Identifying and Characterizing High Dimensional Covariate Shift with Learning Models

by

Tom Ginsberg

A thesis submitted in conformity with the requirements
for the degree of Master of Science

Department of Computer Science
University of Toronto

Identifying and Characterizing High Dimensional Covariate Shift with Learning Models

Tom Ginsberg
Master of Science

Department of Computer Science
University of Toronto
2023

Abstract

The ability to quickly and accurately identify covariate shift at test time is a critical and often overlooked component of safe machine learning systems deployed in high-risk domains. While methods exist for detecting when predictions should not be made on out-of-distribution test examples, identifying distributional level differences between training and test time can help determine when a model should be removed from the deployment setting and retrained. In this thesis, we explore modern and foundational methods for identifying and characterizing distributional shift in high dimensional data, in particular where such data is treated as the covariates to a learning model — this type of distribution shift is known formally as covariate shift. We go on to provide a new definition for *harmful covariate shift* (HCS) that goes beyond ideas from standard learning theory to give a richer insight on when covariate shift may hurt the performance of classification models. Motivated from our definition, we propose a method, the Detectron, to detect HCS based on the discordance between an ensemble of *constrained disagreement classifiers* (CDCs) trained to agree on training data and disagree on test data. We derive a loss function for training CDCs and show that their disagreement rate and predictive entropy represent powerful discriminative statistics for HCS. Furthermore, we present tight finite sample shift detection guarantees in an idealized setting. Empirically, we demonstrate that the CDC learning algorithm produces model with behaviour that aligns well with our desideratum. Finally we showcase the ability of the Detectron to detect HCS with statistical certainty on a variety of high-dimensional datasets. Across numerous domains and modalities, we show state-of-the-art performance compared to existing methods, particularly when the number of observed test samples is small.

Acknowledgements

I would first and foremost like to thank my graduate supervisor Professor Rahul G. Krishnan, for his continuous support and mentorship throughout my studies, as well as Professor Murat Erdogdu for his role as an additional reader for this thesis. I would also like to thank my original graduate cohort, Vahid Balazadeh, and Michael Cooper, for their help brainstorming ideas, useful feedback, research advice, squash playing and all around creation of an amazing lab environment. Additional thanks to others that provided helpful advice and feedback include: Edward De Brouwer, Aslesha Pokhrel, Zhongyuan Liang, Adnan Mohd, Ian Shi, and Stephan Rabanser.

Contents

List of Tables

List of Figures

Chapter 1 Introduction: Covariate Shift

Machine learning models operate on the assumption, albeit incorrectly that they will be deployed on data distributed identically to what they were trained on. The violation of this assumption is known as distribution shift and can often result in significant degradation of performance [Recht et al., 2019; Hendrycks and Dietterich, 2019; Bickel et al., 2009; Rabanser et al., 2019; Otles et al., 2021; Ovadia et al., 2019]. There are several cases where a mismatch between training and deployment data results in very real consequences on human beings. In healthcare, machine learning models have been deployed for predicting the likelihood of sepsis. Yet, as [Habib et al., 2021] show, such models can be miscalibrated for large groups of individuals, directly affecting the quality of care they experience. The deployment of classifiers in the criminal justice system [Hao, 2019], hiring and recruitment pipelines [Dastin, 2018] and self-driving cars [Smiley, 2022] have all seen humans affected by the failures of learning models. The need for methods that quickly detect, characterize and respond to distribution shift is, therefore, a fundamental problem in trustworthy machine learning. For practitioners, regulatory agencies and individuals to have faith in deployed predictive models without the need for laborious manual audits, we need methods for the identification of distribution shift that are *sample-efficient* (identifying shifts from a small number of samples), *informed* (identifying shifts relevant to the domain and learning algorithm), *model-agnostic* (identifying shifts regardless of the functional class of the predictive model) and *statistically sound* (identifying true shifts while avoiding false positives with high-confidence).

In our work, we study a special case of distribution shift, commonly known as *covariate shift*, which considers shifts only in the distribution of input data $\mathcal{P}_X \neq \mathcal{Q}_X$ while the relation between the inputs and outputs remains fixed $\mathcal{P}_{Y|X} = \mathcal{Q}_{Y|X}$. In a standard deployment setting where ground truth labels are not available, estimating any properties of the true distribution $\mathcal{Q}_{Y|X}$ is fundamentally impossible without strong assumptions. As a consequence covariate shift is the only type of distribution shift that can be identified.

We build off recent progress in understanding model performance under covariate shift using the PQ-learning framework [Goldwasser et al., 2020], a framework for selective classifiers that may either predict on or reject a given sample, that provides strong performance guarantees on arbitrary test distributions. Our work uses and extends PQ-learning to develop a practical, model-based hypothesis test, named *the Detectron*, to identify potentially harmful covariate shifts given any existing classification model already in deployment.

Our work makes the following key contributions:

- We show how to construct an ensemble of classifiers that maximize out-of-domain disagreement while behaving consistently in the training domain. We propose the *disagreement cross entropy* for models learned via continuous gradient-based methods (e.g., neural networks), as well as a generalization for those learned via discrete optimization (e.g., random forest).
- We show that the rejection rate and the entropy of the learning ensemble can be used to define a model-aware hypothesis test for covariate shift, the Detectron, that in idealized settings can provably detect covariate shift.
- On high-dimensional image and tabular data, using both neural networks and gradient boosted decision trees, our method outperforms state-of-the-art techniques for detecting covariate shift, particularly when given access to as few as ten test examples.

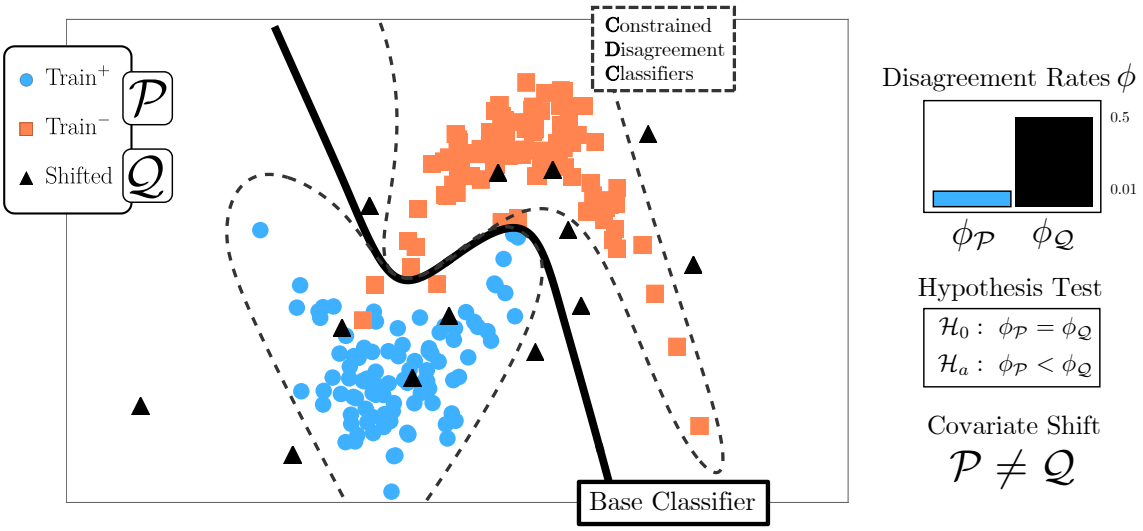

Figure 1.1: **Detectron (PQ Learning for Covariate Shift Detection):** Starting with a base classifier trained on labeled samples from distribution $\mathcal{P}$ we train new ***C**onstrained **D**isagreement **C**lassifiers* (CDCs) on a small set of observed unlabeled samples from a new distribution $\mathcal{Q}$. CDCs aim to maximize classification disagreement on $\mathcal{Q}$ while constrained to agree with the base classifier on $\mathcal{P}$. The rate ϕ that CDCs disagree, as well as their entropy, on test data is a powerful and sample efficient statistic for identifying covariate shift $\mathcal{P} \neq \mathcal{Q}$.

Chapter 2 Background: Covariate Shift

Covariate shift is the tendency for a covariate distribution at test time $p_{\text{test}}(x)$ to differ from that seen during training $p_{\text{train}}(x)$ while the underlying prediction concept y remains fixed [Sugiyama and Kawanabe, 2012]

$$p_{\text{train}}(x) \neq p_{\text{test}}(x) \text{ while } p_{\text{train}}(y|x) = p_{\text{test}}(y|x)$$

The problem of detecting variate shift is important at a fundamental level in machine learning. From a classical learning theory perspective, predictive ML algorithms that operate on unseen data can only claim any performance guarantees when such data is exchangeable with the labeled samples which they were tested on [Haussler, 1990]. The violation of this assumption in general is known as distribution shift. However, in a typical deployment setting where only covariates are observed one cannot detect changes in $p(y)$ or $p(y|x)$ leaving covariate shift as the only remaining phenomenon identifiable from data. The related problems of *label shift* (shift in $p(y)$) and *concept shift* (shift in $p(y|x)$) pose an additional problems for maintaining the robustness of deployed ML systems, but are not the topic of this work. To understand contemporary methods for detecting covariate shift in the high dimensional data that is typical of the covariates found in real-world ML systems we must first take a segway back to the underlying concepts in probability and statistics.

2.1 Two Sample Statistical Testing

If the reader is familiar with the subject of two sample testing they may continue to section 2.2.

The problem of detecting when two distributions $\mathcal{P}$ and $\mathcal{Q}$ over $\mathcal{X}$ are different given only finite samples $\mathbf{P} = \{X_1, \dots X_n\} \overset{\text{iid}}{\sim} \mathcal{P}$ and $\mathbf{Q} = \{\tilde{X}_1, \dots \tilde{X}_m\} \overset{\text{iid}}{\sim} \mathcal{Q}$ is one of the most fundamental problems in statistics. The general problems seeks to rule out a *null hypothesis* $\mathcal{H}_0 : \mathcal{P} = \mathcal{Q}$ in favor of an alternative $\mathcal{H}_\dashv : \mathcal{P} \neq \mathcal{Q}$ at a given significance level α. The significance level is equivalent to the false positive rate or probability of making a type I error, which fixes a threshold for the test such that when $\mathcal{P} = \mathcal{Q}$, $\mathcal{H}_0$ is ruled out with probability no more than α. The power of a statistical test, often denoted as $1 - \beta$ where β is the probability of making a type II error, is the probability that the test correctly rules out $\mathcal{H}_0$ while still maintaining a significance level of α. Will less standard in statistics literature, it is natural to measure the area under the receiver operating characteristics (AUROC) as area under curve $1 - \beta$ as a function of α for $\alpha \in [0, 1]$. For a more

rigours mathematical underpinning of two sample testing see section 2.1 in Schrab et al.. In the following subsections we highlight several important approaches to two sample testing that will form underpinnings of much of the later methodologies.

Kolmogorov–Smirnov test

An elegant and robust two sample test that requires no assumptions on the distributions $\mathcal{P}$ and $\mathcal{Q}$ besides $\mathcal{X} = \mathbb{R}$ is the Kolmogorov–Smirnov test (KS test) [Kolmogorov, 1933]. The KS test leverages the result of the Glivenko–Cantelli theorem [Glivenko, 1933] to construct a test statistic defined by the largest absolute difference between the two empirical cumulative distribution functions across all observed values. As the Glivenko–Cantelli theorem is simple yet fundamental to understanding the KS test we present it in full below:

Theorem 1 (Glivenko–Cantelli [Glivenko, 1933])**.** Assume that $X_1, X_2, \ldots$ are independent and identically-distributed random variables in $\mathbb{R}$ with common cumulative distribution function $F(x)$. The empirical distribution function for $X_1, \ldots, X_n$ is defined by

$$F_n(x) = \frac{1}{n}\sum_{i=1}^{n} I_{[X_i,\infty)}(x) = \frac{1}{n}\left|\{1 \le i \le n \mid X_i \le x\}\right| \tag{2.1}$$

where I_C is the indicator function of the set C. For every (fixed) x, $F_n(x)$ is a sequence of random variables which converge to $F(x)$ almost surely by the strong law of large numbers.

$$\|F_n - F\|_\infty = \sup_{x\in\mathbb{R}} |F_n(x) - F(x)| \longrightarrow 0 \text{ almost surely} \tag{2.2}$$

To perform a KS two sample test, one computes the exact probability under the null hypothesis $\mathcal{P} = \mathcal{Q}$ of obtaining a rarer value compared to the observation of the test statistic

$$D_{\mathbf{P},\mathbf{Q}} \triangleq \sup_{x\in\mathbb{R}} |F_{\mathbf{P}}(x) - F_{\mathbf{Q}}(x)|$$

Where $F_{\mathbf{S}}$ is the empirical CDF for a set of samples $\mathbf{S}$.

Computing the probability of observing a rarer test statistic is non-trivial and in general maps to the problem of counting the number of distinct paths that pass outside the specified diagonal on a $|\mathbf{P}|\times|\mathbf{Q}|$ rectangular grid [Drew et al., 2000]. However, for large $\mathbf{P}$ and $\mathbf{Q}$ asymptotic approximations exist that can greatly reduce the computational complexity. A shortcoming of the KS test is that it sacrifices statistical power in order to be more flexible to handle arbitrary differences between

distributions.

Binomial test

While the KS test present a general approach to univariate two sample testing, we can often use partial knowledge over the distribution that generates our samples to design of a more powerful test; the binomial test is one such example. The test considers a binomially distributed random variable X with unknown rate q i.e., $X \sim \text{Binomial}(n, q)$ for which we observe a single sample x (hence $\mathcal{X} = \{0, \ldots, n\}$). Since the binomial distribution is defined as a sum of i.i.d. Bernoulli random variables with the same rate, x may equivalently be interpreted as a set of n samples of which x are 1 and $n - x$ are 0. We wish to either confirm or rule out that X was sampled from a baseline binomial distribution with known rate p. We do this by computing the probability of observing an event at least as rare as $X = x$ under the null hypothesis that $p = q$. This quantity can be computed exactly using the symmetry of the binomial distribution.

$$\begin{aligned}
\mathbb{P}_{X\sim\text{Bin}(n,p)}(X \text{ is rarer then } x) &= 2 \times \mathbb{P}_{X\sim\text{Bin}(n,p)}(X \geq x) \\
&= 2\sum_{k=x}^{n} \mathbb{P}[X = k] \\
&= 2\sum_{k=x}^{n} (1-p)^{n-k} p^k \binom{n}{k} = 2\frac{B_p(x, n-x+1)}{B(x, n-x+1)}
\end{aligned}$$

Where $B_z(\alpha, \beta)$ is the incomplete Beta function and $B(\alpha, \beta)$ is the beta function. By the convenient construction of the test one can query the likelihood specifically for one-sided alternatives; something that is more practical when there is specific knowledge related to the alternative hypothesis. For example, one can test if a coin is unfairly biased towards heads given an observation of 60/100 heads, in this case the one-sided binomial test admits a p-value of ≈ 0.028 suggesting that we can rule out at the 5% level that the coin is fair in favor of it being biased towards heads.

Do to the simple structure of the binomial test one can derive a simple summation formula for computing the exact statistical power as a function of n, p, q and the significance level α. This quantity corresponds to the probability of observing a p-value of less than or equal to α when

running a binomial test on a

$$\begin{aligned}\mathbb{P}_{Y\sim\text{Bin}(n,q)}\left(\mathbb{P}_{X\sim\text{Bin}(n,p)}(X \geq Y) \leq \frac{\alpha}{2}\right) &= \mathbb{P}_{Y\sim\text{Bin}(n,q)}\left(\sum_{k=k}^{n}\binom{n}{k}(1-p)^{n-k}p^k \leq \frac{\alpha}{2}\right) \\ &= \mathbb{E}_{Y\sim\text{Bin}(n,q)}\left(\left[\sum_{k=k}^{n}\binom{n}{k}(1-p)^{n-k}p^k \leq \frac{\alpha}{2}\right]\right) \\ &= \sum_{y=0}^{n}\binom{n}{y}\left[\sum_{k=y}^{n}\binom{n}{k}(1-p)^{n-k}p^k \leq \frac{\alpha}{2}\right] q^y(1-q)^{n-y}\end{aligned}$$

We verify this formula using a simple monte carlo simulation with $p = .5$, $q = .6$, $n = 100$ and $\alpha = 0.05$. The statistical power and standard error estimated over $10,000$ draws gives 0.461 ± 0.005 whereas the above summation yields a value of 0.4621. One important note is that under this simple construction the binomial test is not a two sample test, but merely a single sample test. However, in a setting where one has access to many more sample in $\mathbf{P}$ compared to that in $\mathbf{Q}$ (which will be the primary setting of the upcoming methodology), one can reasonably estimate the baseline success rate p from $\mathbf{P}$ and use the binomial test as is. In other settings where the assumption of $|\mathbf{P}| \gg |\mathbf{Q}|$ does not hold, one may resort to Barnard's exact test [Erguler, 2016] or the often more powerful alternative Fisher's exact test [Fisher, 1922].

Integral Probability Metrics & Maximum Mean Discrepancy

A general method to define a measure between probability distributions is via an integral probability metric or IPM [Müller, 1997]. An IPM is defined as the supremum over a set of functions F on the absolute difference between the expectations under $\mathcal{P}$ and $\mathcal{Q}$.

$$\text{IPM}_F(P\|Q) = \sup_{f\in F}\left|\mathbb{E}_{x\sim P}\left[f(x)\right] - \mathbb{E}_{x\sim Q}\left[f(x)\right]\right|$$

When F is chosen as the set of all functions with bounded $\infty-$norm $F = \{f : \|f\|_\infty \leq 1\}$ the associated IPM results in the well known total variation distance [Sriperumbudur et al., 2009].

The Maximum Mean Discrepancy (MMD) is formalized as an IPM given F to be the n Reproducing Kernel Hilbert Space $\mathcal{H}_k$ with a kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ [Schrab et al., 2021].

$$\text{MMD}\left(\mathcal{P}, \mathcal{Q}; \mathcal{H}_k\right) = \sup_{f\in\mathcal{H}_k:\|f\|_{\mathcal{H}_k}\leq 1}\left|\mathbb{E}_{X\sim\mathcal{P}}[f(X)] - \mathbb{E}_{Y\sim\mathcal{Q}}[f(Y)]\right|$$

When we chose a characteristic kernel such that $\text{MMD}\left(\mathcal{P}, \mathcal{Q}; \mathcal{H}_k\right) = 0 \iff \mathcal{P} = \mathcal{Q}$ Gretton et al.

construct a quadratic time MMD estimator using samples $\mathbf{P}$ and $\mathbf{Q}$ of sizes n and m respectively:

$$\widehat{\mathrm{MMD}}^2(\mathbf{P},\mathbf{Q};\mathcal{H}_k) = \frac{1}{n(n-1)}\sum_{1\le i\neq i'\le n} k(\mathbf{P}_i,\mathbf{P}_{i'}) + \frac{1}{m(m-1)}\sum_{1\le j\neq j'\le m} k(\mathbf{Q}_j,\mathbf{Q}_{j'}) - \frac{2}{mn}\sum_{i=1}^{n}\sum_{j=1}^{m} k(\mathbf{P}_i,\mathbf{Q}_j) \tag{2.3}$$

Existing work for detecting high dimensional distribution shift using MMD estimators focus on the design of kernels. Rabanser et al. considers the squared exponential kernel $k(x,y) = \exp\left(\sigma^{-1}\|x-y\|^2\right)$, while Liu et al. explore of use of deep neural networks for building more statistically powerful kernels.

While MMD presents an estimator for measuring the difference in distributions, it does not directly lead to a rigorous two sample test with a bounded significance level, to achieve this we will need to turn to the methodology of permutation testing presented in the next subsection.

Permutation Testing

Permutation testing [Fisher, 1935] is a method to guarantee the significance level of any test by first estimating a quantile of the distribution of a test statistic when the null hypothesis $\mathcal{H}_0$ is true ($\mathcal{P} = \mathcal{Q}$). For some tests like KS and Binomial this quantile can be computed analytically or using an exact algorithm, but in the general case it cannot, motivating the need for alternative methods.

Suppose we have a test statistic function $T : \mathcal{X}^n \times \mathcal{X}^m \to \mathbb{R}$. In the context of two sample testing T maps a set $\mathbf{P}$ and $\mathbf{Q}$ to a real value which we assume to be some heuristic measure of the distributional difference between $\mathcal{P}$ and $\mathcal{Q}$ based on the finite sample observations. To construct a one-sided two sample test with T we must first determine a significance threshold τ such that when $\mathcal{P} = \mathcal{Q}$ we have a probability of at least $1-\alpha$ that $T(\mathbf{P},\mathbf{Q}) < \tau$ (or $T(\mathbf{P},\mathbf{Q}) > \tau$ depending on the direction of the test). To estimate τ we take advantage of the exchangability of samples under $\mathcal{H}_0$. If indeed $\mathcal{P} = \mathcal{Q}$ then it should not matter which group of the total $n+m$ samples we call $\mathbf{P}$ and which we call $\mathbf{Q}$, so the strategy is to combine, permute and redivide $\mathbf{P}$ and $\mathbf{Q}$ and feed them into T many times, computing τ as the $1-\alpha$ quantile of all observations of the test statistic. If the test performed on the original partitions exceeds τ (e.g., $T(\mathbf{P},\mathbf{Q}) \ge \tau$) we can rule out $\mathcal{H}_0$ with an exact significance level α. We can compute the power of the test at the level We note that as we are estimating τ from finite data, it itself incurs statistical estimation error which in turn adjusts the true significance level of a test [Tibshirani et al., 2019]. This is a minor detail which is ignored by several modern approaches for two sample testing that employ permutation tests [Liu et al., 2020; Zhao et al., 2022].

2.2 Approaches for High Dimensional Data

Dimensionality Reduction Approaches

Many methods for detecting shift in high-dimensional data simply apply dimensionality reduction (DR) techniques followed by standard two sample tests [Rabanser et al., 2019]. Rabanser et al. perform an evaluation on several DR techniques including, PCA, random feature projections, train and untrained auto-encoders and the output features of a classification model. They on each DR they test the performance of univariate KS-tests on each dimension of the reduced data distribution aggregated using Bonfferoni correction as well as MMD using the squared exponential kernel. Rabanser et al.'s main takeaway is that using the softmax outputs of a pretrained classifier as low dimensional representations for performing KS-tests, a method known as black box shift detection (BBSD) [Lipton et al., 2018], is effective at confidently identifying several synthetic covariate shifts in imaging data (e.g., crops, rotations) given approximately 200 i.i.d samples. However, applying statistical tests to non-invertible representations of data can never guarantee to capture arbitrary covariate shifts, as there may always exist multiple distributions that collapse to the same test statistic [Zhang et al., 2021]. For instance if considering PCA as the DR of choice an adversary could construct a new data distribution $\mathcal{Q}$ where samples are drawn by transforming samples from $\mathcal{P}$ randomly within the null space of the orthogonal projection.

Learning Theoretic Approaches

Ben-David et al. [2006] introduces the earliest theoretic framework for identifying and bounding the effect of covariate shift based on discriminative learning with finite samples. In the foundation work "Detecting Change in Data Streams", Kifer et al. introduce the concept of $\mathcal{A}$-distance as a generalization of the total variation to an arbitrary collection of measurable events $\mathcal{A}$.

$$d_{\mathcal{A}}\left(\mathcal{P}, \mathcal{Q}\right) \triangleq 2 \sup_{A \in \mathcal{A}} \left|\Pr_{\mathcal{D}}[A] - \Pr_{\mathcal{D}'}[A]\right| \tag{2.4}$$

The authors show how various choices of $\mathcal{A}$ can induce certain desired properties, for example when $\mathcal{P}$ and $\mathcal{Q}$ are distribution over the real line and $\mathcal{A}$ is the collection of all intervals $(-\infty, \cdot)$ the $\mathcal{A}$-distance becomes equivalent to the Kolmogorov-Smirnov statistic. Kifer et al. go on to prove tight bounds on various instantiations of the $\mathcal{A}$-distance for detecting shifts in an online setting.

Ben-David et al. develop a learning theoretic extension of Kifer et al. to show that when they chose a class of events whose characteristic functions are functions from a set of binary classifiers F,

the $\mathcal{A}$ distance in connection with VC theory [Vapnik, 1995] allows for finite sample generalization bounds on the performance of arbitrary decision models from F under covariate shift. Ben-David et al. go on to show that the $\mathcal{A}$ distance defined for a binary function class F is equal to

$$d_F(\mathcal{P}, \mathcal{Q}) \triangleq 2\left(1 - 2\min_{f \in F} \text{err}(f)\right) \tag{2.5}$$

where $\min_{f \in F} \text{err}(f)$ is the minimum error that a domain classifier from F can achieve on the task of distinguishing samples from $\mathcal{P}$ and $\mathcal{Q}$ (i.e., if $\mathcal{P} = \mathcal{Q}$ the best domain classifier will have error of 0.5 and $d_F(\mathcal{P}, \mathcal{Q}) = 0$ and if $\mathcal{P}$ and $\mathcal{Q}$ can be perfectly discriminated by some $f \in F$ the $d_F(\mathcal{P}, \mathcal{Q})$ is maximized and equal to 2). This conclusion motivates the classifier two sample test (CTST) [Lopez-Paz and Oquab, 2017] to detect shift by first training a domain classifier $d : \mathcal{X} \to \{0, 1\}$ on a set of samples from $\mathcal{P}$ and $\mathcal{Q}$ then running a binomial test to reject the null hypothesis that d achieves an accuracy of 0.5 on unseen data. Ben-David et al. inspired many ideas for unsupervised domain adaption including the popular method *domain adversarial domain adaptation (DANN)* [Ganin et al., 2016] which aims to learn robust representations of data for classification by explicitly regularizing for a small d_F on the representation space.

Learning Model Based Approaches

More recent approaches for covariate shift detection including, deep kernel MMD [Liu et al., 2020] and H-Divergence [Zhao et al., 2022] focus on training learning models with objectives optimized specifically for statistical testing in a way that goes beyond CTSTs.

Liu et al. show how to train a deep feature extractor $\phi : \mathcal{X} \to \mathbb{R}^d$ to optimize the power of an MMD test using a simple kernel (e.g., squared exponential) whose inputs are feature vectors tom ϕ. Zhao et al. introduce a new family of distributional divergences (H-Divergences) that builds off the concept of H-entropy [DeGroot, 1962]

$$H_\ell \triangleq \inf_{a \in \mathcal{A}} \mathbb{E}_{X \sim \mathcal{P}}[\ell(X, a)]$$

The H-entropy is defined on a distribution $\mathcal{P}$ as the optimal expected action with respect to loss function ℓ and set of possible actions $\mathcal{A}$. For example if ℓ if the reconstruction loss of an autoencoder f_θ and $\mathcal{A}$ is the set of all possible parameter configurations θ, H_ℓ corresponds to the expected reconstruction loss of $f_{\theta^\star}$ where $\theta^\star$ is the optimal choice of parameters.

Zhao et al. present a general definition of the H-Divergence using a function whose arguments are restricted to be $H_\ell\left(\frac{p+q}{2}\right) - H_\ell(p)$ and $H_\ell\left(\frac{p+q}{2}\right) - H_\ell(q)$.

In our work we take a transductive learning approach and construct a method to directly use the structure of a supervised classification problem to improve the statistical power for detecting shifts.

2.3 Out of Distribution Detection

Out of distribution (OOD) detection focuses on identifying when a specific data point x' admits low likelihood under the original training distributions ($p_{\text{train}}(x') \approx 0$)—a useful tool to have at inference time. Ren et al. [2019]; Morningstar et al. [2021] represent a broad class of work that uses density estimation to pose the identification of covariate shift as anomaly detection. However, in finite samples, density estimation for high-dimensional data can be difficult which in turn affects the accuracy of anomaly detection [Zhang et al., 2021]. Still the discipline of OOD detection has seen several recent successes, including ODIN [Liang et al., 2018], Deep Mahalanobis Detectors [Lee et al., 2018] and, Gram Matrices [Sastry and Oore, 2020] which all directly use the predictive model (e.g., information from the intermediate representations of neural networks) to create a real valued score function $\phi : X \to \mathbb{R}$ which attempts to map data points to a real number near zero if the datapoint is in the training distribution and far from zero if the datapoint is out of distribution. Such methods are largely based on heuristics on the manifold of neural networks offering little to no theoretical guarantees on detecting subtle types of covariate shifts encountered in real-world settings. Furthermore, the majority of methods in this space have been designed exclusively for deep neural networks, an uncommon modelling choice particularly for tabular data [Borisov et al., 2021].

Despite these shortcomings, OOD methods can be readily applied for the problem of covariate shift detection under the same principle that governs Integral Probability Metrics (IPMs) [Müller, 1997]; namely that if two distributions are identical, any function should have the same expectation under both distributions.

2.4 Uncertainty Estimation

Related to OOD, uncertainty estimation concerns developing models that identify sources of uncertainty in their predictions [Lakshminarayanan et al., 2017; Ovadia et al., 2019]. Naturally, uncertainty should be large when samples are OOD, however [Ovadia et al., 2019] perform a large-scale empirical comparison of uncertainty estimation methods and find that while deep ensembles generally

provide the best results, the quality of uncertainty estimations, regardless of method, consistently degrades with increasing covariate shift.

2.5 Selective Classification and PQ Learning

Selective classification concerns building classifiers that may either predict on or reject on test samples [Geifman and El-Yaniv, 2019]. Recent work by [Goldwasser et al., 2020] develops a formal framework known as PQ learning which extends probably approximately correct (PAC) learning [Haussler, 1990] to arbitrary test distributions by allowing for selective classification. While PAC learning concerns the development of a classifier with a bounded finite-sample error rate on its training distribution, PQ learning seeks a selective classifier with jointly bounded finite-sample error and rejection rates on arbitrary test distributions. The Rejectron algorithms proposed therein builds an ensemble of models that produce different outputs relative to a perfect baseline on a set of unlabeled test samples. We provide a summary of the original Rejectron algorithm in the supplementary material (see section A.1). PQ-learning represents a major theoretical leap for learning guarantees under covariate shift; however, the majority of the underlying ideas have not been implemented/tested experimentally using real-world data. We show how to build a PQ learner by generalizing the Rejectron algorithm, overcoming several limitations and assumptions made by the original work including extending beyond simple binary classification to general multiclass/multilabel tasks and reducing the number of samples required for learning at each iteration. We go on to show how a PQ learner can be used to characterize covariate shifts in real-world data.

Chapter 3 The Detectron: Identifying Harmful Covariate Shift based on Classifier Disagreement

3.1 Problem Setup

Let $f : X \to \mathbb{R}^N$ be a classifier from a function class F that maps from space of covariates X to a discrete probability distribution over classes $Y = \{1, \ldots, N\}$ i.e.,

$$f(x) = [f(x)_1, f(x)_2, \ldots, f(x)_N]^\top \text{ s.t. } f(x)_i \geq 0 \; \forall \; i \in Y \text{ and } \|f(x)\|_1 = 1$$

We assume f was trained on a dataset of **labeled** samples $\mathbf{P} = \{(x_1, y_1), \ldots, (x_{|\mathbf{P}|}, y_{|\mathbf{P}|})\}$ where each x_i is drawn identically from a distribution $\mathcal{P}$ over X, and each label $y_i \in Y$ corresponds to the ground truth label for the classification problem of interest. In deployment, f is made to predict on new **unlabeled** samples $\mathbf{Q} = \{\tilde{x}_1, \ldots, \tilde{x}_{|\mathbf{Q}|}\}$ from a distribution $\mathcal{Q}$ over X. Our high level goal is to determine whether f may be trusted to do so accurately. More specifically, the problem we address is how to automatically detect, with high statistical power, from only a *small* set of samples $\mathbf{Q}$, if the new covariate distribution $\mathcal{Q}$ has shifted from $\mathcal{P}$. In addition, we wish to detect true shifts efficiently and with a provably correct lower bound on false positive rate (the rate at which we detect shift when in reality there is no shift). The false positive rate is also referred to as the probability of making a type II error, or the significance level and is denoted by the symbol α.

Our problem will differ from a standard two sample testng scenario in several critical ways (1) we may assume access to many more samples from $\mathcal{P}$ compared to $\mathcal{Q}$; motivated by the setting where we wish to detect shift quickly, and hence from as few samples as possible (2) we have access to a robust classifier f that performs reasonably well on held out samples from $\mathcal{P}$ as well as the learning algorithm L used to train it, and (3) we seek to only detect shifts that are likely to reduce classification robustness of f — we refer to this shift characterization *harmful covariate shift* and elaborate on it in the next section.

3.2 Harmful Covariate Shift

A shift in the data distribution is not always harmful. In many practical problems, a practitioner may use domain knowledge to embed invariances with the explicit goal of ensuring the predictive performance of a classifier does not, by construction, change under certain shifts. This may be done directly via translation invariance in convolutional neural networks, permutation invariance in DeepSets [Zaheer et al., 2017] or indirectly via data augmentation or domain adaptation. Other model invariances are often learned naturally due to the structure of a given dataset and associated decision problem.

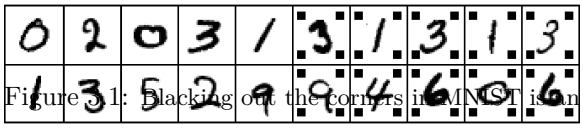

Figure 3.1: Blacking out the corners in MNIST is an example of a *non harmful* covariate shift with respect to a simple CNN that achieves near 100% accuracy on both sets. A likely explanation for why this shift is not harmful is because MNIST images contain content primarily in the center, leading to classifiers implicitly learning an invariance to the corners of the image.

For example, we train a simple CNN using the LeNet architecture on the MNIST dataset as $\mathbf{P}$ and test it on a modified version where a 5×5 pixel region at each corner of image is blacked out $\mathbf{Q}$ (see Figure 3.1). We find that when tested on a set of $10,000$ unseen examples the accuracy ($\pm$ standard error) on the original images is $98.48 \pm 0.12\%$ and $98.09 \pm 0.14\%$ on those with blacked out corners (i.e., nearly identical). A plausible explanation for this observation is that the content useful for explaining MNIST is found near the center of the image, which encourages a robust classifier to learn invariance to the content of the image near the corners. However, in terms of pure statistics one could not argue that there is no covariate shift between $\mathcal{P}$ and $\mathcal{Q}$. To support this claim we run the simplest possible two sample test where we take the mean intensity of each image in $\mathbf{P}$ and $\mathbf{Q}$ using only a sample size of $|\mathbf{P}| = |\mathbf{Q}| = 10$ and run a Kolmogorov Smirnov Test [Hodges, 1958] over $1,000$ permutations to calculate a test power of 0.981 ± 0.004 at the strongest significance level $\alpha = 0$.

The examples presented motivate the point that practical heuristics, and problem structure can lead to models generalizing to a more broad range of distributions than can be characterized by

just the training set. And furthermore, that when two sample tests are decoupled with respect to the heuristics and structures of the domain, they lose their informative abilities. To formalize this notion we define the induced *generalization set* $\mathcal{R}$ which will in general depend on the model architecture, learning algorithm and training dataset. $\mathcal{R}$ is the abstract set of distributions, beyond simply the training distribution, that a model generalizes to. While $\mathcal{R}$ is difficult if not impossible to characterize exactly, we seek a practical method for detecting shift that is explicitly tied to $\mathcal{R}$.

Our approach is based both on PQ learning and intuition from learning theory: if we can find a set of classifiers that achieve the same generalization set $\mathcal{R}$ but behave inconsistently on samples from a distribution $\mathcal{Q}$, then $\mathcal{Q}$ must not be a member $\mathcal{R}$. However, since it is not possible to know $\mathcal{R}$ for a given classifier we introduce a more practical definition of harmful covariate shift based on the learning algorithm and source dataset — whose complex interaction is what induces $\mathcal{R}$.

Definition 1 (Harmful Covariate Shift (HCS))**.** Let F be a family of decision functions that are learnable via a learning algorithm L from a set of samples drawn from a source distribution $\mathcal{P}$. We say a covariate shift from distributions $\mathcal{P} \to \mathcal{Q}$ over X is $(\ell, \alpha, L, \mathcal{P})$-harmful, if there exists a subset of models of two or more models $\mathbf{f} \subseteq F$ that achieve a source domain loss $\mathbb{E}_{\mathcal{P}}[\ell(f(x), x)] \leq \alpha$ for all $f \in \mathbf{f}$ while being more likely to disagree with each other on an unseen sample from $\mathcal{Q}$ compared to $\mathcal{P}$.

$$\begin{aligned} &\exists\ \mathbf{f} \subseteq F, \text{ s.t. } \forall f \in \mathbf{f}\ \ \mathbb{E}_{\mathcal{P}}[\ell(f(x), x)] \leq \alpha \text{ and} \\ &\qquad \mathbb{P}_{x\sim\mathcal{Q}}(\exists\ f_i, f_j \in \mathbf{f} \text{ s.t. } f_i(x) \neq f_j(x)) > \mathbb{P}_{x\sim\mathcal{P}}(\exists\ f_i, f_j \in \mathbf{f} \text{ s.t. } f_i(x) \neq f_j(x)) \end{aligned} \tag{3.1}$$

In plainer words, we define harmful covariate shift based on the existence of multiple *good* models on $\mathcal{P}$ that we assume learn the same generalization set, but that tend to disagree on $\mathcal{Q}$ with greater probability compared to $\mathcal{P}$.

3.3 Constrained Disagreement Classifiers

Our strategy for detecting HCS will be to create an ensemble of *constrained disagreement classifiers* $(g_1, \ldots, g_\aleph)$, classifiers created by the same learning algorithm as f that are constrained to predict consistently (i.e., predict the same as f) on $\mathbf{P}$ but as differently as possible on $\mathbf{Q}$. If $\mathcal{Q}$ is within $\mathcal{R}$ then such an ensemble will fail to predict differently, as the invariances induced by L and $\mathbf{P}$ will not allow a high degree of uncertainty within $\mathcal{R}$. However, when we can find an ensemble that exhibits inconsistent behaviour on $\mathcal{Q}$, there must be covariate shift that explicitly lies outside $\mathcal{R}$. To make

the idea of constrained disagreement classifiers tangible we propose a simple definition which we will translate into a learning algorithm in the following sections.

Definition 2 (Constrained Disagreement Classifier (CDC))**.** A constrained disagreement classifier $g_{(f,\mathbf{P},\mathbf{Q})}$, or simply g or $g_{\mathbf{Q}}$ if f, $\mathbf{P}$ and $\mathbf{Q}$ are clear with context, is a classifier with the following properties:

1. g belongs to the same model class as f and is updated with the same learning algorithm when it observes samples from $\mathcal{P}$;
2. g achieves similar training and held out performance on $\mathcal{P}$ with respect to f;
3. g disagrees maximally with f on elements of dataset $\mathbf{Q}$ while not violating 1 and 2.

Our definition of a CDC aims to explicitly capture the concept of a classifier that learns the same generalization region as f while behaving as inconsistently as possible on $\mathbf{Q}$.

3.4 Domain Classification and Model Capacity

In our background on covariate shift (section 2.2) we discussed the mathematical concept of $\mathcal{A}$ distance from [Kifer et al., 2004; Ben-David et al., 2006] and how it translates to the classifier two sample test [Lopez-Paz and Oquab, 2017]. Naively, the idea of training classifiers to disagree on out of domain data is identical to the simpler concept of domain classification when given models with sufficiently a large learning capacity e.g., deep neural networks. However, vanilla domain classification does not leverage the structure and invariance of the specific problem, which will in general result in a less informative test as we have motivated previously (section 3.2). In Figure 3.2 we explore in a visual toy example how the capacity of CDCs influences the sensitivity for detecting certain distribution shift.

Beyond the concept of shift harmfulness, we will see in our methods and empirical study that unlike domain classifiers, CDCs can easily leverage a pretrained model to result in significantly larger statistical power when detecting shift.

3.5 Connection to PQ Learning

As our work builds on PQ learning, we provide a summary of the original framework and clearly state the distinctions in our methodology. In PQ learning we seek a selective classifier h that achieves

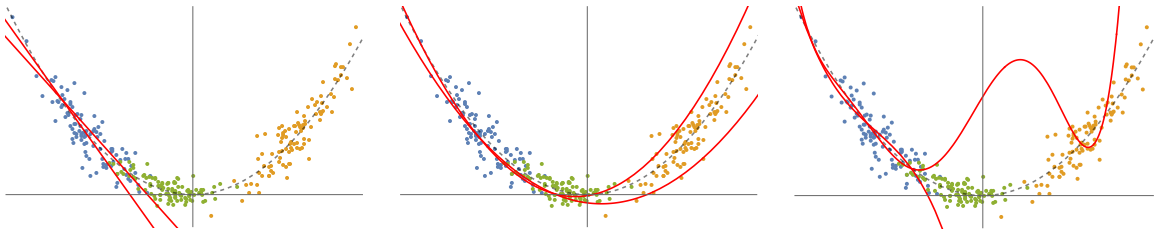

Figure 3.2: Investigating the relation between CDC model complexity and the ability to identify covariate shift. We consider a toy example where the ground truth labels are generated using a quadratic decision boundary, shown as a black dashed line. The blue points correspond to training samples, and the orange and green points correspond to two different covariate shifts, one closer to the training distribution and the other further. (Left) When we choose an underspecified model family (e.g., linear classifiers), there exists no CDC that reports different explanations for the orange and green points. However, if expert knowledge led us to believe that a linear classifier is the true causal predictor for the entire domain, we would consequently not be worried about covariate shift. (Center) When we choose a quadratic function family, there exists enough variation within the space of models that explain the training set to offer different explanations on the distance shift (orange) but not on the near shift (green). An analogy to a more complex learning problem would be that the green region represents a set of datapoints shifted from the source, yet still within the generalization/invariance set of the learning algorithms. (Right) When we learn from an overly expressive function family (polynomials of degree 3+), the space of models that explain the training set can offer different explanations of even near covariate shifts (green points).

a bounded tradeoff between its in distribution rejection rate $\text{rej}_h(\mathbf{x})$ and its out of distribution error $\text{err}_h(\tilde{\mathbf{x}})$ with respect to a ground truth decision function d. Formally, this tradeoff is defined using the following learning theoretic bound.

Definition 3 (PQ learning [Goldwasser et al., 2020])**.** Learner L (ϵ, δ, n)-PQ-learns F if for any distributions $\mathcal{P}, \mathcal{Q}$ over X and any ground truth function $d \in F$, its output $h := L(\mathbf{P}, d(\mathbf{P}), \mathbf{Q})$ satisfies

$$\mathbb{P}_{\mathbf{x}\sim\mathcal{P}^n,\ \tilde{\mathbf{x}}\sim\mathcal{Q}^n}\left[\text{rej}_h(\mathbf{x}) + \text{err}_h(\tilde{\mathbf{x}}) \leq \epsilon\right] \geq 1 - \delta \tag{3.2}$$

L PQ-learns F if L runs in polynomial time and if there is a polynomial p such that L $(\epsilon, \delta, n)-$PQ learns F for every $\epsilon, \delta > 0, n \geq p(1/\epsilon, 1/\delta)$.

Goldwasser et al. propose the Rejectron algorithm for PQ learning in a noiseless binary classification setting with zero training error and access to a perfect empirical risk minimization (ERM) oracle. Rejectron sequentially searches for new classifiers that achieve zero training error while attempting to predict the opposite label on subsets of the unlabeled set $\mathbf{Q}$. We highlight several pitfalls that prevent a practical realization of Rejectron (1) the classification problem must be binary (2) the optimization objective at each step requires an ERM query with $\Omega(|\mathbf{P}|^2)$ samples and (3) most significantly there is no process to control overfitting of large capacity models. By tackling the above issues we derive a practical PQ learning algorithm and propose a powerful hypothesis test to leverage the PQ learner to identify harmful covariate shift.

3.6 Learning to Disagree

To train a classifier to disagree in the binary setting, it suffices to flip the labels. However, in the multi-class classification, it is unclear what a good objective function is. We formulate an explicit loss function that can be minimized via gradient descent to learn a CDC. For classification problems, letting $\hat{y} \coloneqq g(x_i)$ be the predictive distribution over N classes, $\mathbf{f}(x_i) \in \{1, \dots N\}$ the label predicted by f and $[\cdot]$ a binary indicator, we define the *disagreement-cross-entropy* (DCE) $\tilde{\ell}$ as:

$$\tilde{\ell}(\hat{y}, f(x_i)) = \frac{1}{1-N} \sum_{c=1}^{N} [f(x_i) \neq c] \log(\hat{y}_c) \tag{3.3}$$

$\tilde{\ell}$ corresponds to taking the cross entropy of $\hat{y}$ with the uniform distribution over all classes except $f(x_i)$. Since the primary criteria is that $g(x_i)$ disagrees with $f(x_i)$, $\tilde{\ell}$ is designed to minimize the probability of g's prediction for the output of f's while maximizing its overall entropy. Our definition of $\tilde{\ell}$ is significantly more stable to optimize compared to simply maximizing the regular cross entropy as it has a bounded global minimum.

Elaborating on DCE. Consider a model that outputs a distribution $\{p_1, p_2, 1-p_1-p_2\}$ over 3 classes, suppose we would like to train it to **not** output high probability for class 3 i.e., minimize $p_3 = 1 - p_1 - p_2$. An intuitive approach would be to *maximize* the standard cross entropy loss $-\log(1-p_1-p_2)$ that one would equivalently *minimize* if trying to output high probability for class 3. We show in Figure 3.3 that this objective is unstable whereas the DCE $-\frac{1}{2}(\log(p_1) + \log(p_2))$ is convex, and has a bounded minimum corresponding to $p_1 = p_2 = 1/2$. In practice if one wishes a loss function with a minimum of zero it suffices to subtract $\log(N-1)$ from Equation 3.3. A visual description of the DCE is provided in Figure 3.3.

Our goal is to agree on $\mathbf{P}$ and disagree on $\mathbf{Q}$. Consequently, we learn with the loss in Equation 3.4. ℓ denotes the standard cross entropy loss and $\tilde{\ell}$ is the disagreement cross entropy. λ is a scalar parameter that controls the trade off between agreement and disagreement.

$$\mathcal{L}_g(\{x_1, \dots x_n\}) = \frac{1}{n} \left(\sum_{i=1}^{n} \ell(g(x_i), y_i)[x_i \in \mathbf{P}] + \lambda \sum_{i=1}^{n} \tilde{\ell}(g(x_i), f(x_i))[x_i \in \mathbf{Q}] \right) \tag{3.4}$$

When learning CDCs in practice, $\mathcal{L}_g$ should be combined with any additional regularization and data augmentation used in the original training process of f to ensure that we retain the true generalization region of f. Furthermore, training and validation metrics must be closely monitored on unseen samples from $\mathcal{P}$ to ensure that g achieves similar generalization performance on $\mathcal{P}$.

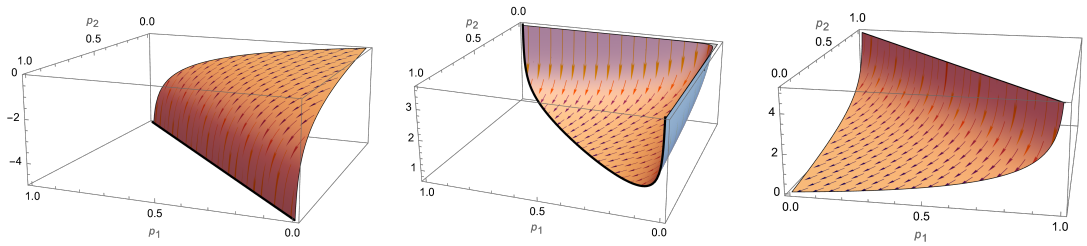

Figure 3.3: Consider a classifier outputs a distribution $\{p_1, p_2, 1-p_1-p_2\}$ over 3 classes. We compare the optimization landscape induced by either (left) minimizing the negative cross entropy for the target $p_3 = 1-p_1-p_2$ (e.g trying not to predict class 3) (center) minimizing our *disagreement cross entropy* (DCE) for the same target and (right) minimizing the regular cross entropy (e.g trying to predict class 3). We observe that naively minimizing the negative cross entropy results in an unbounded minimum, while the DCE is significantly more stable and scales similarly to the regular cross entropy. Gradients are overlaid to help better visualize the 3D geometry.

Choosing λ. In the original formulation of Rejectron, selective classifiers are trained on a dataset consisting of $\mathbf{P}$ replicated $|\mathbf{P}|$ times and $\mathbf{Q}$. Calling an ERM oracle on this data ensures that a misclassification on $\mathbf{P}$ is significantly more costly than one on $\mathbf{Q}$ but requires $\Omega(|\mathbf{P}|^2)$ samples, an impractical number for large datasets. We show that instead we can choose the scalar parameter λ in Equation 3.4 to set learning $\mathbf{P}$ as the primary learning objective and only when it cannot be improved, we allow g to learn how to disagree on $\mathbf{Q}$. The reasoning is a simple counting argument. Suppose agreeing on each sample in $\mathbf{P}$ incurs a reward of 1 and disagreeing with each sample in $\mathbf{Q}$ a reward of λ. To encourage agreement on $\mathbf{P}$ as the primary objective, we set λ such that the extra reward obtained by going from *zero* to *all* disagreements on $\mathbf{Q}$ is less than that achieved with only one extra agreement on $\mathbf{P}$, this gives $\lambda|\mathbf{Q}| < 1$. Practically, we chose $\lambda = 1/(|\mathbf{Q}| + 1)$ and find that no tuning is required.

Generalizing Beyond Cross Entropy. When training models with arbitrary discrete or generally non-differentiable with respect their objective (e.g., random forest), we must find a more general solution for creating CDCs. Such a solution should (1) reduce to the DCE when the model is, in fact, continuous and trained using the standard cross-entropy, and (2) reduces to label flipping when $N = 2$ (binary classification). Our simple solution is to replicate every sample in $\mathbf{Q}$ exactly $N-1$ times and create a unique label for each from the set $\mathcal{S} \coloneqq \{1, \ldots, N\} \backslash \{t\}$ where t is the disagreement target. We also give each a sample a weight of $1/(N-1)$. In the case of $N = 2$, this corresponds to no replication and simply assigning the opposite label. In the case where the model learns by cross-entropy, this generalization corresponds to the DCE in Equation 3.3; we provide a simple proof below.

Starting with the definition of the cross entropy

$$\mathrm{CE}(f(x), y) = -\sum_{c=1}^{N} [c = y] \log(f(x)_c) \tag{3.5}$$

Now we consider the sum of the cross entropy for each label in $\mathcal{S}$:

$$\sum_{y \in \mathcal{S}} \mathrm{CE}(f(x), y) = -\sum_{y \in \mathcal{S}} \sum_{c=1}^{N} [c = y] \log(f(x)_c) \tag{3.6}$$

$$= -\sum_{c=1}^{N} [c \in \mathcal{S}] \log(f(x)_c) \tag{3.7}$$

$$= (N-1)\mathrm{DCE}(f(x), y) \tag{3.8}$$

Hence when giving each sample a weight of $1/(N-1)$ we recover the exact from of DCE.

Ensembling. To learn richer disagreement rules, we create an ensemble of CDCs where the k^{th} model is trained only to disagree on the subset of $\mathbf{Q}$ that has yet to be disagreed on by models 1 through $k-1$. The final disagreement rate $\phi_{\mathbf{Q}}$ is the fraction of unlabelled samples where any CDC provides an alternate decision from f. In what follows we use this rate to detect shift.

3.7 Detecting Shift with Constrained Disagreement

A natural way to apply the concept of constrained disagreement to the identification of covariate shift is to partition $\mathbf{Q}$ into two sets, using the first to train a CDC ensemble and the second to compute an unbiased estimate of its held out disagreement rate $\phi_{\mathcal{Q}}$. We would statistically compare this disagreement rate using a 2×2 exact hypothesis test (e.g., Fisher's or Barnard's) against a baseline estimate for the disagreement rate on $\mathcal{P}$ computed using unseen data. The following shows that this results in a provably correct method to detect shift with high probability using only finite samples.

Theorem 2 (Disagreement implies covariate shift)**.** Let f be a classifier trained on dataset $\mathbf{P}$ consisting of N samples drawn identically from $\mathcal{P}$ and their corresponding labels. Let g be a classifier that is observed to agree (classify identically) with f on $\mathbf{P}$ and disagree on a dataset $\mathbf{Q}$ drawn from $\mathcal{Q}$. If the rate which g disagrees with f on n unseen samples from $\mathcal{Q}$ is greater than that from n unseen samples from $\mathcal{P}$ w.p. greater than $p^\star \coloneqq \frac{1}{2}\left(1 - 4^{-n}\binom{2n}{n}\right)$ there must be covariate shift.

Sketch of Proof. We show that under the null hypothesis where $\mathcal{P} = \mathcal{Q}$ the tightest upper bound

on the probability that g is more likely to disagree on $\mathcal{Q}$ compared to $\mathcal{P}$ is $p^\star$. The contrapositive argument then states if we deem the probability to be greater than $p^\star$ there must be a covariate shift. This result motivates a hypothesis testing approach to determine how probable it is that g is truly more likely to disagree on $\mathcal{Q}$ given only a set of finite observations. The full proof can be found in section A.2.

Remark. While Theorem 2 takes a frequentist's approach to identifying covariate shift, we show in section A.2 that there is a Bayesian formulation which can provide a closed-form and informative lower bound on the probability of covariate shift given an observation of finite sample disagreement rates on $\mathbf{P}$ and $\mathbf{Q}$.

Our theory, while simple, has a limitation that prevents its direct application. Any approach that requires unseen samples from $\mathbf{Q}$ is ill-suited for the low data regime, as it requires splitting $\mathbf{Q}$ leaving an even smaller set for computing the disagreement rate. Estimators from small samples result in high variance and ultimately low statistical power. Since our objective is to detect covariate shift from as few test samples as possible, splitting $\mathbf{Q}$ is not a good option. To tackle this issue practically, we take a transductive approach based on intuition from learning theory: creating learning models to disagree on samples from $\mathbf{Q}$ while generalizing to $\mathcal{R}$ is a far easier task when $\mathcal{Q}$ is not in $\mathcal{R}$. We can therefore use the *relative increase* in disagreement between CDCs on $\mathbf{Q}$ and $\mathbf{P}$ to capture a quantity that is nearly as informative as the unbiased statistic without reducing samples from $\mathbf{Q}$ that we can use.

3.8 The Detectron Test

Our proposed method is to train two CDC ensembles; $\mathbf{g_Q} = \{g_{\mathbf{Q}_1}, \dots g_{\mathbf{Q}_\aleph}\}$ and $\mathbf{g_{P^\star}} = \{g_{\mathbf{P}^\star_1}, \dots g_{\mathbf{P}^\star_\aleph}\}$. First, $g_\mathbf{Q}$ is trained to disagree with f on $\mathbf{Q}$, while still learning the original training objective. Next, enforcing the null hypothesis, we train $g_{\mathbf{P}^\star}$ to disagree on set of unseen samples $\mathbf{P}^\star$ drawn from $\mathcal{P}$ where $n \coloneqq |\mathbf{P}^\star| = |\mathbf{Q}|$. To maximize sample efficiency, we take a transductive approach to detecting shift by analyzing the outputs of $g_\mathbf{Q}$ and $g_\mathbf{P}$ on the sets $\mathbf{Q}$ and $\mathbf{P}^\star$ themselves. We define the ratio of d/n samples that $g_{\mathbf{P}^\star}$ disagrees on with respect to f as $\phi_\mathbf{P}$, and similarly for $g_\mathbf{Q}$ as $\phi_\mathbf{Q}$.

$$\phi_\mathbf{D} \coloneqq \frac{1}{|\mathbf{D}|} \sum_{x \in \mathbf{D}} (1 - [g_i(x) = g_j(x) \;\; \forall \; g_i, g_j \in \mathbf{g_D}]) \tag{3.9}$$

Note that $\phi_\mathbf{P}$ and $\phi_\mathbf{Q}$ are themselves random variables that are sampled using a deterministic the function of the CDC learning algorithm applied to randomly sampled learning sets $\mathbf{P}^\star$ and $\mathbf{Q}$.

Under the null hypothesis, if $\mathcal{Q}$ is not a harmful shift from $\mathcal{P}$, we have $\mathcal{H}_0 : \mathbb{E}[\phi_{\mathbf{Q}}] \leq \mathbb{E}[\phi_{\mathbf{P}}]$ meaning that CDCs are at least as consistent on sets $\mathbf{Q}$ compared to $\mathbf{P}^\star$. The converse of harmful shift is expressed as the one-sided alternative $\mathcal{H}_a : \mathbb{E}[\phi_{\mathbf{Q}}] > \mathbb{E}[\phi_{\mathbf{P}}]$, meaning $g_{\mathbf{Q}}$ is expected to disagree on more samples than $g_{\mathbf{P}^\star}$. See Figure 3.4 for a visual depiction of CDC training when applied to a near distribution shift benchmark.

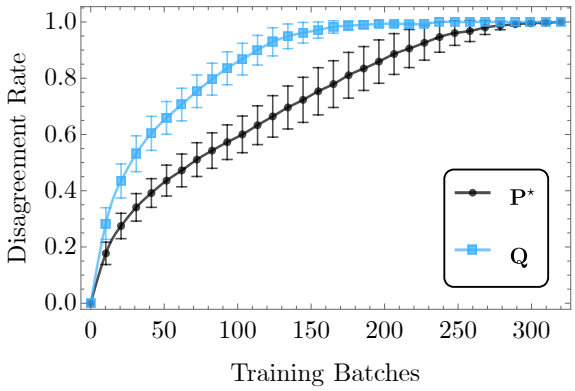

Figure 3.4: **CDC Training Dynamics:** In blue we train CDCs to disagree on a set of 100 samples from CIFAR 10.1 [Recht et al., 2019] ($\mathbf{Q}$) – a near OOD test set for CIFAR 10 – while in black we force CDCs to disagree on the original CIFAR 10 test set ($\mathbf{P}^\star$). We see that even after a small number of training batches the disagree on a significantly larger portion of CIFAR 10.1 compared to CIFAR 10

We refer to this test as *Detectron Disagreement*. To compute the test result at a significance level α we first estimate the null distribution of $\phi_{\mathbf{P}}$ for a fixed sample size n by training K calibration rounds of $g_{\mathbf{P}}$ with different random sets $\mathbf{P}^\star$ of size n. The test result is significant if the observed disagreement rate $\phi_{\mathbf{Q}}$ is greater than the $(1-\alpha)$ quantile of the null distribution.

We consider an additional variant, *Detectron Entropy* (DE), which computes the prediction entropy of each sample under the CDC instead of relying solely on disagreement rates. The intuition for DE draws from the fact that when CDCs satisfy their objective (i.e., in the case of harmful shift) they learn to predict with high entropy on $\mathbf{Q}$ and low entropy on $\mathbf{P}^\star$, resulting in a natural way to distinguish between distributions. The CDC entropy is computed from the mean probabilities over each N classes of the base classifier f and set of k CDCs $g_1, \ldots, g_k$.

$$\text{CDC}_{\text{entropy}}(x) = -\sum_{c=1}^{N} \hat{p}_c \log(\hat{p}_c) \quad \text{where} \quad \hat{p} \coloneqq \frac{1}{k+1}\left(f(x) + \sum_{i=1}^{k} g_i(x)\right) \tag{3.10}$$

We use a KS test to compute a p-value for covariate shift directly on the entropy distributions computed for $\mathbf{Q}$ and $\mathbf{P}^\star$ and guarantee significance using the same strategy as above. The intuition for *Detectron (Entropy)* draws from the fact that when CDCs satisfy their objective (i.e., in the case of harmful shift) they learn to predict with high entropy on $\mathbf{Q}$ and low entropy on $\mathbf{P}^\star$, resulting in a natural way to distinguish between distributions.

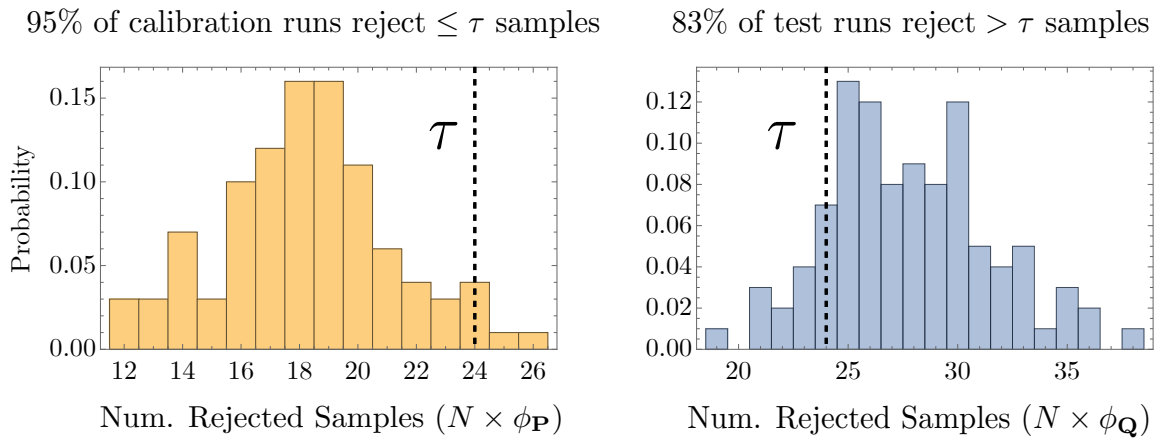

Figure 3.5: **The Detectron disagreement test:** In this example (taken from our experiment where $\mathcal{P}$ = CIFAR10 and $\mathcal{Q}$ = CIFAR10.1 and sample size $N = 50$) pictured we start by training an ensemble of CDCs (we use and ensemble size of 5) to reject/disagree on a set of N unseen samples from the original training distribution ($\mathbf{P}^\star$) while constrained to perform consistently with a base model on the original training and validation sets used to train the base model on CIFAR10. We perform 100 of these calibration runs using different random seeds and samples for $\mathbf{P}^\star$ to estimate a threshold τ such that 95% of the runs reject fewer then τ samples — thereby fixing the significance level of the test to 5%. To estimate the test power, we train CDCs using **the exact same configuration** as the calibration runs except we replace $\mathbf{P}^\star$ with a random set of N samples $\mathbf{Q}$ from $\mathcal{Q}$ (CIFAR 10.1). By averaging the number of runs the reject more than τ samples we can compute the power (or true positive rate) of the test for the configuration.

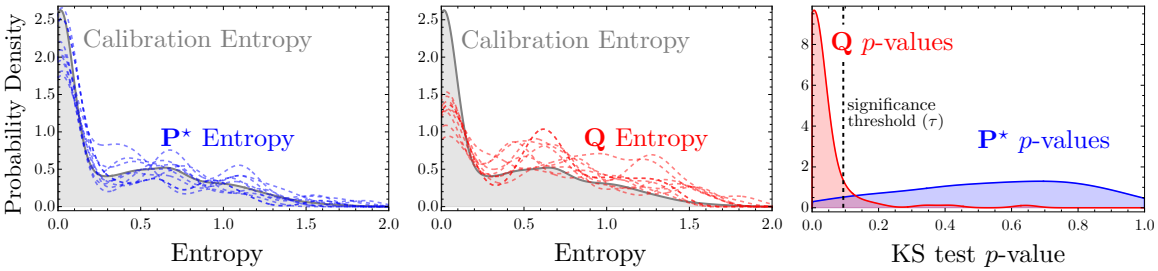

Figure 3.6: **The Detectron entropy test:** Following the same experimental setup as Figure 3.5, we start (left) by computing a KS test between the continuous entropy values for each calibration run $\mathbf{P}^\star$ with the flattened set of entropy values from all other 99 calibration runs. Then (center) we compute a KS test from each test run $\mathbf{Q}$ with a random set of all but one calibration runs. Finally (right), we find a threshold τ on the distribution of p-values obtained from step 1 as the α quantile to guarantee a false positive rate of α. The power of the test is computed as the fraction of p-values computed from 100 test runs $\mathbf{Q}$ that are below τ.

Algorithm 1: The Detectron algorithm for detecting harmful covariate shift

Input: $\mathbf{P}$: labeled dataset, $\mathbf{Q}$: unlabeled dataset, L: learning algorithm,
K: calibration rounds $= 100$, $\aleph$: ensemble size $= 5$, α: significance level $= 0.05$
Output: test result for harmful covariate shift at significance level α

$\mathbf{P}_{\text{train}},\ \mathbf{P}_{\text{val}},\ \mathbf{P}^{\star} \leftarrow \text{Partition}(\mathbf{P})$
$N \leftarrow |\mathbf{Q}|$; $\Phi_{\mathbf{P}} \leftarrow [\,]$
$f \leftarrow L(\mathbf{P}_{\text{train}},\ \mathbf{P}_{\text{val}})$ `// Load or train a base classifier on P`
repeat
  $\mathbf{p}^{\star} \leftarrow \text{RandomSample}(\mathbf{P}^{\star}, N)$
  `// Train an ensemble of CDCs on P*`
  while $n > 0$ *and iterations* $\leq \aleph$ **do**
    $g \leftarrow \text{ConstrainedDisagreement}(L, \mathbf{P}_{\text{train}}, \mathbf{P}_{\text{val}}, \mathbf{p}^{\star}, f)$ `// See Appendix TODO`
    $\mathbf{p}^{\star} \leftarrow \{x \mid x \in \mathbf{p}^{\star} \text{ and } f(x) = g(x)\}$ `// Filter out disagreed on data`
    $\phi_{\mathbf{P}} \leftarrow 1 - |\mathbf{p}^{\star}|/N$ `// Update disagreement rate`
  end
  Append $\phi_{\mathbf{P}}$ to $\Phi_{\mathbf{P}}$
until K *iterations elapse*
`// Train an ensemble of CDCs on Q`
while $n > 0$ *and iterations* $\leq \aleph$ **do**
  $g \leftarrow \text{ConstrainedDisagreement}(L, \mathbf{P}_{\text{train}}, \mathbf{P}_{\text{val}}, \mathbf{Q}, f)$
  $\mathbf{Q} \leftarrow \{x \mid x \in \mathbf{Q} \text{ and } f(x) = g(x)\}$
  $\phi_{\mathbf{Q}} \leftarrow 1 - |\mathbf{Q}|/N$
end
return $\phi_{\mathbf{Q}} > [(1-\alpha)$ *quantile of* $\Phi_{\mathbf{P}}]$

Chapter 4 Applications and Experiments

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetuer eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetuer tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo.

Chapter 5 Discussion

Aliquam lectus. Vivamus leo. Quisque ornare tellus ullamcorper nulla. Mauris porttitor pharetra tortor. Sed fringilla justo sed mauris. Mauris tellus. Sed non leo. Nullam elementum, magna in cursus sodales, augue est scelerisque sapien, venenatis congue nulla arcu et pede. Ut suscipit enim vel sapien. Donec congue. Maecenas urna mi, suscipit in, placerat ut, vestibulum ut, massa. Fusce ultrices nulla et nisl.

Etiam ac leo a risus tristique nonummy. Donec dignissim tincidunt nulla. Vestibulum rhoncus molestie odio. Sed lobortis, justo et pretium lobortis, mauris turpis condimentum augue, nec ultricies nibh arcu pretium enim. Nunc purus neque, placerat id, imperdiet sed, pellentesque nec, nisl. Vestibulum imperdiet neque non sem accumsan laoreet. In hac habitasse platea dictumst. Etiam condimentum facilisis libero. Suspendisse in elit quis nisl aliquam dapibus. Pellentesque auctor sapien. Sed egestas sapien nec lectus. Pellentesque vel dui vel neque bibendum viverra. Aliquam porttitor nisl nec pede. Proin mattis libero vel turpis. Donec rutrum mauris et libero. Proin euismod porta felis. Nam lobortis, metus quis elementum commodo, nunc lectus elementum mauris, eget vulputate ligula tellus eu neque. Vivamus eu dolor.

Chapter 6 Conclusion

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

Nulla mattis luctus nulla. Duis commodo velit at leo. Aliquam vulputate magna et leo. Nam vestibulum ullamcorper leo. Vestibulum condimentum rutrum mauris. Donec id mauris. Morbi molestie justo et pede. Vivamus eget turpis sed nisl cursus tempor. Curabitur mollis sapien condimentum nunc. In wisi nisl, malesuada at, dignissim sit amet, lobortis in, odio. Aenean consequat arcu a ante. Pellentesque porta elit sit amet orci. Etiam at turpis nec elit ultricies imperdiet. Nulla facilisi. In hac habitasse platea dictumst. Suspendisse viverra aliquam risus. Nullam pede justo, molestie nonummy, scelerisque eu, facilisis vel, arcu.

Bibliography

S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira. Analysis of representations for domain adaptation. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006. URL https://proceedings.neurips.cc/paper/2006/file/b1b0432ceafb0ce714426e9114852ac7-Paper.pdf.

S. Bickel, M. Brückner, and T. Scheffer. Discriminative learning under covariate shift. *Journal of Machine Learning Research*, 10(75):2137–2155, 2009. URL http://jmlr.org/papers/v10/bickel09a.html.

V. Borisov, T. Leemann, K. Seßler, J. Haug, M. Pawelczyk, and G. Kasneci. Deep neural networks and tabular data: A survey. *CoRR*, abs/2110.01889, 2021. URL https://arxiv.org/abs/2110.01889.

J. Dastin. Amazon scraps secret ai recruiting tool that showed bias against women. https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G, 2018. (Accessed on 05/11/2022).

M. H. DeGroot. Uncertainty, Information, and Sequential Experiments. *The Annals of Mathematical Statistics*, 33(2):404 – 419, 1962.

J. H. Drew, A. G. Glen, and L. M. Leemis. Computing the cumulative distribution function of the kolmogorov–smirnov statistic. *Computational statistics & data analysis*, 34(1):1–15, 2000.

K. Erguler. Barnard.pdf. https://cran.r-project.org/web/packages/Barnard/Barnard.pdf, 10 2016. (Accessed on 05/24/2022).

R. A. Fisher. On the Interpretation of 2 from Contingency Tables, and the Calculation of P, Jan. 1922. URL https://doi.org/10.2307/2340521.

R. A. Fisher. *The design of Experiments*. 1935.

Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

Y. Geifman and R. El-Yaniv. SelectiveNet: A deep neural network with an integrated reject option. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2151–2159. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/geifman19a.html.

V. Glivenko. Sulla determinazione empirica delle leggi di probabilita. *Gion. Ist. Ital. Attauri.*, 4: 92–99, 1933.

S. Goldwasser, A. T. Kalai, Y. Kalai, and O. Montasser. Beyond Perturbations: Learning Guarantees with Arbitrary Adversarial Test Examples. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 15859–15870. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/b6c8cf4c587f2ead0c08955ee6e2502b-Paper.pdf.

A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

A. R. Habib, A. L. Lin, and R. W. Grant. The Epic Sepsis Model Falls Short—The Importance of External Validation. *JAMA Internal Medicine*, 181(8):1040–1041, 08 2021.

K. Hao. Ai is sending people to jail—and getting it wrong. https://www.technologyreview.com/2019/01/21/137783/algorithms-criminal-justice-ai/, 2019. (Accessed on 05/11/2022).

D. Haussler. Probably approximately correct learning. In *Proceedings of the Eighth National Conference on Artificial Intelligence*, pages 1101–1108. AAAI Press, 1990.

D. Hendrycks and T. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=HJz6tiCqYm.

J. L. Hodges. The significance probability of the smirnov two-sample test. *Arkiv för Matematik*, 3 (5):469–486, 1958. doi: 10.1007/BF02589501. URL https://doi.org/10.1007/BF02589501.

D. Kifer, S. Ben-David, and J. Gehrke. Detecting change in data streams. In *Proceedings of the Thirtieth International Conference on Very Large Data Bases - Volume 30*, VLDB '04, page 180–191. VLDB Endowment, 2004. ISBN 0120884690.

A. Kolmogorov. Sulla determinazione empirica di una lgge di distribuzione. *Inst. Ital. Attuari, Giorn.*, 4:83–91, 1933.

B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6405–6416, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

K. Lee, K. Lee, H. Lee, and J. Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

S. Liang, Y. Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=H1VGkIxRZ.

Z. Lipton, Y.-X. Wang, and A. Smola. Detecting and correcting for label shift with black box predictors. In *International conference on machine learning*, pages 3122–3130. PMLR, 2018.

F. Liu, W. Xu, J. Lu, G. Zhang, A. Gretton, and D. J. Sutherland. Learning deep kernels for non-parametric two-sample tests. In *International conference on machine learning*, pages 6316–6326. PMLR, 2020.

D. Lopez-Paz and M. Oquab. Revisiting classifier two-sample tests. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=SJkXfE5xx.

W. Morningstar, C. Ham, A. Gallagher, B. Lakshminarayanan, A. Alemi, and J. Dillon. Density of states estimation for out of distribution detection. 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 3232–3240. PMLR, 13–15 Apr 2021. URL https://proceedings.mlr.press/v130/morningstar21a.html.

A. Müller. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 29(2):429–443, 1997. ISSN 00018678. URL http://www.jstor.org/stable/1428011.

T. Needham. *Complex Integration: Cauchy's Theorem*, page 377–417. Oxford University Press, 2000.

E. Otles, J. Oh, B. Li, M. Bochinski, H. Joo, J. Ortwine, E. Shenoy, L. Washer, V. B. Young, K. Rao, et al. Mind the performance gap: examining dataset shift during prospective validation. In *Machine Learning for Healthcare Conference*, pages 506–534. PMLR, 2021.

Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. Dillon, B. Lakshminarayanan, and J. Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

S. Rabanser, S. Günnemann, and Z. C. Lipton. Failing loudly: An empirical study of methods for detecting dataset shift. In H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 1394–1406, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/846c260d715e5b854ffad5f70a516c88-Abstract.html.

B. Recht, R. Roelofs, L. Schmidt, and V. Shankar. Do ImageNet classifiers generalize to ImageNet? In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5389–5400. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/recht19a.html.

J. Ren, P. J. Liu, E. Fertig, J. Snoek, R. Poplin, M. Depristo, J. Dillon, and B. Lakshminarayanan. Likelihood ratios for out-of-distribution detection. *Advances in neural information processing systems*, 32, 2019.

C. S. Sastry and S. Oore. Detecting out-of-distribution examples with Gram matrices. 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 8491–8501. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/sastry20a.html.

A. Schrab, I. Kim, M. Albert, B. Laurent, B. Guedj, and A. Gretton. Mmd aggregated two-sample test, 2021. URL https://arxiv.org/abs/2110.15073.

L. Smiley. 'i'm the operator': The aftermath of a self-driving tragedy. https://www.wired.com/story/uber-self-driving-car-fatal-crash/, 2022. (Accessed on 05/11/2022).

B. K. Sriperumbudur, A. Gretton, K. Fukumizu, G. R. G. Lanckriet, and B. Schölkopf. A note on integral probability metrics and ϕ-divergences. *CoRR*, abs/0901.2698, 2009. URL http://arxiv.org/abs/0901.2698.

M. Sugiyama and M. Kawanabe. *Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation*. The MIT Press, 2012. ISBN 0262017091.

R. J. Tibshirani, R. Foygel Barber, E. Candes, and A. Ramdas. Conformal prediction under covariate shift. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

N. G. Ushakov. *Continuity theorems and inversion formulas*. De Gruyter, 2011.

V. N. Vapnik. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., 1995. ISBN 0-387-94559-8.

M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola. Deep sets. *Advances in neural information processing systems*, 30, 2017.

L. Zhang, M. Goldstein, and R. Ranganath. Understanding failures in out-of-distribution detection with deep generative models. In *International Conference on Machine Learning*, pages 12427–12436. PMLR, 2021.

S. Zhao, A. Sinha, Y. He, A. Perreault, J. Song, and S. Ermon. Comparing distributions by measuring differences that affect decision making. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=KB5onONJIAU.

Appendix A Appendix

A.1 Rejectron

We provide a summary of the original Rejectron algorithm for PQ learning [Goldwasser et al., 2020] as it is the primary motivation for our work. Rejectron (algorithm 2) takes as input a labeled training set of n samples $\mathbf{x}$ (iid over $\mathcal{P}$), an unlabeled test set of n samples $\tilde{\mathbf{x}}$ (iid over $\mathcal{Q}$), an error ϵ and a weight Λ. The output is a selective classifier, that predicts according to a base classifier h if the input x is inside some set S and otherwise rejects (abstains from predicting).

$$h|_S(x) := \begin{cases} h(x) & x \in S \\ \text{reject} & x \notin S \end{cases} \tag{A.1}$$

Under several assumptions and a special value $\epsilon^\star$, this selective classifier is guaranteed with high probability to have error less then $2\epsilon^\star$ on $\tilde{\mathbf{x}}$ and a rejection rate below $\epsilon^\star$ on $\mathbf{x}$ (see Theorem 5.7 in Goldwasser et al.).

Algorithm 2: Rejectron [Goldwasser et al., 2020]

Input: train $\mathbf{x} \in X^n$, labels $\mathbf{y} \in Y^n$, test $\tilde{\mathbf{x}} \in X^n$, error $\epsilon \in [0,1]$, weight $\Lambda = n+1$
Output: selective classifier $h|_S$
$h \leftarrow \text{ERM}(\mathbf{x}, \mathbf{y})$ `# assume black box oracle ERM to minimize errors`
for $t = 1, 2, 3, \ldots$ **do**
1. $S_t := \{x \in X : h(x) = c_1(x) = \ldots = c_{t-1}(x)\}$ `# So` $S_1 = X$
2. Choose $c_t \in C$ to maximize $s_t(c) := \text{err}_{\tilde{\mathbf{x}}}\left(h|_{S_t}, c\right) - \Lambda \cdot \text{err}_{\mathbf{x}}(h, c)$ over $c \in C$
3. If $s_t(c_t) \leq \epsilon$, then stop and return $h|_{S_t}$

Rejectron starts by querying an empirical risk minimization (ERM) oracle that uses a 0–1 risk score over a concept class C for a model h that perfectly learns the training dataset. A primary assumption for Rejectron to output a perfect model as well as for it to eventually find a selective classifier that meets the $\epsilon^\star$ bound is that the true decision function (i.e., the function that creates the training labels) is also a member of C. The authors refer to this setting as *realizable*.

On the first iteration of the algorithm, Rejectron finds another model $c_1 \in C$ that jointly maximizes the error with respect to h on $\tilde{\mathbf{x}}$ while minimizing it on $\mathbf{x}$. The authors show that they can efficiently solve this optimization problem using a single ERM query on a dataset of $n^2 + n$ samples (see Lemma 5.1 in Goldwasser et al.).

In every subsequent step $t > 1$ a set S_t is created where all models h through to c_{t-1} agree. An-

other model $c_t \in C$ is found that maximizes the same objective as above but only on the intersection of $\tilde{\mathbf{x}}$ and S_t. Upon termination a selective classifier $h|_{S_t}$ is output.

A.2 Proofs

Theorem 1 (Disagreement implies covariate shift)**.** Let f be a classifier trained on dataset $\mathbf{P}$ consisting of N samples drawn identically from $\mathcal{P}$ and their corresponding labels. Let g be a classifier that is observed to agree (classify identically) with f on $\mathbf{P}$ but and disagree on a dataset $\mathbf{Q}$ drawn from $\mathcal{Q}$. If the rate which g disagrees with f on n unseen samples from $\mathcal{Q}$ is greater then that from n unseen samples from $\mathcal{P}$ w.p greater than $p^\star \coloneqq \frac{1}{2}\left(1 - 4^{-n}\binom{2n}{n}\right)$ there must be covariate shift.

Proof. Let $R_P = \emptyset_1^P + \ldots + \emptyset_n^P$ where each $\emptyset_i^P$ is an i.i.d Bernoulli random variable that describes the probability of g disagreeing with f on an unseen sample from $\mathcal{P}$. Additionally, let $R_Q = \emptyset_1^Q + \ldots + \emptyset_n^Q$ be defined similarly for $\mathcal{Q}$. If $\mathcal{P}$ and $\mathcal{Q}$ are equal then $\emptyset_i^P$ and $\emptyset_i^Q$ are equal by definition and the probability of observing $R_Q > R_P$ is tightly bounded by Equation A.2. This is the tightest upper bound that is not a function of $\mathbb{E}[\emptyset_i^P]$, the proof can be found in Lemma 1.

$$\mathcal{P} = \mathcal{Q} \implies \mathbb{P}\left(R_Q > R_P\right) \leq \frac{1}{2}\left(1 - 4^{-n}\binom{2n}{n}\right) = \frac{1}{2} - O\left(\frac{1}{\sqrt{n}}\right) \tag{A.2}$$

The more helpful contrapositive statement says that if it is sufficiently likely that $R_Q > R_P$, then covariate distributions $\mathcal{P}$ and $\mathcal{Q}$ must not be equal.

$$\mathbb{P}\left(R_Q > R_P\right) > \frac{1}{2}\left(1 - 4^{-n}\binom{2n}{n}\right) \implies \mathcal{P} \neq \mathcal{Q} \tag{A.3}$$

This result naturally lends itself to identifying $\mathcal{P} \neq \mathcal{Q}$ by rejecting an exact statistical hypothesis that $R_Q = R_P$ in favor of the alternative $R_Q > R_P$. □

Lemma 1. Let X and Y be iid binomial random variables with distribution $\mathrm{Bin}(n, p)$ then for all $n \in \mathbb{Z} \geq 0$:

$$P(X > Y) \leq \frac{1}{2}\left(1 - 4^{-n}\binom{2n}{n}\right) < \frac{1}{2} \tag{A.4}$$

Furthermore, eq. (A.4) is the tightest possible bound that does not depend on p.

Proof. Let Z be the distribution $X - Y$, while Z itself is intractable to write down for arbitrary n,

the characteristic function takes a convenient form

$$\phi_Z(t;p) = \mathbb{E}\left[e^{it(x-y)}\right] = \mathbb{E}\left[e^{itx}\right]\mathbb{E}\left[e^{-ity}\right] \tag{A.5}$$

$$= \left(1+p\left(-1+e^{it}\right)\right)^n\left(1+p\left(-1+e^{-it}\right)\right)^n \tag{A.6}$$

$$= \left(-p^2e^{-it} - p^2e^{it} + 2p^2 + pe^{-it} + pe^{it} - 2p + 1\right)^n \tag{A.7}$$

$$= (1 - 2p + p\cos(t) - ip\sin(t) + p\cos(t) + ip\sin(t) \tag{A.8}$$

$$+ 2p^2 - p^2\cos(t) + ip^2\sin(t) - p^2\cos(t) - ip^2\sin(t))^n$$

$$= \left(p^2(2-2\cos(t)) + p(2\cos(t)-2) + 1\right)^n \tag{A.9}$$

Since X and Y are identically distributed $P(Z>0) = P(Z<0)$ and so

$$P(Z>0) = \frac{1}{2}\left(1 - P(Z=0)\right) \tag{A.10}$$

Equation (A.10) suggests that a tight upper bound of the form $P(Z=0) \geq \alpha$ implies a tight lower bound in the form $P(Z>0) \leq (1-\alpha)/2$. To bound $P(Z=0)$ we first write it as an integral expression using the characteristic inversion formula for discrete random variables [Ushakov, 2011]

$$P(Z=0) = \frac{1}{2\pi}\int_{-\pi}^{\pi}\phi_Z(t;p)dt \tag{A.11}$$

Since $\phi_Z(t;p)$ has the form $(a(t)p^2 + b(t)p + 1)^n$ where a and b are real valued functions and $a(t) \geq 0\ \forall t \in \mathbb{R}$ (i.e an integer power of a quadratic equation with positive leading coefficient), then for any choice of t, $\phi_Z(t;p)$ will be globally minimized if and only if $p \to p^\star = -b(t)/\left(2a(t)\right)$. For the particular form of $\phi_Z(t;p)$, $p^\star$ is simply $1/2$

$$p^\star = -\frac{b(t)}{2a(t)} = -\frac{2\cos(t)-2}{2(2-2\cos(t))} = \frac{1}{2} \tag{A.12}$$

This result is intuitive as the variance of a binomial distribution $\text{Bin}(n,p)$ is maximized for any fixed choice of n when $p = 1/2$. We should note that Equation A.12 appears problematic when $\cos(t) = 1$, but in this case $\phi(t;p)$ becomes constant, hence p cannot influence the upper bound. We can now

write the upper bound for $P(Z=0)$

$$P(Z=0) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \phi_Z(t;p)dt \tag{A.13}$$
$$\geq \frac{1}{2\pi}\int_{-\pi}^{\pi} \phi_Z\left(t;\frac{1}{2}\right)dt \tag{A.14}$$
$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} 2^{-n}(\cos(t)+1)^n dt \tag{A.15}$$
$$= 4^{-n}\binom{2n}{n} \tag{A.16}$$

The final expression in eq. (A.16) can be found using the change of variables $z = e^{it}$ and Cauchy's residue theorem [Needham, 2000]

$$I = \frac{1}{2\pi}\int_{-\pi}^{\pi} 2^{-n}(\cos(t)+1)^n dt \tag{A.17}$$
$$= -\frac{i}{2\pi}\oint_{|z|=1} 2^{-n}\frac{1}{z}\left(1+\frac{1+z^2}{2z}\right)^n dz \text{ (using } t \to -i\log z) \tag{A.18}$$
$$= -\frac{i}{2\pi}\oint_{|z|=1} 4^{-n}z^{-n-1}(z+1)^{2n}dz \text{ (simplifying)} \tag{A.19}$$
$$= \text{Res}\left(4^{-n}z^{-n-1}(z+1)^{2n}, 0\right) \quad \text{(applying Cauchy's Theorem)} \tag{A.20}$$
$$= \frac{1}{4^n n!}\lim_{z\to 0}\left(\frac{d^n}{dz^n}(z+1)^{2n}\right) \tag{A.21}$$
$$= \frac{1}{4^n n!}2n(2n-1)(2n-2)\dots(n+1) \tag{A.22}$$
$$= 4^{-n}\binom{2n}{n} \tag{A.23}$$

Combining eq. (A.10) with the bound from eq. (A.16) we arrive at the conjectured upperbound

$$P(X>Y) = P(Z>0) = \frac{1}{2}\left(1-P(Z=0)\right) \tag{A.24}$$
$$\leq \frac{1}{2}\left(1-4^{-n}\binom{2n}{n}\right) \text{ by eq. (A.16)} \tag{A.25}$$
$$\tag{A.26}$$

Finally we may use Sterling's approximation to show that $4^{-n}\binom{2n}{n} \in O\left(\frac{1}{\sqrt{n}}\right)$ and hence converges to 0 as $n \to \infty$ leaving a limiting tight upper bound of 1/2. □

Theorem 3 (Probability of Disagreement: A Bayesian Perspective)**.** Let f be a classifier trained on dataset $\mathbf{P}$ drawn from the distribution $\mathcal{P}$ over X and their corresponding labels. Let g be a classifier observed to agree with f on $\mathbf{P}$ but disagree on a dataset $\mathbf{Q}$ drawn from a distribution $\mathcal{Q}$

over X. We denote the true probabilities that g will disagree with f on a sample from $\mathcal{P}$ and $\mathcal{Q}$ as p and q, respectively. Under a uniform prior $\mathcal{U}(0,1)$ for p and q, if we observe that g disagrees with f on m out of M iid samples from $\mathcal{Q}$ while disagreeing with n out of N iid samples from $\mathcal{P}$, then the posterior probability that g is truly more likely to disagree with f on $\mathcal{Q}$ compared to $\mathcal{P}$:

$$\begin{aligned}\mathbb{P}[q > p] =& 1 - \frac{(M+1)!(N+1)!(m+n+1)!}{(m+1)!n!(M-m)!(m+N+2)!}\times \\ & {}_3F_2(m+1, m-M, m+n+2; m+2, m+N+3; 1)\end{aligned} \tag{A.27}$$

Where ${}_pF_q$ is the generalized hyper-geometric function, implemented in several standard mathematical libraries.

Proof. For simplicity we consider the function $\text{dis} : X \to \{0,1\}$ that outputs 0 if $f(x) = g(x)$ else 1. We define the true disagreement rates $\mathbf{p}$ and $\mathbf{q}$ as

$$\mathbf{p} \coloneqq \mathbb{E}_{x\sim\mathcal{P}}[\text{dis}(x)] \text{ and } \mathbf{q} \coloneqq \mathbb{E}_{x\sim\mathcal{Q}}[\text{dis}(x)] \tag{A.28}$$

Without any *a-priori* knowledge of dis we define the random variables p and q under uniform prior (i.e $p, q \overset{\text{i.i.d}}{\sim} \mathcal{U}(0,1)$) to encode our belief over the true values $\mathbf{p}$ and $\mathbf{q}$. Now we draw N samples as $\mathbf{x} \overset{\text{i.i.d}}{\sim} \mathcal{P}^N$ and M as $\tilde{\mathbf{x}} \overset{\text{i.i.d}}{\sim} \mathcal{Q}^M$ and compute the number of times $\text{dis}(x)$ equals 1 on each set.

$$n \coloneqq \sum_{x\in\mathbf{x}} \text{dis}(x) \text{ and } m \coloneqq \sum_{x\in\tilde{\mathbf{x}}} \text{dis}(x) \tag{A.29}$$

By definition in Equation A.28 we know that n and m are draws from binomial distributions: $\mathcal{N} \sim \text{Bin}(N, p)$ and $\mathcal{M} \sim \text{Bin}(M, q)$ respectively. We can then compute the posterior probability density functions of p and q conditioned on the observations $\mathcal{N} = n$ and $\mathcal{M} = m$ using exact

Bayesian inference.

$$f_{p|n}(x) \coloneqq \mathbb{P}[p = x|\mathcal{N} = n] \tag{A.30}$$

$$= \mathbb{P}[\mathcal{N} = n|p = x] \underbrace{\mathbb{P}[p = x]}_{=1} \left(\int_0^1 \mathbb{P}[\mathcal{N} = n|p = x]dx \right)^{-1} \tag{A.31}$$

$$= \binom{N}{n} x^n (1-x)^{N-n} \left(\int_0^1 \binom{N}{n} x^n (1-x)^{N-n} dx \right)^{-1} \tag{A.32}$$

$$= x^n (1-x)^{N-n} \left(\underbrace{B_x(n+1, -n+N+1)}_{\text{incomplete beta function}} |_{x=0}^{x=1} \right)^{-1} \tag{A.33}$$

$$= x^n (1-x)^{N-n} \left(\underbrace{\frac{n!(N-n)!}{(N+1)N!}}_{x=1} - \underbrace{0}_{x=0} \right)^{-1} \tag{A.34}$$

$$= \binom{N}{n} x^n (1-x)^{N-n} (N+1) \tag{A.35}$$

The integration in Equation A.32 is solved using the definition of the incomplete beta function.

Without loss of generality we may also find $f_{q|m}$

$$f_{q|m}(x) = (M+1)\binom{M}{m} x^m (1-x)^{M-m} \tag{A.36}$$

Given these closed form posterior distributions for p and q we may compute the probability that the true value of q is greater then p

$$\mathbb{P}[q > p|\mathcal{N} = n, \mathcal{M} = m] = \int_{y>x} f_{q|m}(y) f_{p|n}(x) dydx \tag{A.37}$$

$$= \int_0^1 \int_x^1 f_{q|m}(y) f_{p|n}(x) dydx \tag{A.38}$$

$$= \binom{M}{m}\binom{N}{n}(1+M)(1+N) \int_0^1 (1-x)^{-n+N} x^n \int_x^1 (1-y)^{-m+M} y^m dydx \tag{A.39}$$

This integral, while daunting, can easily be solved in closed form using the free online Wolfram Mathematica cloud (result link). The solution is exactly Equation A.27.

$$\mathbb{P}[q > p|\mathcal{N} = n, \mathcal{M} = m] = 1 - \frac{(M+1)!(N+1)!(m+n+1)!}{(m+1)!n!(M-m)!(m+N+2)!} \times \tag{A.40}$$

$${}_3F_2(m+1, m-M, m+n+2; m+2, m+N+3; 1)$$

To gain intuition, a graphical representation of Equation A.40 is provided in Figure A.1. From a

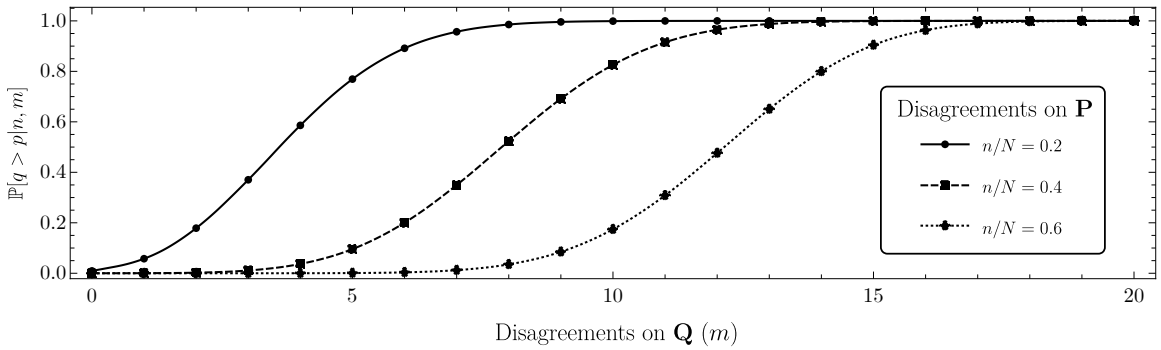

Figure A.1: Belief that the probability q that two classifiers disagree on samples from $\mathcal{Q}$ is greater then the probability p that they disagree on $\mathcal{P}$ given an observation of m disagreements out of M samples from $\mathcal{Q}$ and n of N disagreements on $\mathcal{P}$. We plot this probability for $M = 20$ and $N = 10000$. We observe that even for a small test set size, $M = 20$, we can strongly believe that there is a true difference when m/M is only slightly larger then n/N.

practical standpoint, if a practitioner trains two classifiers f and g that appear to disagree more often on a new dataset than a baseline rate computed on an in-domain test set, they can decide to act on that observation. (e.g., collected more training data) at a particular belief threshold (e.g., probability greater than 80%). Furthermore, there is a natural link between Equation A.40 and covariate shift as exact knowledge of $\mathbf{q} > \mathbf{p}$ by definition implies not only covariate shift $\mathcal{P} \neq \mathcal{Q}$, but a type that is harmful by our original definition in TODO. Therefore, knowing the probability that $q > p$ is a useful measure of how likely, without any additional assumptions, that we are experiencing a harmful covariate shift.

□