Robust Learning with the Hilbert-Schmidt Independence Criterion

Daniel Greenfeld [1] **Uri Shalit** [1]

Abstract

We investigate the use of a non-parametric independence measure, the Hilbert-Schmidt Independence Criterion (HSIC), as a loss-function for learning robust regression and classification models. This loss-function encourages learning models where the distribution of the residuals between the label and the model prediction is statistically independent of the distribution of the instances themselves. This loss-function was first proposed by Mooij et al. (2009) in the context of learning causal graphs. We adapt it to the task of learning for unsupervised covariate shift: learning on a source domain without access to any instances or labels from the unknown target domain, but with the assumption that $p(y|x)$ (the conditional probability of labels given instances) remains the same in the target domain. We show that the proposed loss is expected to give rise to models that generalize well on a class of target domains characterised by the complexity of their description within a reproducing kernel Hilbert space. Experiments on unsupervised covariate shift tasks demonstrate that models learned with the proposed loss-function outperform models learned with standard loss functions, achieving state-of-the-art results on a challenging cell-microscopy unsupervised covariate shift task.

1. Introduction

In recent years there has been much interest in methods for learning *robust models*: models that are learned using certain data but perform well even on data drawn from a distribution which is different from the training distribution. This interest stems from demand for models which can perform under conditions of transfer learning and domain adaptation (Rosenfeld et al., 2018). This is especially relevant as training sets such as labeled image collections are often restricted to a certain setting, time or place, while the learned models are expected to generalize to cases which are beyond the specifics of how the training data was collected.

More specifically, we consider the following learning problem, called *unsupervised covariate shift*. Let (X, Y) be a pair of random variables such that $X \in \mathcal{X}$ and $Y \in \mathcal{Y} \subset \mathbb{R}$, with a joint distribution $P_{\text{source}}(X, Y)$, such that X are the instances and Y the labels. Our goal is, given a training set drawn from $P_{\text{source}}(X, Y)$, to learn a model predicting Y from X that works well on a different, a-priori unknown target distribution $P_{\text{target}}(X, Y)$. In a covariate shift scenario, the assumption is that $P_{\text{target}}(Y \mid X) = P_{\text{source}}(Y \mid X)$ but the marginal distribution $P_{\text{target}}(X)$ can change between source and target. We focus on *unsupervised* covariate shift, where we have no access to samples X or Y from the target domain.

In this paper we propose using a loss function inspired by work in the causal inference community. We consider a model in which the relation between the instance X and its label Y is of the form:

$$Y = f^{\star}(X) + \varepsilon, \quad \varepsilon \perp\!\!\!\perp X, \tag{1}$$

where the variable ε denotes noise which is independent of the distribution of the random variable X.

Given a well-specified model family and enough samples, one can learn $f^{\star}$, in which case there is no need to worry about covariate shift. However, in many realistic cases we cannot expect to have the true model in our model class, nor can we expect to have enough samples to learn the true model even if it is in our model class. While traditional methods rely on unlabeld data from the target domain to reason about $P_{\text{target}}(X)$, throughout this work we do not assume that we have *any* samples from a test distribution, nor that the model is well-specified.

The basic idea presented in this paper is as follows: by Eq. (1) we have $Y - f^{\star}(X) \perp\!\!\!\perp X$. Standard loss functions aim to learn a model $\hat{f}$ such that $\hat{f}(X) \approx Y$, or such that $\hat{p}(Y|X)$ is high. We follow a different approach: learning a model $\hat{f}$ such that $Y - \hat{f}(X)$ is approximately independent of the distribution of X. Specifically, we propose measuring independence using the Hilbert Schmidt Independence Criterion (HSIC): a non-parametric method that does not assume a

[1] Technion Institute of Technology, Haifa, Israe. Correspondence to: Daniel Greenfeld <danielgreenfeld3@gmail.com>, Uri Shalit <urishalit@technion.ac.il>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020. Copyright 2020 by the author(s).

specific noise distribution for ε (Gretton et al., 2005a; 2008). This approach was first proposed by Mooij et al. (2009) in the context of causal inference. As Mooij et al. (2009) point out, this approach can be contrasted with learning with loss functions such as the squared-loss or absolute-loss, which implicitly assume that ε has, respectively, a Gaussian or Laplacian distribution.

Intuitively, covariate shift is most harmful when the target distribution has more mass on areas of $\mathcal{X}$ on which the learned model performs badly. Thus, being robust against unsupervised covariate shift means having no certain subpopulation (of positive measure) on which the model performs particularly badly. This of course might come at a certain cost to the performance on the known source distribution.

The following toy example showcases that standard loss functions do not necessarily incentivize such behaviour. Suppose $\mathcal{X} = \{0, 1\}$, and $Y = X$. Let $P(X = 0) = \varepsilon$, and consider the following hypothesis class: $\mathcal{H} = \{h_1, h_2\}$, where $h_1(x) = 1$ for all x, and $h_2(x) = x - 0.01$. For small values of ε, an algorithm minimizing the mean squared error (or any standard loss) will output h_1. An algorithm relying on the independence criteria on the other hand will output h_2, since its residuals are independent of X. Which is preferable? That depends on the target (test) distribution. If it is the same as the training distribution, h_1 is indeed a better choice. However, if we do not know the target distribution, then h_2 might be the better choice since it will always incur relatively small loss, as opposed to h_1 which might have very poor performance, say if $P(X = 0)$ and $P(X = 1)$ are switched during test time. While pursuing robustness against any change in $P(X)$ is difficult, we will show below that learning with the HSIC-loss provides a natural trade-off between the generalization guarantees on the unknown target, and the complexity of the changes in the target relative to the source distribution.

Our contributions relative to the first proposal by Mooij et al. (2009) are as follows:

1. We prove that the HSIC objective is learnable for an hypothesis class of bounded Rademacher complexity.

2. We prove that minimizing the HSIC-loss minimizes a worst-case loss over a class of unsupervised covariate shift tasks.

3. We provide experimental validation using both linear models and deep networks, showing that learning with the HSIC-loss is competitive on a variety of unsupervised covariate shift benchmarks.

4. We provide code, including a PyTorch (Paszke et al., 2019) class for the HSIC-loss[1].

2. Background and Setup

The Hilbert-Schmidt independence criterion (HSIC), introduced by Gretton et al. (2005a; 2008), is a useful method for testing if two random variables are independent. We give its basics below.

The root of the idea is that while $\mathbb{C}\mathrm{ov}(A, B) = 0$ does not imply that two random variables A and B are independent, having $\mathbb{C}\mathrm{ov}(s(A), t(B)) = 0$ for all bounded continuous functions s and t **does** actually imply independence (Rényi, 1959). Since going over all bounded continuous functions is not tractable, Gretton et al. (2005b) propose evaluating $\sup_{s\in\mathcal{F}, t\in\mathcal{G}} \mathbb{C}\mathrm{ov}\left[s(x), t(y)\right]$ where $\mathcal{F}, \mathcal{G}$ are universal Reproducing Kernel Hilbert Spaces (RKHS). This allows for a tractable computation and is equivalent in terms of the independence property. Gretton et al. (2005a) then introduced HSIC as an upper bound to the measure introduced by Gretton et al. (2005b), showing it has superior performance and is easier to work with statistically and algorithmically.

2.1. RKHS Background

A reproducing kernel Hilbert space $\mathcal{F}$ is a Hilbert space of functions from $\mathcal{X}$ to $\mathbb{R}$ with the following (reproducing) property: there exist a positive definite kernel $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ and a mapping function ϕ from $\mathcal{X}$ to $\mathcal{F}$ s.t. $K(x_1, x_2) = \langle \phi(x_1), \phi(x_2)\rangle_{\mathcal{F}}$. Given two separable (having a complete orthonormal basis) RKHSs $\mathcal{F}$ and $\mathcal{G}$ on metric spaces $\mathcal{X}$ and $\mathcal{Y}$, respectively, and a linear operator $C : \mathcal{F} \to \mathcal{G}$, the Hilbert-Schmidt norm of C is defined as follows:

$$\|C\|_{\text{HS}} = \sum_{i,j} \langle Cu_i, v_j\rangle^2_{\mathcal{G}},$$

where $\{u_i\}$ and $\{u_i\}$ are some orthonormal basis for $\mathcal{F}$ and $\mathcal{G}$ respectively. Here we consider two probability spaces $\mathcal{X}$ and $\mathcal{Y}$ and their corresponding RKHSs $\mathcal{F}$ and $\mathcal{G}$. The mean elements μ_x and μ_y are defined such that $\langle \mu_x, s\rangle_{\mathcal{F}} := \mathbb{E}[\langle \phi(x), s\rangle_{\mathcal{F}}] = \mathbb{E}[s(x)]$, and likewise $\langle \mu_y, t\rangle_{\mathcal{G}} := \mathbb{E}[\langle \psi(y), t\rangle_{\mathcal{G}}] = \mathbb{E}[t(y)]$, where ψ is the embedding from $\mathcal{Y}$ to $\mathcal{G}$. Notice that we can compute the norms of those operators quite easily: $\|\mu_x\|^2_{\mathcal{F}} = \mathbb{E}[K(x_1, x_2)]$ where the expectation is done over i.i.d. samples of pairs from $\mathcal{X}$. For $s \in \mathcal{F}$ and $t \in \mathcal{G}$, their tensor product $s \otimes t : \mathcal{G} \to \mathcal{F}$ is defined as follows: $(s \otimes t)(h) = \langle t, h\rangle_{\mathcal{G}} \cdot s$. The Hilbert-Schmidt norm of the tensor product can be shown to be given by $\|s \otimes t\|^2_{\text{HS}} = \|s\|^2_{\mathcal{F}} \cdot \|t\|^2_{\mathcal{G}}$. Equipped with these definitions, we are ready to define the cross covariance operator $C_{xy} : \mathcal{G} \to \mathcal{F}$:

$$C_{xy} = \mathbb{E}[\phi(x) \otimes \psi(y)] - \mu_x \otimes \mu_y.$$

[1] https://github.com/danielgreenfeld3/XIC.

2.2. HSIC

Consider two random variables X and Y, residing in two metric spaces $\mathcal{X}$ and $\mathcal{Y}$ with a joint distribution on them, and two separable RKHSs $\mathcal{F}$ and $\mathcal{G}$ on $\mathcal{X}$ and $\mathcal{Y}$ respectively. HSIC is defined as the Hilbert Schmidt norm of the cross covariance operator:

$$\text{HSIC}(X, Y; \mathcal{F}, \mathcal{G}) \equiv \|C_{xy}\|^2_{\text{HS}}.$$

Gretton et al. (2005a) show that:

$$\text{HSIC}(X, Y; \mathcal{F}, \mathcal{G}) \geq \sup_{s \in \mathcal{F}, t \in \mathcal{G}} \mathbb{C}\text{ov}\left[s(x), t(y)\right], \qquad (2)$$

an inequality which we use extensively for our results.

We now state Theorem 4 of Gretton et al. (2005a)) which shows the properties of HSIC as an independence test:

Theorem 1 (Gretton et al. (2005a), Theorem 4)**.** *Denote by $\mathcal{F}$ and $\mathcal{G}$ RKHSs both with universal kernels, k, l respectively on compact domains $\mathcal{X}$ and $\mathcal{Y}$. Assume without loss of generality that $\|s\|_\infty \leq 1$ for all $s \in \mathcal{F}$ and likewise $\|t\|_\infty \leq 1$ for all $t \in \mathcal{G}$.*

Then the following holds: $\|C_{xy}\|^2_{HS} = 0 \Leftrightarrow X \perp\!\!\!\perp Y$.

Let $\{(x_i, y_i)\}_{i=1}^n$ be i.i.d. samples from the joint distribution on $\mathcal{X} \times \mathcal{Y}$. The empirical estimate of HSIC is given by:

$$\widehat{\text{HSIC}}\{(x_i, y_i)\}_{i=1}^n; \mathcal{F}, \mathcal{G}) = \frac{1}{(n-1)^2}\textbf{tr}KHLH, \qquad (3)$$

where $K_{i,j} = k(x_i, x_j)$, $L_{i,j} = l(y_i, y_j)$ are kernel matrices for the kernels k and l respectively, and $H_{i,j} = \delta_{i,j} - \frac{1}{n}$ is a centering matrix. The main result of Gretton et al. (2005a) is that the empirical estimate $\widehat{\text{HSIC}}$ converges to HSIC at a rate of $O\left(\frac{1}{n^{1/2}}\right)$, and its bias is of order $O(\frac{1}{n})$.

3. Proposed Method

Throughout this paper, we consider learning functions taking the form $Y = f^\star(X) + \varepsilon$, where X and ε are independent random variables drawn from a distribution $\mathcal{D}$. This presentation assumes the existence of a mechanism tying together X and Y through $f^\star$, up to independent noise factors. A typical learning approach is to set some hypothesis class $\mathcal{H}$, and attempt to solve the following problem:

$$\min_{h \in \mathcal{H}} \mathbb{E}_{X, \varepsilon \sim \mathcal{D}}[\ell(y, h(x))],$$

where ℓ is often the squared loss function in a regression setting, or the cross entropy loss in case of classification.

Here, following Mooij et al. (2009), we suggest using a loss function which penalizes hypotheses whose residual from Y is not independent of the instance X. Concretely, we pose the following learning problem:

$$\min_{h \in \mathcal{H}} HSIC(X, Y - h(X); \mathcal{F}, \mathcal{G}), \qquad (4)$$

where we approximate the learning problem with empirical samples using $\widehat{HSIC}$ as shown in Eq. (3). Unlike typical loss functions, this loss does not decompose as a sum of losses over each individual sample. In Algorithm 1 we present a general gradient-based method for learning with this loss.

Algorithm 1 Learning with HSIC-loss

```
**Input:** samples $\{(x_i, y_i)\}_{i=1}^n$, kernels $k$, $l$, a hypothesis $h_\theta$ parameterized by $\theta$, and a batch size $m > 1$.
**repeat**
    Sample mini-batch $\{(x_i, y_i)\}_{i=1}^m$
    Compute the residuals $r_i^\theta = y_i - h_\theta(x_i)$
    Compute the kernel matrices $K_{i,j} = k(x_i, x_j)$, and $R^\theta_{i,j} = l(r_i^\theta, r_j^\theta)$
    Compute the HSIC-loss on the mini-batch: Loss($\theta$) = $\textbf{tr}(KHR^\theta H)/(m-1)^2$ where $H_{i,j} = \delta_{i,j} - \frac{1}{m}$
    Update: $\theta \leftarrow \theta - \alpha \cdot \nabla\text{Loss}(\theta)$
**until** convergence
Compute the estimated source bias:
$b \leftarrow \frac{1}{n}\sum_{i=1}^n y_i - \frac{1}{n}\sum_{i=1}^n h_\theta(x_i)$
**Output:** A bias-adjusted hypothesis $h(x) = h_\theta(x) + b$
```

As long as the kernel functions $k(\cdot, \cdot)$ and $l(\cdot, \cdot)$ are differentiable, taking the gradient of the HSIC-loss is simple with any automatic differentiation software (Paszke et al., 2019; Abadi et al., 2015). We note that HSIC$(X, Y - h(X); \mathcal{F}, \mathcal{G})$ is exactly the same for any two functions $h_1(X), h_2(X)$ who differ only by a constant. This can be seen by the examining the role of the centering matrix H, or from the invariance of the covariance operator under constant shifts. Therefore, the predictor obtained from solving (4) is determined only up to a constant term. To determine the correct bias, one can infer it from the empirical mean of the observed Y values. We note that it is possible to add a regularization term to the loss function. In our experiments we used standard regularization techniques such as L2 norm weight and early stopping, setting them by standard (source distribution only) cross-validation.

3.1. Understanding the HSIC-loss

Here we provide two additional views on the HSIC-loss, motivating its use in cases which go beyond additive noise.

The first is based on the observation that, up to a constant, the residual $Y - h(X)$ is the gradient of the squared error with respect to $h(X)$. Intuitively, this means that by optimizing for the residual to be independent of X, we ask that the direction and magnitude in which we need to update $h(X)$ to improve the loss is the same regardless of X. Put differently, the gradient of $h(X)$ would be the same for every subset of X. This is also true for classification tasks: consider the outputs of a classification network as logits o which are then transformed by Sigmoid or Softmax operations into a probability vector h. The gradient of the standard cross-entropy loss with respect to o is exactly the residual $Y - h(X)$. Thus, even when not assuming additive noise, requiring that the residual would be independent of X encourages learning a model for which the gradients of the loss have no information about the instances X.

The second interpretation concerns the question of what does it mean for a model $h(X)$ to be optimal with respect to predicting Y from X. One reasonable way to define optimality is when $h(X)$ captures all the available information that X has about the label Y. That is, a classifier is optimal when:

$$Y \perp\!\!\!\perp X \mid h(X). \tag{5}$$

This is also related to the condition implied by recent work on Invariant Risk Minimization (Arjovsky et al., 2019). Optimizing for the condition in equation 5 is difficult because of the conditioning on $h(X)$. We show in the supplemental that attaining the objective encouraged by the HSIC-loss, namely learning a function $h(X)$ such that $Y - h(X) \perp\!\!\!\perp X$, implies the optimality condition 5.

4. Theoretical Results

We now prove several properties of the HSIC-loss, motivating its use as a loss function which emphasizes robustness against distribution shifts. We consider models of the form given in Eq. (1) such that ε has zero mean. Assume that $X \in \mathcal{X}$ and $Y \in \mathcal{Y}$, where $\mathcal{X}, \mathcal{Y}$ are compact metric spa ces. Denote by $\mathcal{F}$ and $\mathcal{G}$ reproducing kernel Hilbert spaces of functions from $\mathcal{X}$ and $\mathcal{Y}$ respectively, to $\mathbb{R}$ s.t. that $\|f\|_{\mathcal{F}} \leq M_{\mathcal{F}}$ for all $f \in \mathcal{F}$ and $\|g\|_{\mathcal{G}} \leq M_{\mathcal{G}}$ for all $g \in \mathcal{G}$. We will use $M_{\mathcal{G}}$ and $M_{\mathcal{F}}$ throughout this section. Omitted proofs can be found in the supplement. Denote by $\tilde{\mathcal{F}}$ and $\tilde{\mathcal{G}}$ the restriction of $\mathcal{F}$ and $\mathcal{G}$ to functions in the unit ball of the respective RKHS. Before we state the results, we give the following useful lemma:

Lemma 2. *Suppose $\mathcal{F}$ and $\mathcal{G}$ are RKHSs over $\mathcal{X}$ and $\mathcal{Y}$, s.t. $\|s\|_{\mathcal{F}} \leq M_{\mathcal{F}}$ for all $s \in \mathcal{F}$ and $\|t\|_{\mathcal{G}} \leq M_{\mathcal{G}}$ for all $t \in \mathcal{G}$. Then the following holds:*

$$\sup_{s \in \mathcal{F}, t \in \mathcal{G}} \mathbb{C}ov[s(X), t(Y)] = M_{\mathcal{F}} \cdot M_{\mathcal{G}} \sup_{s \in \tilde{\mathcal{F}}, t \in \tilde{\mathcal{G}}} \mathbb{C}ov[s(X), t(Y)].$$

4.1. Lower Bound

We first relate HSIC-loss to standard notions of model performance: we show that under mild assumptions, the HSIC-loss is an upper bound to the variance of the residual $f^\star(X) - h(X)$. The additional assumption is that for all $h \in \mathcal{H}$, $f^\star - h$ is in the closure of $\mathcal{F}$, and that $\mathcal{G}$ contains the identity function from $\mathbb{R}$ to $\mathbb{R}$. This means that $M_{\mathcal{F}}$ acts as a measure of complexity of the true function $f^\star$ that we trying to learn. Note however this does not imply that $f^\star \in \mathcal{H}$, but rather this is an assumption on the kernel space used to calculate the HSIC term.

Theorem 3. *Under the conditions specified above:*

$$\mathbb{V}ar(f^\star(X) - h(X)) \leq M_{\mathcal{F}} \cdot M_{\mathcal{G}} \cdot \textbf{\textit{HSIC}}(X, Y - h(X); \tilde{\mathcal{F}}, \tilde{\mathcal{G}}).$$

Recalling the bias-variance decomposition:

$$\mathbb{E}\left[(Y - h(X))^2\right] =$$

$$\mathbb{V}\text{ar}\left(f^\star(X) - h(X)\right) + \left(\mathbb{E}\left[f^\star(X) - h(X)\right]\right)^2 + \mathbb{E}\left[\varepsilon^2\right],$$

we see that the HSIC-loss minimizes the variance part of the mean squared error (MSE). To minimize the entire MSE, the learned function should be adjusted by adding a constant which can be inferred from the empirical mean of Y.

4.1.1. The Realizable Case

If $h \in \mathcal{H}$ has HSIC-loss equal to zero, then up to a constant term, it is the correct function:

Corollary 4. *Under the assumptions of Theorem 3, we have the following:*

$$\textit{HSIC}\left(X, Y - h(X); \tilde{\mathcal{F}}, \tilde{\mathcal{G}}\right) = 0 \Rightarrow h(X) = f^\star(X) + c,$$

almost everywhere.

Proof. From Theorem 3, we have that

$$\text{HSIC}\left(X, Y - h(X); \tilde{\mathcal{F}}, \tilde{\mathcal{G}}\right) = 0 \implies$$
$$\mathbb{V}\text{ar}\left(f^\star(X) - h(X)\right) = 0,$$

therefore $f^\star(X) - h(X)$ must be a constant up to a zero-probability set of X. □

4.2. Robustness Against Covariate Shift

Due to its formulation as a supremum over a large set of functions, the HSIC-loss is an upper bound to a natural notion of robustness. This notion, which will be formalised below, captures the amount by which the performance of a model might change as a result of a covariate shift, where the performance is any that can be measured by some $\ell \in \mathcal{G}$ applied on the residuals $Y - h(X)$. In this subsection we denote the functions in $\mathcal{G}$ as ℓ instead of t, to emphasize that we now think of $\ell(r)$ as possible loss functions acting on the residuals.

We consider two different ways of describing a target distribution which is different from the source. The first is by specifying the density ratio between the target and source distributions. This is useful when the support of the distribution does not change but only the way it is distributed. A second type of covariate shift is due to restricting the support of the distribution to a certain subset. This can be described by an indicator function which states which parts of the source domain are included. The following shows how the HSIC-loss is an upper bound to the degradation in model performance in both covariate shift formulations.

We start with the latter case. For a subset $A \subset \mathcal{X}$ of positive measure, the quantity comparing a model's performance in terms of the loss ℓ on the source distribution and the same model's performance when the target distribution is restricted to A, is as follows:

$$\frac{1}{\mathbb{E}\left[1_A(x)\right]}\mathbb{E}\left[1_A(x)\ell\left(y-h\left(x\right)\right)\right]-\mathbb{E}\left[\ell\left(y-h\left(x\right)\right)\right]$$

For $\delta, c > 0$ let $\mathcal{W}c, \delta$ denote the family of subsets A with source probability at least $c > 0$ s.t. there exists some $s \in \mathcal{F}$ which is δ close to 1_A:

$$\mathcal{W}_{c,\delta}=\{A\subset\mathcal{X}|\exists s\in\mathcal{F}\text{ s.t. }\|1_A-s\|_\infty\le\delta,\,\mathbb{E}\left[1_A(x)\right]\ge c\}.$$

All these subset can be approximately described by functions from $\mathcal{F}$. The complexity of such subsets is naturally controlled by $M_{\mathcal{F}}$.

Theorem 5. *Let $\ell \in \mathcal{G}$ be a non-negative loss function, and let $\delta, c > 0$:*

$$\begin{aligned}&\sup_{A\in\mathcal{W}c,\delta}\frac{1}{\mathbb{E}\left[1_A(x)\right]}\mathbb{E}\left[1_A(x)\ell\left(y-h\left(x\right)\right)\right]\le\\&\frac{M_{\mathcal{F}}M_{\mathcal{G}}HSIC(X,Y-h(X);\tilde{\mathcal{F}},\tilde{\mathcal{G}})}{c}\\&+\left(\frac{2\delta}{c}+1\right)\mathbb{E}\left[\ell\left(y-h\left(x\right)\right)\right].\end{aligned}$$

Theorem 5 states that the degradation in performance due to restricting the support of the distribution to some subset is bounded by terms related to the size of the set and the ability to represent it by $\mathcal{F}$. Compare this to the following naive bound:

$$\sup_{A\in\mathcal{W}c,\delta}\frac{\mathbb{E}\left[1_A(x)\ell\left(y-h\left(x\right)\right)\right]}{\mathbb{E}\left[1_A(x)\right]}\le\frac{\mathbb{E}\left[\ell\left(y-h\left(x\right)\right)\right]}{c}.$$

Failing to account how the loss is distributed across different subsets of $\mathcal{X}$, as done in the HSIC-loss, leads to poor generalization guarantees. Indeed, the naive bound will not be tight for the original function, i.e. $h = f^\star$, but the HSIC based bound will be tight whenever $\delta \ll c$.

Returning to the first way of describing covariate shifts, we denote by $P_{\text{source}}(X)$ the density function of the distribution on $\mathcal{X}$ from which the training samples are drawn, and $P_{\text{target}}(X)$ the density of an unknown target distribution over $\mathcal{X}$.

Theorem 6. *Let $\mathcal{Q}$ denote the set of density functions on $\mathcal{X}$ which are absolutely continuous w.r.t. $P_{source}(X)$, and their density ratio is in $\mathcal{F}$:*

$$\begin{aligned}\mathcal{Q}=\Bigg\{&P_{target}:\mathcal{X}\to\mathbb{R}_{\ge0}\quad s.t.\ \mathbb{E}_{x\sim P_{target}}\left[1\right]=1,\\&\mathbb{E}_{x\sim P_{source}}\left[\frac{P_{target}(x)}{P_{source}(x)}\right]=1,\,\frac{P_{target}}{P_{source}}\in\mathcal{F}\Bigg\}.\end{aligned}$$

Then,

$$\begin{aligned}&\sup_{\substack{P_{target}\in\mathcal{Q}\\ \ell\in\mathcal{G}}}\mathbb{E}_{x\sim P_{target}}[\ell(Y-h(X))]-\mathbb{E}_{x\sim P_{source}}[\ell(Y-h(X))]\\&\le M_{\mathcal{F}}\cdot M_{\mathcal{G}}\cdot HSIC(X,Y-h(X);\tilde{\mathcal{F}},\tilde{\mathcal{G}}),\end{aligned}$$

where HSIC is of course evaluated on the training distribution P_{source}.

Combining Theorem 6 and the lower bound of Theorem 3, we obtain the following result:

Corollary 7. *Under the same assumptions of Theorems 3 and 6, further assume that the square function $x \mapsto x^2$, belongs to $\mathcal{G}$ or its closure. Denote: $\delta_{HSIC}(h) = HSIC(X, Y - h(X); \tilde{\mathcal{F}}, \tilde{\mathcal{G}})$, $MSE_{P_{target}}(h) = \mathbb{E}_{P_{target}}[(Y - h(X))^2]$, $bias_{source}(h) = \mathbb{E}_{P_{source}}[f^\star(x) - h(x)]$, and $\sigma^2 = \mathbb{E}[\varepsilon^2]$. Then:*

$$\begin{aligned}&\sup_{P_{target}\in\mathcal{Q}}MSE_{P_{target}}(h)\\&\le 2M_{\mathcal{F}}\cdot M_{\mathcal{G}}\cdot\delta_{HSIC}(h)+bias_{source}(h)^2+\sigma^2.\end{aligned}$$

Theorem 6 and Corollary 7 show that minimizing HSIC minimizes an upper bound on the worst case loss relative to a class of target distribution whose complexity is determined by the norm of the RKHS $M_{\mathcal{F}}$. Compared to a naive bound based on the infinity norm of the density ratio, this bound is much tighter when considering $f^\star$ for example, and by continuity for functions near it. Further discussion can be found in the supplementary material.

4.3. Learnability: Uniform Convergence

By formulating the HSIC learning problem as a learning problem over pairs of samples from $\mathcal{X} \times \mathcal{X}$ with specially constructed labels, we can reduce the question of HSIC learnability to a standard learning theory problem (Mohri et al. (2018), Ch 3). We use this reduction to prove that it is possible to minimize the HSIC objective on hypothesis classes $\mathcal{H}$ with bounded Rademacher complexity $\mathcal{R}_n(\mathcal{H})$ using a finite sample.

Theorem 8. *Suppose the residuals' kernel k is bounded in $[0, 1]$ and satisfies the following condition: $k(r, r') = \iota(h(x) - h(x'), y - y')$ where $\iota : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}$ is s.t. $\iota(\cdot, y)$ is an L_ι−Lipschitz function for all y. Let $C_1 = \sup_{x,x'} l(x, x')$, $C_2 = \sup_{r,r'} k(r, r')$. Then, with probabil-*

ity of at least $1-\delta$, the following holds for all $h \in \mathcal{H}$:

$$\left| \mathit{HSIC}\left(X, Y - h(X); \mathcal{F}, \mathcal{G}\right) - \widehat{\mathit{HSIC}}\left(\{(x_i, r_i)\}_{i=1}^{n}; \mathcal{F}, \mathcal{G}\right)\right| \\ \leq 3C_1 \left(4L_\iota \mathcal{R}_n(\mathcal{H}) + O\left(\sqrt{\frac{\ln(1/\delta)}{n}}\right)\right) + 3C_2C_1\sqrt{\frac{\ln(2/\delta)}{2n}}.$$

5. Related Work

As mentioned above, Mooij et al. (2009) were the first to propose using the HSIC loss as a means to learn a regression model. However, their work focused exclusively on learning the functions corresponding to edges in a causal graph, and leveraging that to learn causal directions and then the structure of the graph itself. They have not applied the method to domain adaptation or to robust learning, nor did they analyze the qualities of this objective as a loss function.

The literature on robust learning is rapidly growing in size and we cannot hope to cover it all here. Especially relevant papers on robust learning for *unsupervised* domain adaptation are Namkoong and Duchi (2017); Volpi et al. (2018b); Duchi and Namkoong (2018). Volpi et al. (2018b) propose an iterative process whereby the training set is augmented with adversarial examples that are close in some feature space, to obtain a perturbation of the distribution. Namkoong and Duchi (2017) suggest minimizing the variance of the loss in addition to its empirical mean, and employ techniques from learning distributionally robust models. Some recent papers have highlighted strong connections between causal inference and robust learning, see e.g. the works of Heinze-Deml and Meinshausen (2017) and Rothenhäusler et al. (2018). By having some knowledge on the corresponding data generating graph of the problem, Heinze-Deml and Meinshausen (2017) propose minimizing the variance under properties that are presupposed to have no impact on the prediction. A more general means of using the causal graph to learn robust models is given by Subbaswamy and Saria (2018); Subbaswamy et al. (2019), who propose a novel *graph surgery estimator* which specifically takes account of factors in the data which are known apriori to be vulnerable to changes in the distribution. These methods require detailed knowledge of the causal graph and are computationally heavy when the dimension of the problem grows. In (Rothenhäusler et al., 2018), the authors propose using anchors, which are covariates that are known to be exogenous to the prediction problem, to obtain robustness against distribution shifts induced by the anchors. Of course, a large body of work exist on covariate shift learning when there is access to unlabeled test data (see, e.g., Daume III and Marcu (2006); Saenko et al. (2010); Gretton et al. (2009); Tzeng et al. (2017); Volpi et al. (2018a)), however we stress that we do not require such access.

6. Experimental Results

To evaluate the performance of the HSIC loss function, we experiment with synthetic and real-world data. We focus on tasks of unsupervised transfer learning: we train on a one distribution, called the SOURCE distribution, and test on a different distribution, called the TARGET distribution. We assume we have no samples from the target distribution during learning.

6.1. Synthetic Data

As a first evaluation of the HSIC-loss, we experiment with fitting a linear model. We focus on small sample sizes as those often lead to difficulties in covariate shift scenarios. The underlying model in the experiments is $y = \beta^\top x + \varepsilon$ where $\beta \in \mathbb{R}^{100}$ is drawn for each experiment from a Gaussian distribution with $\sigma = 0.1$. In the training phase, x is drawn from a uniform distribution over $[-1, 1]^{100}$. We experimented with ε drawn from one of three distributions: Gaussian, Laplacian, or a shifted exponential: $\varepsilon = 1 - e$ where e is drawn from an exponential distribution $\exp(1)$. In any case, ε is drawn independently from x. In each experiment, we draw $n \in \{2^i\}_{i=5}^{13}$ training samples and train using either using squared-loss, absolute-loss, and HSIC-loss, all with an l_2 regularization term. The SOURCE test set is created in the same manner as the training set was created, while the TARGET test set simulates a covariate shift scenario. This is done by changing the marginal distribution of x from a uniform distribution to a Gaussian distribution over $\mathbb{R}^{100}$. In all cases the noise on the SOURCE and TARGET is drawn from the same distribution. This process is repeated 20 times for each n. When training the models with HSIC-loss, we used batch-size of 32, and optimized using Adam optimizer (Kingma and Ba, 2014). The kernels we chose were radial basis function kernels, with $\gamma = 1$ for both covariates' and residuals' kernels.

Figure 1 presents the results of experiments with Gaussian, Laplacian, and shifted-exponential noise. With Gaussian noise, HSIC-loss performs similarly to squared-loss regression, and with Laplacian noise HSIC-loss performs similarly to absolute-loss regression, where squared-loss is the maximum-likelihood objective for Gaussian noise and absolute-loss is the maximum-likelihood objective for Laplacian noise. In both cases it is reassuring to see that HSIC-loss is on par with the maximum-likelihood objectives. In all cases we see that HSIC-loss is better on the TARGET distribution compared to objectives which are not the maximum likelihood objective. This is true especially in small sample sizes. We believe this reinforces our result in Theorem 6 that the HSIC-loss is useful when we do not know in advance the loss or the exact target distribution.

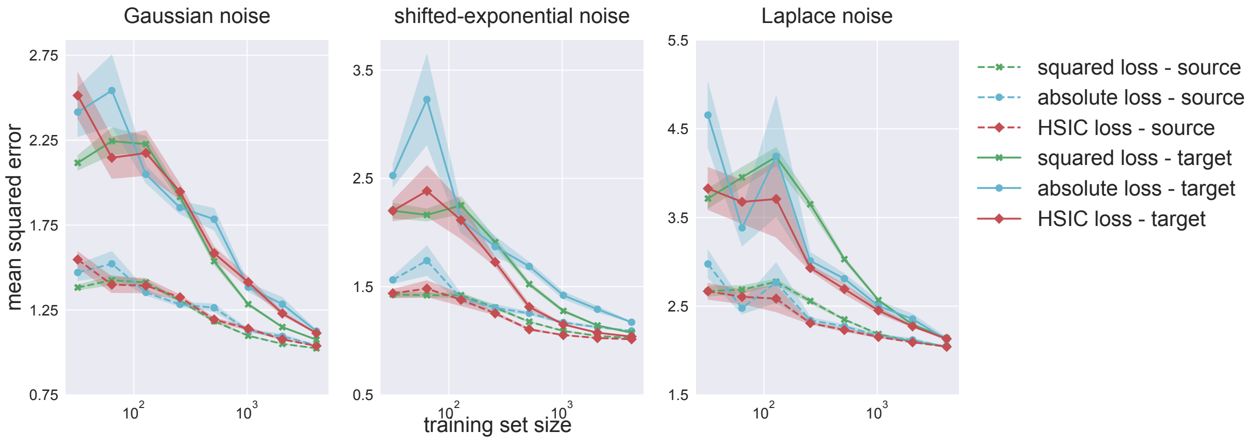

Figure 1. Comparison of models trained with squared-loss, absolute-loss and HSIC-loss. Each point on the graph is the MSE averaged over 20 experiments, and the shaded area represents one standard error from the mean. Dashed lines are the MSE evaluated over the source distribution, solid lines are the MSE evaluated over the target distribution.

6.2. Bike Sharing Dataset

In the bike sharing dataset by Fanaee-T and Gama (2014) from the UCI repository, the task is to predict the number of hourly bike rentals based on the following covariates: temperature, feeling temperature, wind speed and humidity. Consisting of 17,379 samples, the data was collected over two years, and can be partitioned by year and season. This dataset has been used to examine domain adaptation tasks by Subbaswamy et al. (2019) and Rothenhäusler et al. (2018). We adopt their setup, where the SOURCE distribution used for training is three seasons of a year, and the TARGET distribution used for testing is the forth season of the same year, and where the model of choice is linear. We compare with least squares, anchor regression (AR) Rothenhäusler et al. (2018) and Surgery by Subbaswamy et al. (2019).

We ran 100 experiments, each of them was done by randomly sub-sampling 80% of the SOURCE set and 80% of the TARGET set, thus obtaining a standard error estimate of the mean. When training the models with HSIC-loss, we used batch-size of 32, and optimized the loss with Adam (Kingma and Ba, 2014), with learning rate drawn from a uniform distribution over $[0.0008, 0.001]$. The kernels we chose were radial basis function kernels, with $\gamma = 2$ for the covariates' kernel, and $\gamma = 1$ for the residuals' kernel.

We present the results in Table 1. Following the discussion in section 4, we report the *variance* of the residuals in the test set. We can see that training with HSIC-loss results in better performances in 6 out of 8 times. In addition, unlike AR and Surgery, training with HSIC-loss does not require knowledge of the specific causal graph of the problem, nor does it require the training to be gathered from different sources as in AR.

Table 1. Variance results on the bike sharing dataset. Each row corresponds to a training set consisting of three season of that year, and the variance of $Y - h(X)$ on the TARGET set consisting of the forth season is reported. In bold are the best results in each experiment, taking into account one standard error.

Test data	OLS	AR	Surgery	HSIC
(Y1) Season 1	**15.4**±0.02	**15.4**±0.02	15.5±0.03	16.0±0.04
Season 2	23.1±0.03	23.1±0.03	23.7±0.04	**22.9**±0.03
Season 3	28.0±0.03	28.0±0.03	28.1±0.03	**27.9**±0.03
Season 4	23.7±0.03	23.7±0.03	25.6±0.04	**23.6**±0.04
(Y2) Season 1	**29.8**±0.05	**29.8**±0.05	30.7±0.06	30.7±0.07
Season 2	39.0±0.05	39.1±0.05	39.2±0.06	**38.9**±0.04
Season 3	41.7±0.05	41.5±0.05	41.8±0.05	**40.8**±0.05
Season 4	38.7±0.04	**38.6**±0.04	40.3±0.06	**38.6**±0.05

6.3. Rotating MNIST

In this experiment we test the performance of models trained on the MNIST dataset by LeCun et al. (1998) as the SOURCE distribution, and digits which are rotated by an angle θ sampled from a uniform distribution over $[-45, 45]$ as the TARGET distribution. Samples of the test data are depicted in the supplementary material. The standard approach to obtain robustness against such perturbations is to augment the training data with images with similar transformations, as in Schölkopf et al. (1996) for example. However, in practice it is not always possible to know in advance what kind of perturbations should be expected, and therefore it is valuable to develop methods for learning robust models even without such augmentations. We compared training with HSIC-loss to training with cross entropy loss, using three types of architectures. The first is a convolutional neural network (CNN):→ *input* → *conv(dim=32)*→ *conv(dim=64)* → *fully-connected(dim=524)*→ *dropout(p=0.5)* → *fully-connected(dim=10)*. The second is a multi-layered percep-

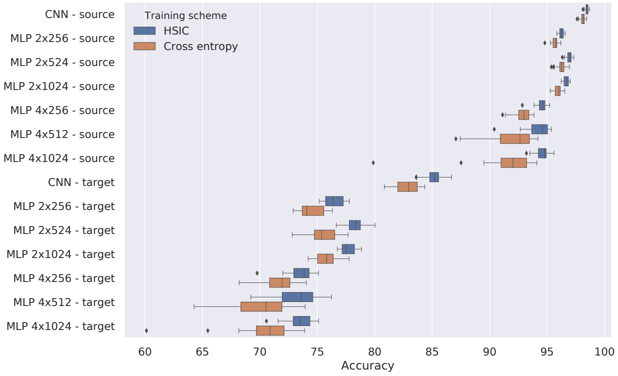

Figure 2. Accuracy on SOURCE and TARGET test sets, with models trained with either cross entropy or HSIC-loss. Plotted are the median, 25th and 75th percentiles.

tron (MLP) with two hidden layers of size $256, 524, 1024$: *input* $\rightarrow$ *fully-connected(dim=256,524,1024)* $\rightarrow$ *fully-connected(dim=256,524,1024)*$\rightarrow$*dropout(p=0.5)* $\rightarrow$ *fully-connected(dim=10)*. The third architecture was also an MLP, except there were four hidden layers instead of two. Each experiment was repeated 20 times, and in every experiment the number of training steps (7 epochs) remained constant for a fair comparison. Each time the training set consisted of 10K randomly chosen samples. The kernels we chose were radial basis function kernels with $\gamma = 1$ for the residuals, and $\gamma = 22$ for the images, chosen according to the heuristics suggested by Mooij et al. (2009). The results are depicted in Figure 2. We see that for all models, moving to the TARGET distribution induces a large drop in accuracy. Yet for all architectures we see that using HSIC-loss gives better performance on the TARGET set compared to using the standard cross-entropy loss.

6.4. Cell Out of Sample Dataset

In the last experiment, we test our approach on the cell out of sample dataset introduced by Lu et al. (2019). This dataset was collected for the purpose of measuring robustness against covariate shift. It consists of 64×64 microscopy images of mouse cells stained with one of seven possible fluorescent proteins (highlighting distinct parts of the cell), and the task is to predict the type of the fluorescent protein used to stain the cell. Learning systems trained on microscopy data are known to suffer from changes in plates, wells and instruments (Caicedo et al., 2017). Attempting to simulate these conditions, the dataset contains four test sets, with increasing degrees of change in the covariates' distribution, as described in Table 2, adopted from Lu et al. (2019). Following (Lu et al., 2019), we trained an 11-layer CNN, DeepLoc, used in (Kraus et al., 2017) for protein subcellular localization. We followed the pre-processing, data augmentation, architecture choice, and training procedures described there, with the exception of using HSIC-loss and different learning rate when using HSIC. When computing HSIC, the kernel width was set to 1 for both kernels. Training was done for 50 epochs on 80% of the SOURCE dataset, and the final model was chosen according to the remaining 20% used as a validation set. The optimization was done with Adam (Kingma and Ba, 2014), with batch size of 128, and exponential decay of the learning rate was used when training with cross-entropy loss. Lu et al. (2019) used data augmentation during training (random cropping and random flips) and test time (prediction is averaged over 5 crops taken from the corners and center image), as this is a common procedure to encourage robustness. We compared HSIC-based models to cross-entropy based models both with and without test time augmentation. We note that (Lu et al., 2019) examined several deep net models and DeepLoc (with cross-entropy training) had the best results.

Table 2. Description of the source and target distributions in the cell out of sample dataset

Dataset	Description	Size
Source	Images from 4 independent plates for each class	41,456
Target1	Held out data	10,364
Target2	Same plates, but different wells	17,021
Target3	2 independent plates for each class, different days	32,596
Target4	1 plate for each class, different day and microscope	30,772

Table 3. Class balanced accuracy on each of the four target distributions. The last row depicts the results of training with cross-entropy as reported in (Lu et al., 2019). HSIC-aug and CE-aug refer to experiments done with test time augmentation.

Training loss	Target1	Target2	Target3	Target4
HSIC	99.2	98.8	93.4	95.3
CE	98.4	98.1	91.7	93.8
HSIC-aug	99.2	98.9	93.4	95.4
CE-aug-(Lu et al., 2019)	98.8	98.5	92.6	94.6

Table 3 depicts the results, showing a clear advantage of the HSIC-based model which is able to achieve new state-of-the-art results in the more difficult TARGET distributions, while preserving the performance in TARGET distributions closer to the SOURCE.

7. Conclusion

In this paper we propose learning models whose errors are independent of their inputs. This can be viewed as a non-parametric generalization of the way residuals are orthogonal to the instances in OLS regression. We prove that the HSIC-loss is learnable in terms of uniform convergence, and show that this loss naturally comes with a strong notion of robustness against changes in the input distribution when the change can be described in a bounded RKHS. The main theoretical limitations of our approach are the assumption that the changes are only in the marginal distribution of X, and that those changes are smooth in some sense. However, we show in our experiments that, in comparison to standard loss functions, the HSIC-loss produces models which perform just as well on the source distribution, and are significantly better on the target distribution, including state-of-the-art results on a challenging benchmark. An interesting future direction is to better understand the connection between this type of loss, originally proposed in the context of learning causal graph structure, and the idea of learning causal models that are expected to be robust against distributional changes (Meinshausen, 2018; Arjovsky et al., 2019).

8. Acknowledgments

The authors would like to thank David Lopez-Paz for helpful discussions. This research was partially supported by the Israel Science Foundation (grant No. 1950/19).

References

M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL http://tensorflow.org/. Software available from tensorflow.org.

M. Arjovsky, L. Bottou, I. Gulrajani, and D. Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

J. C. Caicedo, S. Cooper, F. Heigwer, S. Warchal, P. Qiu, C. Molnar, A. S. Vasilevich, J. D. Barry, H. S. Bansal, O. Kraus, et al. Data-analysis strategies for image-based cell profiling. *Nature methods*, 14(9):849, 2017.

H. Daume III and D. Marcu. Domain adaptation for statistical classifiers. *Journal of artificial Intelligence research*, 26:101–126, 2006.

J. Duchi and H. Namkoong. Learning models with uniform performance via distributionally robust optimization. *arXiv preprint arXiv:1810.08750*, 2018.

H. Fanaee-T and J. Gama. Event labeling combining ensemble detectors and background knowledge. *Progress in Artificial Intelligence*, 2(2-3):113–127, 2014.

A. Gretton, O. Bousquet, A. Smola, and B. Schölkopf. Measuring statistical dependence with hilbert-schmidt norms. In *International conference on algorithmic learning theory*, pages 63–77. Springer, 2005a.

A. Gretton, R. Herbrich, A. Smola, O. Bousquet, and B. Schölkopf. Kernel methods for measuring independence. *Journal of Machine Learning Research*, 6(Dec): 2075–2129, 2005b.

A. Gretton, K. Fukumizu, C. H. Teo, L. Song, B. Schölkopf, and A. J. Smola. A kernel statistical test of independence. In *Advances in neural information processing systems*, pages 585–592, 2008.

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

C. Heinze-Deml and N. Meinshausen. Conditional variance penalties and domain shift robustness. *arXiv preprint arXiv:1710.11469*, 2017.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

O. Z. Kraus, B. T. Grys, J. Ba, Y. Chong, B. J. Frey, C. Boone, and B. J. Andrews. Automated analysis of high-content microscopy data with deep learning. *Molecular systems biology*, 13(4), 2017.

Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

A. Lu, A. Lu, W. Schormann, M. Ghassemi, D. Andrews, and A. Moses. The cells out of sample (coos) dataset and benchmarks for measuring out-of-sample generalization of image classifiers. In *Advances in Neural Information Processing Systems*, pages 1854–1862, 2019.

N. Meinshausen. Causality from a distributional robustness point of view. In *2018 IEEE Data Science Workshop (DSW)*, pages 6–10. IEEE, 2018.

M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of machine learning*. MIT press, 2018.

J. Mooij, D. Janzing, J. Peters, and B. Schölkopf. Regression by dependence minimization and its application to causal inference in additive noise models. In *Proceedings of the 26th annual international conference on machine learning*, pages 745–752. ACM, 2009.

H. Namkoong and J. C. Duchi. Variance-based regularization with convex objectives. In *Advances in Neural Information Processing Systems*, pages 2971–2980, 2017.

A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, pages 8026–8037, 2019.

A. Rényi. On measures of dependence. *Acta mathematica hungarica*, 10(3-4):441–451, 1959.

A. Rosenfeld, R. Zemel, and J. K. Tsotsos. The elephant in the room. *arXiv preprint arXiv:1808.03305*, 2018.

D. Rothenhäusler, N. Meinshausen, P. Bühlmann, and J. Peters. Anchor regression: heterogeneous data meets causality. *arXiv preprint arXiv:1801.06229*, 2018.

K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *European conference on computer vision*, pages 213–226. Springer, 2010.

B. Schölkopf, C. Burges, and V. Vapnik. Incorporating invariances in support vector learning machines. In *Artificial Neural Networks: ICANN 96, LNCS vol. 1112*, pages 47–52, Berlin, Germany, July 1996. Max-Planck-Gesellschaft, Springer. volume 1112 of Lecture Notes in Computer Science.

A. Subbaswamy and S. Saria. Counterfactual normalization: Proactively addressing dataset shift using causal mechanisms. In *34th Conference on Uncertainty in Artificial Intelligence 2018, UAI 2018*, pages 947–957. Association For Uncertainty in Artificial Intelligence (AUAI), 2018.

A. Subbaswamy, P. Schulam, and S. Saria. Preventing failures due to dataset shift: Learning predictive models that transport. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3118–3127, 2019.

E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7167–7176, 2017.

R. Volpi, P. Morerio, S. Savarese, and V. Murino. Adversarial feature augmentation for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5495–5504, 2018a.

R. Volpi, H. Namkoong, O. Sener, J. C. Duchi, V. Murino, and S. Savarese. Generalizing to unseen domains via adversarial data augmentation. In *Advances in Neural Information Processing Systems*, pages 5334–5344, 2018b.