MANDOLINE: Model Evaluation under Distribution Shift

Mayee Chen [*1] **Karan Goel** [*1] **Nimit Sohoni** [*2] **Fait Poms** [1] **Kayvon Fatahalian** [1] **Christopher Ré** [1]

Abstract

Machine learning models are often deployed in different settings than they were trained and validated on, posing a challenge to practitioners who wish to predict how well the deployed model will perform on a target distribution. If an unlabeled sample from the target distribution is available, along with a labeled sample from a possibly different source distribution, standard approaches such as importance weighting can be applied to estimate performance on the target. However, importance weighting struggles when the source and target distributions have non-overlapping support or are high-dimensional. Taking inspiration from fields such as epidemiology and polling, we develop MANDOLINE, a new evaluation framework that mitigates these issues. Our key insight is that practitioners may have prior knowledge about the ways in which the distribution shifts, which we can use to better guide the importance weighting procedure. Specifically, users write simple "slicing functions"—noisy, potentially correlated binary functions intended to capture possible axes of distribution shift—to compute reweighted performance estimates. We further describe a density ratio estimation framework for the slices and show how its estimation error scales with slice quality and dataset size. Empirical validation on NLP and vision tasks shows that MANDOLINE can estimate performance on the target distribution up to 3× more accurately compared to standard baselines.

1 Introduction

Model evaluation is central to the machine learning (ML) pipeline. Ostensibly, the goal of evaluation is for practitioners to determine if their models will perform well when deployed. Unfortunately, standard evaluation falls short of this goal on two counts. First, evaluation data is frequently from a different distribution than the one on which the model will be deployed, for instance due to data collection procedures or distributional shifts over time. Second, practitioners play a passive role in evaluation, which misses an opportunity to leverage their understanding of what distributional shifts they expect and what shifts they want their model to be robust to.

By contrast, fields such as polling (Isakov & Kuriwaki, 2020) and epidemiology (Austin, 2011) "adjust" evaluation estimates to account for such shifts using techniques such as propensity weighting, correcting for differences between treatment and control groups in observational studies (Rosenbaum & Rubin, 1983; D'Agostino Jr, 1998).

Taking inspiration from this, we develop MANDOLINE (Figure 1), a user-guided, theoretically grounded framework for evaluating ML models. Using a labeled validation set from a source distribution and an *unlabeled* set from a target (test) distribution of interest, MANDOLINE computes performance estimates for the target distribution using adjusted estimates on the validation set.

A central challenge is how to account for the shift between source and target distributions when adjusting estimates. A straightforward approach is to use importance weighting (IW)—estimating the density ratio between source and target data to adjust performance. IW works well when the source and target distribution overlap significantly but performs poorly when the distributions' supports have little overlap, as is common under distribution shift. Additionally, IW works well in low dimensions but struggles with large variances of estimated ratios in high dimensional settings (Stojanov et al., 2019).

Our key insight is that practitioners can use their understanding to identify the axes along which the distributions may have changed. Practitioners commonly express this knowledge programmatically by grouping ("slicing") data along such axes for analysis. For example, in sentiment analysis a heuristic may use the word "not" to detect sentence negation (Figure 1). Or, slices can identify critical data subsets (e.g. X-rays of critically-ill patients with chest drains (Oakden-Rayner et al., 2020) or demographic slices when detecting

[*] Equal contribution [1] Department of Computer Science, Stanford University, Stanford, USA [2] Institute for Computational and Mathematical Engineering, Stanford University, Stanford, USA. Correspondence to: Mayee Chen <mfchen@stanford.edu>, Karan Goel <kgoel@cs.stanford.edu>, Nimit Sohoni <nims@stanford.edu>.

Proceedings of the 38^{th} *International Conference on Machine Learning*, PMLR 139, 2021. Copyright 2021 by the author(s).

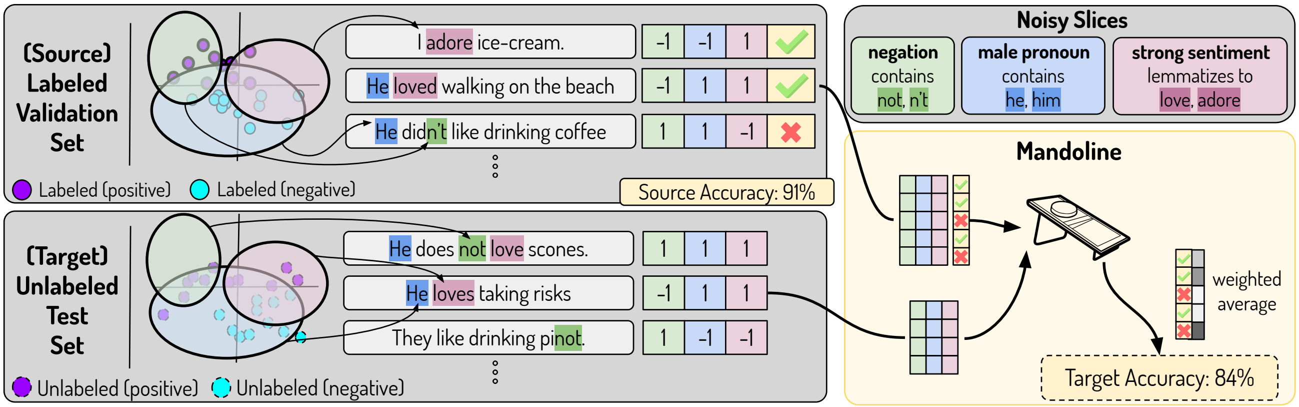

Figure 1. Schematic outlining the MANDOLINE workflow for a sentiment analysis task. *(left)* Given labeled source and unlabeled target data, users write noisy, correlated slicing functions (indicated by color) to create a slice representation for the datasets. *(right)* MANDOLINE uses these slices to perform density ratio estimation. Then, it uses these ratios to output a performance estimate for the target dataset by reweighting the source data.

toxic comments (Borkan et al., 2019)).

MANDOLINE leverages precisely this information: users construct "slicing functions" on the data—either programmatically mapping examples to slices, or using metadata associated with examples to group them. These slices create a common representation in which to project and compare source and target data, reducing dimensionality and mitigating support shift. Compared to standard IW, MANDOLINE simplifies density ratio estimation by relying on source and target slice statistics, rather than raw features.

When the slices accurately capture the relevant distribution shift, MANDOLINE can be instantiated with many standard density ratio estimation methods on the slices. However, since practitioners rely on heuristics to write slicing functions, slices can be noisy, correlated, and incomplete, and direct density ratio estimation does not handle this possible misspecification. We thus represent the distribution of slices as a graphical model to incorporate knowledge of their correlations and incompleteness, and provide a novel extension of the LL-KLIEP method (Tsuboi et al., 2009) that can denoise the density ratio based on the practitioner's confidence and prior knowledge of slice quality. These density ratios are then used to generate importance weights for evaluating models.

Theoretically, we provide a bias-variance decomposition of the error of our reweighting approach. The bias depends on how user-specified slices capture distribution shift, and variance depends on distribution properties and the amount of data. Compared to standard IW, which lacks theoretical guarantees when the variance of the weights is unbounded (Cortes et al., 2010), our results always hold and can describe when MANDOLINE will yield better estimates.

Empirically, we verify that MANDOLINE outperforms standard baselines for density ratio estimation on both synthetic and real-world tasks from natural language processing and computer vision. When slices completely capture shifts without noise, MANDOLINE reduces estimation error by up to 3× compared to standard IW baselines. Even with noisy slices, MANDOLINE exhibits little performance degradation. When slices are under-specified and noisy and do not completely capture large distributional shifts, MANDOLINE continues to outperform baselines by up to 2×.

We conclude with a discussion of slice design. We explore an extremely challenging under-specified setting, where we highlight how clever slice design can reduce estimation error by 2.86×, and that MANDOLINE can flexibly incorporate a strong IW baseline as a slice to match its performance. We also show that MANDOLINE can closely estimate performance drops as large as 23% accuracy points in an "in-the-wild" sentiment analysis setting, which shows the potential of automatically designing slicing functions.[1]

2 Background

We first provide background on IW and density ratio estimation (Section 2.1), and the challenges these approaches face. We then provide a formal setup of our problem of evaluating models under distribution shift (Section 2.2).

Notation. $\mathcal{P}_s$ and $\mathcal{P}_t$ are source and target distributions with respective densities p_s and p_t. $\mathbb{E}_s$ and $\mathbb{E}_t$ are expectations with respect to $\mathcal{P}_s$ and $\mathcal{P}_t$. When a statement applies to both distributions, we refer to their densities collectively as p.

[1] Code for MANDOLINE can be found on GitHub.

2.1 Importance Weighting

Importance weighting (Horvitz & Thompson, 1952) is a general approach for estimating properties of a target random variable X drawn from $\mathcal{P}_t$ given samples $\{x_i\}_{i=1}^n$ from a different source distribution $\mathcal{P}_s$. Since $\mathbb{E}_s\left[\frac{p_t(X)}{p_s(X)}f(X)\right] = \mathbb{E}_t\left[f(X)\right]$ for any function f when supp($\mathcal{P}_t$) $\subseteq$ supp($\mathcal{P}_s$), one can estimate $\mathbb{E}_t\left[f(X)\right]$ with the empirical average $\frac{1}{n}\sum_{i=1}^n \frac{p_t(x_i)}{p_s(x_i)}f(x_i)$. Typically p_s, p_t are unknown, so the density ratio $\frac{p_t(X)}{p_s(X)}$ must be estimated.

Density ratio estimation. The challenge of how to estimate density ratios is well-studied. Estimation of the individual densities to compute the ratios is possible but can lead to poor estimates in high dimensions (Kpotufe, 2017). Instead, most approaches estimate the ratio directly. We discuss several common methods below, although they can all be generalized to fitting the density ratio under the Bregman divergence (Sugiyama et al., 2012b). First, classifier-based approaches (CBIW) use a Bayesian "density-ratio trick" (Hastie et al., 2001; Sugiyama et al., 2012a; Mohamed & Lakshminarayanan, 2016) to cast estimation as binary classification between samples from the source $(z = 0)$ and target $(z = 1)$ distributions. The learned $\frac{p(z=1|x)}{p(z=0|x)}$ is then rescaled to produce a ratio estimate. Kernel mean matching (KMM) matches moments from $\mathcal{P}_t$ to a (parameterized) transformation of the moments of $\mathcal{P}_s$ in a Reproducing Kernel Hilbert Space (RKHS) e.g. with the Gaussian kernel (Gretton et al., 2009). Least-squares importance fitting (LSIF) directly fits the density ratio by minimizing the squared error of a parametrized ratio $r_\phi(x)$ (typically linear or kernel model) compared to $\frac{p_t(x)}{p_s(x)}$ (Kanamori et al., 2009). Finally, the Kullback-Leibler importance estimation procedure (KLIEP) uses $r_\phi(x)$ to construct an approximate distribution $\hat{\mathcal{P}}_t = r_\phi \mathcal{P}_s$ and minimizes the KL-divergence between $\hat{\mathcal{P}}_t$ and $\mathcal{P}_t$ (Sugiyama et al., 2008).

Challenges for IW**.** A common problem when applying IW is *high-variance weights*, which result in poor performance both theoretically and in practice (Cortes et al., 2010). While simple techniques such as self-normalization and weight clipping can be used to reduce variance (Grover et al., 2019), these heuristics do not address the cause of the variance. Instead, we highlight and address two challenges that underlie this problem in IW—learning from high-dimensional, complex data and handling support shift:

1. *High-dimensional data.* Dealing with high-dimensional data is challenging in density ratio estimation, as it is difficult to find well-specified model classes (for CBIW) or data representations (for KMM, LSIF, KLIEP). To remedy this, dimensionality reduction can be used when the distribution shift is restricted to some low-dimensional structured subspace (Sugiyama et al., 2010), but these approaches generally assume a linear subspace.
2. *Support shift.* When there exists some x such that $p_s(x) > 0$ but $p_t(x) = 0$, then $\frac{p_t(x)}{p_s(x)} = 0$. This point is essentially discarded, which reduces the effective number of samples available for IW, and points in the intersection of the support may also have low $p_s(x)$, which results in overweighting a few samples. When there exists some x such that $p_s(x) = 0$ but $p_t(x) > 0$, this x will never be considered in the reweighting—in fact, $\mathbb{E}_s\left[\frac{p_t(X)}{p_s(X)}f(X)\right]$ will *not* equal $\mathbb{E}_t\left[f(X)\right]$ in this case, rendering IW useless in correcting distribution shift. We describe these phenomena as support shift.

2.2 Problem Formulation

We are given a fixed model $f_\theta : \mathcal{X} \to \mathcal{Y}$, a labeled "validation" source dataset $\mathcal{D}_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$, and an unlabeled "reference" target dataset $\mathcal{D}_t = \{x_i^t\}_{i=1}^{n_t}$. $\mathcal{X}, \mathcal{Y}$ denote the domains of the x's and y's, and $\mathcal{D}_t$ and $\mathcal{D}_s$ are drawn i.i.d from $\mathcal{P}_t(\cdot|Y)$ and $\mathcal{P}_s$, respectively. We assume there is no concept drift between the distributions, meaning $p_t(Y|X) = p_s(Y|X)$. Define $\ell_\theta : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ to be a metric of performance of f_θ. Our goal is to evaluate performance on the target population as $\mathcal{L}_t$=$\mathbb{E}_t\left[\ell_\theta(X, Y)\right]$ using labeled samples from $\mathcal{D}_s$ and unlabeled samples from $\mathcal{D}_t$.

Standard IW can estimate $\mathcal{L}_t$ using the density ratio of the features as $\frac{1}{n_s}\sum_{i=1}^{n_s} \frac{p_t(x_i^s)}{p_s(x_i^s)}\ell_\theta(x_i^s, y_i^s)$ using the assumption of no concept drift. However, IW has shortcomings (as discussed in Section 2.1). For instance, it is often *beneficial* to ignore certain features, as the below example illustrates.

Example 2.3 Suppose $p_s(x_1, x_2) \propto \phi_{(\mu_1,\sigma_1^2)}(x_1) \cdot \mathbf{1}(x_2 \in (0, 1))$ and $p_t(x_1, x_2) \propto \phi_{(\mu_2,\sigma_2^2)}(x_1) \cdot \mathbf{1}(x_2 \in (-1, 0))$, where ϕ_{μ,σ^2} is the $\mathcal{N}(\mu, \sigma^2)$ density. x_1 is normal while x_2 is uniformly distributed on $(0, 1)$ and $(-1, 0)$ under p_s and p_t respectively, and x_1, x_2 are independent. Suppose $\ell_\theta(x, y)$ is independent of x_2. The IW estimate of $\mathbb{E}_t\left[\ell_\theta(X, Y)\right]$ will always be zero since $p_t(x) = 0$ for all x in the support of p_s. IW fails due to support shift of the irrelevant x_2, but this can be mitigated by weighting only on x_1, motivating our framework.

3 The MANDOLINE Framework

We present a framework of assumptions (Section 3.1) on $\mathcal{P}_s$ and $\mathcal{P}_t$ that motivate user-specified *slicing functions*, which intend to capture relevant distribution shift. Under these assumptions, we show that weighting based on accurate slicing functions is equivalent to weighting based on features, but mitigates the challenges that standard IW faces by ignoring irrelevant and non-shifting components of the distributions (Section 3.2). We then present a novel density ratio estimation algorithm based on KLIEP (Section 3.3) that accounts for noisy slices.

3.1 Modeling distribution shift

Assume that each sample X can be represented via mappings to four sets of variables $g(X), h(X), a(X), b(X)$. This categorizes information about the data depending on if it is relevant to the learning task and if its distribution changes between $\mathcal{P}_s$ and $\mathcal{P}_t$. $g(X)$ contains relevant properties of the data that are known to the user and undergo distribution shift. $h(X)$ represents "hidden" properties of the data that are also relevant and shifting, but which the user fails to model. $a(X)$ corresponds to the properties that exhibit shift but are irrelevant to the task. Lastly, $b(X)$ are the properties that do not undergo any shift. We state these assumptions formally below.

Assumption 1 (Shift and relevance of data properties)**.**

1. *Representation of X using g, h, a, b: $p_s(X|g, h, a, b) = p_t(X|g, h, a, b) = 1$, and $X \perp\!\!\!\perp Y|g, h, a, b$ for $\mathcal{P}_s$, $\mathcal{P}_t$.*
2. *Shift along g, h, a only: $p_s(X|g, h, a) = p_t(X|g, h, a)$.*
3. *Irrelevance of $a(X)$ to other features and true/predicted labels: $a \perp\!\!\!\perp b|g, h$, $a \perp\!\!\!\perp Y|g, h, b$ for both $\mathcal{P}_s$ and $\mathcal{P}_t$, and changing the value of a does not impact $\ell_\theta(X, Y)$.*

$g(X)$ encapsulates the user's beliefs of what axes the shift between $\mathcal{P}_s$ and $\mathcal{P}_t$ occurs on. Since $g(X)$ may be difficult to model, the user approximates them by designing k *slicing functions* $\widetilde{g}(X) = \{\widetilde{g}_1(X), \dots, \widetilde{g}_k(X)\}$, where each $\widetilde{g}_i : \mathcal{X} \to \{-1, 1\}$ noisily captures an axis $g_i(X)$ via a binary decision. When $h(X)$ is empty, we say that $\widetilde{g}(X)$ is *fully-specified* and otherwise *under-specified*. When $\widetilde{g}(X) = g(X)$, we say that the slices are *perfect* and otherwise *noisy*.

3.2 Importance Weighting based on Slicing Functions

Based on these assumptions, weighting using the relevant shifting properties g and h is sufficient. Theoretically,

Proposition 1. *By Assumption 1,*

$$\mathcal{L}_t = \mathbb{E}_s\left[\frac{p_t(g(X), h(X))}{p_s(g(X), h(X))}\ell_\theta(X, Y)\right].$$

Revisiting Example 2.3 with $g(x) = x_1$, $a(x) = x_2$, and no $h(x)$, $b(x)$, Proposition 1 confirms our intuition that weighting on x_1 is sufficient and reduces support shift.

If $\widetilde{g}$ is perfect and well-specified, using $\frac{p_t(\widetilde{g}(x))}{p_s(\widetilde{g}(x))}$ for weighting corrects the distribution shift without being susceptible to extreme shifts in $a(x)$ and the dimensionality and noise added by $b(x)$. In the more frequent case when $h(x)$ is nonempty, reweighting based on $g(x)$ to estimate $\mathcal{L}_t$ already yields a biased approximation $\mathcal{L}_g = \mathbb{E}_s\left[\frac{p_t(g(X))}{p_s(g(X))}\ell_\theta(X, Y)\right]$. However, as long as the slices are not noisy, the density ratio methods discussed in Section 2.1 can be applied on the slices with well-studied tradeoffs in computational efficiency, estimation error, and robustness to misspecification (Kanamori et al., 2010). When the slices are noisy, the challenge is to learn weights $w(x) = \frac{p_t(g(x))}{p_s(g(x))}$ on g when we only have $\widetilde{g}$ and our datasets, motivating our algorithm for this particular case.

3.3 Density ratio estimation approach

We estimate the density ratio as $\hat{w}(x)$ using $\widetilde{g}$ and our data. We partition $\mathcal{D}_s$ into $\mathcal{D}_{s_1}$ and $\mathcal{D}_{s_2}$ of sizes n_{s_1}, n_{s_2} such that the former is used to learn $\hat{w}(x)$ and the latter is used for evaluation. Our estimate of $\mathcal{L}_t$ is $\hat{\mathcal{L}}_g := \frac{1}{n_{s_2}}\sum_{i=1}^{n_{s_2}} \hat{w}(x_i^{s_2})\ell_\theta(x_i^{s_2}, y_i^{s_2})$. We present a noise-aware extension of LL-KLIEP (Tsuboi et al., 2009) that models g as a latent variable and $p(\widetilde{g}, g)$ as a graphical model. While previous work on latent variable density ratio estimation focuses on the posterior distribution and requires observations of g (Liu et al., 2020), our algorithm allows for denoising by incorporating a user's prior knowledge on the quality of $\widetilde{g}$, which can be viewed as hyperparameters of user confidence.

Graphical Model. Let a graph $G = (\widetilde{g}, E)$ specify dependencies between the slicing functions using standard notions from PGM literature (Lauritzen, 1996; Koller & Friedman, 2009). We assume the user knows dependencies between $\widetilde{g}$, although E can be learned (Ravikumar et al., 2010; Loh & Wainwright, 2013), and assume each $\widetilde{g}_i$ is connected to at most one other $\widetilde{g}_j$. For the joint distribution of $(g, \widetilde{g})$, we augment G by adding an edge from each $\widetilde{g}_i$ to g_i in the following model for both p_s and p_t:

$$p(\widetilde{g}, g; \theta) = \frac{1}{Z_\theta}\exp\left[\sum_{i=1}^k \theta_i g_i + \sum_{i=1}^k \theta_{ii} g_i \widetilde{g}_i + \sum_{(i,j)\in E} \theta_{ij}\widetilde{g}_i\widetilde{g}_j\right], \tag{1}$$

where Z_θ is the log partition function. Note that when $\widetilde{g} = g$, this reduces to an Ising model of g with edgeset E. In Appendix B.1 we show that the marginal distribution of g is then

$$p(g; \psi) = \frac{1}{Z}\exp\left[\sum_{i=1}^k \psi_i g_i + \sum_{(i,j)\in E} \psi_{ij} g_i g_j\right] \tag{2}$$
$$= \frac{1}{Z}\exp(\psi^\top \phi(g)),$$

where Z is a different log partition function, $\phi(g)$ is a representation of the potentials over g, and each element of ψ is a function of θ in (1). Define $\delta = \psi_t - \psi_s$ as the difference in parameters of $p_t(g)$ and $p_s(g)$. Under this model, the density ratio to estimate is $w(x) = \frac{p_t(g(x);\psi_t)}{p_s(g(x);\psi_s)} = \exp\left(\delta^\top \phi(g(x))\right)\frac{Z_s}{Z_t}$.

Latent Variable KLIEP. Based on the parametric form of $w(x)$, KLIEP aims to minimize the KL-divergence between the target distribution $\mathcal{P}_t$ and an estimated $\hat{\mathcal{P}}_t$ with density $\hat{p}_t(g;\delta) = \exp\left(\delta^\top \phi(g)\right) \frac{Z_s}{Z_t} p_s(g)$. Note that since $\hat{p}_t(g;\delta)$ must be a valid density, the log partition ratio $\frac{Z_s}{Z_t}$ is equal to $\frac{1}{\mathbb{E}_s[\exp(\delta^\top \phi(g))]}$. Then, minimizing the KL-divergence between $\mathcal{P}_t$ and $\hat{\mathcal{P}}_t$ is equivalent to solving

$$\text{maximize}_\delta \; \mathbb{E}_t\left[\delta^\top \phi(g)\right] - \log \mathbb{E}_s\left[\exp(\delta^\top \phi(g))\right]. \quad (3)$$

The true distribution of g is unknown, but we can write (3) as $\mathbb{E}_t\left[\mathbb{E}_t\left[\delta^\top \phi(g)|\widetilde{g}\right]\right] - \log \mathbb{E}_s\left[\mathbb{E}_s\left[\exp(\delta^\top \phi(g))|\widetilde{g}\right]\right]$. The outer expectation over $\widetilde{g}$ can be approximated empirically, so in place of (3) we want to maximize $\frac{1}{n_t}\sum_{i=1}^{n_t} \mathbb{E}_t\left[\delta^\top \phi(g)|\widetilde{g}(x_i^t)\right] - \log\left(\sum_{j=1}^{n_{s_1}} \mathbb{E}_s\left[\exp(\delta^\top \phi(g))|\widetilde{g}(x_j^{s_1})\right]\right)$.

Noise Correction. This empirical objective function requires knowledge of $p(g|\widetilde{g})$, which factorizes into $\prod_{i=1}^k p(g_i|\widetilde{g}_i)$. This inspires our noise-aware KLIEP approach: users provide simple 2×2 *correction matrices* σ_s^i, σ_t^i per slice to incorporate their knowledge of slice quality, where $\sigma_s^i(\alpha,\beta) \approx p_s(g_i = \alpha|\widetilde{g}_i = \beta)$ and similarly for σ_t^i. This knowledge can be derived from prior "evaluation" of $\widetilde{g}_i$'s and can also be viewed as a measure of user confidence in their slices. Note that setting each σ^i equal to the identity matrix recovers the noiseless case $\widetilde{g} = g$, which is LL-KLIEP. Our convex optimization problem maximizes

$$\hat{f}_{\text{KLIEP}}(\delta,\sigma) = \frac{1}{n_t}\sum_{i=1}^{n_t} \mathbb{E}_t^\sigma\left[\delta^\top \phi(g)|\widetilde{g}(x_i^t)\right] - \log\left(\sum_{j=1}^{n_{s_1}} \mathbb{E}_s^\sigma\left[\exp(\delta^\top \phi(g))|\widetilde{g}(x_j^{s_1})\right]\right), \quad (4)$$

where $\mathbb{E}_s^\sigma\left[r(g)|\widetilde{g}\right] = \int r(g)\prod_{i=1}^k \sigma_s^i(g_i,\widetilde{g}_i)dg$ for any function $r(g)$. Define $\hat{\delta} = \text{argmax}_\delta\, \hat{f}_{\text{KLIEP}}(\delta,\sigma)$. Then the estimated density ratio $\hat{w}(g)$ is $\frac{n_{s_1}\exp\left(\hat{\delta}^\top \phi(g)\right)}{\sum_{i=1}^{n_{s_1}} \mathbb{E}_s^\sigma\left[\exp(\hat{\delta}^\top \phi(g))|\widetilde{g}(x_i^{s_1})\right]}$, for which we've used an empirical estimate of $\frac{Z_s}{Z_t}$ as well. To produce weights on $\mathcal{D}_s$, we define $\hat{w}(x) = \mathbb{E}_s^\sigma\left[\hat{w}(g)|\widetilde{g}(x)\right]$. Our approach is summarized in Algorithm 1.

Note that our approach can handle *incomplete* slices that do not have full coverage on the dataset. We model $\widetilde{g}_i$ with support $\{-1,0,1\}$ where 0 represents abstention; this can be incorporated into (1) as described in Appendix B.1.

4 Theoretical Analysis

We analyze the performance of our approach by comparing our estimate $\hat{\mathcal{L}}_g$ to the true $\mathcal{L}_t$. We show the error can be decomposed into a bias dependent on the user input and a variance dependent on the true distribution of g, noise correction, and amount of data. We provide an error bound that always holds with high probability, in contrast to standard IW for which no generalization bounds hold for certain distributions.

Algorithm 1 MANDOLINE

```
Input: Datasets $\mathcal{D}_s, \mathcal{D}_t$; slicing functions $\widetilde{g} : \mathcal{X} \to \{-1,1\}^k$, dependency graph $G = (\widetilde{g}, E)$, correction matrices $\sigma_s^i$ and $\sigma_t^i$ for each slice.
Split $\mathcal{D}_s$ into $\mathcal{D}_{s_1}, \mathcal{D}_{s_2}$.
Use $G$'s edgeset to construct representation function $\phi$.
Solve $\hat{\delta} = \text{argmax}_\delta\, \hat{f}_{\text{KLIEP}}(\delta,\sigma)$ defined in (4).
Construct ratio $\hat{w}(g) = \frac{n_{s_1}\exp(\hat{\delta}^\top \phi(g))}{\sum_{j=1}^{n_{s_1}} \mathbb{E}_s^\sigma\left[\exp(\hat{\delta}^\top \phi(g)|\widetilde{g}(x_j^{s_1}))\right]}$.
Return $\hat{\mathcal{L}}_g = \frac{1}{n_{s_2}}\sum_{i=1}^{n_{s_2}} \mathbb{E}_s^\sigma\left[\hat{w}(g)|\widetilde{g}(x_i^{s_2})\right]\ell_\theta(x_i^{s_2}, y_i^{s_2})$.
```

Define a "fake" g estimated from inaccurate σ as $\bar{g}$, where $p(\bar{g}) = \int \sigma(g,\widetilde{g})p(\widetilde{g})d\widetilde{g}$, and $\mathcal{L}_{\bar{g}} = \mathbb{E}_s\left[w(\bar{g}(X))\ell_\theta(X,Y)\right]$, where $w(\bar{g}(X)) = \frac{p_t(\bar{g}(X))}{p_s(\bar{g}(X))}$. Then,

$$|\mathcal{L}_t - \hat{\mathcal{L}}_g| \le \underbrace{|\mathcal{L}_t - \mathcal{L}_g|}_{\text{bias from no } h(X)} + \underbrace{|\mathcal{L}_g - \mathcal{L}_{\bar{g}}|}_{\text{bias from incorrect } \sigma} + \underbrace{|\mathcal{L}_{\bar{g}} - \mathbb{E}_s[\hat{\mathcal{L}}_g]|}_{\text{var. from estimated ratio}} + \underbrace{|\mathbb{E}_s[\hat{\mathcal{L}}_g] - \hat{\mathcal{L}}_g|}_{\text{var. from empirical evaluation}}. \quad (5)$$

Suppose all slices are noisy and the user fails to correct $k' \le k$ of them. Define $\eta_s^{\min}(i), \eta_s^{\max}(i)$ as bounds on the relative error of σ_s^i such that $\left|\frac{p_s(g_i|\widetilde{g}_i) - \sigma_s^i(g_i,\widetilde{g}_i)}{p_s(g_i|\widetilde{g}_i)}\right| \in [\eta_s^{\min}(i), \eta_s^{\max}(i)]$ for all $g_i, \widetilde{g}_i$ per slice. $\eta_t^{\min}(i)$ and $\eta_t^{\max}(i)$ are similarly defined per slice for $\mathcal{P}_t$, and define the total correction ratio as $r = \prod_{i=1}^{k'} \frac{1+\eta_t^{\max}(i)}{1-\eta_s^{\min}(i)}$. Define an upper bound $M = \sup_X \frac{p_t(g(X))}{p_s(g(X))}$ on the density ratio of g.

Theorem 1. *Set $n_{s_1} = n_{s_2} = \frac{n_s}{2}$. Under Assumption 1, with probability at least $1 - \varepsilon$ the accuracy of our estimate of $\mathcal{L}_t$ via noise-aware reweighting is bounded by*

$$|\mathcal{L}_t - \hat{\mathcal{L}}_g| \le 2\|p_t(h|g) - p_s(h|g)\|_{\text{TV}} + rM\sum_{i=1}^{k'}\left(\frac{\eta_t^{\max}(i)}{1-\eta_t^{\min}(i)} + \frac{\eta_s^{\max}(i)}{1-\eta_s^{\min}(i)}\right) + \hat{M}(c_{s,\ell_\theta}+1)\sqrt{\frac{\log(4/\varepsilon)}{n_s}}, \quad (6)$$

where c_{s,ℓ_θ} is a constant dependent on the distribution of ℓ_θ and $\mathcal{P}_s$, and $\hat{M} \xrightarrow{p} rM$ as $n_s \wedge n_t \to \infty$.

We make two observations about Theorem 1.

- The first two terms are the bias from user input and map to the first two terms in (5). The first total variation distance is a bias from not modeling $h(X)$ and describes relevant uncaptured distribution shift. The second term describes the impact of inaccurate correction matrices σ_s, σ_t.

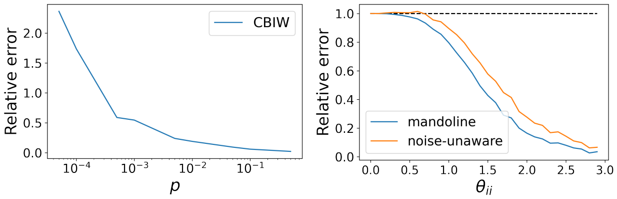

Figure 2. (left) relative error of CBIW in the presence of a spurious shifting feature with probability p (MANDOLINE has a 0.01 error). *(right)* relative error of MANDOLINE vs noise-unaware reweighting as correlation between $g, \widetilde{g}$ increases.

- The last term maps to the third and fourth terms of (5) and describes the variance from learning and evaluating on the data, which collectively with $\hat{M}$ scales in n_t and n_s. This bound depends on the distributions themselves—critically the upper bound on weights, M—and σ_s, σ_t accuracy.

When the user writes perfect, fully-specified slicing functions, the standing bias of the first two terms in Theorem 1 is 0, and our estimate $\hat{\mathcal{L}}_g$ converges to $\mathcal{L}_t$.

Comparison to standard IW. We compare our approach to standard IW in the feature space using our decomposition framework in (5). Standard IW does not utilize slices or user knowledge, so the first two terms of (5) would be 0. The latter two terms depend on the magnitude of the true weights and more generally their variance. In standard IW, the variance of the weights can be high and even unbounded due to support shift or high-dimensional complex data. Even simple continuous distributions can yield bad weights; for instance, Cortes et al. (2010) show a Gaussian distribution shift where the variance of the weights is infinite, leading to inapplicable generalization bounds and poor performance. In contrast, while our approach using $\widetilde{g}$ may incur bias, it will always have weights bounded by at most $\hat{M}$ due to the discrete model and hence have applicable bounds. The variance may also be lower due to how we mitigate problems of support shift and high-dimensional complex data by ignoring $a(X), b(X)$. Therefore, when the bias due to under-specified, noisy $\widetilde{g}$ is less than the additional weight variance of standard IW, our method can significantly reduce the estimation error of $\mathcal{L}_t$ while always providing valid theoretical guarantees.

4.1 Synthetic Experiments

In this section, we provide an example where standard importance weighting struggles while MANDOLINE maintains accurate estimates. We then evaluate how MANDOLINE performs as the noise in the $\widetilde{g}_i$'s varies. Additional details and synthetics are included in Appendix C.1.

Support Shift. Standard IW struggles when the source and target distributions have little overlap. As described in Example 2.3, the density ratio is 0 or undefined when the supports are disjoint. Moreover, if the source has a low-density region where the target density is high, this effectively reduces the sample size by assigning high weight to a few examples. To illustrate, in Figure 2a we generate data with a single g_i and set $\tilde{g}_i = g_i$. We then append a "spurious" feature ($a(X)$) that is 1 with small probability p for source examples, and 1 for *all* target examples. We plot the relative error of CBIW on all features as p decreases to 0; for small values of p this is greater than the difference between source and target. By contrast, MANDOLINE ignores the spurious feature entirely as it is not part of g, achieving a relative error of only 0.01.

Noisy Slicing Functions. In Figure 2b, we show that on data generated from (1), MANDOLINE's noise-aware approach does better than the noise-unaware approach (i.e. setting each σ_t^i, σ_s^i to the identity matrix) as we vary the correlation between g_i and $\widetilde{g}_i$ by adjusting θ_{ii} in (1).

5 Experiments on Real Data

We empirically validate that MANDOLINE can estimate performance under distribution shift on real tasks and datasets.

Experimental Claims. We validate claims about MANDOLINE under the experimental settings described below:

1. **Fully-specified, with perfect slices (Section 5.1).** In the setting when all the factors of distribution shift are known (i.e., $h(X)$ is empty), and the true slice labels are known via metadata annotations (i.e., $\widetilde{g} = g$), MANDOLINE reduces the model performance estimation error by up to 3× over baselines.

2. **Fully-specified, with noisy slices (Section 5.2).** When the slices are noisy / programmatic, but still capture all the shifted variables (i.e., $h(X)$ is still empty but $\widetilde{g} \neq g$), MANDOLINE's performance does not significantly degrade, and it remains competitive with baselines.

3. **Under-specified, with noisy slices (Section 5.3).** For large distribution shifts where only a subset of relevant shifted variables are captured by the noisy programmatic slices (i.e. $h(X)$ is nonempty and $\widetilde{g} \neq g$), MANDOLINE reduces estimation error by up to 2× over baselines.

Tasks. We consider four tasks from computer vision and natural language processing, summarized in Table 1.

Baselines. We compare MANDOLINE against widely used importance weighting baselines on the features. **Direct Source Estimation** (SOURCE) directly uses the estimate from the source distribution for the target. **Classifier-Based Importance Weighting** (CBIW) uses logistic regression on a task-specific feature representation to distinguish samples from the source and target distributions. As noted

Table 1. Summary of real-world tasks and datasets considered.

Task	**Task Labels**	**Distribution Shift**	**Source Data**	**Target Data**	**Slices**
CelebA *image classification*	*male* *vs. female*	↑ blurry images	validation set *(1% blurry)*	perturbed test set *(30% blurry)*	metadata labels *blurry / not blurry*
CivilComments *toxic text classification*	*toxic* *vs. non-toxic*	↑↓ identity proportions *(e.g. ↑ female)*	validation set	perturbed test set	metadata labels *8 identity groups*
SNLI→MNLI *natural language inference*	*entailment, neutral or contradiction*	single-genre → multi-genre examples	SNLI validation set	MNLI matched validation set	programmatic *task model predictions,* *task model entropy*
SNLI$^+$ →HANS$^-$ *natural language inference*	*entailment* *or non-entailment*	↑ lexical overlap with label shift	SNLI validation set + 1% HANS validation set	99% HANS validation set	programmatic *lexical overlap* *noisy non-entailment* *sentence structure* *task model entropy*

by Tsuboi et al. (2009), this baseline returns weights identical to LL-KLIEP on features. **Kernel Mean Matching** (KMM) (Gretton et al., 2009) solves a quadratic program to minimize the discrepancy in feature expectations between the target and a reweighting of the source distribution. **Unconstrained Least-Squares Importance Fitting** (uLSIF) (Kanamori et al., 2009) is a stable version of LSIF that solves a least-squares optimization problem for the density ratio using a linear kernel model. We also run the above baselines on the *slices* to compare against Mandoline's instantiation with KLIEP in the Appendix.

We self-normalize importance weights for all methods to sum to 1 (Owen, 2013). Full experimental details for all experiments are provided in Appendix C.2.

5.1 Fully-Specified with Perfect Slices

First, we consider distribution shift along variables for which annotated metadata is available, i.e. the identities of all the relevant slices are known, and the ground-truth slice labels are known for each example. We consider two tasks in this section: CelebA (images) and CivilComments (text). On CelebA, Mandoline reduces estimation error by 3× over the best baseline (CBIW). On CivilComments, Mandoline and CBIW both exhibit strong performance, returning estimates within 0.12% and 0.03% of the true test accuracy. We describe these experiments below, with results summarized in Table 2.

5.1.1 CelebA

Task. In CelebA, we classify faces as either male or female. Each image comes with metadata annotations. We induce distribution shift by perturbing the CelebA validation and test sets so that the test set has more images annotated as "blurry" (30% of test set vs. only 1% of validation set).

Models. We evaluate ResNet18 and ResNet50 models pretrained on ImageNet and finetuned for 5 epochs on CelebA.

Results. For the ResNet50 models, accuracy is significantly higher on the non-blurry images (97.5%) than for the blurry ones (91%); the target accuracy is thus 95.6%, vs. 97.4% on the source set. We use Mandoline to estimate the test performance, using the provided "blurry" metadata label as our only slice function g_i. The resulting estimate is within 0.16% on average of the true value. In comparison, the mean absolute error of the CBIW estimate is 0.53%, and 1.76% for KMM and uLSIF. Table 2 summarizes these results, along with those for ResNet18, which exhibit similar trends.

5.1.2 CivilComments

Task. The CivilComments dataset (Borkan et al., 2019) contains comments labeled "toxic" or "non-toxic", along with 8 metadata labels on whether a particular identity (male, female, etc.) is referenced in the text. We modify the test set to introduce shift by randomly subsampling examples for each "slice" (subset of data with a given assignment of metadata labels), with different proportions per slice.

Models. We use a standard `bert-base-uncased` model, fine-tuned on CivilComments for 5 epochs.

Results. When the true g_i's are used, Mandoline returns accuracy estimates that are within 0.12% of the true test accuracy. CBIW returns an even better estimate (within 0.03%), while the estimation error of KMM is 1.25% and of uLSIF is 0.39%. By contrast, the raw (unweighted) validation accuracy differs from the test accuracy by 1.6%; thus, both Mandoline and the baselines improve this estimate.

5.2 Fully-Specified with Noisy Slices

Next, we examine the effect of using noisy, user-provided slicing functions in place of perfect metadata labels. Here, we show that this additional noise does not increase estimation error for Mandoline.

We consider the CivilComments task described in the previous section. Instead of using the annotated metadata as our g_i's, we write heuristic slicing functions as a substitute, as one would do in practice if this metadata was unavailable. For each of the identities described in Section 5.1.2, we write a noisy slice $\widetilde{g}_i$ that detects the presence of this identity. For example, we detect the male identity if the words "male", "man", or "men" appear in the text. These simple slicing functions are reasonably accurate compared to the true metadata annotations (average 0.9 F1 score compared to metadata across all 8 slices). With these noisy slices, the accuracy estimate returned by MANDOLINE is within **0.10%** of the true value, *lower* than the error when using metadata (0.16%). This suggests that the noise in our slices is not a major issue, and that our slices in fact better capture the shifts relevant to evaluating the model on the target data.

5.3 Under-Specified with Noisy Slices

Next, we turn to an under-specified setting, where the distribution shift is imperfectly captured by programmatic slices. Here, MANDOLINE is able to reduce average estimation error by up to 2× over the best baseline (Table 3).

Concretely, for natural language inference, we study whether it is possible to estimate performance for the MNLI (Williams et al., 2018) validation set (target) using the SNLI (Bowman et al., 2015) validation set (source). The distribution shift from SNLI→MNLI is substantial, as MNLI was designed to capture far more input variability.

Programmatic Slices. Since the shift from SNLI→MNLI is large, we design slices that can capture general distributional shifts. We construct 3 slicing functions that check (respectively) whether the evaluated classifier predicted one of the 3 classes, and 6 slicing functions that bucket examples based on the entropy of the classifier's outputs. These slices capture how the model's uncertainty or predictions change, which are useful indicators for detecting distributional shift (Hendrycks & Gimpel, 2017). Here, they allow us to perform a *model-specific* estimate adjustment.

Results. We compare 8 off-the-shelf models taken from the HuggingFace Model Hub. Table 3 contains detailed results for estimating both standard accuracy over the 3 NLI classes, as well as an oft-used binary accuracy metric that combines the contradiction and neutral classes. For both metrics, MANDOLINE provides substantially better estimates than baselines while requiring no expensive additional training or fine-tuning. For binary accuracy, MANDOLINE is able to estimate performance on the MNLI dataset with an average error of only 1.6%, when all baselines are *worse* than just using the unadjusted source estimates. MANDOLINE's estimates are also robust across the evaluated models, with a maximum error that is 3.2× lower than the best baseline.

Table 2. Mean absolute estimation error for target accuracy on CELEBA and CIVILCOMMENTS.

METHOD	AVERAGE ESTIMATION ERROR (%)		
	CELEBA		CIVILCOMMENTS
	RESNET18	RESNET50	BERT
SOURCE	1.96%	1.74%	1.62%
CBIW	0.47%	0.53%	**0.03**%
KMM	1.97%	1.76%	1.25%
uLSIF	1.97%	1.76%	0.39%
MANDOLINE	**0.16**%	**0.16**%	0.12%

Table 3. Estimating standard and binary accuracy on MNLI using SNLI. Average and maximum estimation errors for 8 models are reported, with 95% confidence intervals.

METHOD	STANDARD ACCURACY		BINARY ACCURACY	
	AVG. ERROR	MAX. ERROR	AVG. ERROR	MAX. ERROR
SOURCE	6.2% ± 3.8%	15.6%	3.0% ± 2.3%	9.3%
CBIW	5.5% ± 4.5%	17.9%	3.7% ± 2.5%	9.8%
KMM	5.7% ± 3.6%	14.6%	3.3% ± 2.3%	8.7%
uLSIF	6.4% ± 3.9%	16.0%	3.7% ± 2.4%	9.1%
MANDOLINE	**3.6% ± 1.6%**	**5.9%**	**1.6% ± 0.7%**	**2.7%**

Application: Model Selection. We check that MANDOLINE can be used to rank models in terms of their target performance. On 7/8 models, MANDOLINE correctly assesses if the model improves or degrades (see Appendix C.2; Table 9). MANDOLINE has a Kendall-Tau score of 0.786 (p-value 0.006) to the true MNLI performance—only confusing rankings for the top-3 models, which are closely clustered in performance on MNLI (1% performance spread).

6 Slice Design

A natural question raised by our work is: how should we best design slicing functions? Key desiderata for slices are that they should be task-relevant and capture important axes of distribution shift. For many tasks, candidate "slicing functions" in the form of side information are readily available. Here, slices may come directly in the form of metadata (such as for CELEBA), or as user-defined heuristics (such as the keyword matching we use for CIVILCOMMENTS) and are already used for evaluation and monitoring purposes.

There are also software tools to create slices (Goel et al., 2021), and a growing body of work around engineering, discovering and utilizing them (Chen et al., 2019; McCoy et al., 2019b; Ribeiro et al., 2020; Wang et al., 2018), including

Table 4. Reduction in MANDOLINE estimation error for SNLI$^+$ →HANS$^-$, as more slices are added to capture distribution shift.

Slices	**Average Estimation Error**
Lexical Overlap	16.9%
+ *Noisy Non-Entailment*	11.8%
+ *Sentence Structure*	6.9%
+ *Model Entropy*	5.9%
+ CBIW *Slice*	0.6%

automated methods (Polyzotis et al., 2019; Sagadeeva & Boehm, 2021). Our central contribution is to describe how to model and utilize this noisy side information in order to address long-standing challenges in importance weighting.

We investigate how slice design affects the performance of Mandoline. Section 6.1 discusses a challenging under-specified setting, where along with domain knowledge, Mandoline can incorporate standard IW as a slice to achieve a "best-of-both-worlds" performance. Section 6.2 shows that a simple, automated slice design strategy works well out-of-the-box for sentiment classification. These case studies emphasize that understanding slice design is an important direction for future work.

6.1 Tackling Under-Specification with Slice Design

We use a challenging setting to highlight how careful slice design can tackle under-specification. Concretely, we consider distribution shifts when moving from SNLI (Bowman et al., 2015) to HANS (McCoy et al., 2019a). HANS was created to address the lack of diverse *lexical overlap* examples in SNLI, i.e. examples where the hypothesis is created using a subset of words from the premise. Among SNLI examples with lexical overlap, only 1.7% are labeled non-entailment. By contrast, HANS only contains lexical overlap examples, with a 50/50 class balance.

Due to the lack of non-entailment lexical overlap examples, estimating performance on HANS using SNLI is extremely challenging. Therefore, we construct a mixture of SNLI and HANS, moving 1% of HANS to SNLI to create a new source dataset (SNLI$^+$), and keeping the remaining 99% of HANS as the target data (HANS$^-$).

In Table 4 we examine how changing the set of slices affects Mandoline's performance. Just using a single lexical overlap slice yields high estimation error (16.9%), since it does not adjust for the shift in the proportion of non-entailment lexical overlap (1.7% → 50%). To capture this shift, we add noisy slices for contradiction (based on negation and token ordering), sentence structure (word substitutions, length differences, verb tense) and the evaluated model's uncertainty (similar to SNLI→MNLI). These additions further reduce estimation error by 2.86×.

Interestingly, CBIW—when fine-tuned with a `bert-base-uncased` model—learns a classifier that perfectly separates the HANS examples added to SNLI$^+$, giving low estimation error (1.2%). The flexibility of Mandoline allows us to take advantage of this by directly incorporating the CBIW predictions as a slicing function, giving us a "best-of-both-worlds" that achieves extremely low estimation error (0.6%). This highlights a natural strength of Mandoline in being able to easily incorporate information from other methods.

Table 5. Average and maximum estimation errors for target accuracy across 3 models on IMDB, with 95% confidence intervals.

Source → Target Shift	Avg. Error	Max. Error
IMDB → Counterf. IMDB (Kaushik et al., 2020)	3.1% ± 1.4%	4.6%
IMDB → Sentiment 140 (Go et al., 2009)	4.7% ± 0.8%	5.6%
IMDB → Yelp Polarity (Zhang et al., 2015)	3.8% ± 1.2%	4.9%
IMDB → Amazon Polarity (Zhang et al., 2015)	0.2% ± 0.1%	0.3%

6.2 Slice Design "In-the-Wild"

Using sentiment classification on IMDB (Maas et al., 2011), we show that automated slice design can be effective out-of-the-box, *without tuning* Mandoline *at all*. We use only the task-agnostic entropy-based slices described in Section 5.3. Across 3 models, Table 5 shows that we get good estimates when moving from IMDB to varied sentiment datasets. This includes a large shift to Twitter analysis with Sentiment-140, where Mandoline closely estimates a significant absolute performance drop of upto 23% accuracy. Overall our results here and in Section 5.3 show early promise that simple, task-agnostic slices that rely on model entropy can be quite effective.

7 Conclusion

We introduced Mandoline, a framework for evaluating models under distribution shift that utilizes user-specified slicing functions to reweight estimates. When these slicing functions adequately capture the distribution shift, Mandoline can outperform standard IW by addressing issues of support shift and complex, high-dimensional features. We hope that our framework inspires future work on designing and understanding slices and sets the stage for a new paradigm of model evaluation.

Acknowledgements

We gratefully acknowledge the support of NIH under No. U54EB020405 (Mobilize), NSF under Nos. CCF1763315 (Beyond Sparsity), CCF1563078 (Volume to Velocity), and 1937301 (RTML); ONR under No. N000141712266 (Unifying Weak Supervision); the Moore Foundation, NXP, Xilinx, LETI-CEA, Intel, IBM, Microsoft, NEC, Toshiba, TSMC, ARM, Hitachi, BASF, Accenture, Ericsson, Qualcomm, Analog Devices, the Okawa Foundation, American Family Insurance, Google Cloud, Swiss Re, Total, the HAI-AWS Cloud Credits for Research program, the Stanford Data Science Initiative (SDSI),

References

Austin, P. An introduction to propensity score methods for reducing the effects of confounding in observational studies. *Multivariate Behav Res.*, 2011.

Ben-Tal, A., den Hertog, D., Waegenaere, A. D., Melenberg, B., and Rennen, G. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013. ISSN 00251909, 15265501.

Borkan, D., Dixon, L., Sorensen, J., Thain, N., and Vasserman, L. Nuanced metrics for measuring unintended bias with real data for text classification. *WWW*, 2019.

Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 632–642, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1075.

Byrd, J. and Lipton, Z. What is the effect of importance weighting in deep learning? In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 872–881. PMLR, 09–15 Jun 2019.

Chen, V., Wu, S., Ratner, A. J., Weng, J., and Ré, C. Slice-based learning: A programming model for residual learning in critical data slices. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Cortes, C., Mansour, Y., and Mohri, M. Learning bounds for importance weighting. In Lafferty, J., Williams, C., Shawe-Taylor, J., Zemel, R., and Culotta, A. (eds.), *Advances in Neural Information Processing Systems*, volume 23, pp. 442–450. Curran Associates, Inc., 2010.

Cortes, C., Mohri, M., and Medina, A. M. Adaptation based on generalized discrepancy. *Journal of Machine Learning Research*, 20(1):1–30, 2019.

D'Agostino Jr, R. B. Propensity score methods for bias reduction in the comparison of a treatment to a non-randomized control group. *Statistics in medicine*, 17 (19):2265–2281, 1998.

Duchi, J., Glynn, P., and Namkoong, H. Statistics of robust optimization: A generalized empirical likelihood approach, 2018.

Fang, T., Lu, N., Niu, G., and Sugiyama, M. Rethinking importance weighting for deep learning under distribution shift. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 11996–12007. Curran Associates, Inc., 2020.

Go, A., Bhayani, R., and Huang, L. Twitter sentiment classification using distant supervision, 2009.

Goel, K., Rajani, N., Vig, J., Tan, S., Wu, J., Zheng, S., Xiong, C., Bansal, M., and Ré, C. Robustness gym: Unifying the nlp evaluation landscape, 2021.

Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., and Schölkopf, B. Covariate shift by kernel mean matching. *Dataset shift in machine learning*, 3(4): 5, 2009.

Grover, A., Song, J., Kapoor, A., Tran, K., Agarwal, A., Horvitz, E. J., and Ermon, S. Bias correction of learned generative models using likelihood-free importance weighting. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Hastie, T., Tibshirani, R., and Friedman, J. *The Elements of Statistical Learning*. 2001.

Hendrycks, D. and Gimpel, K. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *ArXiv*, abs/1610.02136, 2017.

Hirano, K. and Imbens, G. Estimation of causal effects using propensity score weighting: An application to data on right heart catheterization. *Health Services and Outcomes Research Methodology*, 2(3-4):259–278, December 2001. ISSN 1387-3741. doi: 10.1023/A:1020371312283.

Horvitz, D. and Thompson, D. J. A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47:663–685, 1952.

Hu, W., Niu, G., Sato, I., and Sugiyama, M. Does distributionally robust supervised learning give robust classifiers? In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2029–2037. PMLR, 10–15 Jul 2018.

Isakov, M. and Kuriwaki, S. Towards principled unskewing: Viewing 2020 election polls through a corrective lens from 2016. *Harvard Data Science Review*, 2020.

Kanamori, T., Hido, S., and Sugiyama, M. A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, 10(48):1391–1445, 2009.

Kanamori, T., Suzuki, T., and Sugiyama, M. Theoretical analysis of density ratio estimation. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E93.A(4):787–798, 2010. doi: 10.1587/transfun.E93.A.787.

Kaushik, D., Hovy, E., and Lipton, Z. C. Learning the difference that makes a difference with counterfactually augmented data. *International Conference on Learning Representations (ICLR)*, 2020.

Kim, B., Liu, S., and Kolar, M. Two-sample inference for high-dimensional markov networks, 2019.

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R. L., Beery, S., Leskovec, J., Kundaje, A., Pierson, E., Levine, S., Finn, C., and Liang, P. WILDS: A benchmark of in-the-wild distribution shifts, 2020.

Koller, D. and Friedman, N. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

Kpotufe, S. Lipschitz Density-Ratios, Structured Data, and Data-driven Tuning. In Singh, A. and Zhu, J. (eds.), *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1320–1328, Fort Lauderdale, FL, USA, 20–22 Apr 2017. PMLR.

Lauritzen, S. *Graphical Models*. Clarendon Press, 1996.

Li, F., Morgan, K. L., and Zaslavsky, A. M. Balancing covariates via propensity score weighting. *Journal of the American Statistical Association*, 113(521):390–400, 2018. doi: 10.1080/01621459.2016.1260466.

Li, F., Lam, H., and Prusty, S. Robust importance weighting for covariate shift. In Chiappa, S. and Calandra, R. (eds.), *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 352–362. PMLR, 26–28 Aug 2020.

Lipton, Z., Wang, Y.-X., and Smola, A. Detecting and correcting for label shift with black box predictors. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3122–3130. PMLR, 10–15 Jul 2018.

Liu, S., Takeda, A., Suzuki, T., and Fukumizu, K. Trimmed density ratio estimation. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Liu, S., Zhang, Y., Yi, M., and Kolar, M. Posterior ratio estimation of latent variables, 2020.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Loh, P.-L. and Wainwright, M. J. Structure estimation for discrete graphical models: Generalized covariance matrices and their inverses. *The Annals of Statistics*, 41(6):3022–3049, Dec 2013. ISSN 0090-5364. doi: 10.1214/13-aos1162.

Long, M., Zhu, H., Wang, J., and Jordan, M. I. Unsupervised domain adaptation with residual transfer networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pp. 136–144, Red Hook, NY, USA, 2016. Curran Associates Inc. ISBN 9781510838819.

Maas, A., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pp. 142–150, 2011.

Makiyama, K. densratio. `https://github.com/hoxo-m/densratio`, 2019.

McCoy, R. T., Pavlick, E., and Linzen, T. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *ArXiv*, abs/1902.01007, 2019a.

McCoy, T., Pavlick, E., and Linzen, T. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3428–3448, Florence, Italy, July 2019b. Association for Computational Linguistics. doi: 10.18653/v1/P19-1334.

Mohamed, S. and Lakshminarayanan, B. Learning in implicit generative models. *ArXiv*, abs/1610.03483, 2016.

Nguyen, P., Ramanan, D., and Fowlkes, C. Active testing: An efficient and robust framework for estimating accuracy. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3759–3768, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.

Oakden-Rayner, L., Dunnmon, J., Carneiro, G., and Ré, C. Hidden stratification causes clinically meaningful failures in machine learning for medical imaging. In *Proceedings of the ACM Conference on Health, Inference, and Learning (CHIL)*, 2020.

Owen, A. B. *Monte Carlo theory, methods, and examples*. 2013.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019.

Polo, F. M. and Vicente, R. Covariate shift adaptation in high-dimensional and divergent distributions, 2020.

Polyzotis, N., Whang, S., Kraska, T. K., and Chung, Y. Slice finder: Automated data slicing for model validation. In *Proceedings of the IEEE Int' Conf. on Data Engineering (ICDE), 2019*, 2019.

Rahman, M. M., Kutlu, M., Elsayed, T., and Lease, M. Efficient test collection construction via active learning. In *Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval*, pp. 177–184, 2020.

Ravikumar, P., Wainwright, M. J., and Lafferty, J. D. High-dimensional ising model selection using ℓ_1 -regularized logistic regression. *Ann. Statist.*, 38(3):1287–1319, 06 2010. doi: 10.1214/09-AOS691.

Rhodes, B., Xu, K., and Gutmann, M. U. Telescoping density-ratio estimation. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 4905–4916. Curran Associates, Inc., 2020.

Ribeiro, M. T., Wu, T., Guestrin, C., and Singh, S. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4902–4912, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.442.

Rosenbaum, P. R. and Rubin, D. B. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.

Rosenbaum, P. R. and Rubin, D. B. Constructing a control group using multivariate matched sampling methods that incorporate the propensity score. *The American Statistician*, 39(1):33–38, 1985. ISSN 00031305.

Sagadeeva, S. and Boehm, M. Sliceline: Fast, linear-algebra-based slice finding for ml model debugging. *ACM SIGMOD*, 2021.

Sagawa, S., Koh, P. W., Hashimoto, T. B., and Liang, P. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations (ICLR)*, 2020.

Settles, B. Active learning. *Synthesis lectures on artificial intelligence and machine learning*, 6(1):1–114, 2012.

Stojanov, P., Gong, M., Carbonell, J., and Zhang, K. Low-dimensional density ratio estimation for covariate shift correction. In Chaudhuri, K. and Sugiyama, M. (eds.), *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 3449–3458. PMLR, 16–18 Apr 2019.

Subbaswamy, A., Adams, R., and Saria, S. Evaluating model robustness and stability to dataset shift. In Banerjee, A. and Fukumizu, K. (eds.), *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 2611–2619. PMLR, 13–15 Apr 2021.

Sugiyama, M., Suzuki, T., Nakajima, S., Kashima, H., von Bünau, P., and Kawanabe, M. Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, 60(4):699–746, 2008.

Sugiyama, M., Kawanabe, M., and Chui, P. L. Dimensionality reduction for density ratio estimation in high-dimensional spaces. *Neural Networks*, 23(1):44 – 59, 2010. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2009.07.007.

Sugiyama, M., Suzuki, T., and Kanamori, T. *Density Ratio Estimation in Machine Learning*. 2012a.

Sugiyama, M., Suzuki, T., and Kanamori, T. Density-ratio matching under the bregman divergence: a unified framework of density-ratio estimation. *Annals of the Institute of Statistical Mathematics*, 64:1009–1044, 2012b.

Taskazan, B., Navratil, J., Arnold, M., Murthi, A., Venkataraman, G., and Elder, B. Not your grandfathers test set: Reducing labeling effort for testing, 2020.

Tsuboi, Y., Kashima, H., Hido, S., Bickel, S., and Sugiyama, M. Direct density ratio estimation for large-scale covariate shift adaptation. *Journal of Information Processing*, 17:138–155, 2009. doi: 10.2197/ipsjjip.17.138.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5446.

Williams, A., Nangia, N., and Bowman, S. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association* *for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1112–1122, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1101.

Zhang, X., Zhao, J., and LeCun, Y. Character-level Convolutional Networks for Text Classification . *arXiv:1509.01626 [cs]*, September 2015.