell_k(x)\big)(j)\Big| \\ &\leq 2|k-k'|.\end{aligned} \tag{C.35}$$

Now, we know that because $x \in C_\delta$, then $\langle f^*_{k,1}(x), \psi_1(x)\rangle = \big(\psi_1(x)\big)(i)$ for $i \in \operatorname*{argmax}_{j\in\mathcal{J}} \big(\psi_0(x)\big)(j)$, and $\langle f^*_{k',1}(x), \psi_1(x)\rangle = \big(\psi_1(x)\big)(j)$ for $j \in \operatorname*{argmax}_{k\in\mathcal{I}} \big(\psi_0(x)\big)(j)$. Hence, we can see that $i \in \mathcal{J} \subseteq \mathcal{I}$ and $j \in \mathcal{I}$, and because $\big(\psi_0(x)\big)(i) = \langle f^*_{k,1}(x), \psi_0(x)\rangle = \langle f^*_{k',1}(x), \psi_0(x)\rangle = \big(\psi_0(x)\big)(j)$, and due to ((C.35)) we have

$$\Big|\langle f^*_{k,1}(x), \psi_1(x)\rangle - \langle f^*_{k',1}(x), \psi_1(x)\rangle\Big| \leq 2|k-k'|,$$

as long as $k' \in [k - \delta/2, k)$. Therefore, if we set $\delta = \max\{\delta', \varepsilon/2\}$ we have

$$\big|\langle f^*_{k,1}(x), \psi_1(x)\rangle - \langle f^*_{k',1}(x), \psi_1(x)\rangle\big| \leq \varepsilon/2,$$

and therefore

$$\Big|\mathbb{E}\big[\langle f^*_{k',1}(x),\psi_1(x)\rangle|x\in C_\delta\big]-\mathbb{E}\big[\langle f^*_{k,1}(x),\psi_1(x)\rangle|x\in C_\delta\big]\Big| \\ \leq \mathbb{E}\Big[\big\|\langle f^*_{k,1}(x),\psi_0(x)\rangle-\langle f^*_{k',1}(x),\psi_0(x)\rangle\big\|\Big]\leq \varepsilon/2 \tag{C.36}$$

Finally, we can rewrite $D(k')$ as

$$\begin{aligned} D(k')=\mathbb{E}\big[\langle f^*_{k',1}(x),\psi_0(x)\rangle\big] =&\mathbb{E}\big[\langle f^*_{k',1}(x),\psi_0(x)\rangle|x\in A_k\big]\Pr(x\in A_k)\\ &+\mathbb{E}\big[\langle f^*_{k',1}(x),\psi_0(x)\rangle|x\in B_k\setminus C_\delta\big]\Pr(x\in B_k\setminus C_\delta)\\ &+\mathbb{E}\big[\langle f^*_{k',1}(x),\psi_0(x)\rangle|x\in C_\delta\big]\Pr(x\in C_\delta), \end{aligned}$$

and because of ((C.33)) and ((C.36)), and since the first term is a constant function in terms of $k'$ and for all $k'\in[k-\delta/2,k]$, then we have

$$|D(k')-D(k)|\leq \varepsilon/4+\varepsilon/4+\varepsilon/2=\varepsilon. \tag{C.37}$$

This shows that $D(k')$ is lower semi-continuous around $k'=k$.

5. To prove this part, we first divide $\mathcal{X}$ into two subsets

$$G_{k'}=\Big\{x\in\mathcal{X}:\ \big|\operatorname*{argmax}_i(\ell_{k'}(x))(i)\big|=1\Big\}, \tag{C.38}$$

and $H_{k'}=\mathcal{X}\setminus G_{k'}$. We know that for $x\in G_{k'}$ we have

$$f^*_{k',0}(x)=f^*_{k',1}(x)=\begin{cases}1 & i=\min\{j\in\ \operatorname{argmax}\ell_{k'}(x)\}\\ 0 & \text{otherwise}\end{cases} \tag{C.39}$$

This concludes that

$$\mathbb{E}\big[\langle f^*_{k',0}(x),\psi_1(x)\rangle|x\in G_k\big]=\mathbb{E}\big[\langle f^*_{k',1}(x),\psi_1(x)\rangle|x\in G_k\big]. \tag{C.40}$$

Moreover, let us define the set $\Psi_1^{k'}=\{x\in\mathcal{X}:\exists c\in\mathbb{R},\forall j\in\ \operatorname{argmax}\ell_{k'}(x),\big(\psi_1(x)\big)(j)=c\}$ . We show that sum of the probabilities of $H_{k'}\setminus\Psi_1^{k'}$ is always bounded by $2^d$ for a set of choices for $k'$, or equivalently

$$\sum_{k'\in\mathcal{K}}\Pr(x\in H_{k'}\setminus\Psi_1^{k'})\leq 2^d, \tag{C.41}$$

for all finite or countably infinite choice of $\mathcal{K}\subseteq\mathbb{R}^+$. In fact, we know that for each instance $x$, $\operatorname*{argmax}_{j\in[1:d]}\big(\ell_k(x)\big)(j)$ can take up to $2^d$ cases of all subsets of $\{1,\dots,d\}$.

Therefore, we need to show that there cannot exist two values of $k, k'$ such that for $x \in \big(H_k \setminus \Psi_1^k\big) \cap \big(H_{k'} \setminus \Psi_1^{k'}\big)$ we have

$$\operatorname*{argmax}_j \big(\ell_k(x)\big)(j) = \operatorname*{argmax}_j \big(\ell_{k'}(x)\big)(j). \tag{C.42}$$

If we prove such identity, then due to pigeonhole principle, we have

$$\sum_{k' \in \mathcal{K}} \mathbb{1}_{x \in H_{k'} \setminus \Psi_1^{k'}} \leq 2^d, \tag{C.43}$$

which by integration over all values of $x$ concludes in ((C.41)). We prove this claim by contradiction. If we assume $k, k' \in \mathcal{K}$ such that for $x \in \big(H_k \setminus \Psi_1^k\big) \cap \big(H_{k'} \setminus \Psi_1^{k'}\big)$ the identity ((C.42)) holds, then because $x \in H_k \cap H_{k'}$, then the size of $\operatorname*{argmax}_j \big(\ell_k(x)\big)(j)$ and $\operatorname*{argmax}_j \big(\ell_{k'}(x)\big)(j)$ is at least 2. This concludes that

$$\big(\psi_0(x) - k\psi_1(x)\big)(i) = \big(\psi_0(x) - k\psi_1(x)\big)(j)$$

as well as

$$\big(\psi_0(x) - k'\psi_1(x)\big)(i) = \big(\psi_0(x) - k'\psi_1(x)\big)(j)$$

for all choices of $i, j \in \operatorname{argmax} \ell_k(x)$. As a result, we have

$$(k - k')\Big(\big(\psi_1(x)\big)(i) - \psi_1(x)\big)(j)\Big) = 0,$$

and because $k' \neq k$, we have

$$\big(\psi_1(x)\big)(i) = \psi_1(x)\big)(j),$$

for all $i, j \in \operatorname{argmax} \ell_k(x)$. Therefore, $x \in \Psi_1^{k'}$ and that is a contradiction.

Now that we know that the sum of the probabilities of $\Pr(x \in H_{k'} \setminus \Psi_1^{k'})$ is bounded, we can renormalize that and make a probability measure as

$$g(A) = \frac{\sum_{k \in A, \Pr(x \in H_k \setminus \Psi_1^k) > 0} \Pr(x \in H_k \setminus \Psi_1^k)}{\sum_{k : \Pr(x \in H_k \setminus \Psi_1^k) > 0} \Pr(x \in H_k \setminus \Psi_1^k)}. \tag{C.44}$$

Due to Lemma 15, for all $\varepsilon \geq 0$ we can find a small enough $\delta \geq 0$ such that

$g([k-\delta,k)) \leq \varepsilon/2^{d+1}$, and therefore for all $k' \in [k-\delta,k)$ we have

$$\begin{aligned}\Pr(x \in H_{k'} \setminus \Psi_1^{k'}) &\leq \sum_{t\in[k-\delta,k),\Pr(x\in H_t\setminus\Psi_1^t)>0]} \Pr(x \in H_t \setminus \Psi_1^t) \\ &= g\big([k-\delta,k)\big) \sum_{k:\Pr(x\in H_k\setminus\Psi_1^k)>0} \Pr(x \in H_k \setminus \Psi_1^k) \\ &\leq \frac{\varepsilon}{2^{d+1}} 2^d = \varepsilon/2,\end{aligned}$$

where the last inequality holds because of ((C.41)).

Now, using this and due to ((C.40)), and by defining $g_i(x) = \langle f_{k,i}^*(x), \psi_0(x)\rangle$ for $i = 1,2$, we can bound the difference of $D(k)$ and $C(k)$ as

$$\begin{aligned}\big|D(k) - C(k)\big| &= \Big|\mathbb{E}\big[g_1(x) - g_0(x) | x \in H_{k'}\big]\Pr(x \in H_{k'})\Big| \\ &\leq \Pr(x \in H_{k'} \setminus \Psi_1^{k'} | x \in H_{k'})\Big|\mathbb{E}\big[g_1(x) - g_0(x) | x \in H_{k'} \setminus \Psi_1^{k'}\big]\Big| \\ &\quad + \Pr(x \in H_{k'} \cap \Psi_1^{k'} | x \in H_{k'})\Big|\mathbb{E}\big[g_1(x) - g_0(x) | x \in H_{k'} \cap \Psi_1^{k'}\big]\Big| \\ &\overset{(a)}{\leq} 2(\varepsilon/2) + \Big|\mathbb{E}\big[g_1(x) - g_0(x) | x \in H_{k'} \cap \Psi_1^{k'}\big]\Big| \\ &\overset{(b)}{=} \varepsilon,\end{aligned}$$

where $(a)$ holds because $\|f_{k,0}^* - f_{k,1}^*\|_1 \leq \|f_{k,0}^*\|_1 + \|f_{k,1}^*\|_1 = 2$ and because of Hölder inequality we have $\big|\langle f_{k,0}^*(x) - f_{k,1}^*, \psi_1(x)\rangle\big| \leq \|f_{k,0}^* - f_{k,1}^*\|_1 \|\psi_1(x)\|_\infty \leq 2$. Moreover, to show that $(b)$ holds we know that for $x \in \Psi_1^{k'}$ we have $\big(\psi_1(x)\big)(i) = \big(\psi_1(x)\big)(j)$ for all $i, j \in \operatorname{argmax} \ell_{k'}(x)$. Therefore, because we know $g_0(x) = \big(\psi_1(x)\big)(i)$ for $i \in \operatorname*{argmin}_{j\in \operatorname*{argmax}_l \left(\ell_{k'}(x)\right)(l)} \big(\psi_1(x)\big)(j) \subseteq \operatorname*{argmax}_l \big(\ell_{k'}(x)\big)(l)$ and $g_1(x) = \big(\psi_1(x)\big)(j)$ for $j \in \operatorname*{argmax}_{j\in \operatorname*{argmax}_l \left(\ell_{k'}(x)\right)(l)} \big(\psi_0(x)\big)(j) \subseteq \operatorname*{argmax}_l \big(\ell_{k'}(x)\big)(l)$, we have $g_0(x) = g_1(x)$. The above inequality proves that the limit of $C(k')$ and $D(k')$ for $k' \uparrow k$ are equal and that completes the proof.

□

To prove this theorem, we take the following steps: (i) We show that the set $\mathcal{K}$ has a non-negative member, (ii) we show that the prediction function $f_{k,p}^*(x)$ achieves the inequality constraint tightly, and by Theorem 6 we can conclude that $f_{k,p}^*(x)$ is the optimal solution.

- **step (i)**: It is easy to see that the Bayes optimal solution of the prediction function

in ((4.3)) without any constraint is

$$\big(f^*(x)\big)(i) = \begin{cases} 1 & \big(\psi_0(x)\big)(i) > \big(\psi_0(x)\big)(j) \text{ for all } j \neq i \\ 0 & \big(\psi_0(x)\big)(i) < \max_j \big(\psi_0(x)\big)(j) \\ p_i(x) & \text{otherwise} \end{cases},$$

where $p_i(x) \in \Delta_d$ is an arbitrary vector. We can see that by setting

$$\big(p_i(x)\big)(j) = \begin{cases} 1 & j = \min\{ \underset{t \in \operatorname{argmax} \ell_0(x)}{\operatorname{argmin}} \big(\psi_1(x)\big)(t)\} \\ 0 & \text{otherwise} \end{cases},$$

then the two prediction functions $f^*(x)$ and $f^*_{0,0}(x)$ are equal (See statement of Theorem 7).

Now, in the first and second part of Lemma 16 we have shown that $\mathbb{E}\big[\langle f^*_{k,0}(x), \psi_1(x)\rangle\big]$ is upper semi-continuous and monotonically non-increasing. Therefore, for all $k \in \mathbb{R}^+$ we have

$$\mathbb{E}\big[\langle f^*_{k,0}(x), \psi_1(x)\rangle\big] \leq \mathbb{E}\big[\langle f^*_{0,0}(x), \psi_1(x)\rangle\big] = \mathbb{E}\big[\langle f^*(x), \psi_1(x)\rangle\big].$$

Similarly, we can show that for $k \to \infty$, the solution is equivalent to the Bayes minimizer of

$$f^{**}(x) = \underset{f \in \Delta_d^{\mathcal{X}}}{\operatorname{argmin}} \mathbb{E}\big[\langle f(x), \psi_1(x)\rangle\big].$$

Therefore, since $\delta$ is an interior point of all possible values, it lays on the interval $\big(\mathbb{E}\big[\langle f^{**}(x), \psi_1(x)\rangle\big], \mathbb{E}\big[\langle f^*(x), \psi_1(x)\rangle\big]\big)$, due to the montonicity and upper semi-continuity of $\mathbb{E}\big[\langle f^*_{k,0}, \psi_1(x)\rangle\big]$, we can find $t$ such that

$$\mathbb{E}\big[\langle f^*_{t,0}(x), \psi_1(x)\rangle\big] \leq \delta \leq \lim_{\tau \uparrow t} \mathbb{E}\big[\langle f^*_{k,0}(x), \psi_1(x)\rangle\big]. \tag{C.45}$$

Moreover, this $t$ should be a positive scalar, since otherwise we have

$$\mathbb{E}\big[\langle f^*_{t,0}(x), \psi_1(x)\rangle\big] \geq \mathbb{E}\big[\langle f^*_{0,0}(x), \psi_1(x)\rangle\big] = \mathbb{E}\big[\langle f^*(x), \psi_1(x)\rangle\big] > \delta,$$

which is a contradiction to ((C.45)).

- **step (ii)**: In this step, we consider the following two cases:
  - $C(t)$ **is continuous at** $t$: In this case, ((C.45)) is equivalent to $\delta = C(t) = \mathbb{E}\big[\langle f^*_{t,0}(x), \psi_0(x)\rangle\big]$, which means that the prediction function $f^*_{k,0}(x)$ achieves the constraint tightly, and therefore using Theorem 6 $f^*_{k,0}(x)$ is the optimal

solution.

– $C(t)$ **is discontinuous at** $t$: To show that we can achieve the highest constraint in this case, we first condition the constraint into two events $x \in G_k$ and $x \in \mathcal{X} \setminus G_k$, where $G_k$ is defined in ((C.38)). We know that in the latter case $x \in \mathcal{X} \setminus G_k$, the prediction function $f^*_{k,p}$ can be decomposed into two components

$$f^*_{k,p}(x) = p f^*_{k,1}(x) + (1-p) f^*_{k,0}(x), \tag{C.46}$$

while for $x \in G_k$ the prediction function $f^*_{k,p}(x) = f^*_{k,0}(x) = f^*_{k,1}(x)$ for all $p \in [0,1]$. Therefore, in both cases ((C.46)) holds, and we have

$$\begin{aligned}\mathbb{E}\big[\langle f^*_{k,p}(x), \psi_1(x)\rangle\big] &= \mathbb{E}\big[\langle p f^*_{k,1}(x) + (1-p) f^*_{k,0}(x), \psi_1(x)\rangle\big] \\ &= p\mathbb{E}\big[\langle f^*_{k,1}(x), \psi_1(x)\rangle\big] + (1-p)\mathbb{E}\big[\langle f^*_{k,0}(x), \psi_1(x)\rangle\big] \\ &= pD(k) + (1-p)C(k), \end{aligned} \tag{C.47}$$

where $C(\cdot)$ and $D(\cdot)$ are defined in Lemma 16. Using this lemma, we know that $D(\cdot)$ is lower semi-continuous, and $\lim_{k'\uparrow k} C(k) = \lim_{k'\uparrow k} D(k)$. Therefore, together with ((C.47)) and the definition of $p$ in the statement of theorem, we have

$$\begin{aligned}\mathbb{E}\big[\langle f^*_{k,p}(x), \psi_0(x)\rangle\big] =& p \lim_{k'\uparrow k} C(k') + (1-p)C(k) \\ =& \frac{C(k) - c}{C(k) - \lim_{k'\uparrow k} C(k')} \lim_{k'\uparrow k} C(k') \\ &+ \frac{c - \lim_{k'\uparrow k} C(k')}{C(k) - \lim_{k'\uparrow k} C(k')} C(k) = c. \end{aligned} \tag{C.48}$$

Equivalently, the prediction function achieves the constraint inequality tightly, and therefore by Theorem 6 this is sufficient to be the optimal solution to the constrained optimization problem.

## C.10 Proof of Theorem 8

Through the proof of this theorem, we use (Blumer *et al.*, 1989, Lemma 3.2.3) that implies that the class of multiplications of $k$ binary functions $f_i(x)$ for $i \in [1:k]$ within a hypothesis class with VC dimension $VC(f_i) = d$ itself has a VC dimension that is

bounded as

$$\underbrace{VC(\{\prod_{i=1}^{k} f_i : f_i \in \mathcal{H}_i, VC(\mathcal{H}_i) = d\})}_{\mathcal{H}'} \leq 2dk\log 3k. \tag{C.49}$$

In fact, we use a simple extension to this lemma for which the VC dimension of the functions is not $d$ itself but is bounded above by $d$. In such case we claim that ((C.49)) still holds. The starting point for the proof to this lemma is bounding the size of the restriction $\Pi_{\mathcal{H}}(S) = |\{h \cap S : h \in \mathcal{H}\}|$ for the hypothesis class $\mathcal{H}$ by

$$\Pi_{\mathcal{H}}(S) \leq \big(\frac{em}{d}\big)^d, \tag{C.50}$$

where $VC(\mathcal{H}) = d$ and $m = |S|$. However, this inequality holds for the hypothesis classes that have VC dimensions that are bounded by $d$. The reason is increasingly monotonicity of RHS of ((C.50)). In fact, by obtaining the gradient of $\big(\frac{em}{d}\big)^d$ in terms of $d$ we have

$$\frac{\partial\big(\frac{em}{d}\big)^d}{\partial d} = \frac{\partial\big(e^{d\log em/d}\big)}{\partial d} = (\log em/d - 1)\big(\frac{em}{d}\big)^d,$$

which is nonnegative as long as $m \geq d$. If we particularly set $m^* = 2dk\log 3k$, then $m^* \geq d$ and therefore ((C.50)) holds. Next, similar to the proof of (Blumer *et al.*, 1989, Lemma 3.2.3), we can show that for the set $S$ with size $m^*$ we have

$$\Pi_{\mathcal{H}'}(S) \leq \Pi^k_{\mathcal{H}_1}(S) \leq \big(\frac{em^*}{d}\big)^{dk} \leq 2^{m^*},$$

which means that $S$ cannot be shattered by $\mathcal{H}'$, and therefore the VC dimension of this hypothesis class must be bounded by $m^*$.

We further use the following lemma:

□

**Lemma 17.** *For arbitrary sets of functions* $\{\phi^i_1(x)\}_{i=1}^n$ *and* $\{\phi^i_2(x)\}_{i=1}^n$ *on* $\mathbb{R}$ *and for a given* $d \in \mathbb{R}$ *the hypothesis class*

$$\mathcal{H} = \big\{\prod_{i=1}^{n} \operatorname{sgn}\big(\phi^i_1(x) - k\phi^i_2(x) - d\big) \,:\, k \in \mathbb{R}\big\},$$

*has the VC dimension of at most* 4.

*Proof.* To prove this lemma, we show that the form of the product in the definition of $\mathcal{H}$ reduces to the form of an interval on $\mathbb{R}$, which is known to have VC dimension of 2. In

fact, each term $\text{sgn}(\phi_1^i(x) - k\phi_2^i(x) - d)$ can be rewritten as

$$\begin{aligned}\text{sgn}(\phi_1^i(x) - k\phi_2^i(x) - d) =& \text{sgn}(\tfrac{\phi_1^i(x)-d}{\phi_2^i(x)} - k)\text{sgn}(\phi_2^i(x)) + \text{sgn}(k - \tfrac{\phi_1^i(x)-d}{\phi_2^i(x)})\text{sgn}(-\phi_2^i(x)) \\ &+ \text{sgn}(\phi_1^i(x) - d)\mathbb{I}_{\phi_2^i(x)=0}.\end{aligned}$$

As a result, by multiplying all terms we have

$$\prod_{i=1}^{n} \text{sgn}\big(\phi_1^i(x) - k\phi_2^i(x) - d\big) = \text{sgn}(\min_{i\in\mathcal{A}_x} \tfrac{\phi_1^i(x)-d}{\phi_2(x)} - k)\text{sgn}(k - \max_{i\in\mathcal{B}_x} \tfrac{\phi_1^i(x)-d}{\phi_2(x)}) \prod_{i\in\mathcal{C}_x} \text{sgn}(\phi_1^i(x) - d), \tag{C.51}$$

where $\mathcal{A}_x$, $\mathcal{B}_x$, and $\mathcal{C}_x$ are defined as $\mathcal{A}_x = \{i \in [1:n] : \phi_2^i(x) > 0\}$, $\mathcal{B}_x = \{i \in [1:n] : \phi_2^i(x) < 0\}$, and $\mathcal{C}_x = \{i \in [1:n] : \phi_2^i(x) = 0\}$. Now, we see that the first two terms define an interval for $k \in \big(f_1(x), f_2(x)\big)$ where $f_1(x) = \max_{i\in\mathcal{B}_x} \frac{\phi_1^i(x)-d}{\phi_2^i(x)}$ and $f_2(x) = \min_{i\in\mathcal{A}_x} \frac{\phi_1^i(x)-d}{\phi_2^i(x)}$. Next, we prove that the VC dimension of the hypothesis class of all such functions is less than the VC dimension of $\mathcal{G} = \big\{f : \mathbb{R}\times\mathbb{R} \to \{0,1\} : f(x,y) = \text{sgn}(x - k_1)\text{sgn}(k_2 - y), k_1, k_2 \in \mathbb{R}\big\}$. The reason is that if the aforementioned interval can shatter a set $\mathcal{S}$, then we can find the corresponding values of $f_1(x)$ and $f_2(x)$ for each $x \in \mathcal{S}$, and then form the pair $(x_i, y_i)$ where $x_i = f_1(x)$ and $y_i = f_2(x)$, and by setting $k_1 = k_2 = k$, we can shatter the set $\{(x_i, y_i)\}_{i=1}^{|\mathcal{S}|}$ with $\mathcal{G}$. Note that here all pairs are identical. The reason is that if not, i.e., if $f_1(x) = f_1(x')$ and $f_2(x) = f_2(x')$ for $x, x' \in \mathcal{S}$ and $x \neq x'$, then, for all possible $k$, we have $\text{sgn}(k - f_1(x))\text{sgn}(f_2(x) - k) = \text{sgn}(k - f_1(x'))\text{sgn}(f_2(x') - k)$, and therefore we cannot shatter $\mathcal{S}$ by $\text{sgn}(k - f_1(x))\text{sgn}(f_2(x) - k)$. Therefore, the set $\{(x_i, y_i)\}_{i=1}^{|\mathcal{S}|}$ has the same cardinality of $\mathcal{S}$, which in consequence proves that the VC dimension of all $\text{sgn}(k - f_1(x))\text{sgn}(f_2(x) - k)$ is bounded by $VC(\mathcal{G})$. Moreover, $VC(\mathcal{G}) \leq 4$, since for each 5 points in two-dimensional space, one is in the convex hull of the others, and in case that all others are labeled as 1, the one in the convex hull also must be labeled as 1. As a result, $\mathcal{G}$ cannot shatter 5 points, and therefore $VC(\mathcal{G}) \leq 4$.

Up to now, we have shown that the class of functions equal to the first two terms of ((C.51)) has a VC dimension that is bounded by 4. Next, we show that multiplying a hypothesis class $\mathcal{H}$ with a binary function $\phi(x)$ does not increase the VC dimension of that class. More formally, if we define

$$\mathcal{H} = \{\phi(x)f(x) : f \in \mathcal{H}'\},$$

then $VC(\mathcal{H}) \leq VC(\mathcal{H}')$. The reason is that if we can shatter a set $\mathcal{S}$ using $\mathcal{H}$, then for each member $x \in \mathcal{S}$ there exists two members $f_1, f_2$ of $\mathcal{H}'$ such that $f_1(x) = 1$ and $f_2(x) = 0$. This means that $\phi(x) \neq 0$, because otherwise $f_1(x) = 1$ would not be achievable. Therefore, $\phi(x) = 1$ for all $x \in \mathcal{S}$, and as a result similarly $\mathcal{H}'$ can shatter $\mathcal{S}$, which

proves that $VC(\mathcal{H}) \leq VC(\mathcal{H}')$.

Finally, since we know that the class of all functions in $\mathcal{H}$ is in form of $\operatorname{sgn}(k - f_1(x))\operatorname{sgn}(f_2(x) - k)$ multiplied with a binary function, then we conclude that $VC(\mathcal{H}) \leq 4$. □

To prove the rest of the theorem, we need to show that for all choices of $\hat{k}$ and $\hat{p}$ the difference of the empirical and the true loss is bounded. In fact, we should find a bound in form of

$$\Pr\Big(\sup_{k,p}\big|\mathbb{E}_{S^n}\big[\langle f^*_{k,p}(x), \psi_0(x)\rangle\big] - \mathbb{E}_{\mu}\big[\langle f^*_{k,p}(x), \psi_0(x)\rangle\big]\big| \leq d_n\Big) \geq 1-\varepsilon.$$

Here, we divide the class $\mathcal{X}$ into two subsets $G_k$ and $H_k = \mathcal{X} \setminus G_k$, where $G_k$ is defined in ((C.38)).

Now, using the definition of $f^*_{k,p}(x)$, we know that within $G_k$, the inner-product $\langle f^*_{k,p}(x), \psi_1(x)\rangle$ can be rewritten as

$$\begin{aligned}
\langle f^*_{k,p}(x), \psi_1(x)\rangle &= \Big(\psi_1(x)\Big)\big(\operatorname*{argmax}_i \big(\ell_k(x)\big)(i)\big) \\
&= \sum_{j=1}^{d} \big(\psi_1(x)\big)(j) \prod_{i\neq j} \operatorname{sgn}\Big(\big(\ell_k(x)\big)(j) - \big(\ell_k(x)\big)(i)\Big) \\
&= \sum_{j=1}^{d} \big(\psi_1(x)\big)(j) \underbrace{\prod_{i\neq j} \operatorname{sgn}\Big(\big(\psi_0(x)\big)(j) - \big(\psi_0(x)\big)(0) - k\big[\big(\psi_1(x)\big)(j) - \big(\psi_1(x)\big)(i)\big]\Big)}_{\Phi^k_j(x)}.
\end{aligned}$$

Now, we can condition $x$ on being a member of $G_k$, and therefore the maximum difference between the two empirical and true expectation is as

$$\begin{aligned}
&\sup_{k,p}\Big|\mathbb{E}_{S^n}\big[\langle f^*_{k,p}(x), \psi_1(x)\rangle \,|\, x \in G_k\big] - \mathbb{E}_{\mu}\big[\langle f^*_{k,p}(x), \psi_1(x)\rangle \,|\, x \in G_k\big]\Big| \\
&\leq \sum_{j=1}^{d} \sup_{k,p}\Big|\mathbb{E}_{S^n}\Big[\big(\psi_1(x)\big)(j)\cdot\Phi^k_j(x) \,|\, x \in G_k\Big] - \mathbb{E}_{\mu}\Big[\big(\psi_1(x)\big)(j)\cdot\Phi^k_j(x) \,|\, x \in G_k\Big]\Big|. \quad \text{(C.52)}
\end{aligned}$$

Now, we bound the inner term of ((C.52)) in a high probability setting. To that end, we use Rademacher's inequality in (Shalev-Shwartz and Ben-David, 2014, Theorem 26.5), which shows that maximum difference between the expected value of a function $h \in \mathcal{H}$ over empirical distribution and the true distribution is $2R(\mathcal{H}) + 4c\sqrt{\frac{\ln 4/\varepsilon}{n}}$ where $R(\mathcal{H})$ is the Rademacher's complexity of the class of function $\mathcal{H}$ and $c$ is maximum value that $h$ can take. By defining

$$h(x) := \big(\psi_1(x)\big)(j)\cdot\Phi^k_j(x),$$

we have $c = \|\big(\psi_1(x)\big)(j)\|_\infty \leq 1$. Therefore, we have for all $h$,

$$\sup_{h\in\mathcal{H}} \mathbb{E}_{S^n}\big[h(x)\big] - \mathbb{E}_\mu\big[h(x)\big] \leq 2R(\mathcal{H}) + 4\sqrt{\frac{\ln 4d/\varepsilon}{n}}, \tag{C.53}$$

with probability at least $1-\frac{\varepsilon}{d}$. Now, we can use contraction Lemma (Shalev-Shwartz and Ben-David, 2014, Lemma 26.9) to show that since $\|\big(\psi_1(x)\big)(j)\|_\infty \leq 1$, then $R(\mathcal{H}) \leq R(\mathcal{F})$, where $\mathcal{F} = \{\Phi_j^k(x), k \in \mathbb{R}\}$. Moreover, $\mathcal{F}$ contains functions that are all multiplication of $d-1$ binary functions all in form of

$$\operatorname{sgn}\Big(\big(\psi_1(x)\big)(j) - \big(\psi_1(x)\big)(0) - k\big[\big(\psi_0(x)\big)(j) - \big(\psi_0(x)\big)(i)\big]\Big).$$

Lemma 17 shows that the hypothesis class that contains products of all such function has a VC-dimension that is bounded by 4. As a result, the Rademacher's complexity of $\mathcal{F}$ is bounded using (Mohri *et al.*, 2018, Corollary 3.8, Corollary3.18) as

$$R(\mathcal{F}) \leq \sqrt{\frac{4\log en/4}{n}},$$

and therefore together with ((C.53)) for all $h \in \mathcal{H}$ we have

$$\mathbb{E}_{S^n}\big[h(x)\big] - \mathbb{E}_\mu\big[h(x)\big] \leq 2\sqrt{\frac{4\log en/4}{n}} + 4\sqrt{\frac{\ln 4d/\varepsilon}{n}},$$

with probability at least $1-\frac{\varepsilon}{d}$. Hence, using ((C.52)) we have

$$\begin{aligned}\sup_{k,p}\Big|\mathbb{E}_{S^n}\big[\langle f^*_{k,p}(x), \psi_1(x)\rangle \,|\, x \in G_k\big] - \mathbb{E}_\mu\big[\langle f^*_{k,p}(x), \psi_1(x)\rangle \,|\, x \in G_k\big]\Big| \\ \leq 2d\sqrt{\frac{4\log el/4}{l}} + 4d\sqrt{\frac{\ln 4d/\varepsilon}{l}},\end{aligned} \tag{C.54}$$

with probability at least $1-\varepsilon$. In the last inequality, we used Bonferroni's inequality on $\varepsilon/d$ bad events that each summand of ((C.52)) is not within the concentration bound.

Next, we consider the region $H_k$ in which there are at least two maximizer components of $\ell_k(x)$. In this case, by definition of $\hat{f}_{k,p}(x)$, among these maximizers, we choose the first maximizer of $\psi_0(x)$ with probability $p$ and the first minimizer of $\psi_1(x)$ with probability $1-p$. Therefore, by condition on these cases, and if we define

$$E(k,p) := \Big|\mathbb{E}_{S^n}\big[\langle \hat{f}_{k,p}(x), \psi_1(x)\rangle \,|\, x \in H_k\big] - \mathbb{E}_\mu\big[\langle \hat{f}_{k,p}(x), \psi_1(x)\rangle \,|\, x \in H_k\big]\Big|, \tag{C.55}$$

then we have

$$\sup_{k,p} E(k,p) \leq \sup_{k,p} pE(k,1) + (1-p)E(k,0) \leq \sup_{k,p} E(k,1) + \sup_{k,p} E(k,0). \tag{C.56}$$

Now, to bound $E(k,1)$, we first rewrite the closed-form solution of $\hat{f}_{k,1}(x)$ as

$$\big(\hat{f}_{k,1}(x)\big)(i) = \operatorname{sgn}\Big(\big(\ell_k(x)\big)(i) \geq \max_j \big(\ell_k(x)\big)(j) - d\Big) \prod_{j<i} l_{ij}(x) \prod_{j>i} u_{ij}(x), \tag{C.57}$$

where $l_{ij}(x)$ and $u_{ij}(x)$ are defined as

$$l_{ij}(x) := 1 - \mathbb{I}_{\big(\psi_0(x)\big)(i) \leq \big(\psi_0(x)\big)(j)} \mathbb{I}_{\big(\ell_k(x)\big)(j) \geq \max_t \big(\ell_k(x)\big)(t)},$$

and

$$u_{ij}(x) := 1 - \mathbb{I}_{\big(\psi_0(x)\big)(i) < \big(\psi_0(x)\big)(j)} \mathbb{I}_{\big(\ell_k(x)\big)(j) \geq \max_t \big(\ell_k(x)\big)(t)},$$

respectively. Note that the only difference between the definition of $u_{ij}(x)$ and $l_{ij}(x)$ is that $u_{ij}(x)$ permits the equality of $\big(\psi_0(x)\big)(i)$ with other components, while that is not the case for $l_{ij}(x)$. This difference lets us find the *first* component with the largest value of $\psi_0(x)$.

Now, we can rewrite $\operatorname{sgn}\Big(\big(\ell_k(x)\big)(j) \geq \max_t \big(\ell_k(x)\big)(t)\Big)$ as the product

$$\operatorname{sgn}\Big(\big(\ell_k(x)\big)(j) \geq \max_t \big(\ell_k(x)\big)(t) - d\Big) := \prod_{l \in [1:d]} \operatorname{sgn}\Big(\big(\ell_k(x)\big)(j) \geq \big(\ell_k(x)\big)(l)\Big).$$

As shown in Lemma 17, the class of such function has VC dimension of at most 4. Furthermore, multiplying a hypothesis class with a function such as $\operatorname{sgn}\Big(\big(\psi_0(x)\big)(i) \geq \big(\psi_0(x)\big)(j)\Big)$ and $\operatorname{sgn}\Big(\big(\psi_0(x)\big)(i) > \big(\psi_0(x)\big)(j)\Big)$ does not increase the VC dimension (See proof of Lemma 17, and neither does negation. Therefore, in RHS of ((C.57)) we can count $d$ number of functions, each with a hypothesis class with the VC dimension of at most 4, and therefore using the early discussions in this proof ((C.49)), $\big(\hat{f}_{k,1}(x)\big)(i)$ is within a function class with the VC dimension of at most $8d\log(3d)$. Therefore, similar to ((C.54)) in previous part, we can bound $\sup_{k,p} E(k,1)$ as

$$\begin{aligned}\sup_{k,p} E(k,1) \leq & 2d\sqrt{\frac{8d\log(3d)\log\big(en/\big(8d\log(3d)\big)\big)}{n}} \\ & + 4d\sqrt{\frac{\ln 4d/\varepsilon}{n}},\end{aligned} \tag{C.58}$$

for $l \geq 8d\log(3d)$ with probability at least $1-\varepsilon$.

We can similarly, show that $\sup_{k,p} E(k,0)$ is bounded as

$$\sup_{k,p} E(k,0) \leq 2d\sqrt{\frac{8d\log(3d)\log(en/\big((8n+8)\log(3d)\big)}{n}} + 4d\sqrt{\frac{\ln 4d/\varepsilon}{n}}, \tag{C.59}$$

Therefore, using ((C.54)), ((C.55)), ((C.56)), ((C.58)), ((C.59)), and the application Bonferonni's inequality we have

$$\sup_{k,p}\left|\mathbb{E}_{S^n}\left[\langle f^*_{k,p}(x),\psi_0(x)\rangle\right]-\mathbb{E}_{\mu}\left[\langle f^*_{k,p}(x),\psi_0(x)\rangle\right]\right| \leq 6d\sqrt{\frac{8d\log(3d)\log\frac{el}{(8n+8)\log(3d)}}{l}}+12d\sqrt{\frac{\ln\frac{12d}{\varepsilon}}{l}} \tag{C.60}$$

$$:= d_n(\varepsilon), \tag{C.61}$$

with probability at least $1-\varepsilon$. Therefore, by assuming $\mathbb{E}_{S^n}\left[\langle f^*_{k,p}(x),\psi_1(x)\rangle\right] \leq \alpha - d_n(\varepsilon)$, we assure that $\mathbb{E}_{\mu}\left[\langle f^*_{k,p}(x),\psi_1(x)\rangle\right] \leq \alpha$, with probability at least $1-\varepsilon$, and this completes the proof.

## C.11 Proof of Theorem 9

We first introduce three lemmas that are useful in proving this theorem.

**Lemma 18.** *If $\delta$ is an $\varepsilon$-interior point of the set $\mathcal{C} = \left\{\mathbb{E}_{\mu}\left[\langle f(x),\psi_1(x)\rangle\right] : f \in \Delta_d^{\mathcal{X}}\right\}$, then $\delta$ is $(\varepsilon/2)$-interior point of $\mathcal{D} = \left\{\mathbb{E}_{S^n}\left[\langle f(x),\psi_1(x)\rangle\right] : f \in \Delta_d^{\mathcal{X}}\right\}$ with probability $1-2e^{-\frac{l\varepsilon^2}{4}}$.*

*Proof.* The proof of this lemma is a direct application of Hoeffding's inequality. In fact, for $\|\psi_1\|_\infty \leq C$ that inequality together with Hölder's inequality imply that

$$\Pr\Big(\left|\mathbb{E}_{\mu}\left[\langle f(x),\psi_1(x)\rangle\right]-\mathbb{E}_{S^n}\left[\langle f(x),\psi_1(x)\rangle\right]\right| \geq \varepsilon/2\Big) \leq e^{-\frac{n\varepsilon^2}{4C^2}}.$$

Therefore, if there exists $f_1$ such that $\mathbb{E}_{\mu}\left[\langle f_1(x),\psi_1(x)\rangle\right] = \varepsilon$, then with probability at least $1-e^{-\frac{n\varepsilon^2}{4C^2}}$ we have $\mathbb{E}_{S^n}\left[\langle f_1(x),\psi_1(x)\rangle\right] \in [\varepsilon/2, 3\varepsilon/2]$. Similarly, if $f_2$ exists such that $\mathbb{E}_{\mu}\left[\langle f_1(x),\psi_1(x)\rangle\right] = -\varepsilon$, then with probability $1-e^{-\frac{n\varepsilon^2}{4C^2}}$ we have $\mathbb{E}_{S^n}\left[\langle f_2(x),\psi_1(x)\rangle\right] \in [-3\varepsilon/2, -\varepsilon/2]$. As a result of Bonferroni's inequality, with probability at least $1-$

$2e^{-\frac{n\varepsilon^2}{4C^2}}$ both these events happen, and because of the convexity of the set $\mathcal{D}$ we can say that with such probability all values between $a_0 \in [-3\varepsilon/2, -\varepsilon/2]$ and $a_1 \in [\varepsilon/2, 3\varepsilon/2]$ are in $\mathcal{D}$ too. This, of course at least contains the interval $[-\varepsilon/2, \varepsilon/2]$. □

**Lemma 19.** *Assume that we have an approximation $\hat{\psi}_1(x)$ of $\psi_1(x)$ with the error bounded as $\|\hat{\psi}_1(x) - \psi_1(x)\|_\infty \leq \varepsilon$. Further let $\varepsilon' \in \mathbb{R}^+$ such that $\varepsilon' \geq \varepsilon$. Now, if for $\sigma \in \{-\varepsilon', \varepsilon'\}$ there exists a rule $f \in \Delta_d^{\mathcal{X}}$ such that $\mathbb{E}_\mu\left[\langle f(x), \psi_1(x)\rangle\right] = \delta + \sigma$, then there exists $k \in \mathbb{R}$ as well as $p \in [0,1]$ such that $\mathbb{E}_\mu\left[\langle \hat{f}_{k,p}(x), \hat{\psi}_1(x)\rangle\right] = \delta + \frac{\varepsilon' - \varepsilon}{2}$.*

*Proof.* Firstly, because of Hölder's inequality we know that

$$\left|\mathbb{E}_\mu\left[\langle f(x), \psi_1(x)\rangle\right] - \mathbb{E}_\mu\left[\langle f(x), \hat{\psi}_1(x)\rangle\right]\right| \leq \varepsilon \|f^*_{k,p}(x)\|_1 = \varepsilon,$$

for all $f \in \Delta_d^{\mathcal{X}}$. Therefore, by setting $\sigma = \varepsilon'$ and $\sigma = -\varepsilon'$, we can show that for $f_1 \in \Delta_d^{\mathcal{X}}$ such that

$$\mathbb{E}_\mu\left[\langle f_1(x), \psi_1(x)\rangle\rangle\right] = \delta + \varepsilon',$$

then

$$\mathbb{E}_\mu\left[\langle f_1(x), \hat{\psi}_1(x)\rangle\right] \geq \delta + \varepsilon' - \varepsilon,$$

and where for $f_2 \in \Delta_d^{\mathcal{X}}$

$$\mathbb{E}_\mu\left[\langle f_2(x), \psi_1(x)\rangle\rangle\right] = \delta - \varepsilon',$$

then

$$\mathbb{E}_\mu\left[\langle f_2(x), \hat{\psi}_1(x)\rangle\right] \leq \delta - \varepsilon' + \varepsilon.$$

Now, because of step (iii) of the proof of Theorem 6, we know that the set of constraints for all rules within $\Delta_d^{\mathcal{X}}$ is convex. Therefore, since we can achieve two points $f_1, f_2$ such that the constraint $\mathbb{E}_\mu\left[\langle f_i(x), \hat{\psi}_1(x)\rangle\right]$ can achieve two points above $\delta + \varepsilon' - \varepsilon$ and below $\delta - \varepsilon' + \varepsilon$, then for each $c \in [\delta - \varepsilon' + \varepsilon, \delta + \varepsilon' - \varepsilon]$ there exists $f \in \Delta_d^{\mathcal{X}}$ such that $\mathbb{E}_\mu\left[\langle f(x), \hat{\psi}_1(x)\rangle\right] = c$. Now, let $c = \delta + \frac{\varepsilon' - \varepsilon}{2}$. In the following, we show that there exists $k \in \mathbb{R}$ and $p \in [0,1]$ such that further $\mathbb{E}_\mu\left[\langle \hat{f}_{k,p}(x), \hat{\psi}_1(x)\rangle\right] = c$.

To that end, we first remind that Lemma 16 shows that $\mathbb{E}_\mu\left[\langle \hat{f}_{k,0}(x), \hat{\psi}_1(x)\rangle\right]$ is monotonically non-increasing in terms of $k$. We show that for $k \in \mathbb{R}^-$ we have $\max \hat{\psi}_1(x) - \langle \hat{f}_{k,0}(x), \hat{\psi}_1(x)\rangle \leq -\frac{2}{k}$. The reason is that if $j \in \operatorname{argmax}_l \left(\hat{\psi}_0(x) - k\hat{\psi}_1(x)\right)(l)$ and $j' \in \operatorname{argmax}_l \left(\hat{\psi}_1(x)\right)(l)$, then we have

$$\left(\hat{\psi}_0(x) - k\hat{\psi}_1(x)\right)(j) \geq \left(\hat{\psi}_0(x) - k\hat{\psi}_1(x)\right)(j'),$$

which concludes that

$$-k\big[\big(\hat{\psi}_1(x)\big)(j)-\big(\hat{\psi}_1(x)\big)(j')\big]\geq\big(\hat{\psi}_0(x)\big)(j')-\big(\hat{\psi}_0(x)\big)(j)\geq-2.$$

Therefore, since

$$\mathbb{E}_\mu\big[\langle\ \operatorname{argmax}\hat{\psi}_1(x),\hat{\psi}_1(x)\rangle\big]=\max_{f\in\Delta_d^{\mathcal{X}}}\mathbb{E}_\mu\big[\langle f(x),\hat{\psi}_1(x)\rangle\big]\geq\delta+\varepsilon'-\varepsilon,$$

where the last inequality holds due to the existence of $f_1$, then for $k\leq-8/(\varepsilon'-\varepsilon)$ we have

$$\mathbb{E}_\mu\big[\langle\hat{f}_{k,0}(x),\hat{\psi}_1(x)\rangle\big]\geq\delta+\varepsilon'-\varepsilon-\frac{2}{-8/(\varepsilon'-\varepsilon)}\geq\delta+3\frac{\varepsilon'-\varepsilon}{4}.$$

Similarly, if we let $k\geq 8/(\varepsilon'-\varepsilon)$ we can prove that

$$\mathbb{E}_\mu\big[\langle\hat{f}_{k,0}(x),\hat{\psi}_1(x)\rangle\big]\leq\delta-\varepsilon'+\varepsilon+2l\leq\delta-3\frac{\varepsilon'-\varepsilon}{4}.$$

As a result, the set $\mathcal{C}=\{k:\mathbb{E}_\mu\big[\langle\hat{f}_{k,0}(x),\hat{\psi}_1(x)\rangle\big]\geq c\}$ is non-empty and bounded below by $-\frac{8}{\varepsilon'-\varepsilon}$. Therefore, its infimum exists and is also bounded below by $-\frac{8}{\varepsilon'-\varepsilon}$. Let us name that infimum $\hat{k}$. Now, if $\mathbb{E}_\mu\big[\langle\hat{f}_{k,0,0}(x),\hat{\psi}_1(x)\rangle\big]$ is continuous at $k=\hat{k}$, then we can show that $\mathbb{E}_\mu\big[\langle\hat{f}_{\hat{k},0}(x),\hat{\psi}_1(x)\rangle\big]=c$. If not, then as shown in step (ii) of the proof of Theorem 6, and in particular in ((C.48)), there exists $p$ such that $\mathbb{E}_\mu\big[\langle\hat{f}_{\hat{k},p}(x),\hat{\psi}_1(x)\rangle\big]=c$. This completes the proof. □

**Lemma 20.** *If $\|\hat{\psi}_0-\psi_0\|_\infty\leq\delta_0$ and $\|\hat{\psi}_1-\psi_1\|_\infty\leq\delta_1$, and for $k\in[-K,K]$, and $k'\leq k-\frac{2(\delta_0+K\delta_1)}{T}$ for $T\in\mathbb{R}^+$, then we have*

$$\mathbb{E}\big[\langle\hat{f}_{k,0,0}(x)-f^*_{k',0}(x),\psi_1(x)\rangle\big]\leq T.$$

*Proof.* The proof of this lemma bears similarity to that of Lemma 16. Here too, we define $\hat{\ell}_k(x)=\hat{\psi}_0(x)-k\hat{\psi}_1(x)$. Next, we have

$$\hat{f}_{k,0}(x)=\begin{cases}1 & i=\min\{\quad\operatorname{argmin}\quad\hat{\psi}_1(x)\}\\ & \quad i\in\operatorname*{argmax}_l\big(\hat{\ell}_k(x)\big)(l)\\ 0 & \text{otherwise}\end{cases}.\tag{C.62}$$

Next, we need to show that $\big(\psi_1(x)\big)(j_1)=\langle r_{k',0}(x),\psi_1(x)\rangle\geq\langle\hat{f}_{k,0,0}(x),\psi_0(x)\rangle-T=\big(\psi_0(x)\big)(j_2)-T$. Assume otherwise, meaning that $\big(\psi_1(x)\big)(j_1)<\big(\psi_1(x)\big)(j_2)-T$. In

this case, we have

$$\begin{aligned}
\max \hat{\ell}_k(x) &\overset{(a)}{=} \big(\hat{\ell}_k(x)\big)(j_2) \\
&= \big(\ell_k(x)\big)(j_2) + \big(\hat{\psi}_0(x) - \psi_0(x)\big)(j_2) - k\big(\hat{\psi}_1(x) - \psi_1(x)\big)(j_2) \\
&= \big(\ell_{k'}(x)\big)(j_2) - (k-k')\big(\psi_1(x)\big)(j_2) + \big(\hat{\psi}_0(x) - \psi_0(x)\big)(j_2) \\
&\quad - k\big(\hat{\psi}_1(x) - \psi_1(x)\big)(j_2) \\
&\overset{(b)}{\leq} \big(\ell_{k'}(x)\big)(j_2) - (k-k')\big(\psi_1(x)\big)(j_2) + (\delta_0 + K\delta_1) \\
&\overset{(c)}{<} \big(\ell_{k'}(x)\big)(j_2) - (k-k')\big(\psi_1(x)\big)(j_1) - (k-k')T + (\delta_0 + K\delta_1) \\
&\overset{(d)}{\leq} \big(\ell_{k'}(x)\big)(j_1) - (k-k')\big(\psi_1(x)\big)(j_1) - (k-k')T + (\delta_0 + K\delta_1) \\
&\overset{(e)}{\leq} \big(\ell_{k'}(x)\big)(j_1) - (k-k')\big(\psi_1(x)\big)(j_1) - 2\frac{\delta_0 + K\delta_1}{T}T + (\delta_0 + K\delta_1) \\
&= \big(\ell_{k'}(x)\big)(j_1) - (k-k')\big(\psi_1(x)\big)(j_1) - (\delta_0 + K\delta_1) \\
&= \big(\ell_k(x)\big)(j_1) - (\delta_0 + K\delta_1) \\
&= \big(\hat{\ell}_k(x)\big)(j_1) - (\delta_0 + K\delta_1) - \big(\hat{\psi}_0(x) - \psi_0(x)\big)(j_1) + k\big(\hat{\psi}_1(x) - \psi_1(x)\big)(j_1) \\
&\overset{(f)}{\leq} \big(\hat{\ell}_k(x)\big)(j_1) - (\delta_0 + K\delta_1) + (\delta_0 + K\delta_1) = \big(\hat{\ell}_k(x)\big)(j_1),
\end{aligned}$$

which is a contradiction. Note that $(a)$ holds because of definition of $j_2$ and ((C.62)), $(b)$ holds due to approximation assumptions $\|\hat{\psi}_0 - \psi_0\|_\infty \leq \delta_0$ and $\|\hat{\psi}_1 - \psi_1\|_\infty \leq \delta_1$, $(c)$ holds because of the assumption $\big(\psi_1(x)\big)(j_1) < \big(\psi_1(x)\big)(j_2) - T$, $(d)$ is followed by the definition of $j_1$ on maximizing $\ell_{k'}(x)$, and $(e)$ holds because $k \geq k' + \frac{2(\delta_0 + K\delta_1)}{T}$, and $(f)$ is followed by approximation assumptions. □

In order to prove this theorem, we define a measure of distance between two rules $f_1, f_2 \in \Delta_d^{\mathbb{R}}$ as

$$D_k(f_1, f_2) := \mathbb{E}\big[\langle f_1(x) - f_2(x), \psi_0(x) - k\psi_1(x)\rangle\big]. \tag{C.63}$$

Using this measure of distance, the difference of objectives between two rules $f_1$ and $f_2$ can be written as

$$\begin{aligned}
\mathbb{E}\big[\langle f_1(x), \psi_0(x)\rangle\big] - \mathbb{E}\big[\langle f_2(x), \psi_0(x)\rangle\big] =& D_{k^*}(f_1, f_2) \\
&+ k^*\Big(\mathbb{E}\big[\langle f_1(x), \psi_1(x)\rangle\big] - \mathbb{E}\big[\langle f_2(x), \psi_1(x)\rangle\big]\Big).
\end{aligned} \tag{C.64}$$

Therefore, if two rules achieve similar constraints, and if $D_k(f_1, f_2)$ is small enough, we can prove that the two rules achieve similar objectives too, since $k$ is bounded above by

$K$.

In fact, if we let $f_1(x) = f^*_{k,p}(x)$ and $f_2(x) := \hat{f}_{\hat{k},\hat{p}}$, where $k$ and $p$ are optimal solutions as in Theorem 7, then due to this optimality, and because $\mathbb{E}\big[\langle \hat{f}_{\hat{k},\hat{p}}, \psi_1(x)\rangle\big] \leq \delta$ with probability at least $1-\varepsilon$ as shown in Theorem 8, then LHS of ((C.64)) is positive with at least the same probability. In this proof, we show that how large is that term, and therefore, we show that how sub-optimal is $\hat{f}_{\hat{k},\hat{p}}$ in terms of the objective.

To that end, we first bound the difference between constraints. This bound can be achieved similar to the proof of Theorem 8. In fact, there we showed that if the empirical constraint $\mathbb{E}_{S^n}\big[\langle \hat{f}_{\hat{k},\hat{p}}, \psi_1(x)\rangle\big] \leq \delta - d_n(\pi)$, then using ((C.60)) the true expectation is bounded as $\mathbb{E}_{\mu}\big[\langle \hat{f}_{\hat{k},\hat{p}}, \psi_1(x)\rangle\big] \leq \delta$ with probability at least $1-\pi$. However, ((C.60)) is symmetric in empirical and true constraint, i.e., if we show that $\mathbb{E}_{S^n}\big[\langle \hat{f}_{\hat{k},\hat{p}}, \psi_1(x)\rangle\big] \geq \delta - d_n(\pi)$, then we have $\mathbb{E}_{\mu}\big[\langle \hat{f}_{\hat{k},\hat{p}}, \psi_1(x)\rangle\big] \geq \delta - 2d_n(\pi)$ with probability at least $1-\pi$.

To show $\mathbb{E}_{S^n}\big[\langle \hat{f}_{\hat{k},\hat{p}}, \psi_1(x)\rangle\big] \geq \delta - d_n(\pi)$, we follow three steps, (i) because $\delta$ is $(\varepsilon_l, \varepsilon_u)$-interior point of the set of constraints, i.e., $(\delta-\varepsilon_l, \delta+\varepsilon_u)$ is a subset of all plausible constraints, then $\delta - d_n(\pi)$ is $(\varepsilon_l - d_n(\pi), \varepsilon_u + d_n(\pi))$-interior point. Now, using Lemma 18 and by setting $\varepsilon' = \min\{\varepsilon_l - d_n(\pi), \varepsilon_u + d_n(\pi)\}$ we can show that $\delta - d_n(\pi)$ is $\varepsilon'/2$-interior point of the empirical constraints with probability at least $1-2e^{-\frac{n\varepsilon'^2}{4}}$, (ii) using the first step and assuming $d_n(\pi) \leq \varepsilon_l/2$ we conclude that $\delta - d_n(\pi)$ is $d_n(\pi)/2$-interior point of the empirical constraints with the aforementioned probability, (iii) because of Lemma 19, we conclude that for $\varepsilon = d_n(\pi)/2$, and with probability at least $1-2e^{-\frac{n\varepsilon'^2}{4}}$ there exists $k \in \mathbb{R}$ and $p \in [0,1]$ such that $\mathbb{E}_{S^n}\big[\langle \hat{f}_{k,p}(x), \hat{\psi}_1(x)\rangle\big] = \delta - d_n(\pi) + \frac{d_n(\pi)/2-\varepsilon}{2} = \delta - d_n(\pi)$. As a result of the above discussion we conclude that with probability at least $1-\pi-2e^{-\frac{n\varepsilon'^2}{4}}$ there exists $k$ and $p$ such that $\delta \geq \mathbb{E}\big[\langle \hat{f}_{k,p}(x), \psi_1(x)\rangle\big] \geq \delta - 2d_n(\pi)$. Now, since we know that $\mathbb{E}\big[\langle f_1(x), \psi_1(x)\rangle\big] = \mathbb{E}\big[\langle f^*_{k,p}(x), \psi_1(x)\rangle\big] = \delta$, then we have

$$0 \leq \mathbb{E}\big[\langle f_1(x), \psi_1(x)\rangle\big] - \mathbb{E}\big[\langle f_2(x), \psi_1(x)\rangle\big] \leq 2d_l(\pi), \tag{C.65}$$

with probability at least $1-\pi-2e^{-\frac{n\varepsilon'^2}{4}}$.

The above discussion together with ((C.64)) and the assumption of boundedness of $k$ shows that the difference of objectives is bounded with a high probability, if we bound $D_k(f_1, f_2)$. However, before we proceed with bounding that term, we should derive a relationship between $\hat{k}$ and $k^*$ for the reasons that we see in proving boundedness of $D_k(f_1, f_2)$.

We have already shown that there exists $\hat{p} \in [0,1]$ such that $\delta \geq \mathbb{E}\big[\langle \hat{f}_{\hat{k},\hat{p}}, \psi_1(x)\rangle\big] \geq \delta - 2d_l(\pi)$. Here, Lemma 20 shows that for $k' = k - \frac{2(\delta_0 + K\delta_1)}{T}$ we have $\mathbb{E}\big[\langle f^*_{k',0}, \psi_1(x)\rangle\big] \geq \delta - 2d_l(\pi) - T$ with probability at least $1-\pi-2e^{-\frac{n\varepsilon'^2}{4}}$. Moreover, using symmetry in Lemma 20 and for $k'' = k + \frac{2(\delta_0 + K\delta_1)}{T}$ we have $\mathbb{E}\big[\langle f^*_{k'',0}(x) - \hat{f}_k(x), \hat{\psi}_1(x)\rangle\big] \leq T$. Now, since $\|\psi_1(x) - \hat{\psi}_1(x)\|_\infty \leq \delta_0$, using Hölder's inequality we conclude that $\mathbb{E}\big[\langle f^*_{k'',0}(x) -$

$\hat{f}_k(x), \hat{\psi}_1(x)\rangle\big] \leq T + 2\delta_1$, and consequently $\mathbb{E}\big[\langle f^*_{k'',0}(x), \hat{\psi}_1(x)\rangle\big] \leq \delta + T + 2\delta_0$

Now that we have found a lower-bound on constraint of the rule $f^*_{k-q}(x)$ for $q = \frac{2(\delta_0+K\delta_1)}{T}$, then if we find an upper bound on the constraint of the rule $f^*_{k^*+e}(x)$ for an $e \in \mathbb{R}^+$, then we can use monotonicity of the constraint of $f^*_k$ in terms of $k$ and prove a relationship between $k$ and $k^*$. To that end, we use detection assumption with which we can show that

$$\mathbb{E}\big[\langle f^*_{k^*+\frac{1}{C}(2d_n(\pi)+T)^{1/\gamma}}, \psi_1(x)\rangle\big] \leq \delta - 2d_n(\pi) - T,$$

where we assume that $d_n(\pi) \leq \frac{(C\Delta)^\gamma - T}{2}$. Now, using previous discussions conclude that

$$\mathbb{E}\big[\langle f^*_{k^*+\frac{1}{C}(2d_n(\pi)+T)^{1/\gamma}}, \psi_1(x)\rangle\big] \leq \mathbb{E}\big[\langle f^*_{k',0}, \psi_1(x)\rangle\big],$$

with probability at least $1 - \pi - 2e^{-\frac{n\varepsilon'^2}{4}}$. This together with the first part of Lemma 16 shows that $k' \leq k^* + \frac{1}{C}(2d_n(\pi) + T)^{1/\gamma}$, or equivalently $k \leq k^* + \frac{2(\delta_0+K\delta_1)}{T} + \frac{1}{C}(2d_n(\pi) + T)^{1/\gamma}$ with probability at least $1 - \pi - 2e^{-\frac{n\varepsilon'^2}{4}}$. Since we know that $\gamma$ is clamped above by 1, and using the inequality $(1+x)^a \leq 1 + ax$ for $a \geq 1$ we can substitute the above inequality with $k \leq k^* + \frac{2(\delta_0+K\delta_1)}{T} + \frac{\big(2d_n(\pi)\big)^{1/\gamma}}{C}\big(1 + \frac{T}{\gamma(2d_n(\pi))^{1/\gamma}}\big)$. Now optimizing over $T$ leads in $T = \sqrt{2\gamma C(\delta_0 + K\delta_1)}$, which concludes that $k \leq k^* + \Delta_u k$ with the aforementioned probability, where $\Delta_u k = \frac{\big(2d_n(\pi)\big)^{1/\gamma}}{C} + 2\sqrt{\frac{2(\delta_0+K\delta_1)}{\gamma C}}$, if we have $d_n(\pi) \leq \frac{(C\Delta)^\gamma - \sqrt{2\gamma C(\delta_0+K\delta_1)}}{2}$

Similarly, using sensitivity assumption, we have

$$\mathbb{E}\big[\langle f^*_{k^*+\frac{1}{C}(2\delta_1+T)^{1/\gamma}}(x), \psi_1(x)\rangle\big] \geq \delta + 2\delta_1 + T,$$

where $\frac{(2\delta_1+T)^{1/\gamma}}{C} \leq \Delta$. Next, using previous discussions conclude that

$$\mathbb{E}\big[\langle f^*_{k^*+\frac{1}{C}(2\delta_1+T)^{1/\gamma}}(x), \psi_1(x)\rangle\big] \geq \mathbb{E}\big[\langle f^*_{k'',0}(x), \psi_1(x)\rangle\big],$$

with the aforementioned probability. This, again, together with the first part of Lemma 16 shows that $k'' \geq k^* - \frac{1}{C}(2\delta_1 + T)^{1/\gamma}$, or equivalently $k \geq k^* - \frac{1}{C}(2\delta_1 + T)^{1/\gamma} - \frac{2(\delta_0+K\delta_1)}{T}$. Therefore, by setting $T = \sqrt{2\gamma C(\delta_0 + K\delta_1)}$ we conclude that $k \geq k^* - \Delta_l k$ where $\Delta_l k = \frac{(2\delta_1)^{1/\gamma}}{C} + 2\sqrt{\frac{2(\delta_0+K\delta_1)}{\gamma C}}$, and assuming $\frac{\big(2\delta_1 + \sqrt{2\gamma C(\delta_0+K\delta_1)}\big)^{1/\gamma}}{C} \leq \Delta$.

Next, we turn into bounding $D_{k^*}(f_1, f_2)$. To that end, we first note that

$$t_x(k^*) := \langle f^*_{k^*,p}(x), \psi_0(x) - k^*\psi_1(x)\rangle = \max_i \big(\psi_0(x) - k^*\psi_1(x)\big)(i), \tag{C.66}$$

for all $p \in [0,1]$. This is followed by the definition of $f^*_{k^*,p}(\cdot)$. Similarly, we can show that

$$\hat{t}_x(\hat{k}) := \langle \hat{f}_{\hat{k},p}(x), \hat{\psi}_0(x) - k^* \hat{\psi}_1(x) \rangle = \max_i \big(\hat{\psi}_0(x) - \hat{k}\hat{\psi}_1(x)\big)(i),$$

for all $p \in [0,1]$. Now, we can rewrite $D_{k^*}(f_1, f_2)$ as

$$\begin{aligned}
D_{k^*}(f_1,f_2) &= \mathbb{E}\big[\langle f^*_{k^*,p}(x) - \hat{f}_{\hat{k},\hat{p}}(x), \psi_0 - k^*\psi_1(x)\rangle\big] \\
&= \mathbb{E}[t_x(k^*)] - \mathbb{E}\big[\langle \hat{f}_{\hat{k},\hat{p}}(x), \psi_0 - k^*\psi_1(x)\rangle\big] \\
&= \mathbb{E}[t_x(k^*)] - \mathbb{E}\big[\langle \hat{f}_{\hat{k},\hat{p}}(x), \hat{\psi}_0 - k^*\hat{\psi}_1(x)\rangle\big] \\
&\qquad - \mathbb{E}\big[\langle \hat{f}_{\hat{k},\hat{p}}(x), (\psi_0(x) - \hat{\psi}_0(x)) - k^*(\psi_1(x) - \hat{\psi}_1(x)\rangle\big] \\
&\overset{(a)}{\leq} \mathbb{E}[t_x(k^*)] - \mathbb{E}\big[\langle \hat{f}_{\hat{k},\hat{p}}(x), \hat{\psi}_0 - k^*\hat{\psi}_1(x)\rangle\big] + \delta_0 + K\delta_1 \\
&= \mathbb{E}[t_x(k^*)] - \mathbb{E}[\hat{t}_x(\hat{k})] + (k^* - k)\mathbb{E}\big[\langle \hat{f}_{\hat{k},\hat{p}}(x), \hat{\psi}_0(x)\rangle\big] + \delta_0 + K\delta_1 \\
&\overset{(b)}{\leq} \mathbb{E}[t_x(k^*)] - \mathbb{E}[\hat{t}_x(\hat{k})] + |k^* - \hat{k}| + \delta_0 + K\delta_1,
\end{aligned} \tag{C.67}$$

where $(a)$ and $(b)$ hold due to Hölder's inequality.

Next, we show Lipschitzness of $t(k)$ using its structure. In fact, due to its definition, $t(k)$ is the maximum of a set of lines with $\{t_i = \big(\psi_0(x)\big)(i) - k\big(\psi_1(x)\big)(i)\}_{i=1}^{n+1}$ in terms of $k$ with slope $m_i = -\big(\psi_1(x)\big)(i)$ and $y$-intercept of $b_i = \big(\psi_0(x)\big)(i)$. Therefore, such piecewise-linear function has a Lipschitz factor equal to the maximum slope of the lines, which in here is equal to $\max_i m_i = \max_i \Big|\big(\psi_1(x)\big)(i)\Big| \leq 1$. Therefore, $t(k)$ is a 1-Lipschitz function. Therefore, using ((C.67)) we can bound $D_{k^*}(f_1, f_2)$ as

$$\begin{aligned}
D_{k^*}(f_1,f_2) &\leq \mathbb{E}[t_x(\hat{k}) - \hat{t}_x(\hat{k})] + 2|k^* - \hat{k}| + \delta_0 + K\delta_1 \\
&= \mathbb{E}[\max_i \big(\psi_0(x) - \hat{k}\psi_1(x)\big)(i) - \max_i \big(\hat{\psi}_0(x) - \hat{k}\hat{\psi}_1(x)\big)(i) + 2|k^* - \hat{k}| + \delta_0 + K\delta_1 \\
&\overset{(a)}{\leq} 2|k^* - \hat{k}| + 2(\delta_0 + K\delta_1),
\end{aligned}$$

where $(a)$ holds because each component of $\big(\psi_0(x) - \hat{k}\psi_1(x)\big)$ and $\big(\hat{\psi}_0(x) - \hat{k}\hat{\psi}_1(x)\big)$ is bounded by $\delta_0 + K\delta_1$, and because the maximum operator is a norm, and therefore satisfies sub-additivity. Finally, since we have bounded $\Delta \leq k^* - \hat{k} \leq \Delta_l$ with probability at least $1 - \pi - 2e^{-n\varepsilon'^2/4}$, then we have

$$\begin{aligned}
D_k(f_1,f_2) &\leq 2\max\{\Delta, \Delta_l\} + 2(\delta_0 + K\delta_1) \\
&= 2\frac{\big(2\max\{d_n(\pi), \delta_1\}\big)^{1/\gamma}}{C} + 4\sqrt{\frac{2(\delta_0 + K\delta_1)}{\gamma C}} + 2(\delta_0 + K\delta_1),
\end{aligned}$$

with such probability. This, together with ((C.64)) and ((C.65)) shows that

$$\mathbb{E}\big[\langle f_1(x),\psi_0(x)\rangle\big]-\mathbb{E}\big[\langle f_2(x),\psi_0(x)\rangle\big] \leq 2\frac{\big(2\max\{d_n(\pi),\delta_1\}\big)^{1/\gamma}}{C}+4\sqrt{\frac{2(\delta_0+K\delta_1)}{\gamma C}} + 2(\delta_0+K\delta_1)+2Kd_n(\pi),$$

which completes the proof.

# C.12 Proof of Theorem 16

To prove this theorem, we first prove the following auxiliary lemma

**Lemma 21.** *For $\alpha,\varepsilon\geq 0$, the following holds*

$$\min_{r_i\geq 0,\sum_{i=1}^n r_i\leq\alpha}\sum_{i=1}^n r_i d_i-\min_{r_i\geq 0,\sum_{i=1}^n r_i\leq\alpha+\varepsilon}\sum_{i=1}^n r_i d_i\leq\varepsilon\cdot\max_{i\in[1:n]}|d_i|$$

*Proof of lemma.* We know that for every positive vector $\mathbf{r}$ with $\sum_{i=1}^n r_i\leq\alpha+\varepsilon$, we could rewrite that as a sum of two vectors $\mathbf{r}=\mathbf{r}'+\mathbf{r}''$ for which

$$\sum_{i=1}^n r_i'\leq\alpha,$$

and

$$\sum_{i=1}^n r_i''\leq\varepsilon.$$

As a result, we can rewrite $\min_{r_i\geq 0,\sum_{i=1}^n r_i\leq\alpha+\varepsilon}\sum_{i=1}^n r_i d_i$ as

$$\begin{aligned}\min_{r_i\geq 0,\sum_{i=1}^n r_i\leq\alpha+\varepsilon}\sum_{i=1}^n r_i d_i &\geq \min_{r_i'\geq 0,\sum_{i=1}^n r_i'\leq\alpha}\min_{r_i''\geq 0,\sum_{i=1}^n r_i''\leq\varepsilon}\sum_{i=1}^n (r_i'+r_i'')\cdot d_i\\ &=\min_{r_i'\geq 0,\sum_{i=1}^n r_i'\leq\alpha} r_i' d_i+\min_{r_i''\geq 0,\sum_{i=1}^n r_i''\leq\varepsilon}\sum_{i=1}^n r_i'' d_i.\end{aligned}$$

Hence, we have that

$$\begin{aligned}\min_{r_i\geq 0,\sum_{i=1}^n r_i\leq\alpha+\varepsilon}\sum_{i=1}^n r_i d_i-\min_{r_i'\geq 0,\sum_{i=1}^n r_i'\leq\alpha} r_i' d_i &\geq -\Big|\min_{r_i''\geq 0,\sum_{i=1}^n r_i''\leq\varepsilon}\sum_{i=1}^n r_i'' d_i\Big|\\ &\overset{(a)}{\geq}-\sum_{i=1}^n r_i''\cdot\max_{i\in[1:n]}|d_i|\geq-\varepsilon\cdot\max_{i\in[1:n]}|d_i|,\end{aligned}$$

where $(a)$ holds using Hölder's inequality. □

Next, we know that the optimal deterministic deferral policy should satisfy

$$\begin{aligned}
&\min_{r(x_i)\in\{0,1\},\frac{1}{n}\sum_i r(x_i)\le b} \frac{1}{n}\sum_i r(x_i)\ell_H(x_i,y_i,m_i) + \big(1-r(x_i)\big)\cdot \ell_{AI}(x_i,y_i)\\
&= \frac{1}{n}\sum_i \ell_{AI}(x_i,y_i) + \min_{r(x_i)\in\{0,1\},\frac{1}{n}\sum_{i=1}^n r(x_i)\le b} \frac{1}{n}\sum_{i=1}^n r(x_i)\big(\ell_H(x_i,y_i,m_i)-\ell_{AI}(x_i,y_i)\big)\\
&\overset{(a)}{=} \frac{1}{n}\sum_i \ell_{AI}(x_i,y_i) + \underbrace{\min_{r(x_i)\in\{0,1\},\sum_{i=1}^n r(x_i)\le \lfloor bn\rfloor} \frac{1}{n}\sum_{i=1}^n r(x_i)\big(\ell_H(x_i,y_i,m_i)-\ell_{AI}(x_i,y_i)\big)}_{B},
\end{aligned}$$

where $(a)$ holds because $r(x_i)\in\{0,1\}$ and therefore $\sum r(x_i)\le bn$ if and only if $\sum r(x_i)\le \lfloor bn\rfloor$. Now, we turn to examining $B$. To that end, we first consider the following optimization problem:

$$\min_{r(x_i)\in[0,1],\sum_{i=1}^n r(x_i)\le\lfloor bn\rfloor} \frac{1}{n}\sum_{i=1}^n r(x_i)\big(\ell_H(x_i,y_i,m_i)-\ell_{AI}(x_i,y_i)\big). \tag{C.68}$$

For a minimizer $\mathbf{r}^*$ of the above problem, we could form $\hat{\mathbf{r}}$ as

$$\hat{r}_i = \begin{cases} r_i^*(x_i) & \ell_H(x_i,y_i,m_i)-\ell_{AI}(x_i,y_i)\le 0\\ 0 & \ell_H(x_i,y_i,m_i)-\ell_{AI}(x_i,y_i)>0\end{cases}.$$

One can see that $\hat{\mathbf{r}}(x)$ is also a minimizer of the above problem. Hence, without loss of generality, we assume that there is an optimal deferral policy that has only non-zero value when $(x,y,m)\in A=\{(x,y,m)\in\mathcal{D}:\}$. Furthermore, we know that since $\hat{r}(x_i)\le 1$, then $\sum_i \hat{r}(x_i)\le \min\{\lfloor nb\rfloor,|A|\}$. We argue that this inequality does not hold in a strict form, i.e., we have $\sum_i \hat{r}(x_i)=\min\{\lfloor nb\rfloor,|A|\}$. The reason is that otherwise one can find $r'(x)\in[0,1]^{\mathcal{X}}$ such that $\sum_{(x_i,y_i,m_i)\in A}\hat{f}(x_i)+r'(x_i)=\min\{nb,|A|\}$ and because of negativity of $\ell_H(x_i,y_i,m_i)-\ell_{AI}(x_i,y_i)$, we can strictly reduce the objective function that is a contradiction.

Next, we order $\ell_H(x_i,y_i,m_i)-\ell_{AI}(x_i,y_i)$ increasingly and we name them $d_j$. In fact, we define $k_j$ such that $d_j=\ell_H(x_{k_j},y_{k_j},m_{k_j})-\ell_{AI}(x_{k_j},y_{k_j})$ and that $d_1\le d_2\ldots\le d_{|A|}\le 0$. For the sake of simplicity, we further define $r_j:=r(x_{k_j})$. As a result, the optimization problem in ((C.68)) can be rewritten as

$$\min_{r_i\in[0,1],\sum_{i=1}^n r_i=\min\{\lfloor nb\rfloor,|A|\}} \sum_{i=1}^n r_i d_i.$$

Here, we show that the optimizer of the above problem is $r_i=\mathbb{1}_{i\le\min\{\lfloor nb\rfloor,|A|\}}$. To show

that, we consider $r'_i \in [0,1]$ such that $\sum_{i=1}^{n} r'_i = \min\{\lfloor nb \rfloor, |A|\}$. Then, we have

$$\sum_{i=1}^{n} \mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}} d_i - \sum_{i=1}^{n} r'_i d_i = \sum_{i:\, \mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}} - r'_i < 0} (\mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}} - r'_i)\cdot d_i + \sum_{i:\, \mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}} - r'_i > 0} (\mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}} - r'_i)\cdot d_i. \quad \text{(C.69)}$$

Now, since we know that $\sum_i \mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}} = \sum_i r'_i$, we can define a parameter $Q$ as

$$Q := \sum_{i:\, \mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}} - r'_i > 0} \mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}} - r'_i = \sum_{i:\, \mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}} - r'_i < 0} r'_i - \mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}}. \quad \text{(C.70)}$$

Next, by defining $p_i := \frac{\mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}} - r'_i}{Q}$ for $i$s in which $\mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}} - r'_i > 0$ and $q_i := \frac{r'_i \mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}}}{Q}$ for $i$s in which $\mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}} - r'_i < 0$ and 0 otherwise, we conclude that $\{p_i\}_i$ and $\{q_i\}_i$ are probability mass functions. Hence, using ((C.69)) and ((C.70)), we have

$$\sum_{i=1}^{n} \mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}} d_i - \sum_{i=1}^{n} r'_i d_i = Q\Big( \sum_{i=1}^{\min\{\lfloor nb\rfloor,|A|\}} p_i d_i - \sum_{i=\min\{\lfloor nb\rfloor,|A|\}+1}^{n} q_i d_i \Big).$$

The above identity contains the difference of two expected value over random variables that one is always smaller than the other. As a result, we show that

$$\sum_{i=1}^{n} \mathbb{1}_{i\leq \min\{\lfloor nb\rfloor,|A|\}} d_i - \sum_{i=1}^{n} r'_i d_i \leq 0,$$

which completes the proof.

# Appendix D

# Appendix IV

## D.1 Proof of Example 4

One can find the error probabilities for $h$ and $M$ as

$$\Pr(h \neq Y | Y = y) = p_1,$$

and

$$\Pr(M \neq Y | Y = y) = p_2.$$

Next, we can write the deferral loss as

$$\begin{aligned} L_{\text{def}}(h,r) &= \frac{1}{2}\mathbb{E}\big[r(\mathbf{X})\big]\big[c\Pr(M \neq Y|Y=1) + (1-c)\Pr(M \neq Y|Y=0)\big] \\ &\quad + \frac{1}{2}\mathbb{E}\big[1-r(\mathbf{X})\big]\big[c\Pr(h \neq Y|Y=1) + (1-c)\Pr(h \neq Y|Y=0)\big] \\ &= \frac{p_2\mathbb{E}\big[r(\mathbf{X})\big]}{2} + \frac{p_1\mathbb{E}\big[1-r(\mathbf{X})\big]}{2}. \end{aligned}$$

This shows that the optimal deferral is

$$r^*(x) = \begin{cases} 1 & p_2 \geq p_1 \\ 0 & p_2 < p_1 \end{cases},$$

which does not depend on $c$. Moreover, the optimal deferral loss in this case is

$$\min_{r(\cdot)} L_{\text{def}}(h,r) = \frac{\min\{p_1,p_2\}}{2}.$$

Next, we assume that $f(h,M) = h \vee M$ is the binary OR operation of the two predictions. For analyzing the probability of error $\Pr\big(Y \neq f(h,M)\big)$ we consider the following two cases:

1. $Y = 0$ and $f(h,M) = 1$: In this case, either $h$ or $M$ should be 1, which is equivalent

to either $n_1$ or $n_2$ taking the value of 1. Such probability is equal to

$$\Pr\big(f(h,M) \neq Y | Y = 0\big) = 1 - (1-p_1)(1-p_2).$$

2. $Y = 1$ and $f(h,M) = 0$: In this case, both $n_1$ and $n_2$ must be equal to 1, which has the probability

$$\Pr\big(f(h,M) \neq Y \,|\, Y = 1\big) = p_1 p_2.$$

As a result, the loss of this binary operation as a predictor is

$$L_{\text{OR}}(h) = \frac{1}{2}\big[c p_1 p_2 + (1-c)\big[1 - (1-p_1)(1-p_2)\big]\big].$$

Similarly, one can show that for AND function, we have

$$L_{\text{AND}}(h) = \frac{1}{2}\big[(1-c) p_1 p_2 + c\big[1 - (1-p_1)(1-p_2)\big]\big].$$

It is easy to show that for the case of $c \in [0, \frac{\min\{p_1,p_2\} - p_1 p_2}{p_1 + p_2 - 2p_1 p_2}, 1]$ we have

$$L_{\text{OR}}(h) < L_{\text{def}}(h, r^*) < L_{\text{AND}}(h),$$

for $c \in \big(\frac{\max\{p_1,p_2\} - p_1 p_2}{p_1 + p_2 - p_1 p_2}, 1\big]$ we have

$$L_{\text{AND}}(h) < L_{\text{def}}(h, r^*) < L_{\text{OR}}(h),$$

and for the rest $c \in \big[\frac{\min\{p_1,p_2\} - p_1 p_2}{p_1 + p_2 - 2p_1 p_2}, \frac{\max\{p_1,p_2\} - p_1 p_2}{p_1 + p_2 - 2p_1 p_2}\big]$ we have

$$L_{\text{AND}}(h), L_{\text{OR}}(h) \geq L_{\text{def}}(h, r^*).$$

As a result, we observe that the deferral is optimal if and only if the false negative and the false positive losses are almost equal.

In Figure D.1 we show the average loss of the optimal operation (from 16 possible binary operations on $\{0,1\} \times \{0,1\}$ over training data) and the learn-to-defer method over $N_{\text{iter}} = 100$ seeds, when $p_1$ and $p_2$ are drawn uniformly randomly from $[0,1]$, and the parameters $c$ and $P(Y=1)$ are drawn uniformly randomly from $[0,1]$. One could observe that the deferral method is sub-optimal.

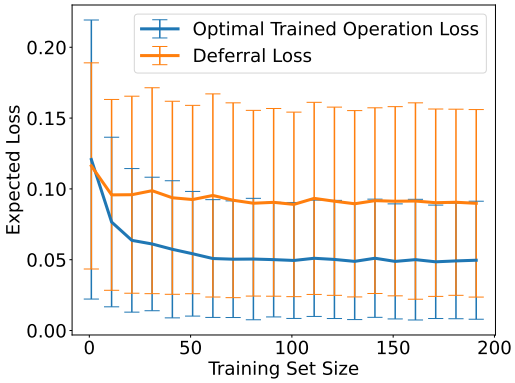



Figure D.1: Average loss of trained operation on two binary decisions of human $M$ and classifier $h$, where the true label is drawn from $Bern(q)$. Here, $M$ and $h$ are two noisy versions of $Y$ with probability of error $p_1$ and $p_2$. The average is over 100 uniformly random drawn probability values $q, p_1$, and $p_2$.

## D.2 Relationship Between Entropic Lower-Bounds

We know that

$$\begin{aligned}
&H(Y|M,r(X)=1)\Pr(r(X)=1)+H(Y|h(X),r(X)=1)\Pr(r(X)=1)\\
&\geq H(Y|M,X,r(X)=1)\Pr(r(X)=1)+H(Y|h(X),X,M,r(X)=1)\Pr(r(X)=1)\\
&\overset{(a)}{=} H(Y|M,X,r(X)=1)\Pr(r(X)=1)+H(Y|X,M,r(X)=1)\Pr(r(X)=1)\\
&\overset{(b)}{=} H(Y|M,X,r(X))\\
&\overset{(c)}{=} H(Y|M,X)
\end{aligned}$$

where $(a)$ holds because

$$\begin{aligned}
I(Y;h(X)|X,M,r(X)=1) &= H(h(X)|X,M,r(X)=1)-H(h(X)|Y,X,M,r(X)=1)\\
&= 0-0=0,
\end{aligned}$$

and $(b)$ is due to the definition of conditional entropy, and $(c)$ holds because

$$I(Y;r(X)|M,X)=H(r(X)|M,X)-H(r(X)|M,X,Y)=0-0=0.$$

## D.3 Proof of Theorem 10

We prove this theorem for the two cases of deferral and DaF system:

- **Deferral System**: We know that the prediction of the deferral system is equal to

$$\hat{Y}_{\text{def}} = r(x) \cdot M + \big(1 - r(x)\big) \cdot h(x). \tag{D.1}$$

Now, to find the lower-bound on the error of the above prediction, we can lower-bound the optimal prediction $\hat{Y}_{\text{opt}}$ of $Y$ based on $\hat{Y}_{\text{def}}$. More formally, optimality of $\hat{Y}_{opt}$ concludes that

$$\Pr(\hat{Y}_{\text{def}} \neq Y) \geq \Pr(\hat{Y}_{\text{opt}} \neq Y). \tag{D.2}$$

Next, we use Fano's inequality (see e.g. Theorem 1 of Scarlett and Cevher (2019)) to lower-bound the RHS of the above inequality as

$$H\Big(\Pr(\hat{Y}_{\text{opt}} \neq Y)\Big) + \Pr(\hat{Y}_{\text{opt}} \neq Y) \log(|\mathcal{Y}| - 1) \geq H(Y|\hat{Y}_{\text{def}}). \tag{D.3}$$

We follow the proof by finding a lower-bound on $H(Y|\hat{Y}_{\text{def}})$. Using the positivity of mutual information, we have

$$\begin{aligned} H(Y|\hat{Y}_{\text{def}}) &= H\big(Y|r(X), \hat{Y}_{\text{def}}\big) + I\big(Y; r(X)|\hat{Y}_{\text{def}}\big) \\ &\geq H\big(Y|r(X), \hat{Y}_{\text{def}}\big) \\ &= \Pr\big(r(X) = 0\big) H\big(Y|\hat{Y}_{\text{def}}, r(X) = 0\big) + \Pr\big(r(X) = 1\big) H\big(Y|\hat{Y}_{\text{def}}, r(X) = 1\big) \\ &= \Pr\big(r(X) = 0\big) H\big(Y|h(X), r(X) = 0\big) + \Pr\big(r(X) = 1\big) H\big(Y|M, r(X) = 1\big), \end{aligned} \tag{D.4}$$

where the last equality is followed by the definition of $\hat{Y}_{\text{def}}$ in ((D.1)). Using ((D.2)), ((D.3)), and ((D.4)), one can prove the theorem in the case of deferral system.

- **DaF system**: In this case, we use Fano's inequality directly. In fact, we have

$$H\Big(\Pr(\hat{Y}_{\text{DaF}} \neq Y)\Big) + \Pr(\hat{Y}_{\text{DaF}} \neq Y) \log(|\mathcal{Y}| - 1) \geq H(Y|\hat{Y}_{\text{DaF}}) \geq H(Y|M, T),$$

where the last inequality is followed by data processing inequality, and that $\hat{Y}$, $(M, X)$, and $Y$ form a Markov chain.

## D.4 Proof of Theorem 11

A Bayes optimal deferral $r^*$, classifier $h^*$, and meta-learner $g^*$ should satisfy

$$(r^*,h^*,g^*) \in \underset{r\in\{0,1,2\}^{\mathcal{X}},h\in\mathcal{Y}^{\mathcal{X}},g\in\mathcal{Y}^{\mathcal{X}\times\mathcal{Y}}}{\operatorname{argmin}} \mathbb{E}\Big[\mathbb{I}_{r(X)=0}\ell\big(h(X),Y\big)+\mathbb{I}_{r(X)=1}\ell_{\text{def}}(M,Y) + \mathbb{I}_{r(X)=2}\ell_{\text{fus}}\big(g(X,M),Y\big)\Big].$$

We start by minimizing over $h$. We know that

$$\underset{h\in\mathcal{Y}^{\mathcal{X}}}{\operatorname{argmin}}\,\mathbb{E}_{X,Y,M}\left[\mathbb{I}_{r(X)=0}\ell\big(h(X),Y\big)+\mathbb{I}_{r(X)=1}\ell_{\text{def}}(M,Y)+\mathbb{I}_{r(X)=2}\ell_{\text{fus}}\big(g(X,M),Y\big)\right] \tag{D.5}$$

$$\overset{(a)}{=} \underset{h\in\mathcal{Y}^{\mathcal{X}}}{\operatorname{argmin}}\,\mathbb{E}_{X,Y}\left[\mathbb{I}_{r(X)=0}\ell\big(h(X),Y\big)\right]$$

$$= \underset{h\in\mathcal{Y}^{\mathcal{X}}}{\operatorname{argmin}}\,\mathbb{E}_{X}\Big[\mathbb{I}_{r(X)=0}\mathbb{E}_{Y|X}\left[\ell\big(h(X),Y\big)\right]\Big], \tag{D.6}$$

where $(a)$ holds because the second and the last term of (D.5) is not a function of $h$. As a result, because $\mathbb{I}_{r(X)=0}\geq 0$, to minimize (D.6), one can choose

$$h^*(x) \in \underset{h\in\mathcal{Y}}{\operatorname{argmin}}\,\mathbb{E}_{Y|X=x}\big[\ell(h,y)\big] \subseteq \underset{h\in\mathcal{Y}}{\operatorname{argmin}}\,\mathbb{I}_{r(x)=0}\mathbb{E}_{Y|X=x}\big[\ell(h,y)\big],$$

which proves ((5.8))

Similarly, one can show that ((5.9)) is the optimal meta-learner.

Next, we find an optimal deferral as

$$r^* \in \underset{r\in\{0,1,2\}^{\mathcal{X}}}{\operatorname{argmin}}\,\mathbb{E}_{X,Y,M}\Big[\mathbb{I}_{r(X)=0}\ell\big(h^*(X),Y\big)+\mathbb{I}_{r(X)=1}\ell_{\text{def}}\big(M,Y\big)+\mathbb{I}_{r(X)=2}\ell_{\text{fus}}\big(g^*(X,M),Y\big)\Big]$$

$$= \underset{r\in\{0,1,2\}^{\mathcal{X}}}{\operatorname{argmin}}\,\mathbb{E}_{X}\Big[\mathbb{I}_{r(X)=0}\mathbb{E}_{M|X}\Big[\mathbb{E}_{Y|X,M}\big[\ell(h^*(X),Y)+\mathbb{I}_{r(X)=1}\mathbb{E}_{Y,M|X}\big[\ell_{\text{def}}\big(M,Y\big)\big] + \mathbb{I}_{r(X)=2}\mathbb{E}_{M|X}\big[\mathbb{E}_{Y|X,M}\big[\ell_{\text{fus}}(g^*(x,m),Y)\big]\big]\Big]$$

$$= \underset{r\in\{0,1,2\}^{\mathcal{X}}}{\operatorname{argmin}}\,\mathbb{E}_{X}\Big[\mathbb{I}_{r(X)=0}\min_{h\in\mathcal{Y}}\mathbb{E}_{Y|X}\big[\ell(h,Y)\big]+\mathbb{I}_{r(X)=1}\mathbb{E}_{Y,M|X}\big[\ell_{\text{def}}\big(M,Y\big)\big] + \mathbb{I}_{r(X)=2}\mathbb{E}_{M|X}\big[\min_{g\in\mathcal{Y}}\mathbb{E}_{Y|X,M}\big[\ell_{\text{fus}}(g,y)\big]\big]\Big], \tag{D.7}$$

where $(a)$ holds using ((5.8)) and ((5.9)) and because $\mathbb{E}_{Y|X=x}\big[\ell(h^*(X),Y)\big]$ is not a function of $r(x)$ or $M$. Finally, one can see that a minimizer of ((D.7)) is as stated in the

theorem.

## D.5 Complementarity of Fusion

Using Theorem 11 one can show that if the losses are defined as

$$\ell_{\text{def}}(g,y) = \mathbb{I}_{g\neq y} + c_{\text{def}},$$

for a fixed cost of deferral and

$$\ell(h,y) = \mathbb{I}_{h\neq y},$$

then the Bayes optimal is obtained using the deferral set

$$\mathcal{T} = \Big\{x : \sum_{i=1}^{|\mathcal{Y}|} \Pr(M=i|X=x) \times \max_{y\in\mathcal{Y}} \Pr(Y=y|M=i,X=x) \geq \max_{y\in\mathcal{Y}} \Pr(Y=y|X=x) + c_{\text{def}}\Big\},$$

and $h^*(x) \in \operatorname*{argmax}_{y\in\mathcal{Y}} \Pr(Y=y|X=x)$ and $g^*(x,m) \in \operatorname*{argmax}_{y\in\mathcal{Y}} \Pr(Y=y|M=i,X=x)$.

Furthermore, in case of $c_{\text{def}} = 0$, we can show that $\mathcal{T} = \mathcal{X}$. The reason is that since

$$\sum_{i=1}^{|\mathcal{Y}|} \Pr(M=i|X=x)\Pr(Y=y|M=i,X=x) = \Pr(Y=y|X=x),$$

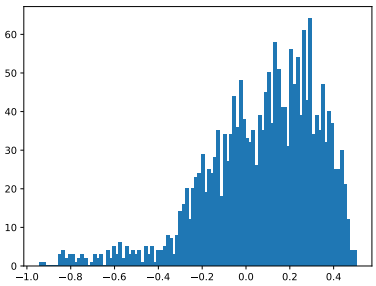
because $\|\vartheta\|_\infty = \max_i \vartheta_i$ for a vector $\vartheta$ is a convex function, and using Jensen's inequality Jensen (1906) (i.e. for a convex function $f$ we have $\mathbb{E}\big[f(X)\big] \geq f\Big(\mathbb{E}\big[X\big]\Big)$ ) we see that

$$\sum_{i=1}^{|\mathcal{Y}|} \Pr(M=i|X=x) \cdot \max_{y\in\mathcal{Y}} \Pr(Y=y|M=i,X=x) \geq \max_{y\in\mathcal{Y}} \Pr(Y=y|X=x). \tag{D.8}$$

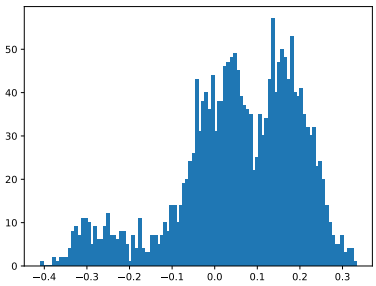
Hence, if there is no cost for deferral, a Bayes optimal DaF system always requests for human advice and use the meta-learner to take the final decision accordingly.

Furthermore, since

$$\max_{y\in\mathcal{Y}} \Pr(Y=y|M=i,X=x) \geq \Pr(Y=i|M=i,X=x), \tag{D.9}$$




(a) Difference of confidence of fusion and classifier




(b) Difference of confidence of fusion and the expert

Figure D.2: Comparing confidences of options in DaF method on test set of CIFAR-10H dataset. The confidences are obtained using CDaF method.

one can conclude that

$$\sum_{i=1}^{|\mathcal{Y}|} \Pr(M=i|X=x) \cdot \max_{y\in\mathcal{Y}} \Pr(Y=y|M=i,X=x)$$

$$\geq \sum_{i=1}^{|\mathcal{Y}|} \Pr(M=i|X=x)\Pr(Y=i|M=i,X=x) \tag{D.10}$$

$$= \Pr(Y=M|X=x), \tag{D.11}$$

and therefore the confidence of the fusion is more than the confidence of human alone.

However, this theoretical result holds only when we observed enough number of data that we can accurately approximate above probabilities. In our experiments, however, we noticed that just training the meta-learner on the instances that human is accurate still needs a certain amount of sample complexity that is not needed if we just defer to the human.

Figure D.2 demonstrates the difference between confidences of fusion, expert, and classifier. We observe that although the confidence of fusion is larger than the expert and classifier in the majority of times, the bounded training size concluded in cases that the classifier confidence and expert confidence are larger than fusion confidence. This phenomenon is particularly notable for the expert confidence.

## D.6 Sufficient Statistics in DaF Methods

**Theorem 17.** *Let us assume that the human decision $M$ and the feature $X$ are independent given the true label. Further, let us assume that $T(X)$ is a sufficient statistics of $X$*

*for estimating M, i.e. we have*

$$I(Y;T(X)) = I(Y;X),$$

*where $I(\cdot,\cdot)$ is the mutual information. Then, $(T(X),M)$ is a sufficient statistics of $(X,M)$ for estimating $Y$.*

*Proof.* First, we upper-bound the mutual information $I(X,M;Y)$ as

$$\begin{aligned}
I(X,M;Y) &\overset{(a)}{=} I(X;Y) + I(M;Y|X) \\
&\overset{(b)}{=} I(T(X);Y) + I(M;Y|X) = I(T(X);Y) + H(M|X) - H(M|Y,X) \\
&\overset{(c)}{\leq} I(T(X);Y) + H(M|T(X)) - H(M|Y,X) \\
&\overset{(d)}{=} I(T(X);Y) + H(M|T(X)) - H(M|Y) \\
&\overset{(e)}{=} I(T(X);Y) + H(M|T(X)) - H(M|Y,T(X)) = I(T(X);Y) + I(M;Y|T(X)) \\
&\overset{(f)}{=} I(M,T(X);Y),
\end{aligned} \tag{D.12}$$

where $(a)$ holds because of chain rule, $(b)$ is because of statistical sufficiency of $T(X)$, $(c)$ holds because marginalization increases the entropy, $(d)$ and $(e)$ hold because

$$0 \leq I(M;T(X)|Y) \leq I(M;X|Y) = 0,$$

which results in $H(M|Y) = H(M|T(X),Y) = H(M|X,Y)$. Finally, $(f)$ is due to chain rule.

Next, we lower-bound $I(X,M;Y)$ as

$$\begin{aligned}
I(X,M;Y) &\overset{(a)}{=} I(M;Y) + I(X;Y|M) = I(M;Y) + H(Y|M) - H(Y|X,M) \\
&\overset{(b)}{\geq} I(M;Y) + H(Y|M) - H(Y|T(X),M) \\
&= I(M;Y) + I(T(X);Y|M) \overset{(c)}{=} I(M,T(X);Y),
\end{aligned} \tag{D.13}$$

where $(a)$ holds using chain rule, $(b)$ holds because conditioning reduces entropy, and $(c)$ is another use of chain rule.

As a result of ((D.12)) and ((D.13)) we show that $I(X,M;Y) = I(T(X),M;Y)$ that completes the sufficient part of the proof.

□

## D.7 Reduction to Confusion Matrix Learning

**Proposition 6.** *Let $X$, $Y$, and $M$ be the corresponding random variables for features, labels, and human decisions. Further, assume that $h(x)$ is the set of confidences of the classifier. The equivalence of the DaF method and that of Kerrigan* et al. *(2021) occurs if and only if the graphical model in Figure 5.3 is the underlying model of the variables.*

*Proof.* First, on one hand we know that the DaF method and that of Kerrigan *et al.* (2021) are equivalent if $\Pr(Y|X,M) = \Pr(Y|h(X),M)$, or because $h(X)$ is a function of $X$ if we have

$$\Pr\big(Y|X,M,h(X)\big) = \Pr(Y|h(X),M), \tag{D.14}$$

that is equivalent to the independence of $Y$ and $X$ given $h(X)$ and $M$, or

$$I(Y;X|h(X),M) = 0. \tag{D.15}$$

The equivalence between this condition and the graphical model in Figure 5.3 can be shown in the following way. Firstly, if we ignore the variable $M$, then the above identity is assigning a Markov chain between $X$, $h(X)$, and $Y$. Now, $M$ can have connection with all three variables. However, since $h(X)$ is a function of $X$, then conditioned on $X$, $M$ should not share any information with $h(X)$. As a result, there does need to be any connection between $X$ and $M$. Conversely, it is easy to show that in this graphical model, conditioning on $M$ and $h(X)$ removes all connections betewen $X$ and $Y$ that proves the claim. □

Note that the above conditional independence results in a conditional independence of human decision and classifier decision given the true label that is assumed in a variety of methods for combining predictions Kerrigan *et al.* (2021); Clemen and Winkler (1999); Jacobs (1995).

**Remark 2.** *Using Fisher factorization Halmos and Savage (1949), we know that $\eta(X)$ is a sufficient statistics of $X$ for estimating $Y$, if the conditional density function of $X$ given $Y$ can be factorized as*

$$f_{X|Y}(X = x|Y = y) = g_1(x)g_2(y,\eta(x)),$$

*for two functions $g_1, g_2$. By setting $\eta(x) = \big[Pr(Y = 1\,|\,X = x), \ldots, Pr(Y = |\mathcal{Y}|\,|X = x)\big]$ we have $g_1(x) = f_X(x)$ and $g_2(y,\vartheta) = \vartheta_y$ for a vector $\vartheta = \big(\vartheta_1, \ldots, \vartheta_{|\mathcal{Y}|}\big)$. As a result, the set of posterior probabilities are sufficient statistics of $X$ for estimating $Y$. Hence, under the conditions of Theorem 17, the pair $(\eta(X),M)$ is sufficient statistics of $(X,M)$ for estimating $Y$. More formally, we have*

$$Pr(Y|X = x, M = m) = Pr(Y|\eta(X) = \eta(x), M = m).$$

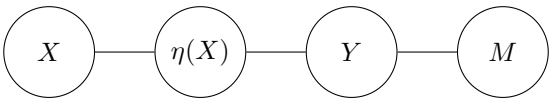



Figure D.3: To substitute features $x$ with $\eta(x) = \big[\Pr(Y=1|X=x),\ldots,\Pr(Y=|\mathcal{Y}|\,|X=x)\big]$ in our analysis, we assume that $X$ and $M$ are conditionally independent given the true label $Y$. We further use the fact that $\eta(X)$ is sufficient statistics of $X$ for estimating $Y$.

*Therefore, in this case the deferral set $\mathcal{T}$ is obtained as*

$$\mathcal{T} = \Big\{x : \sum_{i=1}^{|\mathcal{Y}|} Pr(M=i|X=x)\cdot \min_{g\in\mathcal{Y}} \mathbb{E}_{Y|M=i,\eta(X)=\eta(x)}\big[\ell_{\text{def}}(g,Y)\big] \leq \min_{h\in\mathcal{Y}} \mathbb{E}_{Y|X=x}\big[\ell(h,Y)\big]\Big\},$$

*and the optimal meta-learner is a function of $(\eta(x),m)$ and is obtained as*

$$g^*(\eta(x),m) \in \underset{g\in\mathcal{Y}}{argmin}\, \mathbb{E}_{Y|M=m,\eta(X)=\eta(x)}\big[\ell_{\text{def}}(g,Y)\big].$$

## D.8 An Example of Suboptimality of Kerrigan *et al.* (2021)

Assume that the true label $Y$ for all features is distributed as $Bern(0.8)$. As a result, the best classifier $g(x)$ is

$$g(x) = \underset{y}{\text{argmax}}\, \Pr(Y=y|X=x) = 1, \tag{D.16}$$

which has the accuracy of 80%, i.e. $h(x) = [0.20 \;\; 0.80]$ for all $x$. Furthermore, assume that the human observe the exogenous variable $V$ that is correlated to $X$ and based on which the human can perfectly recover $Y$ for $X \geq 0$, while for $X < 0$ we have the joint probability of human decision and the true label is as following:

$$\begin{aligned}
&\Pr(Y=1,M=1) = 0.6,\\
&\Pr(Y=1,M=0) = 0.2,\\
&\Pr(Y=0,M=1) = 0.10,\\
&\Pr(Y=0,M=0) = 0.10,
\end{aligned} \tag{D.17}$$

which means that the probability of correctness of human is 70%.

Furthermore using the above we have

$$\Pr(M=0)=\frac{1}{2}\Pr(M=0|X\geq 0)+\frac{1}{2}\Pr(M=0|X<0)=\frac{1}{2}(0.30+0.80)=0.55 \tag{D.18}$$

By assuming that $X$ has the same probability of being positive or negative, we have

$$\Pr(Y|M,h(x))=\Pr(Y|M)=\frac{1}{2}\Pr(Y|M,X\geq 0)+\frac{1}{2}\Pr(Y|M,X<0). \tag{D.19}$$

As a result, the method of Kerrigan *et al.* (2021) leads to the accuracy of

$$\begin{aligned}\Pr(\hat{Y}_{\text{ker}}=Y)&=\sum\Pr(M=m)\max_y\big[\frac{1}{2}\Pr(Y=y|M=m,X\geq 0)\\&\qquad+\frac{1}{2}\Pr(Y=y|M=m,X<0)\big] &\text{(D.20)}\\&=\Pr(M=0)\frac{1}{2}(0.333+1)+\Pr(M=1)\frac{1}{2}(1+0.857)\\&=0.65*0.666+0.35*0.928=0.757 &\text{(D.21)}\end{aligned}$$

However, if we use the information within $X$ and using the DaF method we have

$$\begin{aligned}\Pr(\hat{Y}_{\text{DaF}}=Y)&=\sum\frac{1}{2}\Pr(M=m|X\geq 0)\max_y\Pr(Y=y|M=m,X\geq 0)\\&\qquad+\frac{1}{2}\Pr(M=m|X<0)\max_y\Pr(Y=y|M=m,X<0) &\text{(D.22)}\\&=\frac{1}{2}0.8+\frac{1}{2}=0.9. &\text{(D.23)}\end{aligned}$$

Furthermore, an optimal deferral system is

$$r(x)=\begin{cases}1 & x\geq 0\\ 0 & x<0\end{cases} \tag{D.24}$$

The reason is that for $x\geq 0$ we have $\Pr(Y=g(X)|X=x)=0.8$ while $\Pr(Y=M|X=x)=1$. However, for $x<0$ we have $\Pr(Y=g(X)|X=x)=0.8$ while $\Pr(Y=M|X=x)=0.7$. As a result, one can see similar to DaF, the accuracy of the deferral system is

$$\Pr(\hat{Y}_{\text{def}}=Y)=\frac{1}{2}0.8+\frac{1}{2}=0.9. \tag{D.25}$$

This shows that there are cases that our DaF method, and even deferral over-perform the method in Kerrigan *et al.* (2021). The reason is that the latter method simply ignore the variation of human distribution on different instances and substitute that with an average

distribution.

## D.9 Consistency of DaF methods

**Theorem 18.** *Let $\phi(\cdot)$ be a strictly proper binary surrogate function. Further, let $f_1(\theta_1;x) : \mathcal{X} \to \mathbb{R}^{|\mathcal{Y}|}$, $f_2(\theta_2;[x,m]) : \mathcal{X}\times\mathcal{M} \to \mathbb{R}^{|\mathcal{Y}|}$, and $f_3(\theta_3;x) : \mathcal{X} \to \mathbb{R}^{|\mathcal{M}|}$ be three functions that are trained to approximate the label y and human decision m by minimizing*

$$\mathcal{L}(f_1,f_2,f_3,x,y,m) = \underbrace{\phi(f_1^y(x)) + \sum_{y\neq y'} \phi(-f_1^{y'}(x))}_{\mathcal{L}_1(f_1(x),y)} \tag{D.26}$$

$$+ \underbrace{\phi(f_2^y([x,m])) + \sum_{y\neq y'} \phi(-f_2^{y'}([x,m]))}_{\mathcal{L}_2(f_2([x,m]),y)} \tag{D.27}$$

$$+ \underbrace{\phi(f_3^m(x)) + \sum_{m\neq m'} \phi(-f_3^{m'}(x))}_{\mathcal{L}_3(f_3(x),m)}. \tag{D.28}$$

*Then, the above loss is consistent, i.e., we can find Bayes optimal DaF solution using the minimizer of such loss.*

*Proof.* Assume that $\mathcal{F}_1, \mathcal{F}_2$, and $\mathcal{F}_3$ are hypothesis class of all possible functions $f_1, f_2$, and $f_3$. It is easy to show that the solution of risk minimization

$$\operatorname*{argmin}_{f_i\in\mathcal{F}_i} \mathbb{E}\big[\mathcal{L}(f_1,f_2,f_3,X,Y,M)\big] \tag{D.29}$$

contains the solution of

$$f_1(x) = \operatorname*{argmin}_{f\in\mathbb{R}^{|\mathcal{Y}|}} \mathbb{E}_{Y|X=x}[\mathcal{L}_1(f,Y)], \tag{D.30}$$

$$f_2([x,m]) = \operatorname*{argmin}_{f\in\mathbb{R}^{|\mathcal{Y}|}} \mathbb{E}_{Y|X=x,M=m}[\mathcal{L}_2(f,Y)], \tag{D.31}$$

and

$$f_3(x) = \operatorname*{argmin}_{f\in\mathbb{R}^{|\mathcal{M}|}} \mathbb{E}_{Y|X=x}[\mathcal{L}_3(f,M)]. \tag{D.32}$$

This holds because (i) minimum of the sum of two functions are larger than sum of minimum of each function, and (ii) minimum of expected variable of a function is larger than the expected value of minimum of that function (i.e., inverse Jensen's inequality for

the concave function min). Furthermore, a strictly proper composite surrogate assures that the link function $\lambda(\eta)$ defined as

$$\lambda(\eta) = \operatorname*{argmin}_{g\in\mathbb{R}} \eta\phi(g) + (1-\eta)\phi(-g), \tag{D.33}$$

is a strictly increasing function (see Ramaswamy *et al.* (2014)). Therefore, the expected risk of a One-vs-All loss function for a parameter $t \in \mathbb{R}^K$ on a variable $U \in [1:K]$

$$\mathbb{E}_U[\phi(t^U) + \sum_{U'\neq U} \phi(-t^{U'})] = \sum_{U=u} \Pr(U=u)\phi(t^u) + (1-\Pr(U=u))\phi(-t^u), \tag{D.34}$$

has a minimizer that contains the information regarding $\Pr(U=u)$. This holds because the minimizer $t^*$ of the above loss on parameter $t$ is equal to the minimizer of each summand and therefore we have

$$\lambda^{-1}(t^{*u}) = \Pr(U=u). \tag{D.35}$$

This and the previous discussion shows that

$$\lambda^{-1}(f_1^{*y}(x)) = \Pr(Y=y|X=x), \tag{D.36}$$

$$\lambda^{-1}(f_2^{*y}([x,m])) = \Pr(Y=y|X=x,M=m), \tag{D.37}$$

and

$$\lambda^{-1}(f_3^{*m}(x)) = \Pr(M=m|X=x). \tag{D.38}$$

Using these parameters and the definition of extended confidence in Theorem 11 we can find the Bayes optimal DaF solution. □

**Theorem 19.** *For score function $\phi(x) = \log(1+e^{-x})$ and under the circumstance that the meta-learner makes error with probability at most 0.5, LDaF loss is consistent.*

*Proof.* We know that the One-vs-All loss function based on such score function for $t \in \mathbb{R}^{|\mathcal{U}|}$ for estimating $u \in \mathcal{U}$ is obtained as

$$\mathcal{L}(t,u) = \log(1+e^{-t^u}) + \sum_{u'\neq u} \log(1+e^{t^{u'}}), \tag{D.39}$$

and the expected value is calculated as

$$\mathbb{E}_U[\mathcal{L}(t,U)] = \sum_{i=1}^{|\mathcal{U}|} \Pr(U=i)\log(1+e^{-t^i}) + (1-\Pr(U=i))\log(1+e^{t^i}) \tag{D.40}$$

$$= -\sum_{i=1}^{|\mathcal{U}|} \Pr(U=i)\log(\frac{e^{t^i}}{1+e^{t^i}}) + (1-\Pr(U=i))\log(\frac{1}{1+e^{t^i}}) \tag{D.41}$$

$$= \sum_{i=1}^{|\mathcal{U}|} D_{KL}(p_i\|q_i) + h_B(p_i), \tag{D.42}$$

where $p_i := \Pr(U=i)$, $q_i := \frac{e^{t^i}}{1+e^{t^i}}$, and $h_B(p)$ is Shannon binary entropy of a random variable with distribution $Bern(p)$.

Next, we define the LDaF loss function as

$$\mathcal{L}_{LDaF}(f_1,f_2,f_3,x,y,m) = \underbrace{\phi(f_1^y(x)) + \sum_{y\neq y'} \phi(-f_1^{y'}(x))}_{\mathcal{L}_1(f_1(x),y)} \tag{D.43}$$

$$+ \underbrace{\phi(f_2^y([x,m])) + \sum_{y\neq y'} \phi(-f_2^{y'}([x,m]))}_{\mathcal{L}_2(f_2([x,m]),y)} \tag{D.44}$$

$$+ \underbrace{\phi(f_3^e(x)) + \sum_{e\neq e'} \phi(-f_3^{e'}(x))}_{\mathcal{L}_3(f_3(x),e)}, \tag{D.45}$$

where $e := \begin{cases} 1 & \operatorname{argmax} f_2([x,m]) = y \\ 0 & o.w. \end{cases}$.

Similar to SDaF, one can show that the result of the expected risk minimization leads to

$$f_1(x) = \operatorname*{argmin}_f \mathbb{E}_{Y|X=x}[\mathcal{L}_1(f),Y)], \tag{D.46}$$

and

$$f_2(x), f_3(x) = \underset{f_i \in \mathcal{F}_i}{\operatorname{argmin}} \mathbb{E}_{Y,M|X=x}\big[\mathcal{L}_2(f_2([x,M]),Y) + \mathcal{L}_3(f_3(x),E)\big] \tag{D.47}$$

$$\overset{(a)}{=} \underset{f_i \in \mathcal{F}_i}{\operatorname{argmin}} \mathbb{E}_{M|X=x}\big[\sum_{i=1}^{|\mathcal{Y}|} D_{KL}(p'_i(x,M)\|f_2^i(x,M)\big] + D_{KL}(p''(x)\|f_3^1(x)) + D_{KL}(1-p''(x)\|f_3^2(x)) + 2h_B(p''(x))\big], \tag{D.48}$$

where $(a)$ is followed by ((D.42)), $p'_i(x,m) = \Pr(Y = i|X = x, M = m)$, and $p''_i(x) = \Pr(Y = \operatorname{argmax} f_2([x,M])|X = x)$. We can observe that since $D_{KL}(p\|q) \geq 0$ and is zero iff. $p = q$, then the minimizer of the first three terms occur when $p'_i = f_2^i$, $f_3^1 = p''_i$, and $f_3^2 = 1 - p''_i$. Next, we know that $p''_i(x)$ is maximized when $\operatorname{argmax} f_2([x,M]|X = x) = \underset{y}{\operatorname{argmax}} \Pr(Y = y|X = x, M = m)$. Because of this and since $h_B(p)$ is decreasing for $p \in [1/2, 1]$ we conclude that the last term is minimized where $f_2^i = p'_i$. As a result, since the first three terms of above loss and the latter have the same minimizer, then the overall minimizer of the sum of these terms coincides with the mentioned minimizers.

The above discussion proves that the minimizer of $f_1^y(x)$ approximates $\Pr(Y = y|X = x)$, the minimizer of $f_2^y([x,m])$ approximates $\Pr(Y = y|X = x, M = m)$, and the minimizer of $f_3^y([x])$ approximates $p''_i(x)$. Therefore, we can find the extended confidences as in Theorem 11 using these probabilities and obtain Bayes optimal solution. □

## D.10 Defer to Multiple Experts

In this section, we derive the optimal deferral rule to multiple individuals. More specifically, we discuss the two cases for which

- we need to defer the decision to the most suitable candidate, and
- we need to form a committee among a group of individuals

In the following, we express the optimal deferral rule in the first case.

**Theorem 20.** *Assume that for each feature variable X and the true label Y, there a set of human decisions $\{M_i\}_{i=1}^K$. Additionally, it is assumed that deferring to each of these humans incurs an initial cost of $c_i$ for human i, along with a shared decision-based cost of $\ell\big(Y, g_i(M_i,X)\big)$. Moreover, the loss of classifier h is defined as $\ell(h(X),Y)$. In this scenario, the optimal classifier, deferral rule, and meta-learner for each human is as following*

$$h^*(x) = \underset{\hat{y}}{\operatorname{argmin}} \mathbb{E}\big[\ell(\hat{y},Y)|X = x\big], \quad \textit{(classifier)} \tag{D.49}$$

$$r^*(x) = \begin{cases} \underset{i}{argmin}\, \ell_i(x) & \min \ell_i(x) \leq \min_{\hat{y}} \mathbb{E}\big[\ell(\hat{y}, Y)|X = x\big] \\ 0 & o.w. \end{cases}, \quad \textit{(deferral)} \tag{D.50}$$

*and*

$$g_i(m,x) = \underset{g}{argmin}\, \mathbb{E}_{Y|M_i=m,X=x}\big[\ell\big(Y,g\big)\big], \quad \textit{(meta-learner)} \tag{D.51}$$

*where $\ell_i(x)$ is defined as*

$$\ell_i(x) = c_i(x) + \sum_{t=1}^{Y} Pr(M_i = t|X = x) \min_{\hat{y}} \mathbb{E}_{Y|M_i=t,X=x}\big[\ell(Y,\hat{y})\big]. \tag{D.52}$$

*Proof.* We start the proof by writing the expected loss as

$$L_{\text{hyb}}\big(h,r,\{g_i\}_{i=1}^K\big) = \mathbb{E}_X\left[\ell_{\text{hyb}}\big(h(x),r(x),\{g_i(x)\}_{i=1}^K,X\big)\right], \tag{D.53}$$

where $\ell_{\text{hyb}}$ is defined as

$$\ell_{\text{hyb}}\big(\boldsymbol{H},\boldsymbol{R},\{\boldsymbol{G}^i\}_{i=1}^K,x\big) = \mathbb{E}_{Y,\{M_i\}_{i=1}^K|X=x}\left[\mathbb{I}_{\boldsymbol{R}=0}\ell\big(\boldsymbol{H},Y\big) + \sum_{i=1}^{K}\mathbb{I}_{\boldsymbol{R}=i}\big[c_i(X) + \ell\big(Y,\boldsymbol{G}^i_{M_i}\big)\big]\right], \tag{D.54}$$

where $\boldsymbol{H}$, $\boldsymbol{R} \in \{1,\ldots,|\mathcal{Y}|\}$ and $\boldsymbol{G}^i \in \{1,\ldots,|\mathcal{Y}|\}^{\{1,\ldots,|\mathcal{Y}|\}}$. Therefore, the optimal set of classifier, deferral, and meta-learner is the solution

$$h^*,r^*,\{g_i^*\} = \underset{h,r,\{g_i\}}{\text{argmin}}\, L_{\text{hyb}}\big(h,r,\{g_i\}_{i=1}^K\big). \tag{D.55}$$

We note that the optimization is over all measurable functions. As a result, since the optimization of the value of functions on a feature $x_1$ does not change the feasible set for optimizing the functions on feature $x_2$, then we can reduce the optimization to the instance-based optimization problem

$$h(x),r(x),\{g_i(x)\}_{i=1}^K = \underset{\boldsymbol{H},\boldsymbol{R},\{\boldsymbol{G}^i\}_{i=1}^K}{\text{argmin}}\ \ell_{\text{hyb}}\big(\boldsymbol{H},\boldsymbol{R},\{\boldsymbol{G}^i\}_{i=1}^K,x\big). \tag{D.56}$$

Next, by fixing $\boldsymbol{R}$ and minimizing the terms of $\ell_{hyb}$ we have

$$\min_{\boldsymbol{H},\{\boldsymbol{G}^i\}_{i=1}^K} \ell_{\text{hyb}}\big(\boldsymbol{H},\boldsymbol{R},\{\boldsymbol{G}^i\}_{i=1}^K\big) = \min_{\boldsymbol{H}} \mathbb{E}_{Y|X=x}\Big[\mathbb{I}_{\boldsymbol{R}=0}\ell\big(\boldsymbol{H},Y\big)\Big] + \sum_{i=1}^K \min_{\boldsymbol{G}^i} \mathbb{E}_{Y,M_i|X=x}\big[\mathbb{I}_{\boldsymbol{R}=i}\big[c_i(X)+\ell\big(Y,\boldsymbol{G}^i_{M_i}\big)\big]\big], \quad \text{(D.57)}$$

because the choice of $\boldsymbol{G}^i$ and $\boldsymbol{H}$ is independent.

Finally, since $\mathbb{I}_{\boldsymbol{R}=i}$ can only take positive value, we have

$$\boldsymbol{H}^* = \operatorname*{argmin}_{\boldsymbol{H}} \mathbb{E}_{Y|X=x}\Big[\ell\big(\boldsymbol{H},Y\big)\Big] \in \operatorname*{argmin}_{\boldsymbol{H}} \mathbb{E}_{Y|X=x}\Big[\mathbb{I}_{\boldsymbol{R}=0}\ell\big(\boldsymbol{H},Y\big)\Big]. \quad \text{(D.58)}$$

Further, we observe that if

$$\boldsymbol{G}^{*i}_m = \operatorname*{argmin}_t \mathbb{E}_{Y|M_i=m,X=x}\big[\ell\big(Y,t\big)\big], \quad \text{(D.59)}$$

then we have

$$\boldsymbol{G}^{*i} \in \operatorname*{argmin}_{\boldsymbol{G}^i} \mathbb{E}_{M_i|X=x}\mathbb{E}_{Y|M_i,X=x}\big[\ell\big(Y,\boldsymbol{G}^i_{M_i}\big)\big] \quad \text{(D.60)}$$

$$= \operatorname*{argmin}_{\boldsymbol{G}^i} \mathbb{E}_{Y,M_i|X=x}\big[c_i(X)+\ell\big(Y,\boldsymbol{G}^i\big)\big] \in \operatorname*{argmin}_{\boldsymbol{G}^i} \mathbb{E}_{Y,M_i|X=x}\big[\mathbb{I}_{\boldsymbol{R}=i}\big[c_i(X)+\ell\big(Y,\boldsymbol{G}^i\big)\big]\big]. \quad \text{(D.61)}$$

As a result of ((D.56)), ((D.57)), , and ((D.58)) we conclude that ((D.63)) and ((D.65)) hold.

Next, to prove ((D.64)), we assume that $\boldsymbol{R}$ is a function of $x$ and as a result is independent of $Y$ and $M$ given $X$. In this case, we have

$$\boldsymbol{R}^* = \operatorname*{argmin}_{\boldsymbol{R}} \mathbb{I}_{\boldsymbol{R}=0}\mathbb{E}_{Y,M|X=x}\big[\ell\big(\boldsymbol{H}^{\star,Y}\big)\big] + \textstyle\sum_{i=1}^K \mathbb{I}_{\boldsymbol{R}=i}\mathbb{E}_{Y,M|X=x}\big[c_i(X)+\ell\big(Y,\boldsymbol{G}^{*i}\big)\big]\big]. \quad \text{(D.62)}$$

As a result, $\boldsymbol{R}$ get the value to choose the minimum term among the above terms. Finally, using the definition of optimal $\boldsymbol{H}^*$ and $\boldsymbol{G}^{*i}$ as in ((D.58)) and ((D.58)) we conclude ((D.64)). Note that via a very similar argument, we can show that if $\boldsymbol{R}$ is a random variable that is independent of $Y$ and $M$ given $X$, then because in the above we minimize a linear function in terms of $[\Pr(\boldsymbol{R}=i|X=x)]_{i=1}^{|\mathcal{Y}|}$, and because the arguments take values in a simplex, then the optimizer is taking place in a node of this simplex, that proves that the optimal deterministic solution is also an optimal probabilistic solution of $\boldsymbol{R}$.

Next, in case that we can choose a committee among the humans, we can treat the deferral like the previous part, but this time $r(x)$ takes $2^K+1$ values instead of $K+1$ choices, and in that case we have $2^K$ meta-learners too. As a result, the following

proposition is followed readily by the above theorem.

**Proposition 7.** *With the assumption of Theorem 20 and under the condition that we can defer to many experts, i.e. $r(x)$ takes $(K+1)$-dimensional binary vector, and we can combine the decision of all experts i for which $v_i = 1$ using a meta-learner $g_{\boldsymbol{v}}$, then the optimal classifier, meta-learners and deferral are as*

$$h^*(x) = \underset{\hat{y}}{argmin}\,\mathbb{E}\big[\ell(\hat{y},Y)|X=x\big], \quad \textit{(classifier)} \tag{D.63}$$

$$r^*(x) = \begin{cases} \underset{\boldsymbol{v}}{argmin}\,\ell_{\boldsymbol{v}}(x) & \min \ell_{\boldsymbol{v}}(x) \leq \min_{\hat{y}} \mathbb{E}\big[\ell(\hat{y},Y)|X=x\big] \\ 0 & o.w. \end{cases}, \quad \textit{(deferral)} \tag{D.64}$$

*and*

$$g_{\boldsymbol{v}}(\{m_i\}_{i=1}^{\|\boldsymbol{v}\|},x) = \underset{g}{argmin}\,\mathbb{E}_{Y|\{M_{v_i}=m_i\}_{i=1}^{\|\boldsymbol{v}\|},X=x}\big[\ell\big(Y,g\big)\big], \quad \textit{(meta-learner)} \tag{D.65}$$

*where $\ell_{\boldsymbol{v}}(x)$ is defined as*

$$\ell_{\boldsymbol{v}}(x) = \sum_{i:v_i=1} c_i(x) + \sum_{t_1=1}^{|\mathcal{Y}|} \dots \sum_{t_{\|\boldsymbol{v}\|}=1}^{|\mathcal{Y}|} Pr(M_{\boldsymbol{v}} = \boldsymbol{t}|X=x) \min_{\hat{y}} \mathbb{E}_{Y|\{M_{v_i}=t_i\}_{i=1}^{\|\boldsymbol{v}\|},X=x}\big[\ell(Y,\hat{y})\big]. \tag{D.66}$$

To avoid needing of training such large number of networks, we can assume that the deferral occurs to a committee of experts, the result of which is aggregated and then passed through a meta-learner. This, as a result, is equivalent of using Theorem 20 for which $M_i$ is replaced by a function $\text{Aggr}(\cup_{j\in\mathcal{S}} M_j)$ where $\mathcal{S}$ is a set of indices of all non-zero digits in binary representation of $i$. To be able to calculate the optimal solution as in 20, however, it is sufficient to approximate the three probability distribution $\Pr(Y=y|X=x)$, $\Pr(\text{Aggr}(\cup_{j\in\mathcal{S}} M_j)=t|X=x)$, and $\Pr(Y=y|X=x,\text{Aggr}(\cup_{j\in\mathcal{S}} M_j)=t)$. The first term can be readily approximated using a neural network with a consistent surrogate loss. Further, assuming conditional independence of the experts given each instance, the second term can be obtained as

$$\Pr(\text{Aggr}(\cup_{j\in\mathcal{S}} M_j)=t|X=x) = \sum_{m_i\in\mathcal{M}_i} \mathbb{I}_{\text{Aggr}(\cup_{j\in\mathcal{S}} m_j)=t} \prod_{j\in\mathcal{S}} \Pr(M_i=m_i|X=x), \tag{D.67}$$

that can be calculated if we have simulation of each expert, which is similar to what is discussed in a single-expert DaF setting.

Finally, the last probability can be obtained as

$$\Pr(Y=y|X=x,\text{Aggr}(\cup_{j\in\mathcal{S}}M_j)=t)=\frac{\Pr(Y=y,\text{Aggr}(\cup_{j\in\mathcal{S}}M_j)=t|X=x)}{\Pr(\text{Aggr}(\cup_{j\in\mathcal{S}}M_j)=t|X=x)} \tag{D.68}$$

$$=\frac{\sum_{m_i\in\mathcal{M}_i}\mathbb{I}_{\text{Aggr}(\cup_{j\in\mathcal{S}}M_j)=t}\Pr(Y=y,M_1^K=m_1^K|X=x)}{\Pr(\text{Aggr}(\cup_{j\in\mathcal{S}}M_j)=t|X=x)} \tag{D.69}$$

$$=\frac{\sum_{m_i\in\mathcal{M}_i}\mathbb{I}_{\text{Aggr}(\cup_{j\in\mathcal{S}}M_j)=t}\Pr(Y=y|M_1^K=m_1^K,X=x)\prod_{i=1}^K\Pr(M_i=m_i|X=x)}{\Pr(\text{Aggr}(\cup_{j\in\mathcal{S}}M_j)=t|X=x)}. \tag{D.70}$$

As a result, for obtaining the optimal committee, we further need to train the meta-learner to approximate $\Pr(Y=y|M_1^K=m_1^K,X=x)$. As result, with a classifier, $K$ expert simulator, and one meta-learner with a $K\times M$-dimensional input we can obtain the optimal solution for multi-expert setting.

□

## D.11 Comparison of DaF and Learn-to-Defer

In Table D.1, we showed 5 samples within Imagenet-16H dataset for which learn-to-defer method Mozannar and Sontag (2020b) identified incorrectly, while DaF method could correctly classify.

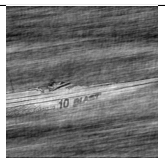
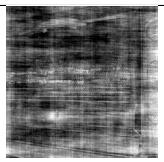
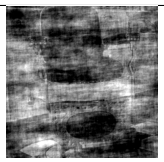
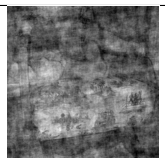
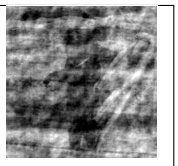
| Image |  |  |  |  |  |
|---|---|---|---|---|---|
| True Label | Boat | Oven | Chair | Chair | Bear |
| DaF | Boat | Oven | Chair | Chair | Bear |
| Learn-to-Defer | Airplane | Airplane | Oven | Elephant | Dog |
| Classifier | Boat | Oven | Chair | Chair | Bear |
| Human | Airplane | Airplane | Oven | Elephant | Dog |
| Meta-Learner | Boat | Oven | Chair | Chair | Bear |
| Defer and Fuse | Yes | Yes | Yes | Yes | Yes |

Table D.1: 5 Examples of correcting learn-to-defer method using DaF

## D.12 Calibration and CDaF

The surrogate loss that is used in LDaF and SDaF method is an OvA loss with $\phi = \log(1+e^{-x})$. Using this loss, as shown in proof of Theorem 19 is equivalent to the sum of binary cross entropy losses on $[\frac{e^{f_i(x)}}{1+e^{f_i(x)}}, \frac{1}{1+e^{f_i(x)}}]$ and the label $\mathbb{I}_{y=i}$ for all $i \in [1 : \texttt{output\_dim}]$, where $f_i(x)$ is the i-th output of network $f$. Now, although each output in the end of training approximates the probability of $\Pr(Y = i|X = x)$, since there is no constraint on joint output, the final $N$-dimensional output can get the value outside of the simplex $\Delta_N$. This, as a result, can lead to errors while finding approximation of expected values and confidences.

In fact, normalizing the outputs is not necessarily helpful too. To show why, assume that one of the outputs (w.l.o.g the last output) is not a good approximation of the true probability, i.e., the true probability is $p$ while the approximation has the bias $\text{TV}_1 = \delta$, and the rest of the probabilities are exact, the total variation of the output probabilities compared to the true probabilities in this case is equal to $\delta$. Now, by normalizing the outputs, we induce $\frac{(1-p)\delta}{1+\delta}$ shift for the other probabilities and convert the last output to $\frac{p+\delta}{1+\delta}$ that has the bias $\frac{\delta(1-p)}{1+\delta}$ with respect to $p$. As a result, the total variation measure in this case is $\text{TV}_2 = \frac{2\delta(1-p)}{1+\delta}$. We observe that $\text{TV}_2 - \text{TV}_1 = \frac{\delta^2+\delta(1-2p)}{1+\delta}$ that is greater than 0 for $p \in [0, 1/2]$ and $\delta \in \mathbb{R}^+$. This shows that in this case normalization effectively reduces the accuracy of output probabilities.

To address this issue, we can train networks for which the output corresponds to set of probabilities sequentially. In fact, in CDaF, in each epoch, we first train the classifier with $N$-dimensional output on label $y$, then we train the meta-learner on the same label, then we train 2-dimensional classifier on binary label $\mathbb{I}_{Y=M}$, and finally we train 2-dimensional classifier on binary label $\mathbb{I}_{Y=\texttt{meta-learner}}$. In all these training processes, since we use softmax layer in the output, we assure that the resulting output is within $\Delta_N$. This, however, cannot be done in OvA methods. In this sense, we argue that CDaF method is more calibrated than the OvA method, and we hope for less error based on this method.

To observe that effect, we trained SDaF on CIFAR-10H dataset and plotted the sum of its outputs in Figure D.4. One can see the shift and variance that such summation has around the value 1.

## D.13 Experiments

### D.13.1 Settings

In the experiment setting, we used 4 datasets that are listed in Table D.2. The expert labels, and size of training, validation, and test data, and batch size corresponding to each data is further listed there.

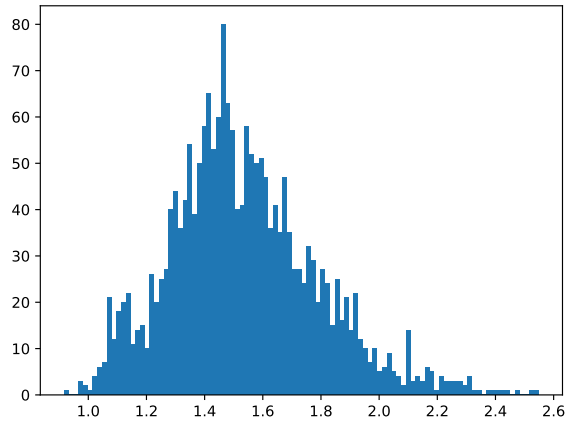



Figure D.4: Sum of output values of the classifier of CIFAR-10H that is trained by SDaF method

Furthermore, the architecture of all the neural networks corresponding to each dataset is listed in Table D.3. Finally, the way these networks are used for different methods are listed in Table D.4. All trainings are done by setting `num_epochs=150`, and using Adam optimizer and cosine annealing learning rate scheduler. All the experiments below are run with Nvidia V100 GPUs coupled with 4 CPUs and 100GBs of RAM. The code of these experiments are avaliable in `https://anonymous.4open.science/r/BeyondDefer-9073/`.

## D.13.2 Coverage Experiments

The accuracy of learn-to-defer methods with respect to the coverage (i.e., 1 - portion of human involvement) is plotted in Figure D.5. We observe that our methods overperform the baselines. Further, the AFE method does not obtain a better accuracy than a random choice between the classifier and meta-learner.

## D.13.3 Deferral Loss experiments

In this experiment, we change the cost that is induced by human involvement and observe the final loss that is induced in our methods compared to other learn-to-defer methods. The demonstration of overal loss in Figure D.6 shows that while our methods in regions of deferral cost for CIFAR-10K and ImageNet-16H overperforms other methods, it stays comparable to other methods otherwise. We suspect that the failure of SDaF and LDaF for $c \in [0, 0.5]$ and for CIFAR-10H dataset is related to uncalibratedness of the outputs as discussed in Appendix D.12. In that case, CDaF shows a clear improvement over those

| Dataset | Train size | Validation size | Test size | Batch size | Expert |
|---|---|---|---|---|---|
| CIFAR-10K Mozannar *et al.* (2023c) | 45k | 5k | 10k | 512 | True label for $K$ classes, and uniformly random otherwise |
| CIFAR-10H Peterson *et al.* (2019) | 7k | 1k | 2k | 1000 | Randomly drawn from the distribution of expert decisions on the instance |
| Imagenet-16H Steyvers *et al.* (2022) Noise Level=110 | 948 | 12 | 240 | 32 | |
| Hatespeech Davidson *et al.* (2017) | 17349 | 2478 | 4956 | 1000 | |

Table D.2: Information regarding the datasets used in experiment section

methods.

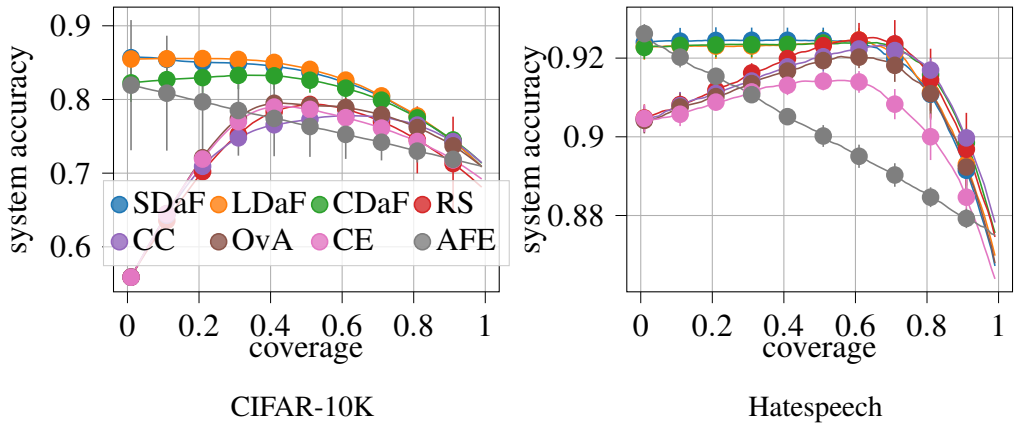
### D.13.4 Imbalanced Cost experiments

In this experiment, we introduce a loss matrix $A$ for which the loss of one-hot encoded prediction $\hat{y}$ of the label $y$ is equal to

$$L = \hat{y}^T A y. \tag{D.71}$$

We draw non-diagonal elements of $A$ uniformly randomly and set the diagonal values equal to 0. Under such setting the methods undergo the process of training as following:

- For LDaF method, the first num_classes outputs of the classifier and all outputs of the meta-learner are trained on one-hot encoded version of the true label. Furthermore, the output `num_classes+1` of the classifier are trained on the expert confidence $1 - m^T A y$ and the output `num_classes+2` of the classifier are trained on the meta-learner confidence $1 - \texttt{meta-learner}^T A y$. In test-time, the optimal decision based on the classifier or meta-learner is taken by minimizing $\hat{y} A p_o$ on values of $\hat{y}$ where $p_o$ is the approximated probability in the output of the classifier or meta-learner network.




CIFAR-10K Hatespeech

Figure D.5: The accuracy vs. coverage of learn-to-defer methods and on CIFAR-10K and Hatespeech, datasets. The bars indicate the standard deviation of the accuracy.

– For CDaF the classifier and meta-learner are trained and used as in LDaF. The confidences $1 - m^T Ay$ and $1 - \texttt{meta-learner}^T Ay$ are trained on two different networks.

– For CC, the classifier is trained as in LDaF and the confidence $1 - m^T Ay$ is trained on a different network. In test-time the confidence of the classifier is obtained by $\max_{\hat{y}} 1 - \hat{y}Ap_h$ where $p_h$ is the approximated probabilities in the output of the classifier.

– For LCE and OvA we perform training similar to CC with the different that the expert confidence is trained on the output `num_classes+1` of the classifier.

We observe in Figure D.7 that our methods outperform all other learn-to-defer methods. In fact, we can observe that CDaF method performs consistently better than other methods on all datasets.

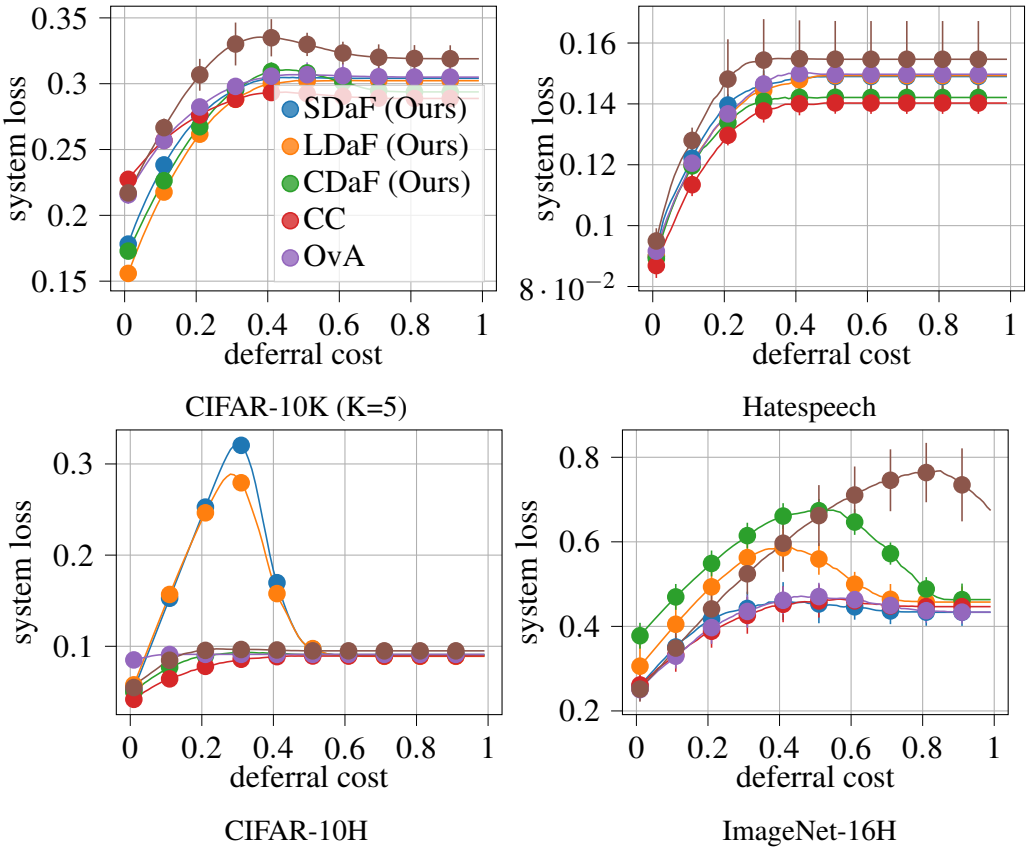



Figure D.6: Average loss vs. deferral cost on Hatespeech, CIFAR-10H, CIFAR-10K , and Imagnet dataset and for 10 random seeds, with bars being standard deviation.

| Dataset | Network1 | Network2 |
|---|---|---|
| CIFAR-10K `num_class=10` | A ReLu convolutional neural network with two convolutional layers with width 50 followed by three fully connected layers with output dimension of 100, 20, and 10 | Same as Network1 just with output dimension `num_class+1` |
| CIFAR-10H `num_class=10` | A WideResNet network that is pre-trained on CIFAR-10 dataset. It contains 25 convolutional layers with 3, 16, 64, 256, and 1024 channels followed by two fully connected layers with output dimension of 50 and 10. | |
| ImageNet-16H `num_class=16` | A DenseNet121 network that is pretrained on ImageNet dataset with output dimension 16. | |
| Hatespeech `num_class=3` | A linear classifier with output dimension 3. | |

| Dataset | Network3 | Network4 | Network5 |
|---|---|---|---|
| CIFAR-10K `num_class=10` | Same as Network1 just with output dimension `num_class+2` | Same as Network1 just with output dimension 2 | Concatenation of Network1 without the last fully connected layer with `num_class` extra dimensions followed by a fully connected layer with output dimension `num_class`. |
| CIFAR-10H num_class=10 | | | |
| ImageNet-16H num_class=16 | | | |
| Hatespeech num_class=3 | | | |

Table D.3: All networks that are used in learn-to-defer tasks based on different datasets.

| Method | Networks |
|---|---|
| LDaF (ours) | Augmented classifier: Network3, meta-learner: Network5 |
| SDaF (ours) | Augmented classifier: Network2, human simulator: Network1, meta-learner: Network5 |
| CDaF (ours) | Classifier: Network1, meta-learner: Network5, expert confidence predictor: Network4, meta-learner confidence predictor: Network4 |
| Active Feature Elicitation Natarajan *et al.* (2018) | Classifier: Network1, meta-learner: Network5 |
| Compare Confidences Raghu *et al.* (2019) | Classifier: Network1, expert confidence predictor: Network4 |
| Reallizable Surrogates Mozannar *et al.* (2023c) | Augmented classifier: Network2 |
| Cross Entropy Mozannar and Sontag (2020b) | Augmented classifier: Network2 |
| One-vs-All Verma and Nalisnick (2022b) | Augmented classifier: Network2 |

Table D.4: The type of networks that are used in each method

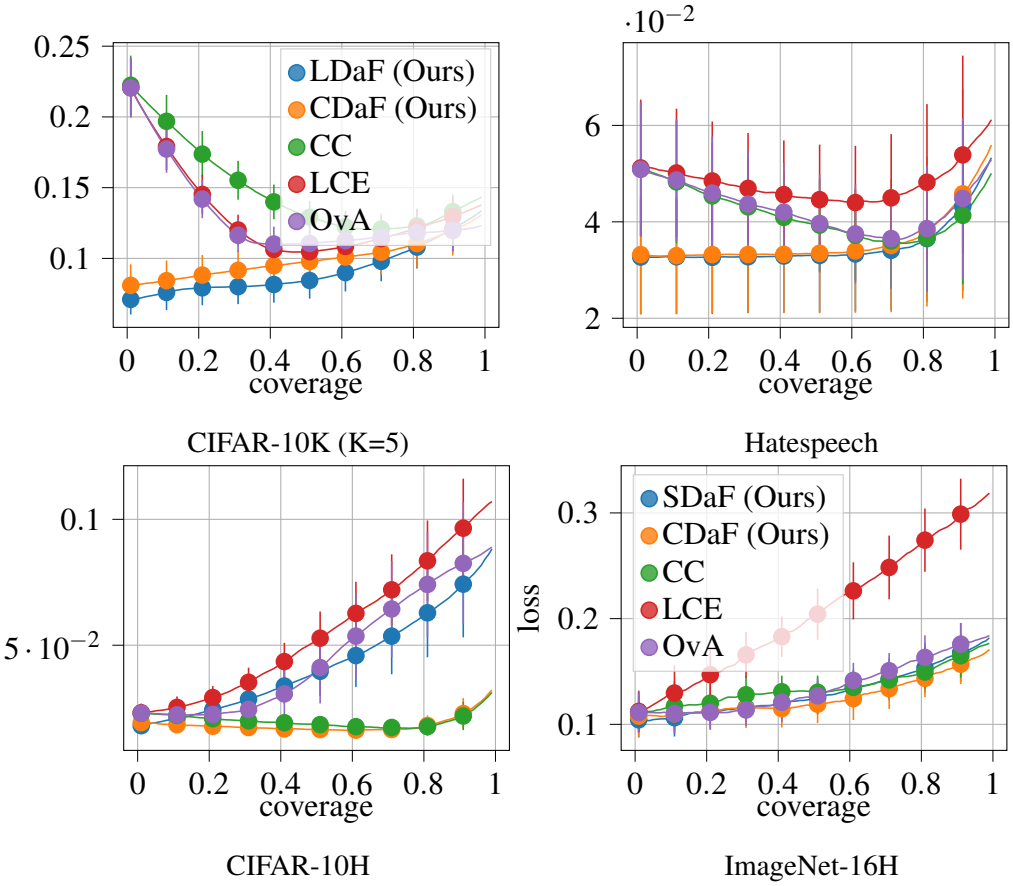



Figure D.7: Average loss vs. coverage on CIFAR-10K, Hatespeech, CIFAR-10H, and ImageNet-16H dataset and for 10 random sets of prediction costs, with bars being standard deviation.

# Bibliography


Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. (2016a). Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318.

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. (2016b). Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, page 308–318, New York, NY, USA. Association for Computing Machinery.

Acs, G., Melis, L., Castelluccia, C., and De Cristofaro, E. (2018). Differentially private mixture of generative neural networks. *IEEE Transactions on Knowledge and Data Engineering*, **31**(6), 1109–1121.

Aronszajn, N. (1950). Theory of reproducing kernels. *Trans Am Math Soc*, **68**(3), 337–404.

Audibert, J.-Y. and Tsybakov, A. B. (2007). Fast learning rates for plug-in classifiers.

Avron, H., Sindhwani, V., Yang, J., and Mahoney, M. W. (2016). Quasi-monte carlo feature maps for shift-invariant kernels. *Journal of Machine Learning Research*, **17**(120), 1–38.

Balog, M., Tolstikhin, I., and Schölkopf, B. (2018). Differentially private database release via kernel mean embeddings. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pages 423–431. PMLR.

Bansal, G., Wu, T., Zhou, J., Fok, R., Nushi, B., Kamar, E., Ribeiro, M. T., and Weld, D. (2021a). Does the whole exceed its parts? the effect of ai explanations on complementary team performance. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–16.

Bansal, G., Nushi, B., Kamar, E., Horvitz, E., and Weld, D. S. (2021b). Is the most accurate ai the best teammate? optimizing ai for teamwork. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 11405–11414.

Bartlett, P. L. and Wegkamp, M. H. (2008). Classification with a reject option using a hinge loss. *Journal of Machine Learning Research*, **9**(8).

Bartlett, P. L., Jordan, M. I., and McAuliffe, J. D. (2006). Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, **101**(473), 138–156.

Beede, E., Baylor, E., Hersch, F., Iurchenko, A., Wilcox, L., Ruamviboonsuk, P., and Vardoulakis, L. M. (2020). A human-centered evaluation of a deep learning system deployed in clinics for the detection of diabetic retinopathy. In *Proceedings of the 2020 CHI conference on human factors in computing systems*, pages 1–12.

Blumer, A., Ehrenfeucht, A., Haussler, D., and Warmuth, M. K. (1989). Learnability and the vapnik-chervonenkis dimension. *Journal of the ACM (JACM)*, **36**(4), 929–965.

Boland, P. J. (1989). Majority systems and the condorcet jury theorem. *Journal of the Royal Statistical Society Series D: The Statistician*, **38**(3), 181–189.

Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J., *et al.* (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, **3**(1), 1–122.

Cao, T., Bie, A., Vahdat, A., Fidler, S., and Kreis, K. (2021). Don't generate me: Training differentially private generative models with sinkhorn divergence. In *NeurIPS*.

Cao, Y., Cai, T., Feng, L., Gu, L., Jinjie, G., An, B., Niu, G., and Sugiyama, M. (2022). Generalizing consistent multi-class classification with rejection to be compatible with arbitrary losses. In *Advances in Neural Information Processing Systems*.

Cao, Y., Mozannar, H., Feng, L., Wei, H., and An, B. (2024). In defense of softmax parametrization for calibrated and consistent learning to defer. *Advances in Neural Information Processing Systems*, **36**.

Chamon, L. F., Paternain, S., Calvo-Fullana, M., and Ribeiro, A. (2022). Constrained learning with non-convex losses. *IEEE Transactions on Information Theory*, **69**(3), 1739–1760.

Charoenphakdee, N., Cui, Z., Zhang, Y., and Sugiyama, M. (2021). Classification with rejection based on cost-sensitive classification. In *International Conference on Machine Learning*, pages 1507–1517. PMLR.

Charusaie, M.-A., Mozannar, H., Sontag, D., and Samadi, S. (2022a). Sample efficient learning of predictors that complement humans. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 2972–3005. PMLR.

Charusaie, M.-A., Mozannar, H., Sontag, D., and Samadi, S. (2022b). Sample efficient learning of predictors that complement humans. In *International Conference on Machine Learning*, pages 2972–3005. PMLR.

Chen, D., Orekondy, T., and Fritz, M. (2020). Gs-wgan: A gradient-sanitized approach for learning differentially private generators. In *Advances in Neural Information Processing Systems 33*.

Chen, R., Xiao, Q., Zhang, Y., and Xu, J. (2015). Differentially private high-dimensional data publication via sampling-based inference. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 129–138.

Chen, R. J., Wang, J. J., Williamson, D. F., Chen, T. Y., Lipkova, J., Lu, M. Y., Sahai, S., and Mahmood, F. (2023a). Algorithmic fairness in artificial intelligence for medicine and healthcare. *Nature biomedical engineering*, **7**(6), 719–742.

Chen, W., Klochkov, Y., and Liu, Y. (2023b). Post-hoc bias scoring is optimal for fair classification. *arXiv preprint arXiv:2310.05725*.

Cheng, X., Cao, Y., Wang, H., Wei, H., An, B., and Feng, L. (2024). Regression with cost-based rejection. *Advances in Neural Information Processing Systems*, **36**.

Chow, C. (1970). On optimum recognition error and reject tradeoff. *IEEE Transactions on information theory*, **16**(1), 41–46.

Chzhen, E., Denis, C., Hebiri, M., Oneto, L., and Pontil, M. (2019). Leveraging labeled and unlabeled data for consistent fair binary classification. *Advances in Neural Information Processing Systems*, **32**.

Clemen, R. T. and Winkler, R. L. (1999). Combining probability distributions from experts in risk analysis. *Risk analysis*, **19**, 187–203.

Corbett-Davies, S., Pierson, E., Feller, A., Goel, S., and Huq, A. (2017). Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd acm sigkdd international conference on knowledge discovery and data mining*, pages 797–806.

Cortes, C., DeSalvo, G., and Mohri, M. (2016). Learning with rejection. In *Algorithmic Learning Theory: 27th International Conference*, pages 67–82. Springer.

Cruz, A. F. and Hardt, M. (2023). Unprocessing seven years of algorithmic fairness. *arXiv preprint arXiv:2306.07261*.

Dantzig, G. B. and Wald, A. (1951). On the fundamental lemma of neyman and pearson. *The Annals of Mathematical Statistics*, **22**(1), 87–93.

Davidson, T., Warmsley, D., Macy, M., and Weber, I. (2017). Automated hate speech detection and the problem of offensive language. In *Eleventh international aaai conference on web and social media*.

De, A., Koley, P., Ganguly, N., and Gomez-Rodriguez, M. (2020). Regression under human assistance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 2611–2620.

De, A., Okati, N., Zarezade, A., and Rodriguez, M. G. (2021). Classification under human assistance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 5905–5913.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Dietrich, F. and Spiekermann, K. (2021). Jury theorems.

Dietterich, T. G. (2000). Ensemble methods in machine learning. In *Multiple Classifier Systems: First International Workshop, MCS 2000 Cagliari, Italy, June 21–23, 2000 Proceedings 1*, pages 1–15. Springer.

Ding, F., Hardt, M., Miller, J., and Schmidt, L. (2021). Retiring adult: New datasets for fair machine learning. *Advances in neural information processing systems*, **34**, 6478–6490.

Donahue, K., Chouldechova, A., and Kenthapadi, K. (2022). Human-algorithm collaboration: Achieving complementarity and avoiding unfairness. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 1639–1656.

Dressel, J. and Farid, H. (2018). The accuracy, fairness, and limits of predicting recidivism. *Science advances*, **4**(1), eaao5580.

Dua, D. and Graff, C. (2017). UCI machine learning repository.

Dwork, C. and Ilvento, C. (2018). Fairness under composition. *arXiv preprint arXiv:1806.06122*.

Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., and Naor, M. (2006). Our data, ourselves: Privacy via distributed noise generation. In *Advances in Cryptology - EUROCRYPT 2006, 25th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, volume 4004 of *Lecture Notes in Computer Science*, pages 486–503. Springer.

Dwork, C., Rothblum, G. N., and Vadhan, S. (2010). Boosting and differential privacy. In *2010 IEEE 51st annual symposium on foundations of computer science*, pages 51–60. IEEE.

Dwork, C., Roth, A., *et al.* (2014). The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, **9**(3–4), 211–407.

El-Yaniv, R. *et al.* (2010). On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, **11**(5).

Fano, R. M. (1961). Transmission of information: A statistical theory of communications. *American Journal of Physics*, **29**(11), 793–794.

Fremlin, D. H. (2000). *Measure theory*, volume 4. Torres Fremlin.

Frigerio, L., de Oliveira, A. S., Gomez, L., and Duverger, P. (2019). Differentially private generative adversarial networks for time series, continuous, and discrete open data. In *ICT Systems Security and Privacy Protection - 34th IFIP TC 11 International Conference, SEC 2019, Lisbon, Portugal, June 25-27, 2019, Proceedings*, pages 151–164.

Gangrade, A., Kag, A., and Saligrama, V. (2021). Selective classification via one-sided prediction. In *International Conference on Artificial Intelligence and Statistics*, pages 2179–2187. PMLR.

Geifman, Y. and El-Yaniv, R. (2017). Selective classification for deep neural networks. *Advances in neural information processing systems*, **30**.

Gillespie, T. (2020). Content moderation, ai, and the question of scale. *Big Data & Society*, **7**(2), 2053951720943234.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial networks. *arXiv preprint arXiv:1406.2661*.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. (2012). A kernel two-sample test. *Journal of Machine Learning Research*, **13**(Mar), 723–773.

Halmos, P. R. and Savage, L. J. (1949). Application of the radon-nikodym theorem to the theory of sufficient statistics. *The Annals of Mathematical Statistics*, **20**(2), 225–241.

Hanneke, S. (2014). Theory of active learning. *Foundations and Trends in Machine Learning*, **7**(2-3).

Harder, F., Adamczewski, K., and Park, M. (2021a). Dp-merf: Differentially private mean embeddings with randomfeatures for practical privacy-preserving data generation. In *International conference on artificial intelligence and statistics*, pages 1819–1827. PMLR.

Harder, F., Adamczewski, K., and Park, M. (2021b). DP-MERF: Differentially private mean embeddings with randomfeatures for practical privacy-preserving data generation. In A. Banerjee and K. Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 1819–1827. PMLR.

Hardt, M., Ligett, K., and Mcsherry, F. (2012). A simple and practical algorithm for differentially private data release. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 2339–2347. Curran Associates, Inc.

Hardt, M., Price, E., and Srebro, N. (2016). Equality of opportunity in supervised learning. *Advances in neural information processing systems*, **29**.

Jacobs, R. A. (1995). Methods for combining experts' probability assessments. *Neural computation*, **7**(5), 867–888.

Jensen, J. L. W. V. (1906). Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta mathematica*, **30**(1), 175–193.

Jiang, X., Osl, M., Kim, J., and Ohno-Machado, L. (2012). Calibrating predictive model estimates to support personalized medicine. *Journal of the American Medical Informatics Association*, **19**(2), 263–274.

Jones, E., Sagawa, S., Koh, P. W., Kumar, A., and Liang, P. (2020). Selective classification can magnify disparities across groups. *arXiv preprint arXiv:2010.14134*.

Kamar, E., Hacker, S., and Horvitz, E. (2012). Combining human and machine intelligence in large-scale crowdsourcing. In *AAMAS*, volume 12, pages 467–474.

Kerrigan, G., Smyth, P., and Steyvers, M. (2021). Combining human predictions with model probabilities via confusion matrices and calibration. *Advances in Neural Information Processing Systems*, **34**.

Killock, D. (2020). Ai outperforms radiologists in mammographic screening. *Nature Reviews Clinical Oncology*, **17**(3), 134–134.

Kittler, J., Hatef, M., Duin, R. P., and Matas, J. (1998). On combining classifiers. *IEEE transactions on pattern analysis and machine intelligence*, **20**(3), 226–239.

Kleinberg, J., Lakkaraju, H., Leskovec, J., Ludwig, J., and Mullainathan, S. (2018). Human decisions and machine predictions. *The quarterly journal of economics*, **133**(1), 237–293.

Klenke, A. (2013). *Probability theory: a comprehensive course*. Springer Science & Business Media.

Kompa, B., Snoek, J., and Beam, A. L. (2021). Second opinion needed: communicating uncertainty in medical machine learning. *NPJ Digital Medicine*, **4**(1), 1–6.

Krishnamurthy, A., Agarwal, A., Huang, T.-K., Daumé III, H., and Langford, J. (2017). Active learning for cost-sensitive classification. In *International Conference on Machine Learning*, pages 1915–1924. PMLR.

Krizhevsky, A., Hinton, G., *et al.* (2009). Learning multiple layers of features from tiny images. *Citeseer*.

Landgrebe, T. and Duin, R. (2005). On neyman-pearson optimisation for multiclass classifiers. In *Proceedings 16th Annual Symposium of the Pattern Recognition Association of South Africa. PRASA*, pages 165–170.

LeCun, Y., Cortes, C., and Burges, C. (2010). Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, **2**.

Ledoux, M. and Talagrand, M. (1991). *Probability in Banach Spaces: isoperimetry and processes*, volume 23. Springer Science & Business Media.

Lee, J. K., Bu, Y., Rajan, D., Sattigeri, P., Panda, R., Das, S., and Wornell, G. W. (2021). Fair selective classification via sufficiency. In *International conference on machine learning*, pages 6076–6086. PMLR.

Lehmann, E. L., Romano, J. P., and Casella, G. (1986). *Testing statistical hypotheses*, volume 3. Springer.

Linsker, R. (1987). Towards an organizing principle for a layered perceptual network. In *Neural information processing systems*.

Liu, S., Cao, Y., Zhang, Q., Feng, L., and An, B. (2024). Mitigating underfitting in learning to defer with consistent losses. In *International Conference on Artificial Intelligence and Statistics*, pages 4816–4824. PMLR.

Loshchilov, I. and Hutter, F. (2017). Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Madras, D., Pitassi, T., and Zemel, R. (2018). Predict responsibly: improving fairness and accuracy by learning to defer. *Advances in Neural Information Processing Systems*, **31**.

Mao, A., Mohri, M., and Zhong, Y. (2024). Predictor-rejector multi-class abstention: Theoretical analysis and algorithms. In *International Conference on Algorithmic Learning Theory*, pages 822–867. PMLR.

McKenna, R., Sheldon, D., and Miklau, G. (2019). Graphical-model based estimation and inference for differential privacy. *arXiv preprint arXiv:1901.09136*.

Mehler, F. G. (1866). Ueber die entwicklung einer function von beliebig vielen variablen nach laplaceschen functionen höherer ordnung.

Metz, C. E. (1978). Basic principles of roc analysis. In *Seminars in nuclear medicine*, volume 8, pages 283–298. Elsevier.

Mohammed, N., Chen, R., Fung, B. C., and Yu, P. S. (2011). Differentially private data release for data mining. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 493–501, New York, NY, USA. ACM.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. (2018). *Foundations of machine learning*. MIT press.

Morishita, T., Morio, G., Horiguchi, S., Ozaki, H., and Nukaga, N. (2022). Rethinking fano's inequality in ensemble learning. In *International Conference on Machine Learning*, pages 15976–16016. PMLR.

Mozannar, H. and Sontag, D. (2020a). Consistent estimators for learning to defer to an expert. In *International Conference on Machine Learning*, pages 7076–7087. PMLR.

Mozannar, H. and Sontag, D. (2020b). Consistent estimators for learning to defer to an expert. In H. D. III and A. Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 7076–7087. PMLR.

Mozannar, H., Lang, H., Wei, D., Sattigeri, P., Das, S., and Sontag, D. (2023a). Who should predict? exact algorithms for learning to defer to humans. In *International conference on artificial intelligence and statistics*, pages 10520–10545. PMLR.

Mozannar, H., Lang, H., Wei, D., Sattigeri, P., Das, S., and Sontag, D. (2023b). Who should predict? exact algorithms for learning to defer to humans. *arXiv preprint arXiv:2301.06197*.

Mozannar, H., Lang, H., Wei, D., Sattigeri, P., Das, S., and Sontag, D. (2023c). Who should predict? exact algorithms for learning to defer to humans. In F. Ruiz, J. Dy, and J.-W. van de Meent, editors, *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pages 10520–10545. PMLR.

Narasimhan, H., Ramaswamy, H. G., Tavker, S. K., Khurana, D., Netrapalli, P., and Agarwal, S. (2022a). Consistent multiclass algorithms for complex metrics and constraints. *arXiv preprint arXiv:2210.09695*.

Narasimhan, H., Jitkrittum, W., Menon, A. K., Rawat, A., and Kumar, S. (2022b). Post-hoc estimators for learning to defer to an expert. *Advances in Neural Information Processing Systems*, **35**, 29292–29304.

Narasimhan, H., Menon, A. K., Jitkrittum, W., and Kumar, S. (2023). Plugin estimators for selective classification with out-of-distribution detection. *arXiv preprint arXiv:2301.12386*.

Narasimhan, H., Menon, A. K., Jitkrittum, W., Gupta, N., and Kumar, S. (2024). Learning to reject meets long-tail learning. In *The Twelfth International Conference on Learning Representations*.

Narayanan, A. and Shmatikov, V. (2008). Robust de-anonymization of large sparse datasets. In *2008 IEEE Symposium on Security and Privacy (sp 2008)*, pages 111–125. IEEE.

Natarajan, S., Das, S., Ramanan, N., Kunapuli, G., and Radivojac, P. (2018). On whom should i perform this lab test next? an active feature elicitation approach. In *IJCAI*, pages 3498–3505.

National Institute for Standards and Techonologies (2018). Nist 2018 differential privacy synthetic data challenge. https://www.nist.gov/ctl/pscr/open-innovation-prize-challenges/past-prize-challenges/2018-differential-privacy-synthetic.

Neyman, J. and Pearson, E. S. (1933). Ix. on the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, **231**(694-706), 289–337.

Neyman, J. and Pearson, E. S. (1936). Contributions to the theory of testing statistical hypotheses. *Statistical research memoirs*.

Ni, C., Charoenphakdee, N., Honda, J., and Sugiyama, M. (2019). On the calibration of multiclass classification with rejection. *arXiv preprint arXiv:1901.10655*.

Nowak-Vila, A., Bach, F., and Rudi, A. (2019). A general theory for structured prediction with smooth convex surrogates. *arXiv preprint arXiv:1902.01958*.

Okati, N., De, A., and Gomez-Rodriguez, M. (2021a). Differentiable learning under triage. *arXiv preprint arXiv:2103.08902*.

Okati, N., De, A., and Rodriguez, M. (2021b). Differentiable learning under triage. *Advances in Neural Information Processing Systems*, **34**, 9140–9151.

Papadimitriou, C. H. and Steiglitz, K. (1998). *Combinatorial optimization: algorithms and complexity*. Courier Corporation.

Papernot, N., Abadi, M., Erlingsson, U., Goodfellow, I., and Talwar, K. (2017). Semi-supervised Knowledge Transfer for Deep Learning from Private Training Data. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Patel, B. N., Rosenberg, L., Willcox, G., Baltaxe, D., Lyons, M., Irvin, J., Rajpurkar, P., Amrhein, T., Gupta, R., Halabi, S., *et al.* (2019). Human–machine partnership with artificial intelligence for chest radiograph diagnosis. *NPJ digital medicine*, **2**(1), 111.

Peterson, J. C., Battleday, R. M., Griffiths, T. L., and Russakovsky, O. (2019). Human uncertainty makes classification more robust. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9617–9626.

Pugh, C. C. and Pugh, C. (2002). *Real mathematical analysis*, volume 2011. Springer.

Qardaji, W., Yang, W., and Li, N. (2014). Priview: practical differentially private release of marginal contingency tables. In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pages 1435–1446.

Raghu, M., Blumer, K., Corrado, G., Kleinberg, J., Obermeyer, Z., and Mullainathan, S. (2019). The algorithmic automation problem: Prediction, triage, and human effort. *arXiv preprint arXiv:1903.12220*.

Rahimi, A. and Recht, B. (2007). Random features for large-scale kernel machines. *NIPS*.

Rahimi, A. and Recht, B. (2008). Random features for large-scale kernel machines. In *Advances in neural information processing systems*, pages 1177–1184.

Rajpurkar, P., Irvin, J., Ball, R. L., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C. P., *et al.* (2018). Deep learning for chest radiograph diagnosis: A retrospective comparison of the chexnext algorithm to practicing radiologists. *PLoS medicine*, **15**(11), e1002686.

Rajpurkar, P., O'Connell, C., Schechter, A., Asnani, N., Li, J., Kiani, A., Ball, R. L., Mendelson, M., Maartens, G., van Hoving, D. J., *et al.* (2020). Chexaid: deep learning assistance for physician diagnosis of tuberculosis using chest x-rays in patients with hiv. *NPJ digital medicine*, **3**(1), 115.

Ramaswamy, H. G., Srinivasan Babu, B., Agarwal, S., and Williamson, R. C. (2014). On the consistency of output code based learning algorithms for multiclass learning problems. In M. F. Balcan, V. Feldman, and C. Szepesvári, editors, *Proceedings of The 27th Conference on Learning Theory*, volume 35 of *Proceedings of Machine Learning Research*, pages 885–902, Barcelona, Spain. PMLR.

Rao, C. R. (1992). Information and the accuracy attainable in the estimation of statistical parameters. In *Breakthroughs in statistics*, pages 235–247. Springer.

Rasmussen, C. E. and Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press.

Rastogi, C., Leqi, L., Holstein, K., and Heidari, H. (2022). A unifying framework for combining complementary strengths of humans and ml toward better predictive decision-making. *arXiv preprint arXiv:2204.10806*.

Reed, M. and Simon, B. (1980). *Methods of modern mathematical physics: Functional analysis*, volume 1. Gulf Professional Publishing.

Rigollet, P. and Tong, X. (2011). Neyman-pearson classification, convexity and stochastic constraints. *Journal of machine learning research*.

Rudin, W. (2013). *Fourier Analysis on Groups: Interscience Tracts in Pure and Applied Mathematics, No. 12*. Literary Licensing, LLC.

Ruggieri, S., Alvarez, J. M., Pugnana, A., Turini, F., *et al.* (2023). Can we trust fair-ai? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 15421–15430.

Sammut, S.-J., Crispin-Ortuzar, M., Chin, S.-F., Provenzano, E., Bardwell, H. A., Ma, W., Cope, W., Dariush, A., Dawson, S.-J., Abraham, J. E., *et al.* (2022). Multi-omic machine learning predictor of breast cancer therapy response. *Nature*, **601**(7894), 623–629.

Sarpatwar, K., Shanmugam, K., Ganapavarapu, V. S., Jagmohan, A., and Vaculin, R. (2019). Differentially private distributed data summarization under covariate shift. In *Advances in Neural Information Processing Systems*, pages 14432–14442.

Scarlett, J. and Cevher, V. (2019). An introductory guide to fano's inequality with applications in statistical estimation. *arXiv preprint arXiv:1901.00555*.

Scott, C. (2007). Performance measures for neyman–pearson classification. *IEEE Transactions on Information Theory*, **53**(8), 2852–2863.

Scott, C. and Nowak, R. (2005). A neyman-pearson approach to statistical learning. *IEEE Transactions on Information Theory*, **51**(11), 3806–3819.

Shalev-Shwartz, S. and Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms*. Cambridge University Press.

Shim, H., Hwang, S. J., and Yang, E. (2018). Joint active feature acquisition and classification with variable-size set encoding. *Advances in neural information processing systems*, **31**.

Slepian, D. (1972). On the symmetrized kronecker power of a matrix and extensions of mehler's formula for hermite polynomials. *SIAM Journal on Mathematical Analysis*, **3**(4), 606–616.

Smola, A., Gretton, A., Song, L., and Schölkopf, B. (2007). A Hilbert space embedding for distributions. In *ALT*, pages 13–31.

Smola, A. J. and Schölkopf, B. (1998). *Learning with kernels*, volume 4. Citeseer.

Snoke, J. and Slavković, A. (2018). pmse mechanism: differentially private synthetic data with maximal distributional similarity. In *International Conference on Privacy in Statistical Databases*, pages 138–159. Springer.

Sriperumbudur, B. K., Fukumizu, K., and Lanckriet, G. R. (2011). Universality, characteristic kernels and rkhs embedding of measures. *Journal of Machine Learning Research*, **12**(7).

Steyvers, M., Tejeda, H., Kerrigan, G., and Smyth, P. (2022). Bayesian modeling of human–ai complementarity. *Proceedings of the National Academy of Sciences*, **119**(11), e2111547119.

Straitouri, E., Singla, A., Meresht, V. B., and Gomez-Rodriguez, M. (2021). Reinforcement learning under algorithmic triage. *arXiv preprint arXiv:2109.11328*.

Sweeney, L. (2002). Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, **10**(05), 571–588.

Tailor, D., Patra, A., Verma, R., Manggala, P., and Nalisnick, E. (2024). Learning to defer to a population: A meta-learning approach. In *International Conference on Artificial Intelligence and Statistics*, pages 3475–3483. PMLR.

Tan, S., Adebayo, J., Inkpen, K., and Kamar, E. (2018). Investigating human+ machine complementarity for recidivism predictions. *arXiv preprint arXiv:1808.09123*.

Tebbe, D. and Dwyer, S. (1968). Uncertainty and the probability of error (corresp.). *IEEE Transactions on Information theory*, **14**(3), 516–518.

Tian, Y. and Feng, Y. (2021). Neyman-pearson multi-class classification via cost-sensitive learning. *arXiv preprint arXiv:2111.04597*.

Tong, X. (2013). A plug-in approach to neyman-pearson classification. *The Journal of Machine Learning Research*, **14**(1), 3011–3040.

Torkzadehmahani, R., Kairouz, P., and Paten, B. (2019). Dp-cgan: Differentially private synthetic data and label generation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Unser, M. and Tafti, P. D. (2014). *An introduction to sparse stochastic processes*. Cambridge University Press.

van Leeuwen, K. G., Schalekamp, S., Rutten, M. J., van Ginneken, B., and de Rooij, M. (2021). Artificial intelligence in radiology: 100 commercially available products and their scientific evidence. *European radiology*, **31**, 3797–3804.

Vapnik, V. and Chervonenkis, A. Y. (1971). On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, **16**(2), 264–280.

Verma, R. and Nalisnick, E. (2022a). Calibrated learning to defer with one-vs-all classifiers. *arXiv preprint arXiv:2202.03673*.

Verma, R. and Nalisnick, E. (2022b). Calibrated learning to defer with one-vs-all classifiers. In K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 22184–22202. PMLR.

Verma, R., Barrejon, D., and Nalisnick, E. (2023). Learning to defer to multiple experts: Consistent surrogate losses, confidence calibration, and conformal ensembles. In F. Ruiz, J. Dy, and J.-W. van de Meent, editors, *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pages 11415–11434. PMLR.

Vermeulen, C., Pagès-Gallego, M., Kester, L., Kranendonk, M., Wesseling, P., Verburg, N., de Witt Hamer, P., Kooi, E., Dankmeijer, L., van der Lugt, J., *et al.* (2023). Ultra-fast deep-learned cns tumour classification during surgery. *Nature*, **622**(7984), 842–849.

Wang, Y.-X., Balle, B., and Kasiviswanathan, S. P. (2019). Subsampled rényi differential privacy and analytical moments accountant. PMLR.

Wilder, B., Horvitz, E., and Kamar, E. (2020). Learning to complement humans. *arXiv preprint arXiv:2005.00582*.

Xian, R., Yin, L., and Zhao, H. (2023). Fair and optimal classification via post-processing. In *International Conference on Machine Learning*, pages 37977–38012. PMLR.

Xiao, H., Rasul, K., and Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*.

Xiao, Y., Xiong, L., and Yuan, C. (2010). Differentially private data release through multidimensional partitioning. In W. Jonker and M. Petković, editors, *Secure Data Management*, pages 150–168, Berlin, Heidelberg. Springer Berlin Heidelberg.

Xie, L., Lin, K., Wang, S., Wang, F., and Zhou, J. (2018a). Differentially private generative adversarial network. *arXiv preprint arXiv:1802.06739*.

Xie, L., Lin, K., Wang, S., Wang, F., and Zhou, J. (2018b). Differentially private generative adversarial network. *CoRR*, **abs/1802.06739**.

Yin, T., Ton, J.-F., Guo, R., Yao, Y., Liu, M., and Liu, Y. (2023). Fair classifiers that abstain without harm. *arXiv preprint arXiv:2310.06205*.

Yoon, J., Jordon, J., and van der Schaar, M. (2019). PATE-GAN: Generating synthetic data with differential privacy guarantees. In *International Conference on Learning Representations*.

Zeng, X., Dobriban, E., and Cheng, G. (2022). Bayes-optimal classifiers under group fairness. *arXiv preprint arXiv:2202.09724*.

Zeng, X., Cheng, G., and Dobriban, E. (2024). Bayes-optimal fair classification with linear disparity constraints via pre-, in-, and post-processing. *arXiv preprint arXiv:2402.02817*.

Zhang, D., McKenna, R., Kotsogiannis, I., Hay, M., Machanavajjhala, A., and Miklau, G. (2018). Ektelo: A framework for defining differentially-private computations. SIGMOD.

Zhang, J. and Bareinboim, E. (2022). Can humans be out of the loop? In *Conference on Causal Learning and Reasoning*, pages 1010–1025. PMLR.

Zhang, J., Cormode, G., Procopiuc, C. M., Srivastava, D., and Xiao, X. (2017). Privbayes: Private data release via bayesian networks. *ACM Transactions on Database Systems (TODS)*, **42**(4), 1–41.

Zhang, T. (2004). Statistical analysis of some multi-category large margin classification methods. *Journal of Machine Learning Research*, **5**(Oct), 1225–1251.

Zhang, Z., Wang, T., Li, N., Honorio, J., Backes, M., He, S., Chen, J., and Zhang, Y. (2021). Privsyn: Differentially private data synthesis. In *30th {USENIX} Security Symposium ({USENIX} Security 21)*.

Zhao, M.-J., Edakunni, N., Pocock, A., and Brown, G. (2013). Beyond fano's inequality: Bounds on the optimal f-score, ber, and cost-sensitive risk and their implications. *The Journal of Machine Learning Research*, **14**(1), 1033–1090.

Zhu, H., Williams, C. K., Rohwer, R., and Morciniec, M. (1997). Gaussian regression and optimal finite dimensional linear models.

Zhu, T., Li, G., Zhou, W., and Yu, P. S. (2017). Differentially private data publishing and analysis: A survey. *IEEE Transactions on Knowledge and Data Engineering*, **29**(8), 1619–1638.