

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

© Copyright 2023 by Tom Ginsberg

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. 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.

We release our code as open source software under a GNU General Public License v3.0.



<https://github.com/tomginsberg/deeptst>

## **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 Erdogan 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, Asic Chen and Stephan Rabanser.

# Contents

<b>1</b>	<b>Introduction</b>	<b>12</b>
<b>2</b>	<b>Background: Covariate Shift</b>	<b>14</b>
2.1	Two Sample Statistical Testing . . . . .	14
2.2	Approaches for High Dimensional Data . . . . .	19
2.3	Out of Distribution Detection . . . . .	21
2.4	Uncertainty Estimation . . . . .	22
2.5	Selective Classification and PQ Learning . . . . .	22
<b>3</b>	<b>Methodology: The Detectron</b>	<b>24</b>
3.1	Problem Setup . . . . .	24
3.2	Harmful Covariate Shift . . . . .	24
3.3	Constrained Disagreement Classifiers . . . . .	26
3.4	Domain Classification and Model Capacity . . . . .	27
3.5	Connection to PQ Learning . . . . .	28
3.6	Learning to Disagree . . . . .	29
3.6.1	Choosing $\lambda$ . . . . .	30
3.6.2	Generalizing Beyond Cross Entropy . . . . .	30
3.6.3	Ensembling . . . . .	31
3.7	Detecting Shift with Constrained Disagreement . . . . .	31
3.8	The Detectron Test . . . . .	32
3.8.1	Detectron Disagreement . . . . .	32
3.8.2	Detectron Entropy . . . . .	33
3.8.3	Detectron Algorithm . . . . .	35

<b>4 Applications and Experiments</b>	<b>36</b>
4.1 Datasets . . . . .	36
4.2 Learning Constrained Disagreement . . . . .	36
4.3 Shift Detection Approach . . . . .	38
4.3.1 Evaluation . . . . .	38
4.3.2 Baselines . . . . .	38
4.4 Shift Detection Experiments . . . . .	39
4.4.1 Increasing the Sample Size . . . . .	40
<b>5 Discussion</b>	<b>41</b>
<b>6 Conclusion, Limitations and Future Work</b>	<b>43</b>
6.1 Conclusion . . . . .	43
6.2 Ethics Statement . . . . .	44
<b>A Appendix</b>	<b>52</b>
A.1 PQ Learning & Rejectron . . . . .	52
A.2 Proofs . . . . .	53
A.3 Constrained Disagreement Loss Details . . . . .	58
A.3.1 Disagreement Cross Entropy . . . . .	58
A.3.2 Disagreement with Overparameterized Models . . . . .	61
A.4 Experimental Details . . . . .	63
A.4.1 Base Classifiers . . . . .	63
A.4.2 Constrained Disagreement Classifiers . . . . .	64
A.4.3 Description of Baseline Methods . . . . .	65
A.5 Datasets . . . . .	66
A.5.1 Sources and Licensees . . . . .	66
A.5.2 Preprocessing, Shift Descriptions and Model Performance . . . . .	67

# List of Tables

4.1	<b>Datasets:</b> We investigate three different forms of covariate shift in two unique data modalities. To verify that these shifts are indeed harmful to the models, we report performance in both the shifted and unshifted domains. Examples and further descriptions of unshifted/shifted splits of each dataset are given in section A.5. . . . .	36
4.2	<b>Results (true positive rate at the 5% significance level) for detection of harmful covariate shift</b> on CIFAR-10.1, Camelyon 17 and UCI Heart Disease benchmarks. The <b>best</b> result for each column is bolded, results that are within <u>2% of the best</u> are underlined and the <i>best baseline</i> method is italicized. . . . .	39

# List of Figures

- 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 *Constrained Disagreement Classifiers* (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  $\phi$  that CDCs disagree, as well as their entropy, on test data is a powerful and sample efficient statistic for identifying covariate shift  $\mathcal{P} \neq \mathcal{Q}$ . . . . . 13
  
- 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. . . . . 25

**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 model family, 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). . . . . 27

**3.3 CDC Training Dynamics:** In blue we train CDCs to disagree on a set of 100 samples from CIFAR 10.1 [Recht et al., 2019] (**Q**) – a near OOD test set for CIFAR 10 – while in black we force CDCs to disagree on the original CIFAR 10 test set (**P\***). Even after a small number of training batches the empirical disagreement rate grows significantly larger CIFAR 10.1 compared to CIFAR 10. The observation that CIFAR 10.1 results in more predictive variance under the CDC algorithm suggest that this covariate shift is harmful . . . . . 33

- 3.4 **The Detectron Disagreement Test:** (taken from our experiment where  $\mathcal{P} = \text{CIFAR 10}$  and  $\mathcal{Q} = \text{CIFAR 10.1}$  and sample size  $m = 50$ ) We start by training an ensemble of CDCs (we use an ensemble size of 5) to disagree on a set of  $m$  unseen samples  $\mathbf{P}^*$  from the original training distribution while constrained to perform consistently with a base model on the original training and validation sets used to train the base model on CIFAR 10. We perform 100 of these calibration runs using different random seeds and samples for  $\mathbf{P}^*$  to estimate a threshold  $\tau$  such that 95% of the runs disagree on fewer than  $\tau$  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  $\mathbf{P}^*$  with a random set of  $m$  samples  $\mathbf{Q}$  from  $\mathcal{Q}$  (CIFAR 10.1). By averaging the number of runs that disagree on more than  $\tau$  samples from  $\mathbf{Q}$  we can compute the power (or true positive rate) of the test as 83%—a state-of-the-art result for this task. . . . . 34
- 3.5 **The Detectron Entropy Test:** Following the same experimental setup as Figure 3.4, we start (left) by computing a KS test between the empirical distributions entropy values for each calibration run  $\mathbf{P}^*$  with the flattened set of entropy values from all other 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  $\tau$  on the distribution of  $p$ -values obtained from step 1 as the lower  $\alpha$  quantile to guarantee a false positive rate of  $\alpha$ . The power of the test is computed as the fraction of  $p$ -values computed from the test runs on  $\mathbf{Q}$  that are below  $\tau$ . . . . . 34
- 4.1 **Ensemble Size vs Properties of Constrained disagreement classifiers on CIFAR-10/10.1:** (Left) We see that for all ensemble sizes, there is lower disagreement on unshifted data (CIFAR-10) compared to disagreement on shifted data (CIFAR-10.1). (Center) Constrained disagreement does not compromise in-distribution performance as computed on unseen data. (Right) As the ensemble grows the selective classification accuracy computed on the set of test examples that all models agree on increases both on in-distribution and out-of-distribution data. Confidence intervals are computed as  $\pm$  one standard deviation across experiments. . . . . 37
- 4.2 **Selective Classification Behaviour of CDCS:** Comparing the accuracy versus rejection rate of selective classification on ID (CIFAR 10) and shifted (CIFAR 10.1) sets we see that the CDC algorithm constructs highly linear selective classifiers. The rejection rate required to reach perfect accuracy on ID data is significantly lower than that required on shifted data. 37

4.3 **True positive rate at the 5% significance level** for the Detectron and baseline methods for detection of covariate shift on the UCI heart disease dataset. The Detectron (Entropy) is shown to uniformly outperform baselines. Confidence intervals are excluded for visual clarity but are found in Table 4.2. . . . . 40

5.1 **Runtime Characteristics:** We train 100 random runs of CDCs on 100 samples from CIFAR 10 and 10.1 and compute the *disagreement statistic* as the difference  $\psi := \mathbb{E}[\phi_{\mathbf{Q}} - \phi_{\mathbf{P}}]$ . While we see that while  $\psi$  peaks near 50 training batches, only 10 batches are required for the Detectron disagreement test to reach an area under the TPR vs FPR curve (AUROC) of nearly 1 (i.e., perfect discrimination). Training CDCs for too long eventually lowers  $\psi$  as  $\mathbb{E}[\phi_{\mathbf{Q}}] \approx \mathbb{E}[\phi_{\mathbf{P}}] \approx 1$  meaning CDCs eventually overfit to disagreeing on all of their data. . . 42

A.1 Belief that the probability  $q$  that two classifiers disagree on samples from  $\mathcal{Q}$  is greater than 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 than  $n/N$ . . . . . 58

A.2 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. . . . . 59

A.3 **CDC Training Dynamics for Overparameterized Models:** Using the same experimental setup as the runtime study in Figure 5.1 we record the associated CDC disagreement rate on both unseen in distribution data  $\mathbf{P}^*$  (CIFAR 10) and a known covariate shift  $\mathbf{Q}$  (CIFAR 10.1). When we trained such an overparameterized model for too long the disagreement rate approaches 1 on both in and out-of-distribution datasets leading to a maximally uninformative test; it is, therefore, crucial to perform early stopping at or before the out-of-distribution disagreement rate reaches 1. . . . . 62

A.4	Cifar 10 vs Cifar 10.1. (Image borrowed from the technical report "Do CIFAR-10 Classifiers Generalize to CIFAR-10?" [Recht et al., 2018]) . . . . .	67
A.5	Samples images from the Camelyon 17 dataset [Veeling et al., 2018]. Using the standard set by the WILDS framework [Koh et al., 2021] we use hospitals 1–3 as the source domain for training and validating models, and hospital 5 as the target domain for assessing distribution shift. . . . .	68
A.6	Marginal distributions for each of the nine variables chosen from the UCI Heart Disease dataset for our experiments. The source domain (yellow) is chosen from the Cleveland and Hungary patient databases, while the shifted target domain (blue) is selected from the Swiss and Long Beach databases. Although the distributions are visually similar, a simple neural network classifier that achieves an AUC of 0.85 on the source domain drops to 0.42 on the target domain. . . . .	69

# 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; Otlés 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  $X$  to a learning model which aims to predict a target variable  $Y$ . In many standard machine learning settings we assume that the model is trained to learn the conditional distribution  $p(Y|X)$  using samples from the joint distribution  $p(X, Y)$ , but is deployed in a new environment where the distribution over covariates no longer matches. Unfortunately, results from standard learning theory only provide relatively loose bounds on the performance of learning models under covariate shift, meaning that even slight shifts can have disastrous outcomes for models in high risk domains.

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 abstain for a given sample. PQ learning introduces a strong set of performance guarantees for such selective classifiers under arbitrary covariate shifts, we summarize the main

results of PQ learning in section 2.5. 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. See Figure 1.1 for a high level overview of our method.

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.

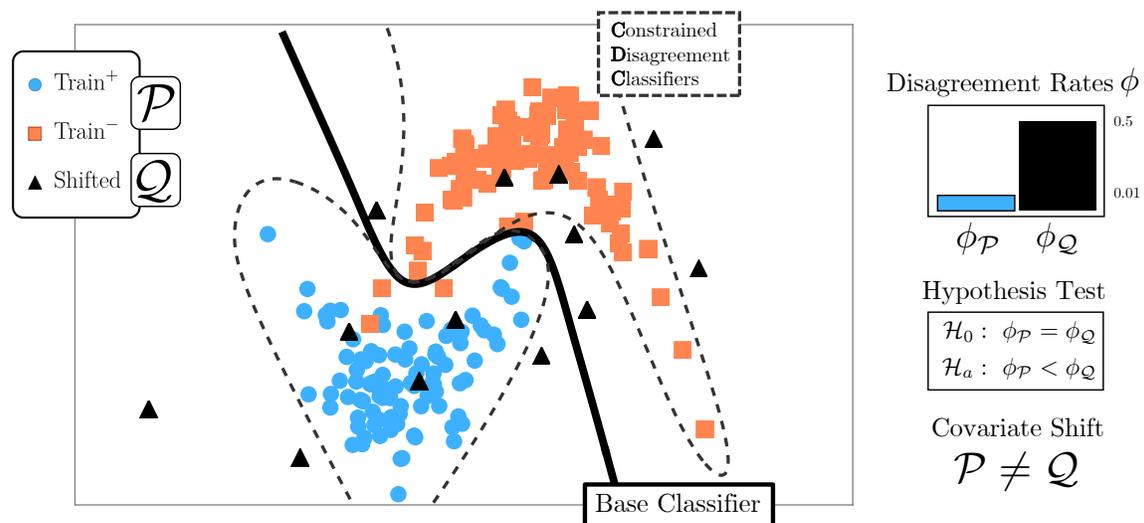


Figure 1.1: **Detectron (PQ Learning for Covariate Shift Detection)**: Starting with a base classifier trained on labeled samples from distribution  $\mathcal{P}$  we train new *Constrained Disagreement Classifiers* (CDCs) on a small set of observed unlabeled samples from a new distribution  $\mathcal{Q}$ . CDCs aim to maximize classification disagreement on  $\mathcal{Q}$  while constrained to agree with the base classifier on  $\mathcal{P}$ . The rate  $\phi$  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 reasonable performance guarantees when such data is exchangeable with the labeled samples which they were trained and 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  $\mathbf{P} = \{X_1, \dots, X_n\} \stackrel{\text{iid}}{\sim} \mathcal{P}$  and  $\mathbf{Q} = \{\tilde{X}_1, \dots, \tilde{X}_m\} \stackrel{\text{iid}}{\sim} \mathcal{Q}$  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  $\mathcal{H}_a : \mathcal{P} \neq \mathcal{Q}$  at a given significance level  $\alpha$ . 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  $\alpha$ . The power of a statistical test, often denoted as  $1 - \beta$  where  $\beta$  is the probability of making a type II error, is the probability that the test correctly rules out  $\mathcal{H}_0$  for a given target distribution  $\mathcal{Q}$  while still maintaining a significance level of  $\alpha$ . While 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  $\alpha$  for  $\alpha \in [0, 1]$ . For a

more rigorous 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 be used in 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  $X_1, X_2, \dots$  are independent and identically-distributed random variables in  $\mathbb{R}$  with common cumulative distribution function  $F(x)$ . The empirical distribution function for  $X_1, \dots, X_n$  is defined by

$$F_n(x) = \frac{1}{n} \sum_{i=1}^n I_{[X_i, \infty)}(x) = \frac{1}{n} |\{1 \leq i \leq n \mid X_i \leq x\}| \quad (2.1)$$

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.

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

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  $\mathcal{X} = \{0, \dots, n\}$ ). 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.

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

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  $\approx 0.028$  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  $\alpha$ . This quantity corresponds to the probability of observing a  $p$ -value of less than or equal to  $\alpha$  when

running a binomial test on sample from a random variable  $X \sim \text{Bin}(n, q)$  against a baseline rate  $p$ .

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

We use  $\mathbb{I}[\cdot]$  as a binary indicator. 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  $0.461 \pm 0.005$  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  $\mathcal{Q}$ .

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

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  $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  [Schrab et al., 2021].

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

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

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

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

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  $k(x, y) = \exp(-\sigma^{-1}\|x - y\|^2)$ , 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  $T : \mathcal{X}^n \times \mathcal{X}^m \rightarrow \mathbb{R}$ . 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  $\tau$  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  $\tau$  we take advantage of the exchangeability 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  $\tau$  as the  $1 - \alpha$  quantile of all observations of the test statistic. If the test performed on the original partitions exceeds  $\tau$  (e.g.,  $T(\mathbf{P}, \mathbf{Q}) \geq \tau$ ) we can rule out  $\mathcal{H}_0$  with an exact significance level  $\alpha$ . We note that as we are estimating  $\tau$  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, trained and untrained auto-encoders and the output features of a classification model. On each DR method 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  $\mathcal{A}$ .

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

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

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

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  $d : \mathcal{X} \rightarrow \{0, 1\}$  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  $\phi_\omega : \mathcal{X} \rightarrow \mathbb{R}^d$  to optimize the power of an MMD test using a simple kernel. The proposed learnable kernel  $k_\omega$  is a linear combination of a kernel  $\kappa$  that operates on the featurizer data and another kernel  $q$  that operates on the raw data:

$$k_\omega(x, y) = (1 - \epsilon)\kappa(\phi_\omega(x), \phi_\omega(y)) + \epsilon q(x, y) \quad (2.6)$$

the choice of kernels  $k$  and  $q$  as well as the value for  $\epsilon \in [0, 1]$  are considered as hyperparameters. In practice the authors use squared exponential kernels with a lengthscale of 0.1 and  $\epsilon = 10^{-10}$ . The parameters  $\omega$  of the deep kernel are learned to optimize the MMD objective in Equation 2.3 on splits of source and test data and test significance is computed using a permutation test on the unseen splits.

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  $\ell$  and set of possible actions  $\mathcal{A}$ . For example, if  $\ell$  is the reconstruction loss of an autoencoder  $f_\theta$  and  $\mathcal{A}$  is the set of all possible parameter configurations  $\theta$ ,  $H_\ell$  corresponds to the expected reconstruction loss of  $f_{\theta^*}$  where  $\theta^*$  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  $H_\ell\left(\frac{p+q}{2}\right) - H_\ell(q)$ . The authors propose using an empirical estimate of the H-divergence directly as a test statistic for performing a two sample test with significance bounded through permutation.

While deep kernels and H-divergence operate in a purely unlabeled/unsupervised setting, in our work we 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  $\phi : X \rightarrow \mathbb{R}$  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. Hence testing for a difference in expected OOD scores between a source and target dataset is a valid hypothesis test for covariate. However, as OOD scores are designed to only look at a single sample level, instead of treating a set of samples as a whole one would not expect such a test to yield high statistical power for near distribution shifts.

## 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]. A non exhaustive list of approaches to model uncertainty using predictive models are Gaussian Processes [Williams and Rasmussen, 1995], measuring the predictive variance or entropy of Deep Ensembles [Lakshminarayanan et al., 2017] or Monte Carlo Dropout [Gal and Ghahramani, 2015], Bayesian Neural Networks [Goan and Fookes, 2020], and Evidential Learning [Amini et al., 2020; Sensoy et al., 2018]. Naturally, uncertainty should be large when samples are OOD implying that one could consider testing for covariate shift using the same principle as identified for OOD scores. 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 sup-

plementary 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 Methodology: The Detectron

## 3.1 Problem Setup

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

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

We denote the prediction of a model  $f(x)$  as  $\hat{f}(x) := \arg \max_{i \in \{1, \dots, N\}} f(x)_i$  which corresponds to the class with the maximum predicted probability. We assume  $f$  was trained on a dataset of **labeled** samples  $\mathbf{P} = \{(x_i, y_i)\}_{i=1}^n$  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  $\mathbf{Q} = \{\tilde{x}_i\}_{i=1}^m$  ( $\tilde{x}_i$  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 such a way that is easily distinguishable with respect to a set of classifiers that fit the training set. To give further intuition, if we can create multiple models that perform reliably and consistently on the training set  $\mathbf{P}$ , but disagree on  $\mathbf{Q}$ , not only is exchangeability between  $\mathbf{P}$  and  $\mathbf{Q}$  violated, but no claims can be made as to what the *correct* predictions on  $\mathbf{Q}$  should be.

Our problem will differ from a standard two sample testing 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* (e.g. could realistically be deployment) on held out samples from  $\mathcal{P}$  as well as the learning algorithm  $L$  used to train it. (3) We seek to only detect shifts that could 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.

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  $5 \times 5$  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 ob-



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.

ervation 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  $0.981 \pm 0.004$  at the strongest significance level  $\alpha = 0$ .

The MNIST example motivates 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 informal idea of the *generalization set*  $\mathcal{R}$  as the set of distributions, beyond simply the training distribution, that a model generalizes to.  $\mathcal{R}$  is an abstract object which may in general depend on the model architecture/function class  $F$ , learning algorithm  $L$  and training dataset  $\mathbf{P}$ . 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 on PQ learning, intuition from learning theory and observations in learning dynamics under distribution shift. A one sentence summary is: if we can find a set of classifiers

$\{g_1, \dots, g_N\} \subseteq F$  that achieve the same generalization set as  $f$  but behave inconsistently with respect to  $f$  on samples from a distribution  $\mathcal{Q}$ , then  $\mathcal{Q}$  must not be a member  $\mathcal{R}$ . 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  $\mathcal{P} \rightarrow \mathcal{Q}$  over  $X$  is  $(\ell, \alpha, L, \mathcal{P})$ -harmful, if there exists a finite subset of two or more models  $\mathbf{f} \subseteq F$  that achieve a source domain loss  $\mathbb{E}_{\mathcal{P}}[\ell(f(x), y)] \leq \alpha$  for all  $f \in \mathbf{f}$  and are more likely to disagree with each other on an unseen sample from  $\mathcal{Q}$  compared to  $\mathcal{P}$ .

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

In plainer words, we define harmful covariate shift based on the existence of multiple *good* models on  $\mathcal{P}$  whose respective invariances define the 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*  $\{g_1, \dots, g_N\}$ , 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  $\mathbf{P}$  by definition do 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:

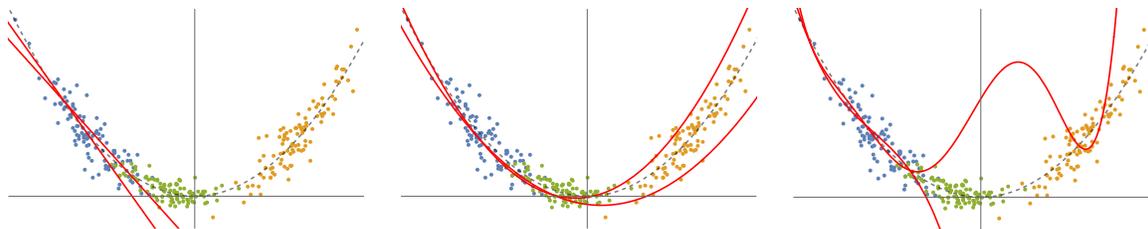
1.  $g$  belongs to the same model class as  $f$  and is updated with the same learning algorithm when it observes samples from  $\mathcal{P}$ ;

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.



**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 model family, 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).

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  $h|_S$  that achieves 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, a selective classifier predicts according to a base classifier  $h$  if the input  $x$  is inside some set  $S$  and otherwise rejects/abstains from predicting.

$$h|_S(x) := \begin{cases} h(x) & x \in S \\ \text{reject} & x \notin S \end{cases} \quad (3.2)$$

The error and rejection rate are defined for a selective classifier  $h|_S$  with respect to a distribution  $\mathcal{P}$  as:

$$\text{rej}_h(S) := \mathbb{P}_{x \sim \mathcal{P}} [x \notin S] \quad \text{err}_h(S) := \mathbb{P}_{x \sim \mathcal{P}} [h(x) \neq y \wedge x \in S] \quad (3.3)$$

Where  $y = d(x)$  is the ground truth label for  $x$ . The *empirical* rejection and error rates for a set of samples  $\mathbf{x} = \{x_i\}_{i=1}^n$  and corresponding labels  $\mathbf{y} = \{y_i\}_{i=1}^n$  are similarly defined as:

$$\text{rej}_h(\mathbf{x}) := \frac{1}{n} \sum_{i=1}^n \delta_{x_i \notin S} \quad \text{err}_h(\mathbf{x}) := \frac{1}{\sum_{i=1}^n \delta_{x_i \in S}} \sum_{i=1}^n (\delta_{x_i \neq y_i} \times \delta_{x_i \in S}) \quad (3.4)$$

Using this construction Goldwasser et al. define PQ learning with the following learning theoretic bound.

**Definition 3** (PQ learning [Goldwasser et al., 2020]). Learner  $L$   $(\epsilon, \delta, n)$ -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

$$\mathbb{P}_{\mathbf{x} \sim \mathcal{P}^n, \tilde{\mathbf{x}} \sim \mathcal{Q}^n} [\text{rej}_h(\mathbf{x}) + \text{err}_h(\tilde{\mathbf{x}}) \leq \epsilon] \geq 1 - \delta \quad (3.5)$$

$L$  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  $g(x)$  be the predictive distribution over  $N$  classes,  $\hat{f}(x) \in \{1, \dots, N\}$  the label predicted by  $f$  and  $\mathbb{I}[\cdot]$  a binary indicator, we define the *disagreement-cross-entropy* (DCE) as:

$$\tilde{\ell}_{\text{DCE}}(g(x), \hat{f}(x)) = \frac{1}{1-N} \sum_{c=1}^N \mathbb{I}[\hat{f}(x) \neq c] \log(g(x)_c) \quad (3.6)$$

Equation 3.6 corresponds to taking the cross entropy of  $g(x)$  with the uniform distribution over all classes except  $f(x_i)$ . Since the primary criteria is that  $g(x)$  disagrees with  $f(x)$ ,  $\tilde{\ell}_{\text{DCE}}$  is designed to minimize the probability of  $g$ 's prediction for the output of  $f$ 's while maximizing its overall entropy. Our definition of  $\tilde{\ell}_{\text{DCE}}$  is significantly more stable to optimize compared to simply maximizing the regular cross entropy as it has a bounded global minimum. See further details in section A.3.

Our goal is to agree on  $\mathbf{P}$  and disagree on  $\mathbf{Q}$ . Consequently, we learn with the loss in Equation 3.7.  $\ell$  denotes the standard cross entropy loss and  $\tilde{\ell}_{\text{DCE}}$  is the disagreement cross entropy.  $\lambda$  is a scalar parameter that controls the trade off between agreement and disagreement.

$$\mathcal{L}_{\text{CDC}}(\mathbf{P}, \mathbf{Q}) = \frac{1}{|\mathbf{P}| + |\mathbf{Q}|} \left( \underbrace{\sum_{(x,y) \in \mathbf{P}} \ell(g(x), y)}_{\text{predict labels on source data}} + \lambda \underbrace{\sum_{\tilde{x} \in \mathbf{Q}} \tilde{\ell}_{\text{DCE}}(g(\tilde{x}), f(\tilde{x}))}_{\text{disagree with } f \text{ on target data}} \right) \quad (3.7)$$

When learning CDCs in practice,  $\mathcal{L}_{\text{CDC}}$  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  $\mathcal{P}$ .

### 3.6.1 Choosing $\lambda$

We show that we can choose the scalar parameter  $\lambda$  in Equation 3.7 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  $\mathbf{Q}$ , thus relaxing the original constrained optimization objective to a tractable weighted objective. 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  $\lambda$ . To encourage agreement on  $\mathbf{P}$  as the primary objective, we set  $\lambda$  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.

### 3.6.2 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  $\mathcal{S} := \{1, \dots, N\} \setminus \{t\}$  where  $t$  is the disagreement target. We also give each 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 3.6; we provide a simple proof below.

*Proof.* Starting with the definition of the cross entropy

$$\text{CE}(f(x), y) = - \sum_{c=1}^N \mathbb{I}[c = y] \log(f(x)_c) \quad (3.8)$$

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

$$\sum_{y \in \mathcal{S}} \text{CE}(f(x), y) = - \sum_{y \in \mathcal{S}} \sum_{c=1}^N \mathbb{I}[c = y] \log(f(x)_c) \quad (3.9)$$

$$= - \sum_{c=1}^N \mathbb{I}[c \in \mathcal{S}] \log(f(x)_c) \quad (3.10)$$

$$= (N - 1) \tilde{\ell}_{\text{DCE}}(f(x), y) \quad (3.11)$$

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

### 3.6.3 Ensembling

To learn richer disagreement rules, we create an ensemble of CDCs where the  $k^{\text{th}}$  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  $2 \times 2$  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  $\mathbf{P}$  drawn 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  $\mathcal{P}$  with probability greater than  $p^* := \frac{1}{2} (1 - 4^{-n} \binom{2n}{n})$  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  $p^*$ . The contrapositive argument then states if we deem the probability to be greater than  $p^*$  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 (Theorem 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. Furthermore for small samples sizes the ability to find the classifier  $g$  that meets of predicate of Theorem 2 becomes incredibly difficult. Since our objective is to detect covariate shift from as few test samples as possible, splitting  $\mathbf{Q}$  and computing disagreement on unseen data will not result in a powerful test. To tackle this issue practically, we take a transductive approach based on intuition from learning theory and observed dynamics of CDC learning. We claim that training 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  $\mathcal{R}$ . Easiness in this setting is captured by the fact that standard learning algorithms tend to converge significantly quicker to the CDC objective when  $\mathcal{P} \neq \mathcal{Q}$ . We can therefore use the *relative increase* in disagreement between CDCs *during training* on  $\mathbf{Q}$  and  $\mathbf{P}$  to capture a quantity that is more informative than the unbiased statistic without reducing samples from  $\mathbf{Q}$  that we can use.

## 3.8 The Detectron Test

### 3.8.1 Detectron Disagreement

Our proposed method is to train two CDC ensembles of size  $\aleph$  denoted by  $\mathbf{g}_{\mathbf{Q}} := \{g_{\mathbf{Q}}^1, \dots, g_{\mathbf{Q}}^{\aleph}\}$  and  $\mathbf{g}_{\mathbf{P}^*} := \{g_{\mathbf{P}^*}^1, \dots, g_{\mathbf{P}^*}^{\aleph}\}$ . First,  $\mathbf{g}_{\mathbf{Q}}$  is trained to disagree with  $f$  on  $\mathbf{Q}$ , while constrained to agree on the original training set  $\mathbf{P}$ . Next, enforcing the null hypothesis, we train  $\mathbf{g}_{\mathbf{P}^*}$  to disagree on set of **unseen samples**  $\mathbf{P}^*$  drawn from  $\mathcal{P}$  where  $m := |\mathbf{P}^*| = |\mathbf{Q}|$ . To maximize sample efficiency, we take a transductive approach to detecting shift by analyzing the outputs of  $\mathbf{g}_{\mathbf{Q}}$  and  $\mathbf{g}_{\mathbf{P}}$  on the sets  $\mathbf{Q}$  and  $\mathbf{P}^*$  themselves by calculating their *empirical disagreement* during training.

We define the empirical disagreement between a base classifier  $f$ , an ensemble of classifiers  $\mathbf{g} := \{g^1, \dots, g^{\aleph}\}$ , and an unlabeled dataset  $\mathbf{D} := \{x_i\}_{i=1}^n$  as the proportion of samples in  $\mathbf{D}$  where any  $g \in \mathbf{g}$  predicts differently from  $f$ :

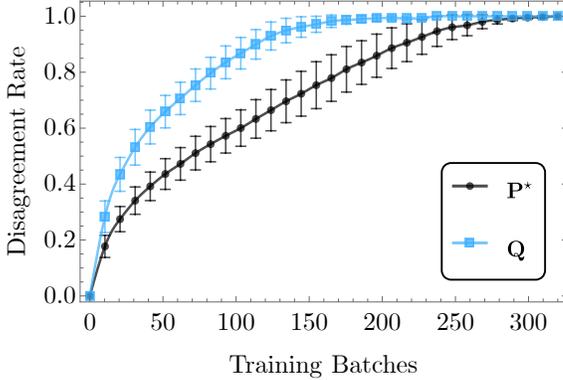
$$\phi_{\mathbf{D}}(f, \mathbf{g}) := \frac{1}{|\mathbf{D}|} \sum_{x \in \mathbf{D}} \left( 1 - \mathbb{I} \left[ \bigwedge_{g \in \mathbf{g}} \hat{f}(x) = \hat{g}(x) \right] \right) \quad (3.12)$$

The fraction of samples that  $\mathbf{g}_{\mathbf{P}^*}$  disagrees on with respect to  $f$  on  $\mathbf{P}^*$  is denoted using shorthands  $\phi_{\mathbf{P}^*} := \phi_{\mathbf{P}^*}(f, \mathbf{g}_{\mathbf{P}^*})$ , and similarly for  $\mathbf{g}_{\mathbf{Q}}$  as  $\phi_{\mathbf{Q}}$ . Note that  $\phi_{\mathbf{P}^*}$  and  $\phi_{\mathbf{Q}}$  are random variables that obey intractable distributions governed by the dynamics of the CDC learning algorithm and disagreement sets  $\mathbf{P}^*$  and  $\mathbf{Q}$ . The observed low variance and large divergence between the distributions of  $\phi_{\mathbf{P}^*}$  and  $\phi_{\mathbf{Q}}$  under covariate shift forms the backbone the Detectron test. Under the null

hypothesis  $\mathcal{H}_0$ , if  $\mathcal{Q}$  is not a harmful shift from  $\mathcal{P}$ , then  $\phi_{\mathcal{Q}}$  will not grow more rapidly than  $\phi_{\mathcal{P}^*}$

$$\mathcal{H}_0 : \mathbb{E}[\phi_{\mathcal{Q}}] \leq \mathbb{E}[\phi_{\mathcal{P}^*}], \quad \mathcal{H}_a : \mathbb{E}[\phi_{\mathcal{Q}}] > \mathbb{E}[\phi_{\mathcal{P}^*}] \quad (3.13)$$

The converse of harmful shift is expressed as the one-sided alternative  $\mathcal{H}_a$ , meaning disagreeing on  $\mathcal{Q}$  is easier than  $\mathcal{P}^*$ . See Figure 3.3 for a visual depiction of CDC training dynamics when applied to the near distribution shift benchmark CIFAR 10/10.1 [Recht et al., 2019].



**Figure 3.3: CDC Training Dynamics:** In blue we train CDCs to disagree on a set of 100 samples from CIFAR 10.1 [Recht et al., 2019] ( $\mathcal{Q}$ ) – a near OOD test set for CIFAR 10 – while in black we force CDCs to disagree on the original CIFAR 10 test set ( $\mathcal{P}^*$ ). Even after a small number of training batches the empirical disagreement rate grows significantly larger CIFAR 10.1 compared to CIFAR 10. The observation that CIFAR 10.1 results in more predictive variance under the CDC algorithm suggest that this covariate shift is harmful

We refer to the test in Equation 3.13 as *Detectron Disagreement* (Det-D). To compute the test result at a significance level  $\alpha$  we first estimate the null distribution of  $\phi_{\mathcal{P}}$  for a fixed sample size  $m$  by training  $K$  calibration rounds of  $\mathbf{g}_{\mathcal{P}}$  with different random sets  $\mathcal{P}^*$  of size  $m$ . The test result is significant if the observed disagreement rate  $\phi_{\mathcal{Q}}$  is greater than the  $(1 - \alpha)$  quantile of the null distribution. This process is explained visually in Figure 3.4.

### 3.8.2 Detectron Entropy

We consider an additional variant, *Detectron Entropy* (Det-E), which computes the mean prediction entropy of the CDC instead of relying solely on disagreement rates. The intuition for Det-E 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  $\mathcal{Q}$  and low entropy on  $\mathcal{P}^*$ , resulting in a natural way to distinguish between distributions. The CDC entropy is computed for a sample  $x$  from the mean probabilities over each  $N$  classes of the base classifier  $f$  and set of CDCs  $\mathbf{g}$ .

$$\text{CDC}_{\text{entropy}}(x) = - \sum_{c=1}^N \hat{p}_c \log(\hat{p}_c) \quad \text{where} \quad \hat{p}_c := \frac{1}{|\mathbf{g}| + 1} \left( f(x)_c + \sum_{g \in \mathbf{g}} g(x)_c \right) \quad (3.14)$$

We use a KS test to compute a  $p$ -value for covariate shift directly on the entropy distributions computed for  $\mathcal{Q}$  and  $\mathcal{P}^*$  and guarantee significance using the same strategy as Det-D (This process

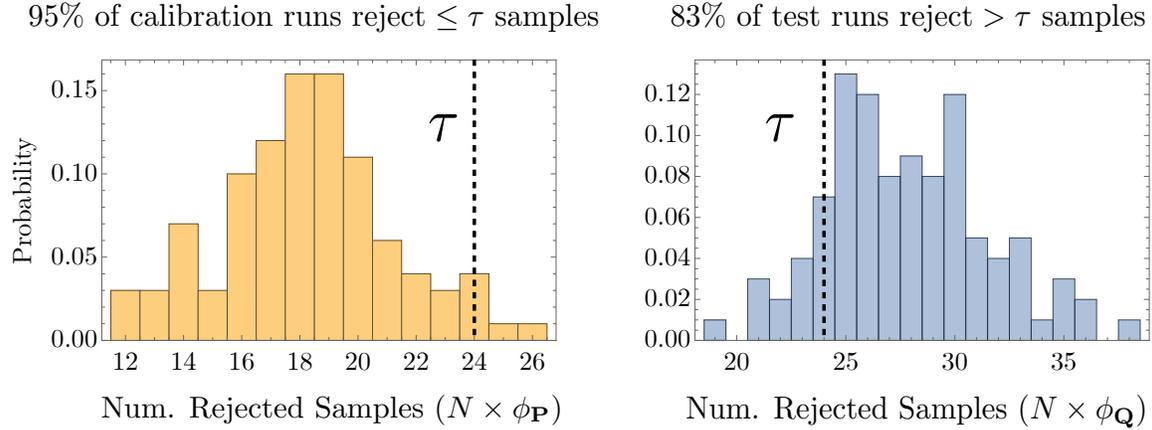


Figure 3.4: **The Detectron Disagreement Test:** (taken from our experiment where  $\mathcal{P} = \text{CIFAR 10}$  and  $\mathcal{Q} = \text{CIFAR 10.1}$  and sample size  $m = 50$ ) We start by training an ensemble of CDCs (we use an ensemble size of 5) to disagree on a set of  $m$  unseen samples  $\mathbf{P}^*$  from the original training distribution while constrained to perform consistently with a base model on the original training and validation sets used to train the base model on CIFAR 10. We perform 100 of these calibration runs using different random seeds and samples for  $\mathbf{P}^*$  to estimate a threshold  $\tau$  such that 95% of the runs disagree on fewer than  $\tau$  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  $\mathbf{P}^*$  with a random set of  $m$  samples  $\mathbf{Q}$  from  $\mathcal{Q}$  (CIFAR 10.1). By averaging the number of runs that disagree on more than  $\tau$  samples from  $\mathbf{Q}$  we can compute the power (or true positive rate) of the test as 83%—a state-of-the-art result for this task.

is explained visually in Figure 3.5). 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  $\mathbf{P}^*$ , resulting in a natural way to distinguish between distributions.

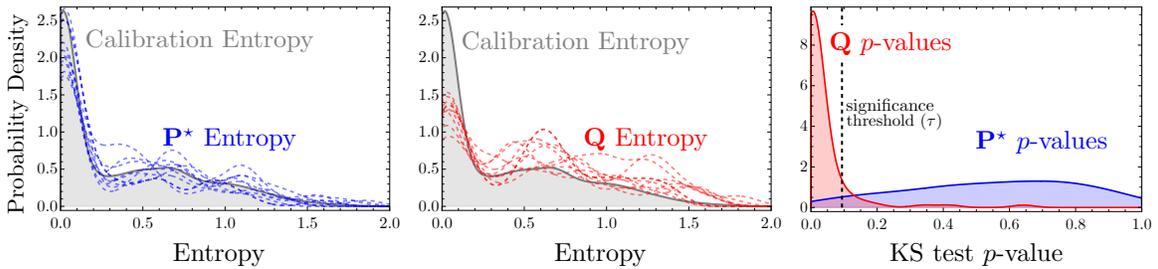


Figure 3.5: **The Detectron Entropy Test:** Following the same experimental setup as Figure 3.4, we start (left) by computing a KS test between the empirical distributions entropy values for each calibration run  $\mathbf{P}^*$  with the flattened set of entropy values from all other calibration runs. 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  $\tau$  on the distribution of  $p$ -values obtained from step 1 as the lower  $\alpha$  quantile to guarantee a false positive rate of  $\alpha$ . The power of the test is computed as the fraction of  $p$ -values computed from the test runs on  $\mathbf{Q}$  that are below  $\tau$ .

### 3.8.3 Detectron Algorithm

Concluding our methodology, we present a formal algorithmic description of the Detectron Disagreement test below in algorithm 1. The function `ConstrainedDisagreement(loss, train, val, test, model)` refers to fine-tuning the base classifier  $f$  to minimize the CDC loss in Equation 3.7. The details/hyperparameters of this step are discussed in subsection A.4.2.

We release our code as open source software under a GNU General Public License v3.0.



<https://github.com/tomginsberg/deeptst>

---

**Algorithm 1:** The Detectron algorithm for detecting harmful covariate shift

---

**Input:**  $\mathbf{P}$ : labeled dataset,  $\mathbf{Q}$ : unlabeled dataset,  $\ell$ : loss function,  
 $K$ : calibration rounds = 100,  $\aleph$ : ensemble size = 5,  $\alpha$ : significance level = 0.05  
**Output:** test result for harmful covariate shift at significance level  $\alpha$

$\mathbf{P}_{\text{train}}, \mathbf{P}_{\text{val}}, \mathbf{P}^* \leftarrow \text{Partition}(\mathbf{P})$   
 $m \leftarrow |\mathbf{Q}|$ ;  $\Phi_{\mathbf{P}^*} \leftarrow []$   
 $f \leftarrow L(\mathbf{P}_{\text{train}}, \mathbf{P}_{\text{val}})$  // Load or train a base classifier on  $\mathbf{P}$

**repeat**  
   $\mathbf{p}^* \leftarrow \text{RandomSample}(\mathbf{P}^*, m)$   
  // Train an ensemble of CDCs on  $\mathbf{P}^*$   
  **while**  $|\mathbf{p}^*| > 0$  and iterations  $\leq \aleph$  **do**  
     $g \leftarrow \text{ConstrainedDisagreement}(\ell, \mathbf{P}_{\text{train}}, \mathbf{P}_{\text{val}}, \mathbf{p}^*, f)$  // Optimize Equation 3.7  
     $\mathbf{p}^* \leftarrow \{x \mid x \in \mathbf{p}^* \text{ and } f(x) = g(x)\}$  // Filter out disagreed on data  
     $\phi_{\mathbf{P}^*} \leftarrow 1 - |\mathbf{p}^*|/m$  // Update disagreement rate  
  **end**  
  Append  $\phi_{\mathbf{P}^*}$  to  $\Phi_{\mathbf{P}^*}$   
**until**  $K$  iterations elapse  
// Train an ensemble of CDCs on  $\mathbf{Q}$   
**while**  $\mathbf{Q} > 0$  and iterations  $\leq \aleph$  **do**  
   $g \leftarrow \text{ConstrainedDisagreement}(\ell, \mathbf{P}_{\text{train}}, \mathbf{P}_{\text{val}}, \mathbf{Q}, f)$   
   $\mathbf{Q} \leftarrow \{x \mid x \in \mathbf{Q} \text{ and } f(x) = g(x)\}$   
   $\phi_{\mathbf{Q}} \leftarrow 1 - |\mathbf{Q}|/m$   
**end**  
**return**  $\phi_{\mathbf{Q}} > [(1 - \alpha) \text{ quantile of } \Phi_{\mathbf{P}^*}]$

---

# Chapter 4 Applications and Experiments

## 4.1 Datasets

Our experiments are carried out on natural distribution shifts across multiple domains, modalities, and model types. We use the *CIFAR-10.1* dataset [Recht et al., 2019] where shift comes from subtle changes in the dataset creation processes, the *Camelyon17 dataset* [Veeling et al., 2018] for metastases detection in histopathological slides from multiple hospitals, as well as the *UCI heart disease* dataset [Janosi et al., 1988] which contains tabular features collected across international health systems and indicators of heart disease.

A summary of the three datasets used as well as the description of shifts and their impacts on base model performance is provided in Table 4.1 below.

Domain/Task	Dataset	Shift	Metric	(Unshifted)	(Shifted)
Natural Images <i>Object classification</i>	CIFAR-10/10.1 [Recht et al., 2019]	Data Collection Process	Accuracy	0.87 (Resnet18)	0.77 (Resnet18)
Histopathological Images <i>Metastases Detection</i>	Camelyon-17 [Veeling et al., 2018]	Different Hospitals	Accuracy	0.93 (Resnet18)	0.81 (Resnet18)
Tabular Medical Data <i>Angiographic Status</i>	UCI Heart Disease [Janosi et al., 1988]	Different Countries	AUROC	0.88 (xgboost) 0.85 (MLP)	0.70 (xgboost) 0.42 (MLP)

Table 4.1: **Datasets:** We investigate three different forms of covariate shift in two unique data modalities. To verify that these shifts are indeed harmful to the models, we report performance in both the shifted and unshifted domains. Examples and further descriptions of unshifted/shifted splits of each dataset are given in section A.5.

## 4.2 Learning Constrained Disagreement

Before discussing the results of the Detectron test, we begin by investigating the CDC learning objective (Equation 3.7) in isolation to verify that it meets our desired criteria in 2. We training ensembles of 10 CDCs using the *disagreement cross entropy* (DCE) with CIFAR-10 as  $\mathcal{P}$  and CIFAR-10.1 as  $\mathcal{Q}$  for 100 random runs at a sample size of 50. We use an ImageNet pretrained residual network fine-tuned on CIFAR 10 as our basemodel  $f$ ; see section A.4 for full training details. CDCs

are initialized from  $f$  and use the same training algorithm/loss function when updated on samples from  $\mathcal{P}$  (property 1 in Def. 2). The results in Figure 4.1 empirically validates minimizing the DCE as a CDC learning objective. The first observation is that when an unseen set is drawn from a shifted distribution  $\mathcal{Q}$ , the empirical disagreement rate  $\phi_{\mathcal{Q}}$  grows significantly larger than the baseline disagreement rate  $\phi_{\mathcal{P}}$  (property 3). Next, we see that CDCs preserve accuracy on unseen data from the training distribution, regardless of the dataset they are trained to disagree on (property 2). Finally, an additional observation shows that as the ensemble size increases (and disagreed upon points are removed) we see that the selective classification accuracy on the agreed upon points increases. Furthermore, the concave behaviour of the selective classification curve indicates that the that CDCs quickly disagree on many data samples that would have been misclassified by  $f$ , while in later iterations fewer of such samples are identified (see Figure 4.2 for a direct comparison of accuracy vs rejection rate of CDCs).

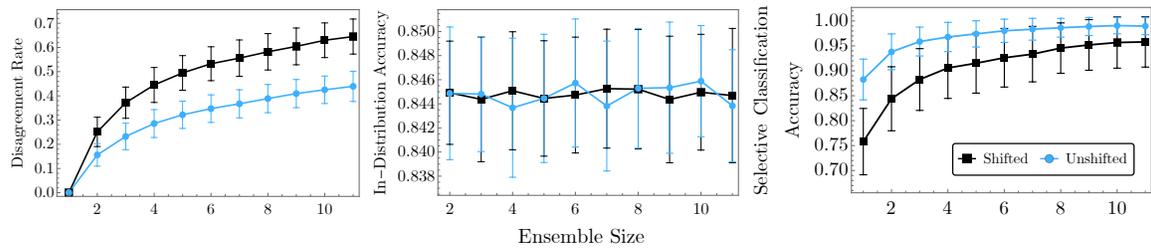


Figure 4.1: **Ensemble Size vs Properties of Constrained disagreement classifiers on CIFAR-10/10.1:** (Left) We see that for all ensemble sizes, there is lower disagreement on unshifted data (CIFAR-10) compared to disagreement on shifted data (CIFAR-10.1). (Center) Constrained disagreement does not compromise in-distribution performance as computed on unseen data. (Right) As the ensemble grows the selective classification accuracy computed on the set of test examples that all models agree on increases both on in-distribution and out-of-distribution data. Confidence intervals are computed as  $\pm$  one standard deviation across experiments.

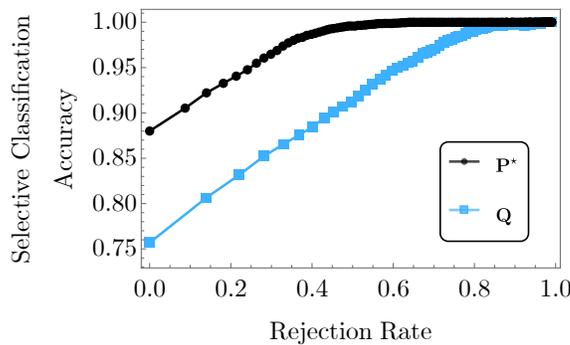


Figure 4.2: **Selective Classification Behaviour of CDCS:** Comparing the accuracy versus rejection rate of selective classification on ID (CIFAR 10) and shifted (CIFAR 10.1) sets we see that the CDC algorithm constructs highly linear selective classifiers. The rejection rate required to reach perfect accuracy on ID data is significantly lower than that required on shifted data.

### 4.3 Shift Detection Approach

We evaluate the Detectron in a standard two-sample testing scenario similar to prior work [Zhao et al., 2022]. Given two datasets  $\mathbf{P} = \{(x_i, y_i)\}_{i=1}^n$  ( $x_i$  drawn from  $\mathcal{P}$ ) and  $\mathbf{Q} = \{\tilde{x}_i\}_{i=1}^m$  ( $\tilde{x}_i$  drawn from  $\mathcal{Q}$ ) and classifier  $f$ , we seek to rule out the null hypothesis ( $\mathcal{P} = \mathcal{Q}$ ) at the 5% significance level. To guarantee fixed significance we employ a permutation test by first sampling from the distribution of test statistics derived by the Detectron where the null hypothesis  $\mathcal{P} = \mathcal{Q}$  holds (i.e.,  $\mathbf{Q}$  is drawn  $\mathcal{P}$ ). Next, we compute a threshold over the empirical test statistic distribution that sets the false positive rate to 5% (see Figure 3.4). This step can be performed before deployment as it only requires access to  $\mathbf{P}$ . To mimic deployment settings where we wish to identify covariate shift quickly, we assume access to far more samples from  $\mathcal{P}$  compared to  $\mathcal{Q}$ . For each dataset, we begin by training a base classifier on the unshifted dataset. We evaluate the detection of covariate shift on 100 randomly selected test sets of  $n = 10, 20$  and  $50$  samples from  $\mathcal{Q}$ . In all cases, we train a maximum ensemble size of 5 (parameter  $\aleph$  in algorithm 1). To prevent CDCs from catastrophically overfitting (see subsection A.3.2) in the case of small test set sizes, we perform early stopping if in-distribution validation performance drops by over 5% from the measured performance of the base classifier. Hyperparameters and training details for all models can be found in section A.4.

#### 4.3.1 Evaluation

We report the *True Positive Rate at 5% Significance Level (TPR@5)* aggregated over 100 randomly selected sets  $\mathbf{Q}$ . This signifies how often our method correctly identifies covariate shift ( $\mathcal{P} \neq \mathcal{Q}$ ) while only incorrectly identifying shift 5% of the time. This is also referred to as the statistical power of a test where the significance level or probability of making a type I error is 5%.

#### 4.3.2 Baselines

We compare the Detectron against several methods for OOD detection, uncertainty estimation and covariate shift detection. *Deep Ensembles* Ovdia et al. [2019] using both (1) *disagreement* and (2) *entropy* scoring methods as a direct ablation to the CDC approach (3) *Black Box Shift Detection (BBSD)* [Lipton et al., 2018]. (4) *Relative Mahalanobis Distance (RMD)* [Ren et al., 2021]. (5) *Classifier Two Sample Test (CTST)* [Lopez-Paz and Oquab, 2017]. (6) *Deep Kernel MMD (MMD-D)* [Liu et al., 2020]. (7) *H Divergence (H-Div)* [Zhao et al., 2022]. For more information on baselines see Appendix subsection A.4.3.

## 4.4 Shift Detection Experiments

We present statistical power (TPR@5) results for sample sizes of 10, 20 and 50 using the Detectron (algorithm 1) on all datasets are shown in Table 4.2. We report the mean and standard error of TPR@5 computed on 100 samples for each trial.

$ \mathbf{Q} $	CIFAR 10.1			Camelyon 17			UCI Heart Disease		
	10	20	50	10	20	50	10	20	50
BBSD	.07 ± .03	.05 ± .02	.12 ± .03	.16 ± .04	.38 ± .05	.87 ± .03	.13 ± .03	.22 ± .04	.46 ± .05
Rel. Mahalanobis	.05 ± .02	.03 ± .03	.04 ± .02	.16 ± .04	.40 ± .05	<i>.89 ± .03</i>	.11 ± .03	.36 ± .05	.66 ± .05
Deep Ensemble (Dis)	.23 ± .04	.40 ± .05	<i>.74 ± .04</i>	.10 ± .03	.11 ± .03	.13 ± .03	.02 ± .01	.00 ± .00	.32 ± .05
Deep Ensemble (Entropy)	<i>.33 ± .05</i>	<i>.52 ± .05</i>	.68 ± .05	.14 ± .03	.26 ± .04	.82 ± .04	.13 ± .03	.32 ± .05	.64 ± .05
CTST	.03 ± .02	.04 ± .02	.04 ± .02	.11 ± .03	.59 ± .05	.59 ± .05	<i>.15 ± .04</i>	<i>.51 ± .05</i>	<u><i>.98 ± .01</i></u>
MMD-D	.24 ± .04	.10 ± .03	.05 ± .02	<i>.42 ± .05</i>	<i>.62 ± .05</i>	.69 ± .05	.09 ± .03	.12 ± .03	.27 ± .04
H-Div	.02 ± .01	.05 ± .02	.04 ± .02	.03 ± .02	.07 ± .03	.23 ± .04	<i>.15 ± .04</i>	.26 ± .04	.37 ± .05
<b>Detectron (Dis)</b>	<b>.37 ± .05</b>	<u><i>.54 ± .05</i></u>	.83 ± .04	<b>.97 ± .02</b>	<b>1.0 ± .00</b>	.96 ± .02	.24 ± 0.04	.57 ± 0.05	.82 ± 0.04
<b>Detectron (Entropy)</b>	<u><i>.35 ± .05</i></u>	<b>.56 ± .05</b>	<b>.92 ± .03</b>	<b>.97 ± .02</b>	<b>1.0 ± .00</b>	<b>1.0 ± .00</b>	<b>.45 ± .05</b>	<b>.88 ± 0.03</b>	<b>1.0 ± .00</b>

Table 4.2: **Results (true positive rate at the 5% significance level) for detection of harmful covariate shift** on CIFAR-10.1, Camelyon 17 and UCI Heart Disease benchmarks. The **best** result for each column is bolded, results that are within 2% of the best are underlined and the *best baseline* method is italicized.

### 4.4.1 Increasing the Sample Size

To expand on Table 4.2 we show an extended analysis of the performance of the Detectron on the UCI Heart Disease dataset. Using a sample size ranging from 10 to 100 we compute the TPR@5 averaged over 100 trials and plot the results in Figure 4.3. As the Detectron is model agnostic we use gradient boosted trees (XGBoost [Chen and Guestrin, 2016]) for Detectron and CTST methods while the remaining baselines which require neural network models use a 2 layer MLP that achieves similar in-distribution performance.

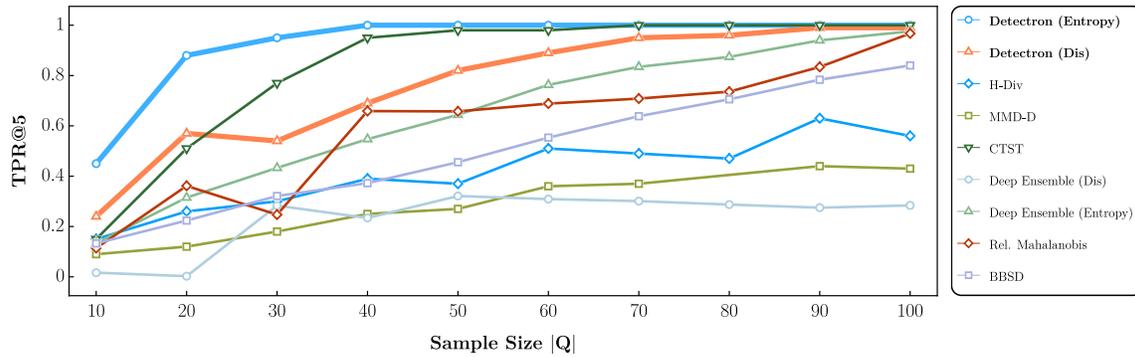


Figure 4.3: **True positive rate at the 5% significance level** for the Detectron and baseline methods for detection of covariate shift on the UCI heart disease dataset. The Detectron (Entropy) is shown to uniformly outperform baselines. Confidence intervals are excluded for visual clarity but are found in Table 4.2.

# Chapter 5 Discussion

## Overall Performance

We observe in the bottom rows of Table 4.2 that Detectron methods outperform all baselines across all three tasks. This confirms our intuition that designing distribution tests based specifically on available data and the outputs of learning algorithms is a promising avenue for improving existing methods in the high dimensional/low data regime.

## Sample Efficiency

For more severe shifts (Camelyon and UCI), we see in Table 4.2 the most significant improvements over baselines in the lowest sample regime (10 data points). The fine-grained result in Figure 4.3 shows that CTST catches up to Detectron at 40 samples while deep ensemble, BBSD, and Mahalanobis catch up at 100.

## Disagreement vs Entropy

For the experiments on imaging datasets with deep neural networks Detectron (Disagreement) often performs nearly as well as Detectron (Entropy), while Detectron (Entropy) is strictly superior for the UCI dataset. While we recommend entropy as the method to maximize test power, disagreement is a more interpretable statistic as it correlates well with the portion of misclassified samples (see (right) Figure 4.1).

## Comparison to Baselines

Amongst the baselines, there is no clear best method. Although on average, ensemble entropy is superior on CIFAR, MMD-D on Camelyon, and CTST on UCI. Our method may be thought of as a combination of ensembles, CTST, and H-Divergence. As ensembles, we leverage the variation in outputs between a set of classifiers; as CTST, we learn in a domain adversarial setting; and as H-Divergence, we compute a test statistic based on data that a model was trained on. Lastly, while MMD-D and H-Divergence were shown to be the previous state-of-the-art, their performance was validated only on larger sample sizes ( $\geq 200$ ).

## On Tabular Data

The Detectron shows promise for deployment on tabular datasets (bottom right of Table 4.2 and Figure 4.3), where (1) the computational cost of training models is low, (2) the model agnostic

nature of the Detectron is beneficial as random forests often outperform neural networks in tabular data [Borisov et al., 2021], and (3) based on our discussions with medical professionals, the ability to detect covariate shift from small test sizes is of particular interest in the healthcare domain where population shift is a constant problem burden for maintaining the reliability of deployed models.

### On Computational Cost

Our method is more computationally expensive than some existing methods for detecting shifts such as BBSD and Mahalanobis Scores, but is similar complexity to other approaches such as Ensembles, MMD-D and H-Divergence which may require training multiple deep models. However, as the Detectron leverages a pretrained model already in deployment, we find in practice that only a small number of training rounds are required to create each CDC. For instance, on CIFAR 10/10.1 a CDC ensemble of 5 models using a ResNet 18 architecture can train in under  $\approx 2m$  using an unoptimized PyTorch implementation on 1 GPU. Furthermore, looking at the runtime behavior in Figure 5.1 we see that while allowing for more computation time increases the fidelity of the Detectron, only a small number of training batches may be required to achieve a desirable level of statistical significance.

In scenarios where the deployed classifier is deemed high-risk (e.g. healthcare, justice system, education) where each data point is a decision that affects a human being, we believe the additional computational expense is justified for an accurate and sensitive assessment of whether the classifier needs updating. Having established the utility, accelerating the Detectron as well as building a deeper understanding of the runtime performance tradeoffs, is fertile ground for future work.

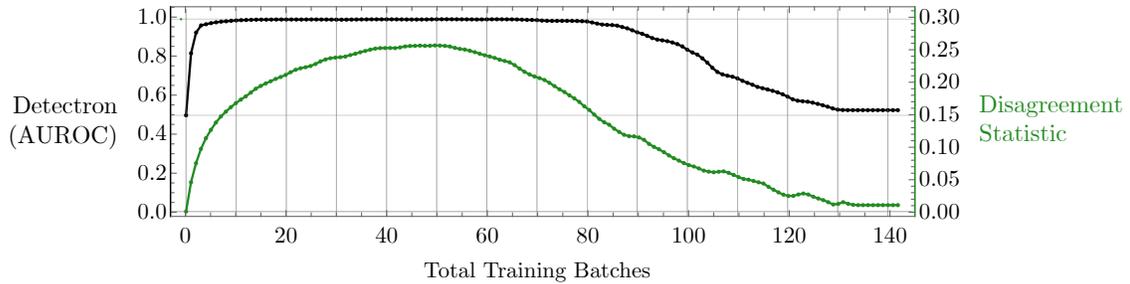


Figure 5.1: **Runtime Characteristics:** We train 100 random runs of CDCs on 100 samples from CIFAR 10 and 10.1 and compute the *disagreement statistic* as the difference  $\psi := \mathbb{E}[\phi_{\mathbf{Q}} - \phi_{\mathbf{P}}]$ . While we see that while  $\psi$  peaks near 50 training batches, only 10 batches are required for the Detectron disagreement test to reach an area under the TPR vs FPR curve (AUROC) of nearly 1 (i.e., perfect discrimination). Training CDCs for too long eventually lowers  $\psi$  as  $\mathbb{E}[\phi_{\mathbf{Q}}] \approx \mathbb{E}[\phi_{\mathbf{P}}] \approx 1$  meaning CDCs eventually overfit to disagreeing on all of their data.

# Chapter 6 Conclusion, Limitations and Future Work

## 6.1 Conclusion

Our work presents a practical application of PQ learning towards creating a method capable of detecting covariate shifts given a pre-existing classifier. On both neural networks and random forests, we showcase the efficacy of our method in being sensitive enough to detect covariate shift using a small number of unlabelled examples across several real-world datasets. We remark on several characteristics of our algorithm that represent potential directions for future work:

### On Generalization and Model Complexity

The definition of the Detectron specifies the use of the same function class for identifying shift as is used in the original prediction problem. The Detectron may fail to detect harmful shifts in cases where the base model is learned from an underspecified function class. Appendix A.3.2 provides additional context, examples and ways to mitigate this limitation. The precise relationships between model complexity, generalization error, and test power are interesting directions for future work.

### Beyond Classification

Our work here focuses on the case of classification, however, we believe there is a viable extension of our work to regression models where constrained predictors are explicitly learned to maximize test error according to the existing metric, such as mean squared error. We leave this exploration for future work.

### Beyond Covariate Shifts

While covariate shift is the only type of shift that can be discovered from unlabeled data, building learning-based methods to identify label and concept shift is another direction for future work.

## 6.2 Ethics Statement

The speed of the adoption of ML into risk scenarios raises a critical need for methods that ensure trustworthiness and reliability. However, over-reliance on such methods brings about critical ethical considerations. As we have seen, Detectron is highly sensitive to picking out discriminative features of data distributions, and, as such, its usage may prevent practitioners from deploying models in new environments. As a result, the individuals in those environments may become subject to unfair treatment. For instance, if Detectron determines a model trained on hospital A safe to deploy in hospitals B and C, but not in D, the individuals in population D may experience a lower level of care. In a real example, Detectron, when tested on a model trained on a subset of light-skinned celebrities (CelebA dataset–Liu et al. [2015]), quickly raises the alarm when given images of those that are not light-skinned. While Detectron can help mitigate potential disasters encountered by deploying models in hazardous domains, it should not be an excuse for practitioners to avoid collecting richer and more diverse datasets as a primary strategy to ensure model reliability.

# Bibliography

- A. Amini, W. Schwarting, A. Soleimany, and D. Rus. Deep evidential regression. *Advances in Neural Information Processing Systems*, 33:14927–14937, 2020.
- 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>.
- D. Chaki, A. Das, and M. I. Zaber. A comparison of three discrete methods for classification of heart disease data. *Bangladesh Journal of Scientific and Industrial Research*, 50:293–296, 2015.
- T. Chen and C. Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 785–794, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4232-2. doi: 10.1145/2939672.2939785. URL <http://doi.acm.org/10.1145/2939672.2939785>.
- 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. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- 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  $\chi^2$  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. Gal and Z. Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning, 2015. URL <https://arxiv.org/abs/1506.02142>.
- 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.
- E. Goan and C. Fookes. Bayesian neural networks: An introduction and survey. In *Case Studies in Applied Bayesian Data Science*, pages 45–87. Springer International Publishing, 2020. doi: 10.1007/978-3-030-42553-1\_3. URL [https://doi.org/10.1007/978-3-030-42553-1\\_3](https://doi.org/10.1007/978-3-030-42553-1_3).
- 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>.
- W. R. Inc. Mathematica, Version 13.0.0. URL <https://www.wolfram.com/mathematica>. Champaign, IL, 2021.
- A. Janosi, W. Steinbrunn, M. Pfisterer, and R. Detrano. Heart Disease. UCI Machine Learning Repository, 1988.
- 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.
- D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In Y. Bengio and Y. LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL <http://arxiv.org/abs/1412.6980>.

- D. P. Kingma and M. Welling. Auto-encoding variational bayes. In Y. Bengio and Y. LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL <http://arxiv.org/abs/1312.6114>.
- P. W. Koh, S. Sagawa, H. Marklund, S. M. Xie, M. Zhang, A. Balsubramani, W. Hu, M. Yasunaga, R. L. Phillips, I. Gao, T. Lee, E. David, I. Stavness, W. Guo, B. Earnshaw, I. Haque, S. M. Beery, J. Leskovec, A. Kundaje, E. Pierson, S. Levine, C. Finn, and P. Liang. Wilds: A benchmark of in-the-wild distribution shifts. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 5637–5664. PMLR, 18–24 Jul 2021. URL <https://proceedings.mlr.press/v139/koh21a.html>.
- A. Kolmogorov. Sulla determinazione empirica di una legge di distribuzione. *Inst. Ital. Attuari, Giorn.*, 4:83–91, 1933.
- A. Krizhevsky, V. Nair, and G. Hinton. The cifar-10 dataset. *online: http://www.cs.toronto.edu/kriz/cifar.html*, 55(5), 2014.
- 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.

- Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.
- 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>.
- S. Marcel and Y. Rodriguez. Torchvision the machine-vision package of torch. In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, page 1485–1488, New York, NY, USA, 2010. Association for Computing Machinery. ISBN 9781605589336. doi: 10.1145/1873951.1874254. URL <https://doi.org/10.1145/1873951.1874254>.
- 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 cifar-10 classifiers generalize to cifar-10?, 2018. URL <https://arxiv.org/abs/1806.00451>.
- 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.
- J. Ren, S. Fort, J. Liu, A. G. Roy, S. Padhy, and B. Lakshminarayanan. A simple fix to mahalanobis distance for improving near-ood detection. *ICML workshop on Uncertainty and Robustness in Deep Learning*, 2021. URL <http://www.gatsby.ucl.ac.uk/~balaji/udl2021/accepted-papers/UDL2021-paper-007.pdf>.
- 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>.
- M. Sensoy, L. Kaplan, and M. Kandemir. Evidential deep learning to quantify classification uncertainty. *Advances in neural information processing systems*, 31, 2018.
- 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  $\phi$ -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.
- B. S. Veeling, J. Linmans, J. Winkens, T. Cohen, and M. Welling. Rotation equivariant CNNs for digital pathology. June 2018.
- C. Williams and C. Rasmussen. Gaussian processes for regression. *Advances in neural information processing systems*, 8, 1995.
- M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Póczos, 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 PQ Learning & 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  $\epsilon$  and a weight  $\Lambda$ . 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).

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

Under several assumptions and a special value  $\epsilon^*$ , this selective classifier is guaranteed with high probability to have error less than  $2\epsilon^*$  on  $\tilde{\mathbf{x}}$  and a rejection rate below  $\epsilon^*$  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**  $t = 1, 2, 3, \dots$  **do**

1.  $S_t := \{x \in X : h(x) = c_1(x) = \dots = c_{t-1}(x)\}$  # So  $S_1 = X$
  2. Choose  $c_t \in C$  to maximize  $s_t(c) := \text{err}_{\tilde{\mathbf{x}}}(h|_{S_t}, c) - \Lambda \cdot \text{err}_{\mathbf{x}}(h, c)$  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  $\epsilon^*$  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

Below are the full proofs for Theorem 1 and 2 from section 3.7.

**Theorem 1** (Disagreement implies covariate shift). Let  $f$  be a classifier trained on dataset  $\mathbf{P}$  drawn 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  $\mathcal{P}$  with probability greater than  $p^* := \frac{1}{2} \left(1 - 4^{-n} \binom{2n}{n}\right)$  there must be covariate shift.

*Proof.* Let  $R_P = \emptyset_1^P + \dots + \emptyset_n^P$  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  $R_Q = \emptyset_1^Q + \dots + \emptyset_n^Q$  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.

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

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.

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

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  $R_Q > R_P$ .  $\square$

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

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

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

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

$$= (1 + p(-1 + e^{it}))^n (1 + p(-1 + e^{-it}))^n \quad (\text{A.6})$$

$$= (-p^2 e^{-it} - p^2 e^{it} + 2p^2 + p e^{-it} + p e^{it} - 2p + 1)^n \quad (\text{A.7})$$

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

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

$$= (p^2(2 - 2 \cos(t)) + p(2 \cos(t) - 2) + 1)^n \quad (\text{A.9})$$

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

$$P(Z > 0) = \frac{1}{2} (1 - P(Z = 0)) \quad (\text{A.10})$$

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]

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

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  $p \rightarrow p^* = -b(t)/(2a(t))$ . For the particular form of  $\phi_Z(t; p)$ ,  $p^*$  is simply  $1/2$

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

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  $P(Z = 0)$

$$P(Z = 0) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \phi_Z(t; p) dt \quad (\text{A.13})$$

$$\geq \frac{1}{2\pi} \int_{-\pi}^{\pi} \phi_Z\left(t; \frac{1}{2}\right) dt \quad (\text{A.14})$$

$$= \frac{1}{2\pi} \int_{-\pi}^{\pi} 2^{-n} (\cos(t) + 1)^n dt \quad (\text{A.15})$$

$$= 4^{-n} \binom{2n}{n} \quad (\text{A.16})$$

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]

$$I = \frac{1}{2\pi} \int_{-\pi}^{\pi} 2^{-n} (\cos(t) + 1)^n dt \quad (\text{A.17})$$

$$= -\frac{i}{2\pi} \oint_{|z|=1} 2^{-n} \frac{1}{z} \left(1 + \frac{1+z^2}{2z}\right)^n dz \quad (\text{using } t \rightarrow -i \log z) \quad (\text{A.18})$$

$$= -\frac{i}{2\pi} \oint_{|z|=1} 4^{-n} z^{-n-1} (z+1)^{2n} dz \quad (\text{simplifying}) \quad (\text{A.19})$$

$$= \text{Res}(4^{-n} z^{-n-1} (z+1)^{2n}, 0) \quad (\text{applying Cauchy's Theorem}) \quad (\text{A.20})$$

$$= \frac{1}{4^n n!} \lim_{z \rightarrow 0} \left( \frac{d^n}{dz^n} (z+1)^{2n} \right) \quad (\text{A.21})$$

$$= \frac{1}{4^n n!} 2n(2n-1)(2n-2) \dots (n+1) \quad (\text{A.22})$$

$$= 4^{-n} \binom{2n}{n} \quad (\text{A.23})$$

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

$$P(X > Y) = P(Z > 0) = \frac{1}{2} (1 - P(Z = 0)) \quad (\text{A.24})$$

$$\leq \frac{1}{2} \left( 1 - 4^{-n} \binom{2n}{n} \right) \quad \text{by eq. (A.16)} \quad (\text{A.25})$$

$$(\text{A.26})$$

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  $n \rightarrow \infty$  leaving a limiting tight upper bound of 1/2.  $\square$

**Theorem 2** (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  $\mathcal{P}$ :

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

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

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

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

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

$$n := \sum_{x \in \mathbf{x}} \text{dis}(x) \text{ and } m := \sum_{x \in \tilde{\mathbf{x}}} \text{dis}(x) \quad (\text{A.29})$$

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.

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

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

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

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

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

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

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

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

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

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

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

$$= \int_0^1 \int_x^1 f_{q|m}(y) f_{p|n}(x) dy dx \quad (\text{A.38})$$

$$= \binom{M}{m} \binom{N}{n} (1+M)(1+N) \int_0^1 (1-x)^{-n+N} x^n \int_x^1 (1-y)^{-m+M} y^m dy dx \quad (\text{A.39})$$

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

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

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

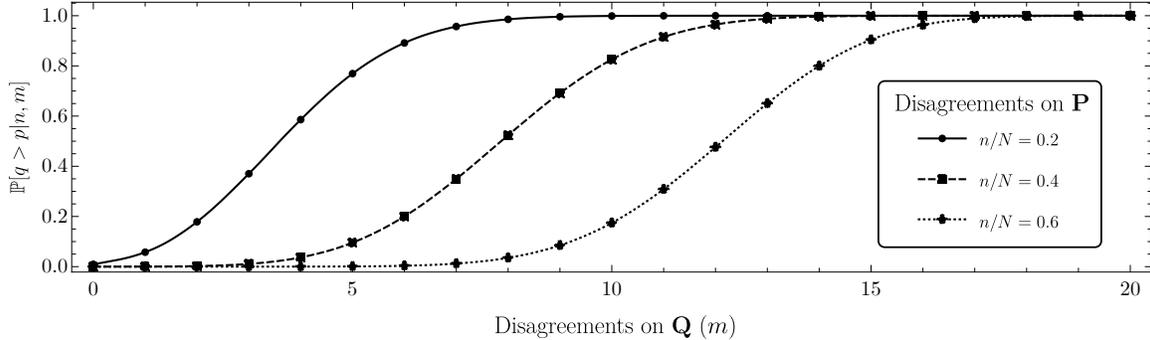


Figure A.1: Belief that the probability  $q$  that two classifiers disagree on samples from  $\mathcal{Q}$  is greater than the probability  $p$  that they disagree on  $\mathcal{P}$  given an observation of  $m$  disagreements out of  $M$  samples from  $\mathcal{Q}$  and  $n$  of  $N$  disagreements 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 than  $n/N$ .

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

□

### A.3 Constrained Disagreement Loss Details

We elaborate on the technical details of the objective functions required for training constrained disagreement classifiers.

#### A.3.1 Disagreement Cross Entropy

**Intuition.** In our methodology, we propose the *disagreement cross entropy* (DCE) as a simple loss function that encourages a classifier to disagree with a target label while otherwise outputting a high entropy prediction. DCE is equivalent to simply flipping the target label and computing the regular cross-entropy in the binary classification case.

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 A.2 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 3.6. A visual description of the DCE is provided in Figure A.2.

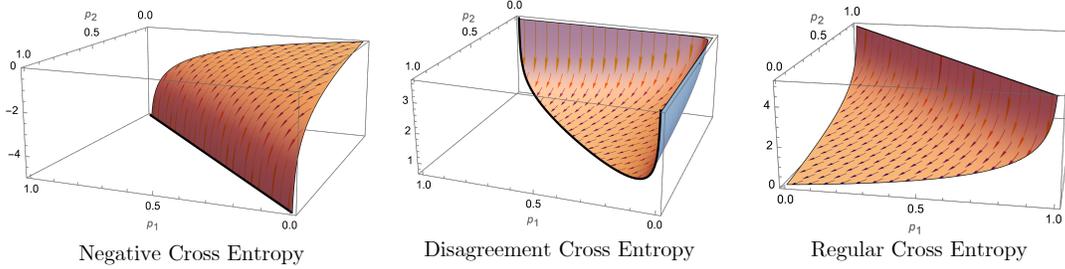


Figure A.2: Consider a classifier outputs a distribution  $\{p_1, p_2, 1 - p_1 - p_2\}$  over 3 classes. We compare the optimization landscape induced by either (left) minimizing the negative cross entropy for the target  $p_3 = 1 - p_1 - p_2$  (e.g trying not to predict class 3) (center) minimizing 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.

### Validity of the DCE Measure

We introduce a definition of a *valid disagreement loss function* and show that the DCE loss satisfies it.

**Definition 4** (A Valid Disagreement Loss Function). Let  $P$  be the set of all probability vectors of length  $N$  whose maximum index is a fixed target label  $y \in \{1, \dots, N\}$ . Similarly, let  $Q$  be the set of all probability vectors whose maximum index is **not** uniquely equal to  $y$ . For instance, in the three-dimensional case with  $y = 3$

$$P = \{(p_1, p_2, 1 - p_1 - p_2) \text{ s.t. } | \max(p_1, p_2) < 1 - p_1 - p_2\}$$

$$Q = \{(p_1, p_2, 1 - p_1 - p_2) \text{ s.t. } | \max(p_1, p_2) \geq 1 - p_1 - p_2\}$$

$$\text{and } 0 \leq p_1, p_2 \leq 1 \wedge p_1 + p_2 \leq 1$$

A score  $S(\mathbf{p}; y)$  is proper for disagreement if

$$\forall \mathbf{p} \in P \min_{\mathbf{q} \in Q} S(\mathbf{q}; y) \leq S(\mathbf{p}; y) \quad (\text{A.41})$$

which is to say that for every probability vector in  $P$  there exists a probability vector in  $Q$  that achieves a score at least as low.

**Theorem 3** (Validity of DCE). The Disagreement Cross Entropy Loss is a valid disagreement loss function.

*Proof.* We show that the minimum DCE for a probability vector  $\mathbf{q} \in Q$  is  $\log(N - 1)$  while the minimum DCE for  $\mathbf{p} \in P$  is  $\log(N)$ . First note that by definition of the DCE in Equation 3.6 the unique probability vector  $\mathbf{v}$  that globally minimizes it with respect to a target class  $y$  is the vector with elements  $1/(N - 1)$  for all indices  $\mathbf{v}_i$  where  $i \neq y$  and  $\mathbf{v}_y = 0$ .  $\mathbf{v}$  is a member of  $Q$  because it does not have a unique maximum at position  $y$ . Computing the DCE of  $\mathbf{v}$  with respect to the target  $y$  gives:

$$\begin{aligned} \text{DCE}(\mathbf{v}, y) &= \frac{1}{1 - N} \sum_{i=1}^N \log(\mathbf{v}_i) \mathbb{I}[i \neq y] \\ &= \frac{1}{1 - N} \log\left(\frac{1}{N - 1}\right) \times (N - 1) \\ &= \log(N - 1) \end{aligned}$$

Next note that since  $\mathbf{p}_y$  does not contribute to minimizing the DCE which is in the form  $\sum_{p \in \mathbf{p} \setminus \mathbf{p}_y} \log(p)$  the minimal solution must minimize the term  $\mathbf{p}_y$ . However to enforce  $\mathbf{p} \in P$  we must have that  $\forall i \in \{1, \dots, N\} \setminus y, \mathbf{p}_y > \mathbf{p}_i$ . Hence, minimizing  $\mathbf{p}_y$  while making sure  $\mathbf{p} \in P$  enforces  $\mathbf{p}_y > 1/N$  for some  $\epsilon > 0$  and all other  $p_i < 1/N$  since if  $\mathbf{p}_y \leq 1/N$  we could also chose some  $p_i \geq 1/N$  and produce a vector  $\mathbf{p} \in Q$ . Without loss of generality choosing  $\mathbf{p}_y = 1/N + \epsilon$  for some small  $\epsilon > 0$  we must chose all other  $\mathbf{p}_i = 1/N - \epsilon_i$  s.t. all  $\epsilon_i > 0$  and  $\sum_{i \in \{1, \dots, N\} \setminus y} \epsilon_i = \epsilon$ . This gives a DCE of:

$$\begin{aligned} \text{DCE}(\mathbf{p}, y) &= \frac{1}{1 - N} \sum_{i=1}^N \log(\mathbf{p}_i) \mathbb{I}[i \neq y] = \frac{1}{1 - N} \sum_{i \in \{1, \dots, N\} \setminus y} \log\left(\frac{1}{N} - \epsilon_i\right) \\ \text{where } \lim_{\epsilon \rightarrow 0} \frac{1}{1 - N} \sum_{i \in \{1, \dots, N\} \setminus y} \log\left(\frac{1}{N} - \epsilon_i\right) &= \log(N) \end{aligned}$$

For all vectors  $\mathbf{p} \in P$  we cannot achieve a DCE of less than  $\log(N)$  but the minimum DCE in  $Q$  is  $\log(N - 1)$  which is less than  $\log(N)$ , hence proving that the DCE is a valid scoring rule for

disagreement. □

### Implementation

We provide a simple PyTorch style implementation for batched computation of the CDC objective in Equation 3.7 given batch of logits and targets. We assume `logits` is a floating point vector of  $(\text{batch\_size} \times N)$  logit values, `targets` is a integer vector (`batch_size`) of target classes, `mask` is a 0-1 mask of (`batch_size`) that is 1 at index  $i$  if `logits[i]` is  $\mathbf{P}$  and 0 otherwise.

```
import torch, import torch.nn.functional as F
def cdc_loss(logits, targets, mask, weight=1/(size of test set + 1)):
    # select data in P
    logits_p, targets_p = logits[mask], targets[mask]
    # select data in Q
    logits_q, targets_q = logits[1-mask], targets[1-mask]
    # cross entropy on P with targets
    loss_p = F.cross_entropy(logits_p, targets_p, reduction=None)
    # DCE on Q
    zero_hot = 1 - F.one_hot(targets_q, num_classes=N)
    loss_q = (logits_q * zero_hot).sum(dim=1) / (1 - N)
    + torch.logsumexp(logits_q)
    # Weighted mean
    return torch.cat([loss_p, weight * loss_q]).mean()
```

### A.3.2 Disagreement with Overparameterized Models

Many existing tests for covariate shift that rely on overparameterized models do not perform well in the small sample regime due to catastrophic overfitting. In this section, we explain how this phenomenon, if ignored, can also be catastrophic for the Detectron but is easily fixable with a simple early stopping technique. We recall the main hypothesis on which the Detectron is built:

*Given a base classifier  $f \in F$  that is well fit to a training dataset  $\mathbf{P}$ , it is easier to learn another classifier  $g \in F$  that disagrees with  $f$  on unlabeled data  $\mathbf{Q}$ , but agrees with  $f$  on  $\mathbf{P}$  when the distribution of  $\mathbf{Q}$  is far from that of  $\mathbf{P}$ .*

Restated with the ideas introduced in this work, the above translates to the CDC objective being more easily satisfied when there is a harmful shift. This hypothesis is exemplified in Figure A.3 to the right, where we see that the CDC disagreement rate grows significantly more rapidly when  $\mathbf{Q}$  is chosen to be out-of-distribution (blue curve) versus in distribution (black curve). However, as the models are over-parameterized, the disagreement rate eventually reaches 1 independently of whether we use the true OOD data  $\mathbf{Q}$  or the ID data  $\mathbf{P}^*$ . Since our test is computed based on the disagreement rate  $\phi_{\mathbf{Q}}$  falling above  $1 - \alpha$  quantile of the calibration distribution of  $\phi_{\mathbf{P}}$  the test will be highly uninformative if  $\phi_{\mathbf{Q}} = \phi_{\mathbf{Q}} = 1$ .

Note that it is critical to use the exact same training algorithm on both the given test set  $\mathbf{Q}$  and the in-distribution calibration set  $\mathbf{P}^*$  to ensure the statistical soundness of the Detectron, so one can tune the CDC learning algorithm on either  $\mathbf{Q}$  or  $\mathbf{P}^*$  to achieve a desired result then apply the exact same algorithm to the other set and test for statistical differences. In our experiments with overparameterized models, we picked a relatively low number of training iterations to use (see subsection A.4.2 for details) practically eliminating the issue of overfitting. However, in general, one should fix a training budget for CDCs so that they are far from reaching an ID disagreement rate of  $\phi_{\mathbf{P}} = 1$ . Knowing the exact level of  $\phi_{\mathbf{P}}$  to stop at will depend on the dataset and learning algorithm used but as a general guideline we suggest  $\approx 0.5$  to allow for a large gap between ID and OOD disagreement rates while giving enough of a compute budget to achieve non-trivial results.

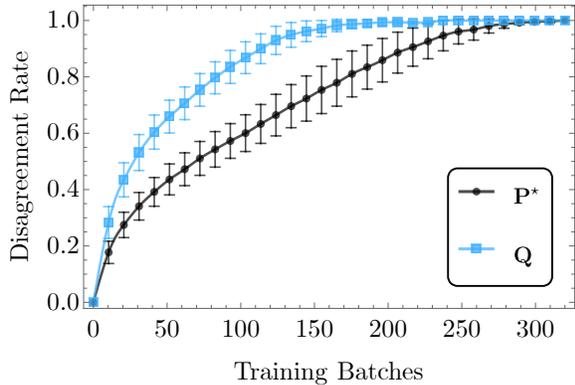


Figure A.3: **CDC Training Dynamics for Over-parameterized Models:** Using the same experimental setup as the runtime study in Figure 5.1 we record the associated CDC disagreement rate on both unseen in distribution data  $\mathbf{P}^*$  (CIFAR 10) and a known covariate shift  $\mathbf{Q}$  (CIFAR 10.1). When we trained such an overparameterized model for too long the disagreement rate approaches 1 on both in and out-of-distribution datasets leading to a maximally uninformative test; it is, therefore, crucial to perform early stopping at or before the out-of-distribution disagreement rate reaches 1.

## A.4 Experimental Details

### A.4.1 Base Classifiers

For each dataset used in our experiments, we begin by training a *base classifier* on the source domain portion of the dataset to use in subsequent experiments and baselines. For a brief description of the datasets used and base classifiers, see ?? and for a more detailed description of each dataset as well as what we have considered precisely as the source and shifted domains, see subsection A.5.2.

**CIFAR 10.** We use a standard Resnet18 model pre-trained on ImageNet [Deng et al., 2009] made available in the torchvision library [Marcel and Rodriguez, 2010] although we reinitialize the last network layer to have an output size of 10. We use stochastic gradient descent (SGD) with a base learning rate of 0.1,  $L_2$  regularization of  $5 \times 10^{-4}$ , momentum of 0.9, a batch size of 128 and a cosine annealing learning rate schedule with a maximum 200 iterations stepped once per epoch for a total of 200 epochs. We use the standard CIFAR-10 training split normalized by its mean ( $\mu = [0.4914, 0.4822, 0.4465]$ ) and standard deviation ( $\sigma = [0.2023, 0.1994, 0.2010]$ ). Every epoch, we randomly crop each image to a size of  $32 \times 32$  after applying a 0 padding of four pixels to each spatial dimension, and we apply a horizontal flip with probability 0.5. This model achieves a test performance of 87%. While this score is far from state-of-the-art on CIFAR-10, our goal is not to construct a perfect model. We wish to create a *reasonably good* model as an example of a model that could realistically be deployed in real-world settings. When training deep ensembles, we only vary the random seed in the range  $[0, \dots, 4]$ .

**Camelyon 17.** We follow a similar approach to CIFAR 10. However, we use two output features (for binary classification of cancerous or benign pathology), a batch size of 512, the ADAM optimizer [Kingma and Ba, 2015] with a base learning rate of 0.001,  $L_2$  regularization of  $10^{-5}$  and a total of 5 training epochs for which we select the model with the best validation accuracy. This model achieves a test accuracy of 0.93. When training deep ensembles, we only vary the random seed in the range  $[0, \dots, 4]$ .

**UCI Heart Disease.** We train both neural networks and gradient boosted trees using the XGboost library [Chen and Guestrin, 2016]. For the neural network model, we use a simple MLP with an input dimension of 9, 3 hidden layers of size 16 with ReLU activation followed by a 30% dropout layer and a linear layer to 2 outputs (heart disease present or not). We use 358 samples for training and 120 for validation. We train for a maximum of 1000 epochs and select the model with the highest AUC on the validation set, performing early stopping if the validation AUC has not increased in over 100

epochs. This model achieves a test AUC computed on 119 samples of 0.85. As with CIFAR 10 and Camelyon 17, we only vary the random seed in the range  $[0, \dots, 9]$  when training deep ensembles. Note that we chose a larger ensemble size here as models are fairly cheap to train. Another important trick when using small  $\mathbf{Q}$  sizes is to sample all of  $\mathbf{Q}$  in each batch filling the best with a random set of samples from  $\mathbf{P}$ . Our procedure artificially inflates the size of  $\mathbf{Q}$  so the hyperparameter  $\lambda$  must account for this by picking up an extra multiplicative factor equal to  $(\text{batches per epoch})^{-1}$ .

When training gradient boosted trees using XGboost we employ standard library parameters ( $\eta = 0.1$ , `eval_metric=auc`, `max_depth= 6`, `subsample= 0.8`, `colsample_bytree=0.8`, `min_child_weight= 1`, `objective=binary:logistic`, `num_round= 10`). This model while taking less than 5s to train achieves a test AUC of 0.88.

#### A.4.2 Constrained Disagreement Classifiers

We expand on the experimental details for learning constrained disagreement classifiers. When training a CDC  $g_{(f, \mathbf{P}, \mathbf{Q})}$  we start by creating a new dataset that combines all elements of the labeled set  $\mathbf{P}$  and the unlabeled set  $\mathbf{Q}$  with pseudo labels inferred by the base classifier  $f$ . We store a single bit for each sample in the combined dataset to indicate if a sample was originally drawn from  $\mathbf{P}$  or  $\mathbf{Q}$ . When training CDCs with neural networks, we use the DCE loss (*Equation 3.7*) under similar semantics as the pseudo-code implementation provided above. When training discrete models, we resort to our generalized approach in subsection 3.6.2. To reduce training time we initialize  $g$  using the exact same architecture/weights as  $f$  and apply the exact same optimization algorithm/learning rate used to train  $f$  (see subsection A.4.1). For CIFAR 10, we train each CDC for a maximum of 10 epochs performing early stopping if the model drops in in-distribution validation performance by over 5%.

We enforce the early stopping criteria to help prevent CDCs from overfitting to the disagreement loss when the target dataset has not come from a harmfully shifted domain. The intuition is the following: under the null, if a target dataset  $\mathbf{Q}$  comes from the same distribution as a training dataset  $\mathbf{P}$ , then learning to disagree with  $f$  on  $\mathbf{Q}$  while constrained to agree on all of  $\mathbf{P}$  can only be solved by overfitting to predict with high entropy on the specific examples in  $\mathbf{Q}$ , versus learning a distinct pattern that distinguishes the distributions. Forcing a model to predict with high entropy on a subset of in-distribution datapoints can only hurt its associated in-distribution generalization, a phenomenon which we can directly assess by measuring validation performance.

The details for training CDCs on Camelyon 17 are the same as those described for CIFAR 10,

however due to the large training set size (302436 samples) we simply select a random subset of size 50,000 as  $\mathbf{P}$  at each epoch – a number we experimentally deemed as sufficient to achieve low in-distribution generalization error. When training CDCs on the UCI Heart Disease dataset, we use XGBoost [Chen and Guestrin, 2016] with the same hyperparameters described in subsection A.4.1.

For the runtime experiment presented in Figure 5.1 we train each CDC for only one batch, where each batch contains a set of 100 samples  $\mathbf{Q}$  and is filled up to a batch size of 512 with random samples from  $\mathbf{P}_{\text{train}}$ . After every batch we eliminate all samples where the CDC disagrees with the base predictions. We continue this for a maximum of 150 batches, but perform early stopping if 10 batches pass without at least one sample getting disagreed on.

### A.4.3 Description of Baseline Methods

We compare the Detectron against several methods for OOD detection, uncertainty estimation and covariate shift detection found in recent literature.

1. *Deep Ensembles* shown by Ovadia et al. [2019] to provide the most accurate estimates of predictive confidence under covariate shift. To compare directly with Detectron we test both the disagreements rates and the entropy distributions of the ensemble. For disagreement, we use a binomial test against a success probability estimated from the iid validation data  $\mathbf{P}^*$ , and for entropy we use a KS-test between the ensemble entropy as computed by Equation 3.14 (but using a regular ensemble instead of a CDC ensemble) on  $\mathbf{Q}$  vs  $\mathbf{P}^*$ .
2. *Black Box Shift Detection (BBS)* [Lipton et al., 2018] is overall best method across numerous synthetic benchmarks for covariate shift detection evaluated by Rabanser et al.. We follow the same evaluation and perform a univariate KS test on each dimension of the softmax output of the base classifier between  $\mathbf{Q}$  and a held out set from the training distribution. Bonferroni correction is used to compute a single  $p$ -value as the minimum value divided by the number of tests. We guarantee significance using the same permutation approach described in section 3.8.
3. *Relative Mahalanobis Distance (RMD)* [Ren et al., 2021] (a method designed specifically for identifying near OOD samples) using the penultimate layer of a pretrained model. We test for covariate shift by performing a KS test directly on the distribution of RMD confidence scores derived on  $\mathbf{Q}$  and  $\mathbf{P}^*$ .
4. *Classifier two sample test (CTST)* [Lopez-Paz and Oquab, 2017]. Using the same architecture as and initialization as the base classifier we reconfigure the output layer, and we train a

domain classifier on half the test data with source data labeled as 0 and test data as 1. We then test this models accuracy on the other half of the test data and compare its performance to random chance using a binomial test. While this method is technically sound it is not suitable for the low data regime where learning a domain classifier on half the test data is unlikely to generalize beyond random performance on the other half.

5. *Deep Kernel MMD* [Liu et al., 2020]. We use the authors original source code available at <https://github.com/fengliu90/DK-for-TST> to perform the deep kernel MMD test.
6. *H-Divergence* [Zhao et al., 2022]. Most similar to our approach, this work proposes a two sample test based on the output of a learning model after training on either source or target data. Specifically, the authors fit a model to both the source dataset  $\mathcal{P}$ , the target dataset  $\mathcal{Q}$  and a uniform mixture  $(\mathcal{P} + \mathcal{Q})/2$ . Under the null hypothesis  $\mathcal{P} = \mathcal{Q}$  the loss in each case is equal in expectation. However, when  $\mathcal{P} \neq \mathcal{Q}$ , the generalized entropy of the mixture distribution may be be larger. In practice the authors fit three VAE [Kingma and Welling, 2014] models and compute the test statistic  $\ell((\mathcal{P} + \mathcal{Q})/2) - \min(\ell(\mathcal{Q}), \ell(\mathcal{P}))$ , where  $\ell$  is the VAE loss computed as a sum of the binary cross entropy reconstruction loss and the KL divergence regularizer. We perform 100 runs where the null hypothesis (e.g. sample  $\mathbf{Q}$  from  $\mathcal{P}$ ) and one where it does not. Significance is determined in the standard way be observing if the true test statistic exceeds the 95<sup>th</sup> percentile of the test statistic distribution under the null hypothesis. Unfortunately this method, while state of the art on several benchmarks including the MNIST vs Fake MNIST two sample test, demonstrated low utility on more complex tasks with smaller sample sizes. After a discussion with the authors, we attempted to improve the results by first pre-training the VAE to produce valid samples and reconstructions under the source distribution and computing the H-Divergence statistic after fine-tuning. Despite this effort, we still was low statistical significance with small sample sizes likely due to the noisy nature of training VAE’s in the low data regime. We use the authors original source code available here <https://github.com/a7b23/H-Divergence>.

## A.5 Datasets

### A.5.1 Sources and Licensees

- **CIFAR-10** (MIT License Copyright (c) 2013 Valay Shah)

- **Camelyon-17** (CC0 1.0 Universal Public Domain Dedication)
- **UCI Heart Disease** (Creative Commons Attribution 4.0 International)

### A.5.2 Preprocessing, Shift Descriptions and Model Performance

We provide full details on the three datasets used in our experiments, including any preprocessing steps and what splits we considered as source domain  $\mathcal{P}$  and target domain  $\mathcal{Q}$ .

**CIFAR 10/10.1.** We use the well known CIFAR 10 dataset [Krizhevsky et al., 2014] as the source domain for training base classifiers (subsection A.4.1) and the new CIFAR 10 test set CIFAR 10.1 containing 2000 class balanced images [Recht et al., 2019] as a source of harmful distribution shift. Although the images in CIFAR 10.1 appear to be visually very similar to CIFAR 10 most classifiers trained on CIFAR 10 drop significantly in performance (3% to 15% [Recht et al., 2019]) when tested on CIFAR 10.1.

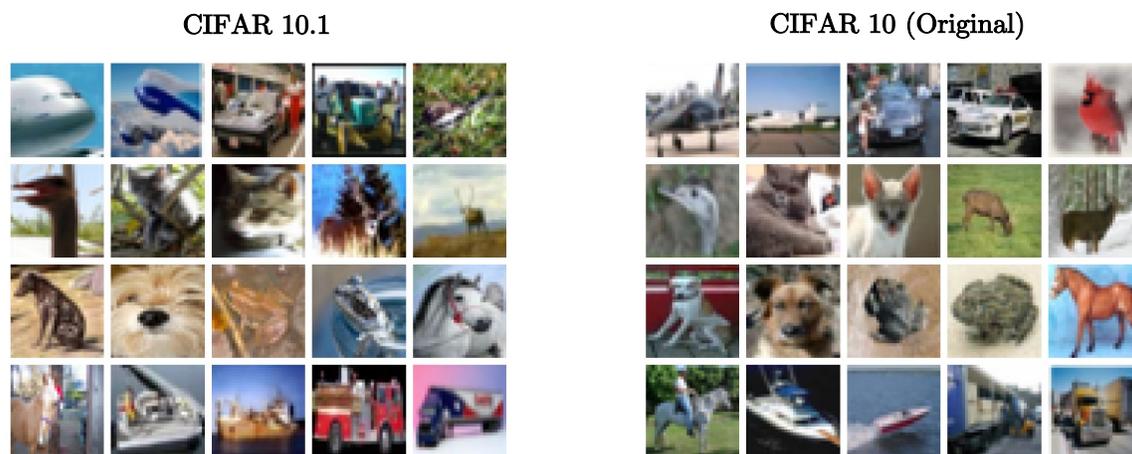


Figure A.4: Cifar 10 vs Cifar 10.1. (Image borrowed from the technical report ”Do CIFAR-10 Classifiers Generalize to CIFAR-10?” [Recht et al., 2018])

**Camelyon 17.** As described by the original authors [Veeling et al., 2018] the Camelyon benchmark is a new and challenging image classification dataset consisting of 327,680 color images ( $96 \times 96$ ) extracted from histopathologic scans of lymph node sections. Each image is annotated with a binary label indicating the presence of metastatic tissue in the center  $32 \times 32$  pixel region. In our experiments, we use the WILDS [Koh et al., 2021] framework to facilitate download, preprocessing, as well as source/target splits for Camelyon. As shown in Figure A.5, the source domain is chosen as data from hospitals 1, 2, 3. In contrast, the test domain is collected from hospital 5, which visually shows significantly higher contrast due to different data acquisition equipment/methods.

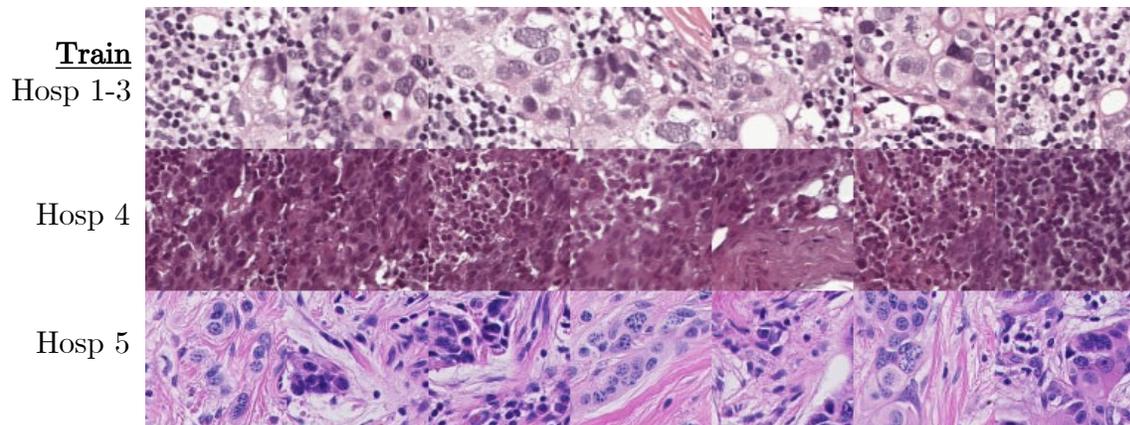


Figure A.5: Samples images from the Camelyon 17 dataset [Veeling et al., 2018]. Using the standard set by the WILDS framework [Koh et al., 2021] we use hospitals 1–3 as the source domain for training and validating models, and hospital 5 as the target domain for assessing distribution shift.

**UCI Heart Disease.** The UCI Heart Disease (UCI-HD) dataset [Janosi et al., 1988] consists of 76 attributes collected from four unique patient databases in Cleveland, Hungary, Switzerland, and the VA Long Beach. We select nine features out of the commonly used 14 to minimize the portion of missing values. These features are {age, sex, chest pain type, resting blood pressure, serum cholesterol, fasting blood sugar, resting electrocardiographic results, maximum heart rate achieved, exercise-induced angina}. The prediction task is to determine the diagnosis of heart disease (also known as angiographic disease status), which is given in a range from 0–4, where 0 indicates healthy and 1–4 indicates a severity level based on the narrowing of major blood vessels. Following prior work [Chaki et al., 2015] we only consider the simplified binary classification task for differentiating patients with a normal angiographic status (label of 0) from those with abnormal status (label  $> 0$ ). We select the source domain as the Cleveland and Hungary databases and the target domain as the Switzerland and VA Long Beach databases. A graphical overview of the marginal feature distributions for the source and target domain is shown in Figure A.6.

To allow for out-of-the-box training of deep neural networks on the UCI HD dataset, we use the missing value synthesis functional in Wolfram Mathematica [Inc.]. The algorithm uses density estimation and mode finding on conditioned distributions to synthesize missing values. See the language guide page titled [Synthesize Missing Values in Numeric Data](#) for a more detailed description. To aid in future research, we provide a copy of our processed dataset in [here](#).

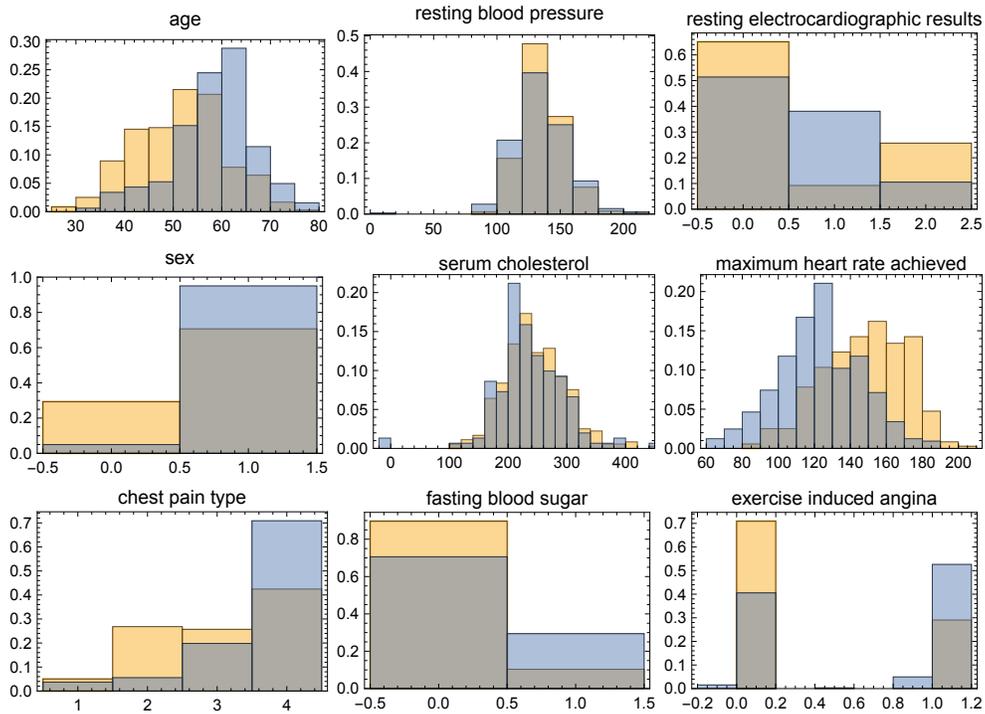


Figure A.6: Marginal distributions for each of the nine variables chosen from the UCI Heart Disease dataset for our experiments. The source domain (yellow) is chosen from the Cleveland and Hungary patient databases, while the shifted target domain (blue) is selected from the Swiss and Long Beach databases. Although the distributions are visually similar, a simple neural network classifier that achieves an AUC of 0.85 on the source domain drops to 0.42 on the target domain.