Domain Generalization by Marginal Transfer Learning

Gilles Blanchard BLANCHARD@MATH.UNI-POTSDAM.DE
Institut für Mathematik
Universität Potsdam

Aniket Anand Deshmukh ANIKETDE@UMICH.EDU
Electrical Engineering and Computer Science
University of Michigan

Ürun Dogan URUNDOGAN@GMAIL.COM
Microsoft Research

Gyemin Lee GYEMIN@SEOULTECH.AC.KR
Dept. Electronic and IT Media Engineering
Seoul National University of Science and Technology

Clayton Scott CLAYSCOT@UMICH.EDU
Electrical Engineering and Computer Science
University of Michigan

Editor: NA

Abstract

Domain generalization is the problem of assigning class labels to an unlabeled test data set, given several labeled training data sets drawn from similar distributions. This problem arises in several applications where data distributions fluctuate because of biological, technical, or other sources of variation. We develop a distribution-free, kernel-based approach that predicts a classifier from the marginal distribution of features, by leveraging the trends present in related classification tasks. This approach involves identifying an appropriate reproducing kernel Hilbert space and optimizing a regularized empirical risk over the space. We present generalization error analysis, describe universal kernels, and establish universal consistency of the proposed methodology. Experimental results on synthetic data and three real data applications demonstrate the superiority of the method with respect to a pooling strategy.

Keywords: Domain Generalization, Kernel Methods, Kernel Approximation

1. Introduction

Is it possible to leverage the solution of one classification problem to solve another? This is a question that has received increasing attention in recent years from the machine learning community, and has been studied in a variety of settings, including multi-task learning, covariate shift, and transfer learning. In this work we study domain generalization, another setting in which this question arises, and one that incorporates elements of the three aforementioned settings and is motivated by many practical applications.

To state the problem, let $\mathcal{X}$ be a feature space and $\mathcal{Y}$ a space of labels to predict. For a given distribution P_{XY} on $\mathcal{X} \times \mathcal{Y}$, we refer to the X marginal distribution P_X as simply the

marginal distribution, and the conditional $P_{XY}(Y|X)$ as the posterior distribution. There are N similar but distinct distributions $P_{XY}^{(i)}$ on $\mathcal{X} \times \mathcal{Y}$, $i = 1, \ldots, N$. For each i, there is a training sample $S_i = (X_{ij}, Y_{ij})_{1 \leq j \leq n_i}$ of iid realizations of $P_{XY}^{(i)}$. There is also a test distribution P_{XY}^T that is similar to but again distinct from the "training distributions" $P_{XY}^{(i)}$. Finally, there is a test sample $(X_j^T, Y_j^T)_{1 \leq j \leq n_T}$ of iid realizations of P_{XY}^T, but in this case the labels Y_j are not observed. The goal is to correctly predict these unobserved labels. Essentially, given a random sample from the marginal test distribution P_X^T, we would like to predict the corresponding labels.

Domain generalization, which has also been referred to as learning to learn or lifelong learning, may be contrasted with other learning problems. In multi-task learning, only the training distributions are of interest, and the goal is to use the similarity among distributions to improve the training of individual classifiers (Thrun, 1996; Caruana, 1997; Evgeniou et al., 2005). In our context, we view these distributions as "training tasks," and seek to generalize to a new distribution/task.[1]. In the covariate shift problem, the marginal test distribution is different from the marginal training distribution(s), but the posterior distribution is assumed to be the same (Bickel et al., 2009). In our case, both marginal and posterior test distributions can differ from their training counterparts (Quionero-Candela et al., 2009).

Finally, in transfer learning, it is typically assumed that at least a few labels are available for the test data, and the training data sets are used to improve the performance of a standard classifier, for example by learning a metric or embedding which is appropriate for all data sets (Ando and Zhang, 2005; Rettinger et al., 2006). In our case, no test labels are available, but we hope that through access to multiple training data sets, it is still possible to obtain collective knowledge about the "labeling process" that may be transferred to the test distribution. Some authors have considered transductive transfer learning, which is similar to the problem studied here in that no test labels are available. However, existing work has focused on the case $N = 1$ and typically relies on the covariate shift assumption (Arnold et al., 2007).

We propose a distribution-free, kernel-based approach to domain generalization, based on the hypothesis that information about task is encoded in its marginal distribution. Our methodology is shown to yield a consistent learning procedure, meaning that the generalization error tends to the best possible as the sample sizes $N, \{n_i\}, n_T$ tend to infinity. We also offer a thorough experimental study validating the proposed approach on a synthetic data set, and on three real-world datasets, including comparisons to a simple pooling approach.

The general probabilistic framework we adopt to theoretically analyze the proposed algorithm is to assume that the training task generating distributions $P_{XY}^{(i)}$ as well as the test task distribution P_{XY}^T are themselves drawn i.i.d. from a probability distribution μ over the set $\mathfrak{P}_{\mathcal{X} \times \mathcal{Y}}$ of probability distributions on $\mathcal{X} \times \mathcal{Y}$. This two-stage task sampling model (a task distribution P_{XY} is sampled from μ, then training examples (X, Y) are sampled from P_{XY}) was first introduced in the seminal work of Baxter (1997, 2000), which also proposed a general learning-theoretical analysis of the model. A generic approach to the problem is to consider a family of hypothesis spaces, and use the training tasks in order to select

1. The terminology appears to vary. Here we call a specific distribution P_{XY} a *task*, but the terms *domain* or *environment* are also common in the literature.

in that family an hypothesis space that is optimally suited to learning tasks sampled from μ; roughly speaking, this means finding a good trade-off between the complexity of said class and its approximation capabilities for tasks sampled from μ, in an average sense. The information gained by finding a well-adapted hypothesis space can lead to a significantly improved label efficiency of learning a new task. A related approach is to learn directly the task sampling distribution μ (Carbonell et al., 2013).

The family of hypothesis spaces under consideration can be implicit, as when learning a feature representation or metric that is suitable for all tasks. In the context of (reproducing) kernel methods, this has been studied under the form of learning a linear transform or projection in kernel space (Maurer, 2009), and learning the kernel itself (Pentina and Ben-David, 2015).

A related approach is to find a transformation (feature extraction) of X so that transformed marginal distributions approximately match across tasks; the underlying assumption is that this allows to find some common information between tasks. This idea has been combined with the principle of kernel mean mapping (which represents entire distributions as points in a Hilbert space) to compare distributions (Pan et al., 2011; Muandet et al., 2013; Maurer et al., 2013; Pentina and Lampert, 2014; Grubinger et al., 2015; Ghifary et al., 2017), generally to find a projection in kernel space realizing a suitable compromise between matching of transformed marginal distributions and preserving of information between input and label. It has also been proposed to match task distributions by optimal transport of marginal distributions (Courty et al., 2016).

In the present paper, our aim is to learn to predict the classifier for a given task from the marginal distribution; for this we will use the principle of kernel mean mapping as well. Still, our ansatz is different from the previously discussed methods, because instead of transforming the data to match distributions, we aim to learn how the task-dependent hypothesis (e.g., a linear classifier) transforms as a function of the marginal. In this sense our approach is a complement to these other algorithms rather than a competitor. Indeed, after our initial conference publication (Blanchard et al., 2011), our methodology was successfully applied in conjunction with the feature transformation proposed by Muandet et al. (2013).

2. Motivating Application: Automatic Gating of Flow Cytometry Data

Flow cytometry is a high-throughput measurement platform that is an important clinical tool for the diagnosis of blood-related pathologies. This technology allows for quantitative analysis of individual cells from a given cell population, derived for example from a blood sample from a patient. We may think of a flow cytometry data set as a set of d-dimensional attribute vectors $(X_j)_{1\leq j\leq n}$, where n is the number of cells analyzed, and d is the number of attributes recorded per cell. These attributes pertain to various physical and chemical properties of the cell. Thus, a flow cytometry data set is a random sample from a patient-specific distribution.

Now suppose a pathologist needs to analyze a new (test) patient with data $(X_j^T)_{1\leq j\leq n_T}$. Before proceeding, the pathologist first needs the data set to be "purified" so that only cells of a certain type are present. For example, lymphocytes are known to be relevant for the diagnosis of leukemia, whereas non-lymphocytes may potentially confound the analysis. In

other words, it is necessary to determine the label $Y_j^T \in \{-1, 1\}$ associated to each cell, where $Y_j^T = 1$ indicates that the j-th cell is of the desired type.

In clinical practice this is accomplished through a manual process known as "gating." The data are visualized through a sequence of two-dimensional scatter plots, where at each stage a line segment or polygon is manually drawn to eliminate a portion of the unwanted cells. Because of the variability in flow cytometry data, this process is difficult to quantify in terms of a small subset of simple rules. Instead, it requires domain-specific knowledge and iterative refinement. Modern clinical laboratories routinely see dozens of cases per day, so it would be desirable to automate this process.

Since clinical laboratories maintain historical databases, we can assume access to a number (N) of historical (training) patients that have already been expert-gated. Because of biological and technical variations in flow cytometry data, the distributions $P_{XY}^{(i)}$ of the historical patients will vary. In order to illustrate the flow cytometry gating problem, we use the NDD dataset from the FlowCap-I challenge.[2] For example, Fig. 1 shows exemplary two-dimensional scatter plots for two different patients – see caption for details. Despite differences in the two distributions, there are also general trends that hold for all patients. Virtually every cell type of interest has a known tendency (e.g., high or low) for most measured attributes. Therefore, it is reasonable to assume that there is an underlying distribution (on distributions) governing flow cytometry data sets, that produces roughly similar distributions thereby making possible the automation of the gating process.

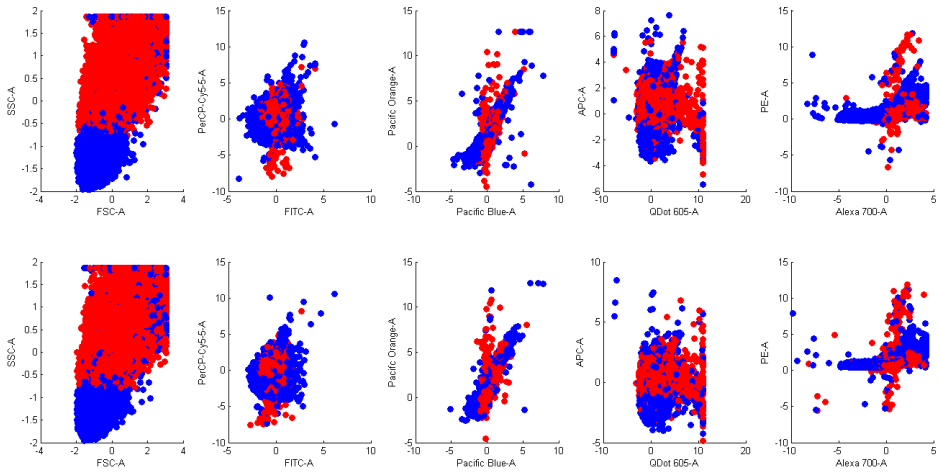

Figure 1: Two-dimensional projections of multi-dimensional flow cytometry data. Each row corresponds to a single patient, and each column to a particular two-dimensional projection. The distribution of cells differs from patient to patient. The colors indicate the results of gating, where a particular type of cell, marked dark (blue), is separated from all other cells, marked bright (red). Labels were manually selected by a domain expert.

2. We will revisit this data set in Section 7.5 where details are given.

3. Formal Setting

Let $\mathcal{X}$ denote the observation space and $\mathcal{Y} \subseteq \mathbb{R}$ the output space. We assume that we observe N samples $S_i = (X_{ij}, Y_{ij})_{1 \leq j \leq n_i}$, $i = 1, \ldots, N$.

Let $\mathfrak{P}_{\mathcal{X} \times \mathcal{Y}}$ denote the set of probability distributions on $\mathcal{X} \times \mathcal{Y}$, $\mathfrak{P}_{\mathcal{X}}$ the set of probability distributions on $\mathcal{X}$ (which we call "marginals"), and $\mathfrak{P}_{\mathcal{Y}|\mathcal{X}}$ the set of conditional probabilities of Y given X (also known as Markov transition kernels from X to Y, which we also call "posteriors"). The disintegration theorem (see for instance Kallenberg (2002), Theorem 6.4) tells us that (under suitable regularity properties, e.g., $\mathcal{X}$ is a Polish space) any element $P_{XY} \in \mathfrak{P}_{\mathcal{X} \times \mathcal{Y}}$ can be written as a product $P_{XY} = P_X \bullet P_{Y|X}$, with $P_X \in \mathfrak{P}_{\mathcal{X}}$, $P_{Y|X} \in \mathfrak{P}_{Y|X}$, that is to say,

$$\mathbb{E}_{(X,Y) \sim P_{XY}} \left[h(X, Y) \right] = \int \left(\int h(x, y) P_{Y|X}(dy | X = x) \right) P_X(dx)$$

for any integrable function $h : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$. The space $\mathfrak{P}_{\mathcal{X} \times \mathcal{Y}}$ is endowed with the topology of weak convergence and the associated Borel σ-algebra.

It is assumed that there exists a distribution μ on $\mathfrak{P}_{\mathcal{X} \times \mathcal{Y}}$, where $P_{XY}^{(1)}, \ldots, P_{XY}^{(N)}$ are i.i.d. realizations from μ, and the sample S_i is made of n_i i.i.d. realizations of (X, Y) following the distribution $P_{XY}^{(i)}$. Now consider a test sample $S^T = (X_j^T, Y_j^T)_{1 \leq j \leq n_T}$, whose labels are not observed by the user. A decision function is a function $f : \mathfrak{P}_{\mathcal{X}} \times \mathcal{X} \mapsto \mathbb{R}$ that predicts $\widehat{y} = f(\widehat{P}_X^T, x)$, where $\widehat{P}_X^T = \frac{1}{T} \sum_{j=1}^T \delta_{X_j^T}$ is the empirical marginal distribution of the test sample and x is any given test point (which can belong to the test sample or not). If $\ell : \mathbb{R} \times \mathcal{Y} \mapsto \mathbb{R}_+$ is a loss, and predictions on the test sample are given by $\widehat{Y}_j^T = f(\widehat{P}_X^T, X_j^T)$, then the empirical average loss incurred on the test sample is $\frac{1}{n_T} \sum_{j=1}^{n_T} \ell(\widehat{Y}_j^T, Y_j^T)$. Based on this, we define the average generalization error of a decision function over test samples of size n_T,

$$\mathcal{E}(f, n_T) := \mathbb{E}_{P_{XY}^T \sim \mu} \mathbb{E}_{S^T \sim (P_{XY}^T)^{\otimes n_T}} \left[\frac{1}{n_T} \sum_{i=1}^{n_T} \ell(f(\widehat{P}_X^T, X_i^T), Y_i^T) \right]. \tag{1}$$

An important point of the analysis is that, at training time as well as at test time, the marginal distribution P_X for a sample is only known through the sample itself, that is, through the empirical marginal $\widehat{P}_X$. As is clear from equation (1), because of this the generalization error also depends on the test sample size n_T. As n_T grows, $\widehat{P}_X^T$ will converge to P_X^T (in the sense of weak convergence). This motivates the following generalization error when we have an infinite test sample, where we then assume that the true marginal P_X^T is observed:

$$\mathcal{E}(f, \infty) := \mathbb{E}_{P_{XY}^T \sim \mu} \mathbb{E}_{(X^T, Y^T) \sim P_{XY}^T} \left[\ell(f(P_X^T, X^T), Y^T) \right]. \tag{2}$$

To gain some insight into this risk, let us decompose μ into two parts, μ_X which generates the marginal distribution P_X, and $\mu_{Y|X}$ which, conditioned on P_X, generates the posterior

$P_{Y|X}$. Denote $\widetilde{X} = (P_X, X)$. We then have

$$\begin{aligned}\mathcal{E}(f,\infty) &= \mathbb{E}_{P_X\sim\mu_X}\mathbb{E}_{P_{Y|X}\sim\mu_{Y|X}}\mathbb{E}_{X\sim P_X}\mathbb{E}_{Y|X\sim P_{Y|X}}\left[\ell(f(\widetilde{X}),Y)\right]\\ &= \mathbb{E}_{P_X\sim\mu_X}\mathbb{E}_{X\sim P_X}\mathbb{E}_{P_{Y|X}\sim\mu_{Y|X}}\mathbb{E}_{Y|X\sim P_{Y|X}}\left[\ell(f(\widetilde{X}),Y)\right]\\ &= \mathbb{E}_{(\widetilde{X},Y)\sim Q^\mu}\left[\ell(f(\widetilde{X}),Y)\right].\end{aligned}$$

Here Q^μ is the distribution that generates $\widetilde{X}$ by first drawing P_X according to μ_X, and then drawing X according to P_X. Similarly, Y is generated, conditioned on $\widetilde{X}$, by first drawing $P_{Y|X}$ according to $\mu_{Y|X}$, and then drawing Y from $P_{Y|X}$. From this last expression, we see that the risk is like a standard supervised learning risk based on $(\widetilde{X}, Y) \sim Q^\mu$. Thus, we can deduce properties that are known to hold for supervised learning risks. For example, in the binary classifications setting, if the loss is the 0/1 loss, then $f^*(\widetilde{X}) = 2\tilde{\eta}(\widetilde{X}) - 1$ is an optimal predictor, where $\tilde{\eta}(\widetilde{X}) = \mathbb{E}_{Y\sim Q^\mu_{Y|\widetilde{X}}}\left[\mathbf{1}_{\{Y=1\}}\right]$. More generally,

$$\mathcal{E}(f,\infty) - \mathcal{E}(f^*,\infty) = \mathbb{E}_{\widetilde{X}\sim Q^\mu_{\widetilde{X}}}\left[\mathbf{1}_{\{\mathrm{sign}(f(\widetilde{X}))\neq\mathrm{sign}(f^*(\widetilde{X}))\}}|2\tilde{\eta}(\widetilde{X})-1|\right].$$

Our goal is a learning rule that asymptotically predicts as well as the global minimizer of (2), for a *general* loss ℓ. By the above observations, consistency with respect to a general ℓ (thought of as a surrogate) will imply consistency for the 0/1 loss, provided ℓ is classification calibrated (Bartlett et al., 2006). Despite the similarity to standard supervised learning in the infinite sample case, we emphasize that the learning task here is different, because the realizations $(\widetilde{X}_{ij}, Y_{ij})$ are neither independent nor identically distributed.

Finally, we note that there is a condition where for μ-almost all test distribution P^T_{XY}, the decision function $f^*(P^T_X, .)$ (where f^* is the global minimizer of (2)) coincides with an optimal Bayes decision function for P^T_{XY} (although *no labels from this test distribution are observed*). This condition is simply that the posterior $P_{Y|X}$ is (μ-almost surely) a function of P_X (in other terms: that with the notation introduced above, $\mu_{Y|X}(P_X)$ is a Dirac measure for μ-almost all P_X). Although we will *not* be assuming this condition throughout the paper, observe that it is implicitly assumed in the motivating application presented in Section 2, where an expert labels the data points by just looking at their marginal distribution.

Lemma 1 *For a fixed distribution P_{XY}, and a decision function $g : \mathcal{X} \to \mathbb{R}$, let us denote $\mathcal{R}(g, P_{XY}) = \mathbb{E}_{(X,Y)\sim P_{XY}}\left[\ell(g(X), Y)\right]$ and*

$$\mathcal{R}^*(P_{XY}) := \min_{g:\mathcal{X}\to\mathbb{R}} \mathcal{R}(g, P_{XY}) = \min_{g:\mathcal{X}\to\mathbb{R}} \mathbb{E}_{(X,Y)\sim P_{XY}}\left[\ell(g(X), Y)\right]$$

the corresponding optimal (Bayes) risk for the loss function ℓ. Assume that μ is a distribution on $\mathfrak{P}_{\mathcal{X}\times\mathcal{Y}}$ such that μ-a.s. it holds $P_{Y|X} = F(P_X)$ for some deterministic mapping F. Let f^* be a minimizer of the risk (2). Then we have for μ-almost all P_{XY}:

$$\mathcal{R}(f^*(P_X, .), P_{XY}) = \mathcal{R}^*(P_{XY})$$

and

$$\mathcal{E}(f^*, \infty) = \mathbb{E}_{P_{XY}\sim\mu}\left[\mathcal{R}^*(P_{XY})\right].$$

Proof For any $f : \mathfrak{P}_{\mathcal{X}} \times \mathcal{X} \to \mathbb{R}$, one has for all P_{XY}: $\mathcal{R}(f(P_X, .), P_{XY}) \geq \mathcal{R}^*(P_{XY})$. For any fixed $P_X \in \mathfrak{P}_{\mathcal{X}}$, consider $P_{XY} := P_X \bullet F(P_X)$ and $g^*(P_X)$ a Bayes decision function for this joint distribution. Pose $f(P_X, x) := g^*(P_X)(x)$. Then f coincides for μ-almost all P_{XY} with a Bayes decision function for P_{XY}, achieving equality in the above inequality. The second equality follows by taking expectation over $P_{XY} \sim \mu$. ■

4. Learning Algorithm

We consider an approach based on kernels. The function $k : \Omega \times \Omega \to \mathbb{R}$ is called a *kernel* on Ω if the matrix $(k(x_i, x_j))_{1 \leq i,j \leq n}$ is symmetric and positive semi-definite for all positive integers n and all $x_1, \ldots, x_n \in \Omega$. It is well-known that if k is a kernel on Ω, then there exists a Hilbert space $\tilde{\mathcal{H}}$ and $\tilde{\Phi} : \Omega \to \tilde{\mathcal{H}}$ such that $k(x, x') = \langle \tilde{\Phi}(x), \tilde{\Phi}(x') \rangle_{\tilde{\mathcal{H}}}$. While $\tilde{\mathcal{H}}$ and $\tilde{\Phi}$ are not uniquely determined by k, the Hilbert space of functions (from Ω to $\mathbb{R}$) $\mathcal{H}_k = \{\langle v, \tilde{\Phi}(\cdot) \rangle_{\tilde{\mathcal{H}}} : v \in \tilde{\mathcal{H}}\}$ is uniquely determined by k, and is called the reproducing kernel Hilbert space (RKHS) of k.

One way to envision $\mathcal{H}_k$ is as follows. Define $\Phi(x) := k(\cdot, x)$, which is called the *canonical feature map* associated with k. Then the span of $\{\Phi(x) : x \in \Omega\}$, endowed with the inner product $\langle \Phi(x), \Phi(x') \rangle = k(x, x')$, is dense in $\mathcal{H}_k$. We also recall the *reproducing property*, which states that $\langle f, \Phi(x) \rangle = f(x)$ for all $f \in \mathcal{H}_k$.

For later use, we introduce the notion of a *universal* kernel. A kernel k on a compact metric space Ω is said to be *universal* when its RKHS is dense in $\mathcal{C}(\Omega)$, the set of continuous functions on Ω, with respect to the supremum norm. Universal kernels are important for establishing universal consistency of many learning algorithms. We refer the reader to Steinwart and Christmann (2008) for additional background on kernels.

Several well-known learning algorithms, such as support vector machines and kernel ridge regression, may be viewed as minimizers of a norm-regularized empirical risk over the RKHS of a kernel. A similar development has also been made for multi-task learning (Evgeniou et al., 2005). Inspired by this framework, we consider a general kernel algorithm as follows.

Consider the loss function $\ell : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}_+$. Let $\overline{k}$ be a kernel on $\mathfrak{P}_X \times \mathcal{X}$, and let $\mathcal{H}_{\overline{k}}$ be the associated RKHS. For the sample S_i, let $\widehat{P}_X^{(i)} = \frac{1}{n_i} \sum_{j=1}^{n_i} \delta_{X_{ij}}$ denote the corresponding empirical X distribution. Also consider the extended input space $\mathfrak{P}_X \times \mathcal{X}$ and the extended data $\widetilde{X}_{ij} = (\widehat{P}_X^{(i)}, X_{ij})$. Note that $\widehat{P}_X^{(i)}$ plays a role analogous to the task index in multi-task learning. Now define

$$\widehat{f}_\lambda = \operatorname*{arg\,min}_{f \in \mathcal{H}_{\overline{k}}} \frac{1}{N} \sum_{i=1}^{N} \frac{1}{n_i} \sum_{j=1}^{n_i} \ell(f(\widetilde{X}_{ij}), Y_{ij}) + \lambda \|f\|^2 . \tag{3}$$

4.1 Specifying the kernels

In the rest of the paper we will consider a kernel $\overline{k}$ on $\mathfrak{P}_{\mathcal{X}} \times \mathcal{X}$ of the product form

$$\overline{k}((P_1, x_1), (P_2, x_2)) = k_P(P_1, P_2) k_X(x_1, x_2), \tag{4}$$

where k_P is a kernel on $\mathfrak{P}_{\mathcal{X}}$ and k_X a kernel on $\mathcal{X}$.

Furthermore, we will consider kernels on $\mathfrak{P}_{\mathcal{X}}$ of a particular form. Let k'_X denote a kernel on $\mathcal{X}$ (which might be different from k_X) that is measurable and bounded. We define the *kernel mean embedding* $\Psi : \mathfrak{P}_{\mathcal{X}} \to \mathcal{H}_{k'_X}$:

$$P_X \mapsto \Psi(P_X) := \int_{\mathcal{X}} k'_X(x, \cdot) dP_X(x). \tag{5}$$

This mapping has been studied in the framework of "characteristic kernels" (Gretton et al., 2007a), and it has been proved that universality of k'_X implies injectivity of Ψ (Gretton et al., 2007b; Sriperumbudur et al., 2010).

Note that the mapping Ψ is linear. Therefore, if we consider the kernel $k_P(P_X, P'_X) = \langle \Psi(P_X), \Psi(P'_X) \rangle$, it is a linear kernel on $\mathfrak{P}_{\mathcal{X}}$ and cannot be a universal kernel. For this reason, we introduce yet another kernel $\mathfrak{K}$ on $\mathcal{H}_{k'_X}$ and consider the kernel on $\mathfrak{P}_{\mathcal{X}}$ given by

$$k_P(P_X, P'_X) = \mathfrak{K}\left(\Psi(P_X), \Psi(P'_X)\right) . \tag{6}$$

Note that particular kernels inspired by the finite dimensional case are of the form

$$\mathfrak{K}(v, v') = F(\left\|v - v'\right\|), \tag{7}$$

or

$$\mathfrak{K}(v, v') = G(\left\langle v, v' \right\rangle), \tag{8}$$

where F, G are real functions of a real variable such that they define a kernel. For example, $F(t) = \exp(-t^2/(2\sigma^2))$ yields a Gaussian-like kernel, while $G(t) = (1 + t)^d$ yields a polynomial-like kernel. Kernels of the above form on the space of probability distributions over a compact space $\mathcal{X}$ have been introduced and studied in Christmann and Steinwart (2010). Below we apply their results to deduce that $\overline{k}$ is a universal kernel for certain choices of k_X, k'_X, and $\mathfrak{K}$.

4.2 Relation to other kernel methods

By choosing $\overline{k}$ differently, one can recover other existing kernel methods. In particular, consider the class of kernels of the same product form as above, but where

$$k_P(P_X, P'_X) = \begin{cases} 1 & P_X = P'_X \\ \tau & P_X \neq P'_X \end{cases}$$

If $\tau = 0$, the algorithm (3) corresponds to training N kernel machines $f(\widehat{P}_X^{(i)}, \cdot)$ using kernel k_X (e.g., support vector machines in the case of the hinge loss) on each training data set, independently of the others (note that this does not offer any generalization ability to a new dataset). If $\tau = 1$, we have a "pooling" strategy that, in the case of equal sample sizes n_i, is equivalent to pooling all training data sets together in a single data set, and running a conventional supervised learning algorithm with kernel k_X (*i.e.*, this corresponds to trying to find a single "one-fits-all" prediction function which does not depend on the marginal). In the intermediate case $0 < \tau < 1$, the resulting kernel is a "multi-task kernel," and the algorithm recovers a multitask learning algorithm like that of Evgeniou et al. (2005). We

compare to the pooling strategy below in our experiments. We also examined the multi-task kernel with $\tau < 1$, but found that, as far as generalization to a new unlabeled task is concerned, it was always outperformed by pooling, and so those results are not reported. This fits the observation that the choice $\tau = 0$ does not provide any generalization to a new task, while $\tau = 1$ at least offers some form of generalization, if only by fitting the same decision function to all datasets.

In the special case where all labels Y_{ij} are the same value for a given task, and k_X is taken to be the constant kernel, the problem we consider reduces to "distributional" classification or regression, which is essentially standard supervised learning where a distribution (observed through a sample) plays the role of the feature vector. Our analysis techniques could easily be specialized to this problem.

5. Learning Theoretic Study

Although the regularized estimation formula (3) defining $\widehat{f}_\lambda$ is standard, the generalization error analysis is not, since the $\widetilde{X}_{ij}$ are neither identically distributed nor independent (Szabo et al., 2015). We begin with a generalization error bound that establishes uniform estimation error control over functions belonging to a ball of $\mathcal{H}_{\overline{k}}$. We then discuss universal kernels, and finally deduce universal consistency of the algorithm.

To simplify somewhat the presentation, we assume below that all training samples have the same size $n_i = n$. Also let $\mathcal{B}_k(r)$ denote the closed ball of radius r, centered at the origin, in the RKHS of the kernel k. We will consider the following assumptions on the loss function and on the kernels:

(L) The loss function $\ell : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}_+$ is L_ℓ-Lipschitz in its first variable and bounded by B_ℓ.

(K-A) The kernels k_X, k'_X and $\mathfrak{K}$ are bounded respectively by constants $B_k^2, B_{k'}^2 \geq 1$, and $B_{\mathfrak{K}}^2$. In addition, the canonical feature map $\Phi_{\mathfrak{K}} : \mathcal{H}_{k'_X} \to \mathcal{H}_{\mathfrak{K}}$ associated to $\mathfrak{K}$ satisfies a Hölder condition of order $\alpha \in (0, 1]$ with constant $L_{\mathfrak{K}}$, on $\mathcal{B}_{k'_X}(B_{k'})$:

$$\forall v, w \in \mathcal{B}_{k'_X}(B_{k'}) : \qquad \|\Phi_{\mathfrak{K}}(v) - \Phi_{\mathfrak{K}}(w)\| \leq L_{\mathfrak{K}} \|v - w\|^\alpha . \tag{9}$$

Sufficient conditions for (9) are described in Section A.2. As an example, the condition is shown to hold with $\alpha = 1$ when $\mathfrak{K}$ is the Gaussian-like kernel on $\mathcal{H}_{k'_X}$. The boundedness assumptions are also clearly satisfied for Gaussian kernels.

Theorem 2 (Uniform estimation error control) *Assume conditions* **(L)** *and* **(K-A)** *hold. If $P_{XY}^{(1)}, \ldots, P_{XY}^{(N)}$ are i.i.d. realizations from μ, and for each $i = 1, \ldots, N$, the sample $S_i = (X_{ij}, Y_{ij})_{1 \leq j \leq n}$ is made of i.i.d. realizations from $P_{XY}^{(i)}$, then for any $R > 0$, with probability at least $1 - \delta$:*

$$\sup_{f \in \mathcal{B}_{\overline{k}}(R)} \left| \frac{1}{N} \sum_{i=1}^{N} \frac{1}{n} \sum_{j=1}^{n} \ell(f(\widetilde{X}_{ij}), Y_{ij}) - \mathcal{E}(f, \infty) \right|$$
$$\leq c \left(R B_k L_\ell \left(B_{k'} L_{\mathfrak{K}} \left(\frac{\log N + \log \delta^{-1}}{n} \right)^{\frac{\alpha}{2}} + B_{\mathfrak{K}} \frac{1}{\sqrt{N}} \right) + B_\ell \sqrt{\frac{\log \delta^{-1}}{N}} \right), \tag{10}$$

where c is a numerical constant.

Proof [sketch] The full proofs of this and other results are given in Section A. We give here a brief overview. We use the decomposition

$$\begin{aligned}
&\sup_{f\in\mathcal{B}_{\overline{k}}(R)}\left|\frac{1}{N}\sum_{i=1}^{N}\frac{1}{n_i}\sum_{j=1}^{n_i}\ell(f(\widetilde{X}_{ij}),Y_{ij})-\mathcal{E}(f,\infty)\right| \\
&\qquad\leq \sup_{f\in\mathcal{B}_{\overline{k}}(R)}\left|\frac{1}{N}\sum_{i=1}^{N}\frac{1}{n_i}\sum_{j=1}^{n_i}\Big(\ell(f(\widehat{P}_X^{(i)},X_{ij}),Y_{ij})-\ell(f(P_X^{(i)},X_{ij}),Y_{ij})\Big)\right| \\
&\qquad\quad+\sup_{f\in\mathcal{B}_{\overline{k}}(R)}\left|\frac{1}{N}\sum_{i=1}^{N}\frac{1}{n_i}\sum_{j=1}^{n_i}\ell(f(P_X^{(i)},X_{ij}),Y_{ij})-\mathcal{E}(f,\infty)\right| \\
&\qquad=:(I)+(II).
\end{aligned}$$

Bounding (I), using the Lipschitz property of the loss function, can be reduced to controlling

$$\left\|f(\widehat{P}_X^{(i)},.)-f(P_X^{(i)},.)\right\|_\infty,$$

conditional to $P_X^{(i)}$, uniformly for $i=1,\ldots,N$. This can be obtained using the reproducing property of the kernel $\overline{k}$, the convergence of $\Psi(\widehat{P}_X^{(i)})$ to $\Psi(P_X^{(i)})$ as a consequence of Hoeffding's inequality in a Hilbert space, and the other assumptions (boundedness/Hölder property) on the kernels.

Concerning the control of the term (II), it can be decomposed in turn into the convergence conditional to $(P_X^{(i)})$, and the convergence of the conditional generalization error. In both cases, a standard approach using the Azuma-McDiarmid inequality (McDiarmid, 1989) followed by symmetrization and Rademacher complexity analysis on a kernel space (Koltchinskii, 2001; Bartlett and Mendelson, 2002) can be applied. For the first part, the random variables are the (X_{ij},Y_{ij}) (which are independent conditional to $(P_X^{(i)})$); for the second part, the i.i.d. variables are the $(P_X^{(i)})$ (the (X_{ij},Y_{ij}) being integrated out).

■

Next, we turn our attention to universal kernels (see Section 4 for the definition). A relevant notion for our purposes is that of a normalized kernel. If k is a kernel on Ω, then

$$k^*(x,x'):=\frac{k(x,x')}{\sqrt{k(x,x)k(x',x')}}$$

is the associated *normalized* kernel. If a kernel is universal, then so is its associated normalized kernel. For example, the exponential kernel $k(x,x')=\exp(\kappa\langle x,x'\rangle_{\mathbb{R}^d})$, $\kappa>0$, can be shown to be universal on $\mathbb{R}^d$ through a Taylor series argument. Consequently, the Gaussian kernel

$$k_\sigma(x,x'):=\frac{\exp(\frac{1}{\sigma^2}\langle x,x'\rangle)}{\exp(\frac{1}{2\sigma^2}\|x\|^2)\exp(\frac{1}{2\sigma^2}\|x'\|^2)}$$

is universal, being the normalized kernel associated with the exponential kernel with $\kappa = 1/\sigma^2$. See Steinwart and Christmann (2008) for additional details and discussion.

To establish that $\overline{k}$ is universal on $\mathfrak{P}_{\mathcal{X}} \times \mathcal{X}$, the following lemma is useful.

Lemma 3 *Let Ω, Ω' be two compact spaces and k, k' be kernels on Ω, Ω', respectively. Then if k, k' are both universal, the product kernel*

$$\overline{k}((x, x'), (y, y')) := k(x, y)k'(x', y')$$

is universal on $\Omega \times \Omega'$.

Several examples of universal kernels are known on Euclidean space. For our purposes, we also need universal kernels on $\mathfrak{P}_{\mathcal{X}}$. Fortunately, this was studied by Christmann and Steinwart (2010). Some additional assumptions on the kernels and feature space are required:

(K-B) k_X, k'_X, $\mathfrak{K}$, and $\mathcal{X}$ satisfy the following:

- $\mathcal{X}$ is a compact metric space
- k_X is universal on $\mathcal{X}$
- k'_X is continuous and universal on $\mathcal{X}$
- $\mathfrak{K}$ is universal on any compact subset of $\mathcal{H}_{k'_X}$.

Adapting the results of (Christmann and Steinwart, 2010), we have the following.

Theorem 4 (Universal kernel) *Assume condition* **(K-B)** *holds. Then, for k_P defined as in* (6)*, the product kernel $\overline{k}$ in* (4) *is universal on $\mathfrak{P}_{\mathcal{X}} \times \mathcal{X}$.*

Furthermore, the assumption on $\mathfrak{K}$ is fulfilled if $\mathfrak{K}$ is of the form (8)*, where G is an analytical function with positive Taylor series coefficients, or if $\mathfrak{K}$ is the normalized kernel associated to such a kernel.*

Proof By Lemma 3, it suffices to show $\mathfrak{P}_{\mathcal{X}}$ is a compact metric space, and that $k_P(P_X, P'_X)$ is universal on $\mathfrak{P}_{\mathcal{X}}$. The former statement follows from Theorem 6.4 of Parthasarathy (1967), where the metric is the Prohorov metric. We will deduce the latter statement from Theorem 2.2 of Christmann and Steinwart (2010). The statement of Theorem 2.2 there is apparently restricted to kernels of the form (8), but the proof actually only uses that the kernel $\mathfrak{K}$ is universal on any compact set of $\mathcal{H}_{k'_X}$. To apply Theorem 2.2, it remains to show that $\mathcal{H}_{k'_X}$ is a separable Hilbert space, and that Ψ is injective and continuous. Injectivity of Ψ is equivalent to k'_X being a characteristic kernel, which follows from the assumed universality of k'_X (Sriperumbudur et al., 2010). The continuity of k'_X implies separability of $\mathcal{H}_{k'_X}$ (Steinwart and Christmann (2008), Lemma 4.33) as well as continuity of Ψ (Christmann and Steinwart (2010), Lemma 2.3 and preceding discussion). Now Theorem 2.2 of (Christmann and Steinwart, 2010) may be applied, and the results follows.

The fact that kernels of the form (8), where G is analytic with positive Taylor coefficients, are universal on any compact set of $\mathcal{H}_{k'_X}$ was established in the proof of Theorem 2.2 of the same work (Christmann and Steinwart, 2010). ■

As an example, suppose that $\mathcal{X}$ is a compact subset of $\mathbb{R}^d$. Let k_X and k'_X be Gaussian kernels on $\mathcal{X}$. Taking $G(t) = \exp(t)$, it follows that $\mathfrak{K}(P_X, P'_X) = \exp(\langle \Psi(P_X), \Psi(P'_X) \rangle_{\mathcal{H}_{k'_X}})$ is universal on $\mathfrak{P}_\mathcal{X}$. By similar reasoning as in the finite dimensional case, the Gaussian-like kernel $\mathfrak{K}(P_X, P'_X) = \exp(-\frac{1}{2\sigma^2} \|\Psi(P_X) - \Psi(P'_X)\|^2_{\mathcal{H}_{k'_X}})$ is also universal on $\mathfrak{P}_\mathcal{X}$. Thus the product kernel is universal on $\mathfrak{P}_\mathcal{X} \times \mathcal{X}$.

From Theorems 2 and 4, we may deduce universal consistency of the learning algorithm. Furthermore, we can weaken the assumption on the loss relative to Theorem 2. In particular, universal consistency does not require that the loss be bounded, and therefore holds for unbounded losses such as the hinge and logistic losses.

Corollary 5 (Universal consistency) *Let ℓ be Lipschitz in its first variabe such that*

$$\sup_{y \in \mathcal{Y}} \ell(0, y) < \infty. \tag{11}$$

Further assume that conditions **(K-A)** *and* **(K-B)** *are satisfied. Assume that as* $\min(N, n) \to \infty$, $N = \mathcal{O}(n^\gamma)$ *for some* $\gamma > 0$, *and* $\frac{N}{\log n} \to \infty$. *Also let* $\lambda = \lambda(N, n)$ *be a sequence such that as* $\min(N, n) \to \infty$, $\lambda(N, n) \to 0$ *and*

$$\lambda \min\left(\frac{N}{\log n}, \left(\frac{n}{\log n}\right)^\alpha\right) \to \infty.$$

Then

$$\mathcal{E}(\widehat{f}_{\lambda(N,n)}, \infty) \to \inf_{f : \mathfrak{P}_\mathcal{X} \times \mathcal{X} \to \mathbb{R}} \mathcal{E}(f, \infty)$$

almost surely.

The proof of the corollary relies on the bound established in Theorem 2, the universality of $\overline{k}$ established in Theorem 4, and otherwise relatively standard arguments. The assumption (11) always holds for classification, and it holds for regression, for example, when $\mathcal{Y}$ is compact and $\ell(0, y)$ is continuous as a function of y.

6. Implementation

Implementation of the algorithm in (3) relies on techniques that are similar to those used for other kernel methods, but with some variations. The first subsection illustrates how, for the case of hinge loss, the optimization problem corresponds to a certain cost-sensitive support vector machine. Subsequent subsections focus on more scalable implementations based on approximate feature mappings.

6.1 Representer theorem and hinge loss

For a particular loss ℓ, existing algorithms for optimizing an empirical risk based on that loss can be adapted to the marginal transfer setting. We now illustrate this idea for the case of the hinge loss, $\ell(t, y) = \max(0, 1 - yt)$. To make the presentation more concise, we will employ the extended feature representation $\widetilde{X}_{ij} = (\widehat{P}_X^{(i)}, X_{ij})$, and we will also "vectorize"

the indices (i, j) so as to employ a single index on these variables and on the labels. Thus the training data are $(\widetilde{X}_i, Y_i)_{1 \le i \le M}$, where $M = \sum_{i=1}^{N} n_i$, and we seek a solution to

$$\min_{f \in \mathcal{H}_{\overline{k}}} \sum_{i=1}^{M} c_i \max(0, 1 - Y_i f(\widetilde{X}_i)) + \frac{1}{2} \|f\|^2 .$$

Here $c_i = \frac{1}{\lambda N n_m}$, where m is the smallest positive integer such that $i \le n_1 + \cdots + n_m$. By the representer theorem (Steinwart and Christmann, 2008), the solution of (3) has the form

$$\widehat{f}_\lambda = \sum_{i=1}^{M} r_i \overline{k}(\widetilde{X}_i, \cdot)$$

for real numbers r_i. Plugging this expression into the objective function of (3), and introducing the auxiliary variables ξ_i, we have the quadratic program

$$\begin{aligned} \min_{r,\xi} \ & \frac{1}{2} r^T \overline{K} r + \sum_{i=1}^{M} c_i \xi_i \\ \text{s.t. } & Y_i \sum_{j=1}^{M} r_j \overline{k}(\widetilde{X}_i, \widetilde{X}_j) \ge 1 - \xi_i, \ \forall i \\ & \xi_i \ge 0, \ \forall i, \end{aligned}$$

where $\overline{K} := (\overline{k}(\widetilde{X}_i, \widetilde{X}_j))_{1 \le i,j \le M}$. Using Lagrange multiplier theory, the dual quadratic program is

$$\begin{aligned} \max_{\alpha} \ & -\frac{1}{2} \sum_{i,j=1}^{M} \alpha_i \alpha_j Y_i Y_j \overline{k}(\widetilde{X}_i, \widetilde{X}_j) + \sum_{i=1}^{M} \alpha_i \\ \text{s.t. } & 0 \le \alpha_i \le c_i \ \forall i, \end{aligned}$$

and the optimal function is

$$\widehat{f}_\lambda = \sum_{i=1}^{M} \alpha_i Y_i \overline{k}(\widetilde{X}_i, \cdot).$$

This is equivalent to the dual of a cost-sensitive support vector machine, without offset, where the costs are given by c_i. Therefore we can learn the weights α_i using any existing software package for SVMs that accepts example-dependent costs and a user-specified kernel matrix, and allows for no offset. Returning to the original notation, the final predictor given a test X-sample S^T has the form

$$\widehat{f}_\lambda(\widehat{P}_X^T, x) = \sum_{i=1}^{N} \sum_{j=1}^{n_i} \alpha_{ij} Y_{ij} \overline{k}((\widehat{P}_X^{(i)}, X_{ij}), (\widehat{P}_X^T, x))$$

where the α_{ij} are nonnegative. Like the SVM, the solution is often sparse, meaning most α_{ij} are zero.

Finally, we remark on the computation of $k_P(\widehat{P}_X, \widehat{P}'_X)$. When $\mathfrak{K}$ has the form of (7) or (8), the calculation of k_P may be reduced to computations of the form $\left\langle \Psi(\widehat{P}_X), \Psi(\widehat{P}'_X) \right\rangle$. If $\widehat{P}_X$ and $\widehat{P}'_X$ are empirical distributions based on the samples $X_1, \ldots, X_n$ and $X'_1, \ldots, X'_{n'}$, then

$$\begin{aligned}\left\langle \Psi(\widehat{P}_X), \Psi(\widehat{P}'_X) \right\rangle &= \left\langle \frac{1}{n}\sum_{i=1}^{n} k'_X(X_i, \cdot), \frac{1}{n'}\sum_{j=1}^{n'} k'_X(X'_j, \cdot) \right\rangle \\ &= \frac{1}{nn'}\sum_{i=1}^{n}\sum_{j=1}^{n'} k'_X(X_i, X'_j).\end{aligned}$$

Note that when k'_X is a (normalized) Gaussian kernel, $\Psi(\widehat{P}_X)$ coincides (as a function) with a smoothing kernel density estimate for P_X.

6.2 Approximate Feature Mapping for Scalable Implementation

Assuming $n_i = n$, for all i, the computational complexity of a nonlinear SVM solver is between $O(N^2n^2)$ and $O(N^3n^3)$ (Joachims, 1999; Chang and Lin, 2011). Thus, standard nonlinear SVM solvers may be insufficient when either or both N and n are very large.

One approach to scaling up kernel methods is to employ approximate feature mappings together with linear solvers. This is based on the idea that kernel methods are solving for a linear predictor after first nonlinearly transforming the data. Since this nonlinear transformation can have an extremely high- or even infinite-dimensional output, classical kernel methods avoid computing it explicitly. However, if the feature mapping can be approximated by a finite dimensional transformation with a relatively low-dimensional output, one can directly solve for the linear predictor, which can be accomplished in $O(Nn)$ time (Hsieh et al., 2008).

In particular, given a kernel $\overline{k}$, the goal is to find an approximate feature mapping $\overline{z}(\tilde{x})$ such that $\overline{k}(\tilde{x}, \tilde{x}') \approx \overline{z}(\tilde{x})^T\overline{z}(\tilde{x}')$. Given such a mapping $\overline{z}$, one then applies an efficient linear solver, such as Liblinear (Fan et al., 2008), to the training data $(\overline{z}(\tilde{X}_{ij}), Y_{ij})_{ij}$ to obtain a weight vector w. The final prediction on a test point $\tilde{x}$ is then $w^T\overline{z}(\tilde{x})$. As described in the previous subsection, the linear solver may need to be tweaked, as in the case of unequal sample sizes n_i, but this is usually straightforward.

Recently, such low-dimensional approximate future mappings $z(x)$ have been developed for several kernels. We examine two such techniques in the context of marginal transfer learning, the Nyström approximation (Williams and Seeger, 2001; Drineas and Mahoney, 2005) and random Fourier features. The Nyström approximation applies to any kernel method, and therefore extends to the marginal transfer setting without additional work. On the other hand, we give a novel extension of random Fourier features to the marginal transfer learning setting (for the case of all Gaussian kernels), together with performance analysis.

6.2.1 Random Fourier Features

The approximation of Rahimi and Recht is based on Bochner's theorem, which characterizes shift invariant kernels (Rahimi and Recht, 2007).

Theorem 6 *A continuous kernel $k(x,y) = k(x-y)$ on $\mathbb{R}^d$ is positive definite iff $k(x-y)$ is the Fourier transform of a finite positive measure $p(w)$, i.e.,*

$$k(x-y) = \int_{\mathbb{R}^d} p(w) e^{jw^T(x-y)} dw \,. \tag{12}$$

If a shift invariant kernel $k(x-y)$ is properly scaled then Theorem 6 guarantees that $p(w)$ in (12) is a proper probability distribution.

Random Fourier features (RFFs) approximate the integral in (12) using samples drawn from $p(w)$. If $w_1, w_2, ..., w_L$ are i.i.d. draws from $p(w)$,

$$\begin{aligned}
k(x-y) &= \int_{\mathbb{R}^d} p(w) e^{jw^T(x-y)} dw \\
&= \int_{\mathbb{R}^d} p(w) \cos(w^T x - w^T y) dw \\
&\approx \frac{1}{L} \sum_{i=1}^{L} \cos(w_i^T x - w_i^T y) \\
&= \frac{1}{L} \sum_{i=1}^{L} \cos(w_i^T x)\cos(w_i^T y) + \sin(w_i^T x)\sin(w_i^T y) \\
&= \frac{1}{L} \sum_{i=1}^{L} [\cos(w_i^T x), \sin(w_i^T x)]^T [\cos(w_i^T y), \sin(w_i^T y)] \\
&= z_w(x)^T z_w(y) \,,
\end{aligned} \tag{13}$$

where $z_w(x) = \frac{1}{\sqrt{L}}[\cos(w_1^T x), \sin(w_1^T x), ..., \cos(w_L^T x), \sin(w_L^T x)] \in \mathbb{R}^{2L}$ is an approximate nonlinear feature mapping of dimensionality $2L$. In the following, we extend the RFF methodology to the kernel $\bar{k}$ on the extended feature space $\mathfrak{P}_{\mathcal{X}} \times \mathcal{X}$. Let $X_1, \ldots, X_{n_1}$ and $X'_1, \ldots, X'_{n_2}$ be i.i.d. realizations of P_X and P'_X respectively, and let $\widehat{P}_X$ and $\widehat{P}'_X$ denote the corresponding empirical distributions. Given $x, x' \in \mathcal{X}$, denote $\tilde{x} = (\widehat{P}_X, x)$ and $\tilde{x}' = (\widehat{P}'_X, x')$. The goal is to find an approximate feature mapping $\bar{z}(\tilde{x})$ such that $\bar{k}(\tilde{x}, \tilde{x}') \approx \bar{z}(\tilde{x})^T \bar{z}(\tilde{x}')$. Recall that

$$\bar{k}(\tilde{x}, \tilde{x}') = k_P(\widehat{P}_X, \widehat{P}'_X) k_X(x, x');$$

specifically, we consider k_X and k'_X to be Gaussian kernels and the kernel on distributions k_P to have the Gaussian-like form

$$k_P(\widehat{P}_X, \widehat{P}'_X) = \exp\left\{ \frac{1}{2\sigma_P^2} \|\Psi(\widehat{P}_X) - \Psi(\widehat{P}'_X)\|^2_{H_{k'_X}} \right\}.$$

As noted earlier in this section, the calculation of $k_P(\widehat{P}_X, \widehat{P}'_X)$ reduces to the computation of

$$\langle \Psi(\widehat{P}_X), \Psi(\widehat{P}'_X) \rangle = \frac{1}{n_1 n_2} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} k'_X(X_i, X'_j). \tag{14}$$

We use Theorem 6 to approximate k'_X and thus $\langle \Psi(\widehat{P}_X), \Psi(\widehat{P}'_X) \rangle$. Let $w_1, w_2, ..., w_L$ be i.i.d. draws from $p'(w)$, the inverse Fourier transform of k'_X. Then we have:

$$\begin{aligned}
\langle \Psi(\widehat{P}_X), \Psi(\widehat{P}'_X) \rangle &= \frac{1}{n_1 n_2} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} k'_X(X_i, X'_j) \\
&\approx \frac{1}{L n_1 n_2} \sum_{l=1}^{L} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \cos(w_l^T X_i - w_l^T X'_j) \\
&= \frac{1}{L n_1 n_2} \sum_{l=1}^{L} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} [\cos(w_l^T X_i)\cos(w_l^T X'_j) + \sin(w_l^T X_i)\sin(w_l^T X'_j)] \\
&= \frac{1}{L n_1 n_2} \sum_{l=1}^{L} \{ \sum_{i=1}^{n_1} [\cos(w_l^T X_i), \sin(w_l^T X_i)]^T \sum_{j=1}^{n_2} [\cos(w_l^T X'_j), \sin(w_l^T X'_j)] \} \\
&= Z_P(\widehat{P}_X)^T Z_P(\widehat{P}'_X),
\end{aligned}$$

where

$$Z_P(\widehat{P}_X) = \frac{1}{n_1 \sqrt{L}} \sum_{i=1}^{n_1} \Big[\cos(w_1^T X_i), \sin(w_1^T X_i), ..., \cos(w_L^T X_i), \sin(w_L^T X_i) \Big], \tag{15}$$

and $Z_P(\widehat{P}'_X)$ is defined analogously with n_1 replaced by n_2. For the proof of Theorem 7, let z'_X denote the approximate feature map corresponding to k'_X, which satisfies $Z_P(\widehat{P}_X) = \frac{1}{n_1} \sum_{i=1}^{n_1} z'_X(X_i)$.

Note that the lengths of the vectors $Z_P(\widehat{P}_X)$ and $Z_P(\widehat{P}'_X)$ are $2L$. To approximate $\bar{k}$ we may write

$$\begin{aligned}
\bar{k}(\tilde{x}, \tilde{x}') &\approx \exp \frac{-\|Z_P(\widehat{P}_X) - Z_P(\widehat{P}'_X)\|^2_{\mathbb{R}^{2L}}}{2\sigma_P^2} \cdot \exp \frac{-\|x - x'\|^2_{\mathbb{R}^d}}{2\sigma_X^2} \\
&= \exp \frac{-(\sigma_X^2 \|Z_P(\widehat{P}_X) - Z_P(\widehat{P}'_X)\|^2_{\mathbb{R}^{2L}} + \sigma_P^2 \|x - x'\|^2_{\mathbb{R}^d})}{2\sigma_P^2 \sigma_X^2} \\
&= \exp \frac{-(\|\sigma_X Z_P(\widehat{P}_X) - \sigma_X Z_P(\widehat{P}'_X)\|^2_{\mathbb{R}^{2L}} + \|\sigma_P x - \sigma_P x'\|^2_{\mathbb{R}^d})}{2\sigma_P^2 \sigma_X^2} \\
&= \exp \frac{-\|(\sigma_X Z_P(\widehat{P}_X), \sigma_P x) - (\sigma_X Z_P(\widehat{P}'_X), \sigma_P x')\|^2_{\mathbb{R}^{2L+d}}}{2\sigma_P^2 \sigma_X^2}
\end{aligned} \tag{16}$$

This is also a Gaussian kernel, now on $\mathbb{R}^{2L+d}$. Again by applying Theorem 6, we have

$$\bar{k}(\widehat{P}_X, X), (\widehat{P}'_X, X')) \approx \int_{\mathbb{R}^{2L+d}} p(v) e^{jv^T((\sigma_X Z_P(P_X), \sigma_P X) - (\sigma_X Z_P(P'_X), \sigma_P X'))} dv.$$

Let $v_1, v_2, ..., v_q$ be drawn i.i.d. from $p(v)$, the inverse Fourier transform of the Gaussian kernel with bandwidth $\sigma_P \sigma_X$. Let $u = (\sigma_X Z_P(\widehat{P}_X), \sigma_P x)$ and $u' = (\sigma_X Z_P(\widehat{P}'_X), \sigma_P x')$.

Then

$$\begin{aligned}\bar{k}(\tilde{x},\tilde{x}') &\approx \frac{1}{Q}\sum_{q=1}^{Q}\cos(v_q^T(u-u')) \\ &= \bar{z}(\tilde{x})^T\bar{z}(\tilde{x}'),\end{aligned}$$

where

$$\bar{z}(\tilde{x}) = \frac{1}{\sqrt{Q}}[\cos(v_1^T u),\sin(v_1^T u),...,\cos(v_Q^T u),\sin(v_Q^T u)] \in \mathbb{R}^{2Q} \tag{17}$$

and $\bar{z}(\tilde{x}')$ is defined similarly.

This completes the construction of the approximate feature map. The following result, which uses Hoeffding's inequality and generalizes a result of Rahimi and Recht (2007), says that the approximation achieves any desired approximation error with very high probability as $L, Q \to \infty$.

Theorem 7 *Let L be the number of random features to approximate the kernel on distributions and Q be the number of features to approximate the final product kernel. For any $\epsilon_l > 0$, $\epsilon_q > 0$, $\tilde{x} = (\widehat{P}_X, x)$, $\tilde{x}' = (\widehat{P}'_X, x')$,*

$$P(|\bar{k}(\tilde{x},\tilde{x}') - \bar{z}(\tilde{x})^T\bar{z}(\tilde{x}')| \geq \epsilon_l + \epsilon_q) \leq 2\exp\Big(-\frac{Q\epsilon_q^2}{2}\Big) + 6n_1n_2\exp\Big(-\frac{L\epsilon^2}{2}\Big), \tag{18}$$

where $\epsilon = \frac{\sigma_P^2}{2}\log(1+\epsilon_l)$, σ_P is the bandwidth parameter of the Gaussian-like kernel k_P, and n_1 and n_2 are the sizes of the empirical distributions $\widehat{P}_X$ and $\widehat{P}'_X$, respectively.

The above results holds for fixed $\tilde{x}$ and $\tilde{x}'$. Following again Rahimi and Recht (2007), one can use an ϵ-net argument to prove a stronger statement for every pair of points in the input space simultaneously. They show

Lemma 8 *Let $\mathcal{M}$ be a compact subset of $\mathbb{R}^d$ with diameter $r = \mathrm{diam}(\mathcal{M})$ and let D be the number of random Fourier features used. Then for the mapping defined in (13), we have*

$$P\Big(\sup_{x,y\in\mathcal{M}}|z_w(x)^T z_w(y) - k(x-y)| \geq \epsilon\Big) \leq 2^8\Big(\frac{\sigma r}{\epsilon}\Big)^2\exp\Big(\frac{-D\epsilon^2}{2(d+2)}\Big),$$

where $\sigma = \mathbb{E}[w^T w]$ is the second moment of the Fourier transform of k.

Our RFF approximation of $\bar{k}$ is grounded on Gaussian RFF approximations on Euclidean spaces, and thus, the following result holds by invoking Lemma 8, and otherwise following the argument of Theorem 7.

Theorem 9 *Using the same notations as in Theorem 7 and Lemma 8,*

$$\begin{aligned}&P\Big(\sup_{x,x'\in\mathcal{M}}|\bar{k}(\tilde{x},\tilde{x}') - \bar{z}(\tilde{x})^T\bar{z}(\tilde{x}')| \geq \epsilon_l + \epsilon_q\Big) \\ &\qquad\leq 2^8\Big(\frac{\sigma'_X r}{\epsilon_q}\Big)^2\exp\Big(\frac{-Q\epsilon_q^2}{2(d+2)}\Big) + 2^9 3n_1n_2\Big(\frac{\sigma_P\sigma_X r}{\epsilon_l}\Big)^2\exp\Big(\frac{-L\epsilon_l^2}{2(d+2)}\Big)\end{aligned} \tag{19}$$

where σ'_X is the width of kernel k'_X in Eqn. (14) and σ_P and σ_X are the widths of kernels k_P and k_X respectively.

6.2.2 Nyström Approximation

Like random Fourier features, the Nyström approximation is a technique to approximate kernel matrices. Unlike random Fourier features, for the Nyström approximation, the feature maps are data-dependent. Also, in the last subsection, all kernels were assumed to be shift invariant. With the Nyström approximation there is no such assumption.

For a general kernel k, the goal is to find a feature mapping $z : \mathbb{R}^d \to \mathbb{R}^L$, where $L > d$, such that $k(x, x') \approx z(x)^T z(x')$. Let r be the target rank of the final approximated kernel matrix, and m be the number of selected columns of the original kernel matrix. In general $r \leq m \ll n$.

Given data points $x_1, \ldots, x_n$, the Nyström method approximates the kernel matrix by first sampling m data points $x'_1, x'_2, ..., x'_m$ without replacement from the original sample, and then constructing a low rank matrix by $\widehat{K}_r = K_b \widehat{K}^{-1} K_b^T$, where $K_b = [k(x_i, x'_j)]_{n\times m}$, and $\widehat{K} = [k(x'_i, x'_j)]_{m\times m}$. Hence, the final approximate feature mapping is

$$z_n(x) = \widehat{D}^{-\frac{1}{2}} \widehat{V}^T [k(x, x'_1), ..., k(x, x'_m)], \tag{20}$$

where $\widehat{D}$ is the eigenvalue matrix of $\widehat{K}$ and $\widehat{V}$ is the corresponding eigenvector matrix.

The Nyström approximation holds for any positive definite kernel, but random Fourier features can be used only for shift invariant kernels. On the other hand, random Fourier features are very easy to implement and have a lower time complexity than the Nyström method (where one has to find the eigenvalue decomposition). Moreover, the Nyström method is useful only when the kernel matrix is low rank. In our experiments, we use random Fourier features when all kernels are Gaussian and the Nyström method otherwise.

7. Experiments

This section empirically compares our marginal transfer learning method with pooling. One implementation of the pooling algorithm was mentioned in Section 4.2, where k_P is taken to be a constant kernel. Another implementation is to put all the training data sets together and train a single conventional kernel method. The only difference between the two implementations is that in the former, weights of $1/n_i$ are used for examples from training task i. In almost all of our experiments below, the various training tasks have the same sample sizes, in which case the two implementations coincide. The only exception is the fourth experiment when we use all training data, in which case we use the second of the two implementations mentioned above.

We consider three classification problems ($\mathcal{Y} = \{-1, 1\}$), for which the hinge loss is employed, and one regression problem ($\mathcal{Y} \subset \mathbb{R}$), where the ϵ-insensitive loss is employed. Thus, the algorithms implemented are natural extensions of support vector classification and regression to marginal transfer learning.

7.1 Model Selection

The various experiments use different combinations of kernels. In all experiments, linear kernels $k(x_1, x_2) = {x_1}^T x_2$ and Gaussian kernels $k_\sigma(x_1, x_2) = \exp\big(-\frac{||x_1 - x_2||^2}{2\sigma^2}\big)$ were used.

The bandwidth σ of each Gaussian kernel and the regularization parameter λ of the machines were selected by grid search. For model selection, five-fold cross-validation has

been used. In order to stabilize the cross-validation procedure, it was repeated 5 times over independent random splits into folds (Kohavi et al., 1995). Thus, candidate parameter values were evaluated on the 5×5 validation sets and the configuration yielding the best average performance was selected. If any of the chosen hyper-parameters was at the grid boundary, the grid was extended accordingly, i.e., the same grid size has been used, however, the center of grid has been assigned to the previously selected point. The grid used for kernels was $\sigma \in \left(10^{-2}, 10^{4}\right)$ with logarithmic spacing, and the grid used for the regularization parameter was $\lambda \in \left(10^{-1}, 10^{1}\right)$ with logarithmic spacing.

7.2 Synthetic Data Experiment

To illustrate the proposed method, a synthetic problem was constructed. The synthetic data generation algorithm is given in Algorithm 1. In brief, for each classification task, the data are uniformly supported on an ellipse, with the major axis determining the labels, and the rotation of the major axis randomly generated in a 90 degree range for each task. One random realization of this synthetic data is shown in Figure 2. This synthetic dataset perfectly satisfies the assumptions of marginal transfer learning, because the Bayes classifier for a task is uniquely determined by the marginal distribution of the features, i.e. Lemma 1 applies (and the optimal error $\mathcal{E}(f^*, \infty)$ is zero). On the other hand, observe that the expectation of each X distribution is the same regardless of the task and thus does not provide any relevant information, so that taking into account at least second order information is needed to achieve marginal transfer.

To analyse the effects of number of examples per task (n) and number of tasks (N), we constructed 12 synthetic data sets by taking combinations $N \times n$ where $N \in \{16, 64, 256\}$ and $n \in \{8, 16, 32, 256\}$. For each synthetic dataset, the test set contains 10 tasks and each task contains one million data points. All kernels are taken to be Gaussian, and the random Fourier features speedup is used. The results are shown in Table 3. The marginal transfer learning method significantly outperforms the baseline pooling method. Furthermore, the performance of the transfer learning approach improves as N and n increase, as expected. The pooling method, however, does no better than random guessing regardless of N and n.

In the remaining experiments, the marginal distribution does not perfectly characterize the optimal decision function, but still provides some information to offer improvements over pooling.

7.3 Parkinson's disease telemonitoring dataset

We test our method in the regression setting using the Parkinson's disease telemonitoring dataset, which is composed of a range of biomedical voice measurements using a telemonitoring device from 42 people with early-stage Parkinson's. The recordings were automatically captured in the patients' homes. The aim is to predict the clinician's Parkinson's disease symptom score for each recording on the unified Parkinson's disease rating scale (UPDRS) (Tsanas et al., 2010). Thus we are in a regression setting, and employ the ϵ-insensitive loss from support vector regression. All kernels are taken to be Gaussian, and the random Fourier features speedup is used.

There are around 200 recordings per patient. We randomly select 7 test users and then vary the number of training users N from 10 to 35 in steps of 5, and we also vary the number

Algorithm 1: Synthetic Data Generation

input : N: Number of tasks, n: Number of training examples per task
output: Realization of synthetic dataset for N Tasks
for $i = 1$ *to* N **do**

- sample rotation α_i uniformly in $\left[\frac{\pi}{4}, \frac{3\pi}{4}\right]$;
- Take an ellipse whose major axis is aligned with the horizontal axis, and rotate it by an angle of α_i about its center;
- Sample n points X_{ij}, $j = 1, \ldots, n$ uniformly at random from the rotated ellipse;
- Label the points according to their position with respect to the major axis i.e. the points that are on the left of the major axis are considered as class 1 and the points on the right of the major axis are considered as class -1.

end

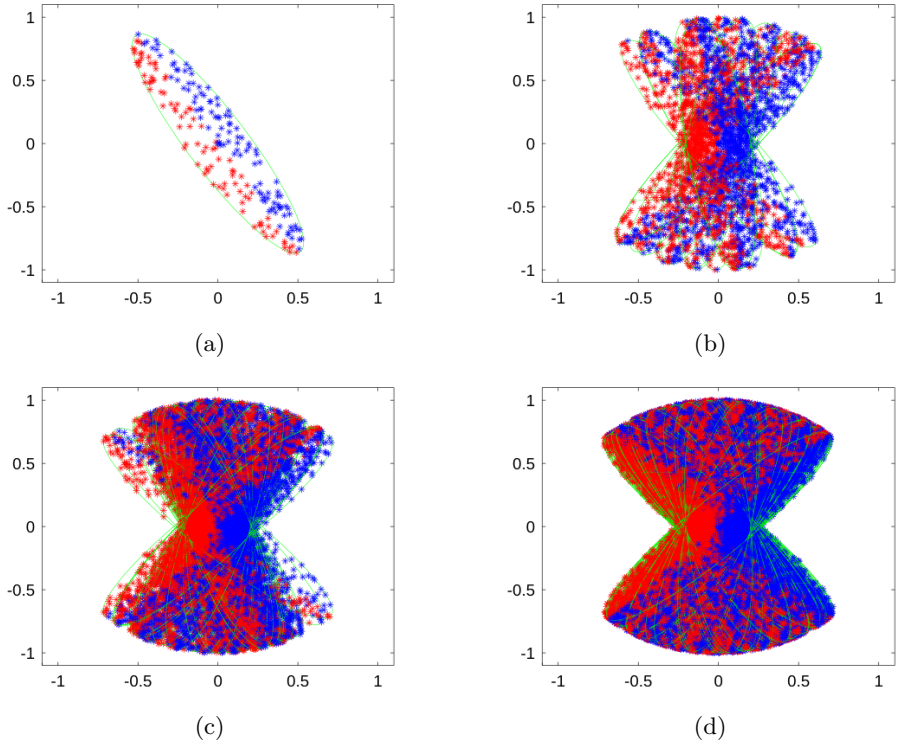

Figure 2: Plots of synthetic datasets (red and blue points represent negative and positive classes) for different settings: (a) Random realization of a single task with 256 training examples per task. Plots (b), (c) and(d) are random realizations of synthetic data with 256 training examples for 16, 64 and 256 tasks.

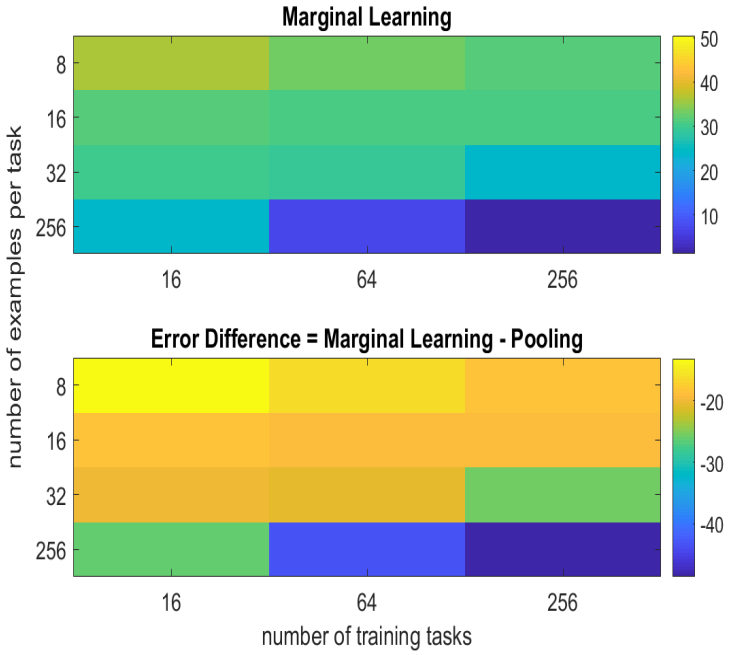

Figure 3: Synthetic dataset: Classification error rates for proposed method and difference with baseline for different experimental settings, i.e., number of examples per task and number of tasks.

of training examples n per user from 20 to 100. We repeat this process several times to get the average errors which are shown in Fig 4 and Tables 3 and 4 (found in the appendix). The marginal transfer learning method clearly outperforms pooling, especially as N and n increase.

7.4 Satellite Classification

Microsatellites are increasingly deployed in space missions for a variety of scientific and technological purposes. Because of randomness in the launch process, the orbit of a microsatellite is random, and must be determined after the launch. One recently proposed approach is to estimate the orbit of a satellite based on radiofrequency (RF) signals as measured in a ground sensor network. However, microsatellites are often launched in bunches, and for this approach to be successful, it is necessary to associate each RF measurement vector with a particular satellite. Furthermore, the ground antennae are not able to decode unique identifier signals transmitted by the microsatellites, because (a) of constraints on the satellite/ground antennae links, including transmission power, atmospheric attenuation, scattering, and thermal noise, and (b) ground antennae must have low gain and low directional specificity owing to uncertainty in satellite position and dynamics. To address this problem, recent work has proposed to apply our marginal transfer learning methodology (Sharma and Cutler, 2015).

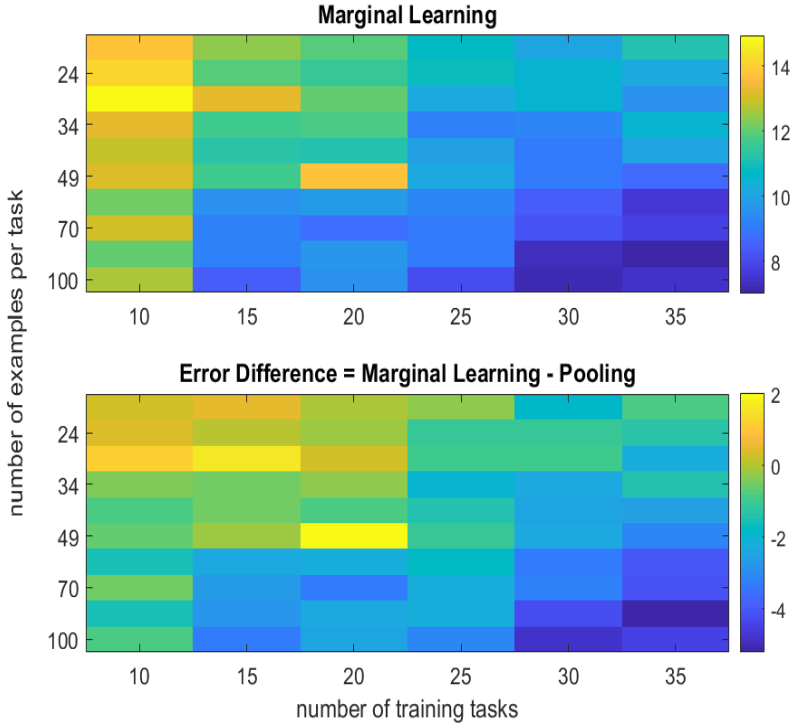

Figure 4: Parkinson's disease telemonitoring dataset: Root mean square error rates for proposed method and difference with baseline for different experimental settings, i.e., number of examples per task and number of tasks.

As a concrete instance of this problem, suppose two microsatellites are launched together. Each launch is a random phenomenon and may be viewed as a task in our framework. For each launch i, training data (X_{ij}, Y_{ij}), $j = 1, \ldots, n_i$, are generated using a highly realistic simulation model, where X_{ij} is a feature vector of RF measurements across a particular sensor network and at a particular time, and Y_{ij} is a binary label identifying which of the two microsatellites produced a given measurement. By applying our methodology, we can classify unlabeled measurements X_j^T from a new launch with high accuracy. Given these labels, orbits can subsequently be estimated using the observed RF measurements. We thank Srinagesh Sharma and James Cutler for providing us with their simulated data, and refer the reader to their paper for more details on the application (Sharma and Cutler, 2015).

To demonstrate this idea, we analyzed the data from Sharma and Cutler (2015) for $T = 50$ launches, viewing up to 40 as training data and 10 as testing. We use Gaussian kernels and the RFF kernel approximation technique to speed up the algorithm. Results are shown in Fig 5 (tables given in the appendix). As expected, the error for the proposed method is much lower than for pooling, especially as N and n increase.

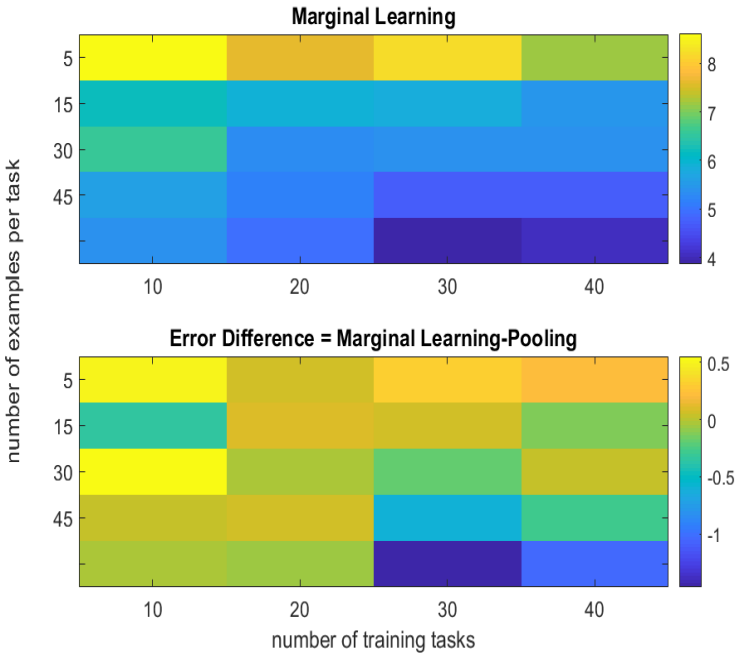

Figure 5: Satellite dataset: Classification error rates for proposed method and difference with baseline for different experimental settings, i.e., number of examples per task and number of tasks.

7.5 Flow Cytometry Experiments

We demonstrate the proposed methodology for the flow cytometry auto-gating problem, described in Sec. 2. The pooling approach has been previously investigated in this context by Toedling et al. (2006). We used a dataset that is a part of the FlowCAP Challenges where the ground truth labels have been supplied by human experts (Aghaeepour et al., 2013). We used the so-called "Normal Donors" dataset. The dataset contains 8 different classes and 30 subjects. Only two classes (0 and 2) have consistent class ratios, so we have restricted our attention to these two.

The corresponding flow cytometry data sets have sample sizes ranging from 18,641 to 59,411, and the proportion of class 0 in each data set ranges from 25.59 to 38.44%. We randomly selected 10 tasks as test tasks and used exactly the same tasks over all experiments. We varied the number of tasks in the training data from 5 to 20 with an additive step size of 5, and the number of training examples per task from 32 to 16384 with a multiplicative step size of 2. We repeated this process 10 times to get the average errors which are shown in Fig. 6 and Tables 7 and 8. The kernel k_P was Gaussian, and the other two were linear. The Nyström approximation was used to achieve an efficient implementation.

For nearly all settings the proposed method has a smaller error rate than the baseline. Furthermore, for the marginal transfer learning method, when one fixes the number of training examples and increases the number of tasks then the classification error rate drops.

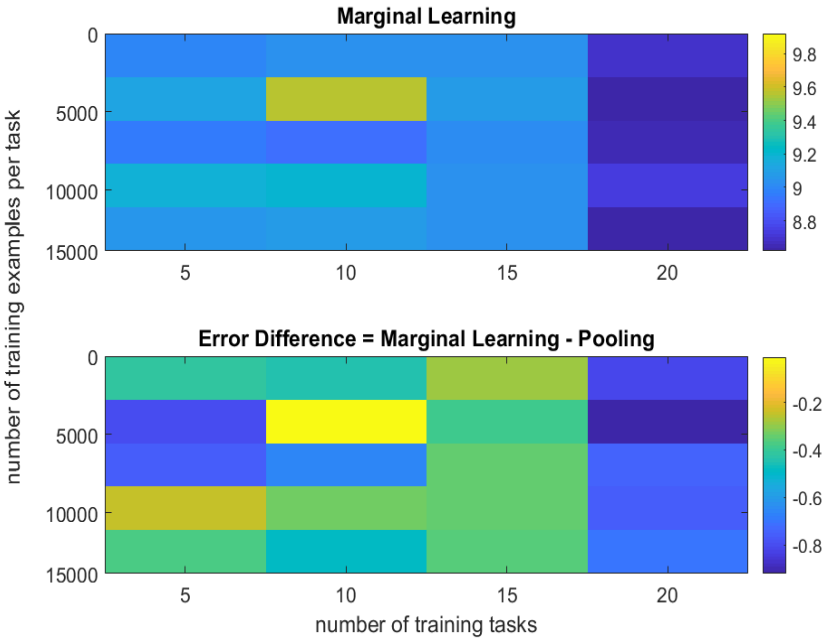

Figure 6: Flow Cytometry Dataset: Classification error rates for proposed method and difference with baseline for different experimental settings, i.e., number of examples per task and number of tasks.

8. Discussion

Our approach to marginal transfer learning relies on the extended input pattern $\widetilde{X} = (P_X, X)$. Thus, we study the natural algorithm of minimizing a regularized empirical loss over a reproducing kernel Hilbert space associated with the extended input domain $\mathfrak{P}_{\mathcal{X}} \times \mathcal{X}$. We also establish universal consistency. For this we present a novel generalization error analysis under the inherent non-iid sampling plan, and construct a universal kernel on $\mathfrak{P}_{\mathcal{X}} \times \mathcal{X}$. A detailed implementation based on novel approximate feature mappings is also presented. On one synthetic and three real-world datasets, the transfer learning approach clearly outperforms a pooling baseline.

Several future directions exist. From an application perspective, the need for adaptive classifiers arises in many applications, especially in biomedical applications involving biological and/or technical variation in patient data. Examples include brain computer interfaces and patient monitoring. For example, when electrocardiograms are used to continuously monitor cardiac patients, it is desirable to classify each heartbeat as irregular or not. Given the extraordinary amount of data involved, automation of this process is essential. However, irregularities in a test patient's heartbeat will differ from irregularities of historical patients, hence the need to adapt to the test distribution (Wiens, 2010).

From a theoretical and methodological perspective, several questions are of interest. We would like to specify conditions on μ, the distribution-generating distribution, that are favorable for generalization (beyond the simple condition discussed in Lemma 1).

We can also ask how the methodology and analysis can be extended to the context where a small number of labels are available for the test distribution, as is commonly assumed in transfer learning. In this setting, two approaches are possible. The simplest one is to use the same optimization problem (3), where we include additionally the labeled examples of the test distribution. However, if several test samples are to be treated in succession, and we want to avoid a full, resource-consuming re-training using all the training samples each time, an interesting alternative is the following: learn once a function $f_0(P_X, x)$ using the available training samples via (3); then, given a partially labeled test sample, learn a decision function on this sample only via the usual kernel norm regularized empirical loss minimization method, but replace the usual regularizer term $\|f\|^2$ by $\|f - f_0(P_x, .)\|^2$ (note that $f_0(P_x, .) \in \mathcal{H}_{\overline{k}}$). In this sense, the marginal-adaptive decision function learned from the training samples would serve as a "prior" for learning on the test data.

Future work may also consider the multiclass setting, and other asymptotic regimes, e.g., where $\{n_i\}, n_T$ do not tend to infinity, or they tend to infinity much slower than N.

Appendix A. Proofs

This section contains technical details for the proofs of the announced results.

A.1 Proof of Theorem 2

We control the difference between the training loss and the idealized test loss via the following decomposition:

$$\begin{aligned}
&\sup_{f\in\mathcal{B}_{\overline{k}}(R)}\left|\frac{1}{N}\sum_{i=1}^{N}\frac{1}{n_i}\sum_{j=1}^{n_i}\ell(f(\widetilde{X}_{ij}),Y_{ij})-\mathcal{E}(f,\infty)\right| \\
&\qquad\leq \sup_{f\in\mathcal{B}_{\overline{k}}(R)}\left|\frac{1}{N}\sum_{i=1}^{N}\frac{1}{n_i}\sum_{j=1}^{n_i}\Big(\ell(f(\widehat{P}_X^{(i)},X_{ij}),Y_{ij})-\ell(f(P_X^{(i)},X_{ij}),Y_{ij})\Big)\right| \\
&\qquad\quad+\sup_{f\in\mathcal{B}_{\overline{k}}(R)}\left|\frac{1}{N}\sum_{i=1}^{N}\frac{1}{n_i}\sum_{j=1}^{n_i}\ell(f(P_X^{(i)},X_{ij}),Y_{ij})-\mathcal{E}(f,\infty)\right| \\
&\qquad=: (I)+(II).
\end{aligned}$$

A.1.1 Control of term (I)

Using the assumption that the loss ℓ is L_ℓ-Lipschitz in its first coordinate, we can bound the first term as follows:

$$\begin{aligned}
(I)&\leq L_\ell\sup_{f\in\mathcal{B}_{\overline{k}}(R)}\frac{1}{N}\sum_{i=1}^{N}\frac{1}{n_i}\sum_{i=1}^{n_i}\left|f(\widehat{P}_X^{(i)},X_{ij})-f(P_X^{(i)},X_{ij})\right| \\
&\leq L_\ell\sup_{f\in\mathcal{B}_{\overline{k}}(R)}\frac{1}{N}\sum_{i=1}^{N}\left\|f(\widehat{P}_X^{(i)},.)-f(P_X^{(i)},.)\right\|_\infty
\end{aligned}$$

We now use the following result:

Lemma 10 *Assume the general conditions in* **(K-A)** *hold. Let P_X be an arbitrary distribution on $\mathcal{X}$ and $\widehat{P}_X$ denote an empirical distribution on $\mathcal{X}$ based on an iid sample of size n from P_X. Then with probability at least $1-\delta$ over the draw of this sample, it holds that*

$$\sup_{f\in\mathcal{B}_{\overline{k}}(R)}\left\|f(\widehat{P}_X^{(i)},.)-f(P_X^{(i)},.)\right\|_\infty\leq 3RB_kB_{k'}L_{\mathfrak{K}}\left(\frac{\log 2\delta^{-1}}{n}\right)^{\frac{\alpha}{2}}.$$

Proof Let $X_1,\ldots,X_n$ denote the n-sample from P_X. Let us denote Φ'_X the canonical mapping $x\mapsto k'_X(x,.)$ from $\mathcal{X}$ into $\mathcal{H}_{k'_{\mathcal{X}}}$. We have for all $x\in\mathcal{X}$, $\|\Phi'_X(x)\|\leq B_{k'}$, so, as a consequence of Hoeffding's inequality in a Hilbert space (see, e.g., (Pinelis and Sakhanenko, 1985)) we have with probability $1-\delta$:

$$\left\|\Psi(P_X)-\Psi(\widehat{P}_X)\right\|=\left\|\frac{1}{n}\sum_{i=1}^{n}\Phi'_X(X_i)-\mathbb{E}_{X\sim P_X}\left[\Phi'_X(X)\right]\right\|\leq 3B_{k'}\sqrt{\frac{\log 2\delta^{-1}}{n}}. \tag{21}$$

On the other hand, using the reproducing property of the kernel $\overline{k}$, we have for any $x \in \mathcal{X}$ and $f \in \mathcal{B}_{\overline{k}}(R)$:

$$
\begin{aligned}
&|f(\widehat{P}_X, x) - f(P_X, x)| \\
&\qquad = \left|\left\langle \overline{k}((\widehat{P}_X, x), .) - \overline{k}((P_X, x), .), f \right\rangle\right| \\
&\qquad \leq \|f\| \left\| \overline{k}((\widehat{P}_X, x), .) - \overline{k}((P_X, x), .) \right\| \\
&\qquad \leq R k_X(x,x)^{\frac{1}{2}} \Big(\mathfrak{K}(\Psi(P_X), \Psi(P_X)) \\
&\qquad\qquad + \mathfrak{K}(\Psi(\widehat{P}_X), \Psi(\widehat{P}_X)) - 2\mathfrak{K}(\Psi(P_X), \Psi(\widehat{P}_X)) \Big)^{\frac{1}{2}} \\
&\qquad \leq R B_k \left\| \Phi_{\mathfrak{K}}(\Psi(P_X)) - \Phi_{\mathfrak{K}}(\Psi(\widehat{P}_X)) \right\| \\
&\qquad \leq R B_k L_{\mathfrak{K}} \left\| \Psi(P_X) - \Psi(\widehat{P}_X) \right\|^{\alpha},
\end{aligned}
$$

where we have used the fact that for all $P \in \mathfrak{P}_{\mathcal{X}}$, $\|\Psi(P)\| \leq \int_{\mathcal{X}} \|k'_X(x, \cdot)\| \, dP_X(x) \leq B_{k'}$, so that $\Psi(P) \in \mathcal{B}_{k'_X}(B_{k'})$. Combining with (21) gives the result. ■

Conditionally to the draw of $(P_X^{(i)})_{1 \leq i \leq N}$, we can now apply this lemma to each $(P_X^{(i)}, \widehat{P}_X^{(i)})$ then the union bound over $i = 1, \ldots, N$ to get that with probability $1 - \delta$ (conditionally to $(P_X^{(i)})_{1 \leq i \leq N}$, and thus also unconditionally):

$$
(I) \leq 3 R B_k B_{k'} L_\ell L_{\mathfrak{K}} \left(\frac{\log \delta^{-1} + \log 2N}{n} \right)^{\frac{\alpha}{2}}.
$$

A.1.2 Control of term (II)

First define the conditional (idealized) test error for a given test distribution P_{XY}^T as

$$
\mathcal{E}(f, \infty | P_{XY}^T) := \mathbb{E}_{(X^T, Y^T) \sim P_{XY}^T} \left[\ell(f(P_X^T, X^T), Y^T) \right]. \tag{22}
$$

We can further decompose (II) as

$$
\begin{aligned}
(II) &\leq \frac{1}{N} \sum_{i=1}^{N} \frac{1}{n_i} \sum_{j=1}^{n_i} \Big(\ell(f(P_X^{(i)}, X_{ij}), Y_{ij}) - \mathcal{E}(f, \infty | P_{XY}^{(i)}) \Big) \\
&\quad + \frac{1}{N} \sum_{i=1}^{N} \Big(\mathcal{E}(f, \infty | P_{XY}^{(i)}) - \mathcal{E}(f, \infty) \Big) \\
&=: (IIa) + (IIb).
\end{aligned}
$$

We recall in what follows that the loss function ℓ is positive and bounded by the constant B_ℓ, and that the kernel $\mathfrak{K}$ is bounded by $B_{\mathfrak{K}}^2$.

Control of term (IIa). We study term (IIa) conditional to $(P_{XY}^{(i)})_{1\leq i\leq N}$. In this case, note that for this conditional distribution, the variables $(X_{ij},Y_{ij})_{ij}$ are now independent (but not identically distributed) variables. We can thus apply the Azuma-McDiarmid inequality (McDiarmid, 1989) to the function

$$\zeta((X_{ij},Y_{ij})_{ij}) := \sup_{f\in\mathcal{B}_{\overline{k}}(R)} \frac{1}{N}\sum_{i=1}^{N}\frac{1}{n_i}\sum_{j=1}^{n_i}\left(\ell(f(P_X^{(i)},X_{ij}),Y_{ij}) - \mathcal{E}(f,\infty|P_{XY}^{(i)})\right).$$

We deduce that with probability $1-\delta$ over the (conditional, then also unconditional) draw of $(X_{ij},Y_{ij})_{ij}$, it holds

$$\left|\zeta - \mathbb{E}\left[\zeta\middle|(P_{XY}^{(i)})_{1\leq i\leq N}\right]\right| \leq \sqrt{C_\zeta \log\delta^{-1}}\,;$$

where

$$C_\zeta := \frac{B_\ell^2}{N^2}\sum_{i=1}^{N}\frac{1}{n_i}\,;$$

note that when all n_is are equal to n, this simplifies to

$$C_\zeta := \frac{B_\ell^2}{Nn}\,.$$

Next, to bound $\mathbb{E}\left[\zeta\middle|(P_{XY}^{(i)})_{1\leq i\leq N}\right]$, we can use relatively standard Rademacher complexity analysis. Denote $(\varepsilon_{ij})_{1\leq i\leq N,1\leq j\leq n_i}$ iid Rademacher variables (independent from everything else). We have

$$\begin{aligned}
&\mathbb{E}\left[\zeta\middle|(P_{XY}^{(i)})_{1\leq i\leq N}\right]\\
&= \mathbb{E}_{(X_{ij},Y_{ij})}\left[\frac{1}{N}\sup_{f\in\mathcal{B}_{\overline{k}}(R)}\sum_{i=1}^{N}\frac{1}{n_i}\sum_{j=1}^{n_i}\left(\ell(f(P_X^{(i)},X_{ij}),Y_{ij}) - \mathcal{E}(f,\infty|P_{XY}^{(i)})\right)\middle|(P_{XY}^{(i)})_{1\leq i\leq N}\right]\\
&\leq \frac{2}{N}\mathbb{E}_{(X_{ij},Y_{ij})}\mathbb{E}_{(\varepsilon_{ij})}\left[\sup_{f\in\mathcal{B}_{\overline{k}}(R)}\sum_{i=1}^{N}\frac{1}{n_i}\sum_{j=1}^{n_i}\varepsilon_{ij}\left(\ell(f(P_X^{(i)},X_{ij}),Y_{ij})\right)\middle|(P_{XY}^{(i)})_{1\leq i\leq N}\right]\\
&\leq \frac{2RL_\ell B_k B_{\mathfrak{K}}}{N}\sqrt{\sum_{i=1}^{N}\frac{1}{n_i}}\,.
\end{aligned}$$

The first inequality is a standard symmetrization argument. The last inequality is a variation (with possibly unequal weights $1/n_i$) on the standard bound (see (Bartlett and Mendelson, 2002), Theorem 7 and Lemma 22) for the Rademacher complexity of a Lipschitz loss function on the ball of radius R of $\mathcal{H}_{\overline{k}}$, the kernel $\overline{k}$ being bounded by $B_k^2 B_{\mathfrak{K}}^2$. In case all n_is are equal, this boils down to

$$\mathbb{E}\left[\zeta\middle|(P_{XY}^{(i)})_{1\leq i\leq N}\right] \leq 2L_\ell R B_X B_{\mathfrak{K}}\sqrt{\frac{1}{Nn}}\,.$$

Control of term (IIb). Since the $(P_{XY}^{(i)})_{1\leq i\leq N}$ are iid, we can apply the Azuma-McDiarmid inequality to the function

$$\xi((P_{XY}^{(i)})_{1\leq i\leq N}) := \sup_{f\in\mathcal{B}_{\overline{k}}(R)} \frac{1}{N}\sum_{i=1}^{N}\Big(\mathcal{E}(f,\infty|P_{XY}^{(i)}) - \mathcal{E}(f,\infty)\Big)\,,$$

obtaining that with probability $1-\delta$ over the draw of $(P_{XY}^{(i)})_{1\leq i\leq N}$, it holds

$$|\xi - \mathbb{E}\left[\xi\right]| \leq B_\ell\sqrt{\frac{\log\delta^{-1}}{2N}}\,;$$

Rademacher complexity analysis for bounding $\mathbb{E}\left[\xi\right]$: below, we will denote (X_i,Y_i) a (single) draw from distribution $P_{XY}^{(i)}$ (and these draws are independent). We also denote $(\varepsilon_i)_{1\leq i\leq N}$ iid Rademacher variables (independent from everything else). We have

$$\begin{aligned}
\mathbb{E}\left[\xi\right] &= \mathbb{E}_{(P_{XY}^{(i)})_{1\leq i\leq N}}\Bigg[\sup_{f\in\mathcal{B}_{\overline{k}}(R)}\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{(X,Y)\sim P_{XY}^{(i)}}\left[\ell(f(P_X,X),Y)\right] \\
&\qquad\qquad - \mathbb{E}_{P_{XY}\sim\mu}\mathbb{E}_{(X,Y)\sim P_{XY}}\left[\ell(f(P_X,X),Y)\right]\Bigg] \\
&\leq \frac{2}{N}\mathbb{E}_{(P_{XY}^{(i)})_{1\leq i\leq N}}\mathbb{E}_{(\varepsilon_i)_{1\leq i\leq N}}\left[\sup_{f\in\mathcal{B}_{\overline{k}}(R)}\sum_{i=1}^{N}\varepsilon_i\mathbb{E}_{(X_i,Y_i)\sim P_{XY}^{(i)}}\left[\ell(f(P_X^{(i)},X_i),Y_i)\right]\right] \\
&\leq \frac{2}{N}\mathbb{E}_{(P_{XY}^{(i)})_{1\leq i\leq N}}\mathbb{E}_{(X_i,Y_i)_{1\leq i\leq N}}\mathbb{E}_{(\varepsilon_i)_{1\leq i\leq N}}\left[\sup_{f\in\mathcal{B}_{\overline{k}}(R)}\sum_{i=1}^{N}\varepsilon_i\ell(f(P_X^{(i)},X_i),Y_i)\right] \\
&\leq \frac{2RL_\ell B_k B_{\mathfrak{K}}}{\sqrt{N}}\,.
\end{aligned}$$

The first inequality is a standard symmetrization argument. In the second inequality, the inner expectation on the (X_i,Y_i) is pulled outwards. The last inequality is a standard bound for the Rademacher complexity of a Lipschitz loss function on the ball of radius R of $\mathcal{H}_{\overline{k}}$, the kernel $\overline{k}$ being bounded by $B_k^2 B_{\mathfrak{K}}^2$.

Combining all of the above inequalities, we obtained the announced result of the theorem.

A.2 Regularity conditions for the kernel $\mathfrak{K}$

We investigate sufficient conditions on the kernel $\mathfrak{K}$ to ensure the regularity condition (9). Roughly speaking, the regularity of the feature mapping of a reproducing kernel is "one half" of the regularity of the kernel in each of its variables. The next result considers the situation where $\mathfrak{K}$ is itself simply a Hölder continuous function of its variables.

Lemma 11 *Let $\alpha\in(0,\frac{1}{2}]$. Assume that the kernel $\mathfrak{K}$ is Hölder continuous of order 2α and constant $L_{\mathfrak{K}}^2/2$ in each of its two variables on $\mathcal{B}_{k'_X}(B_{k'})$. Then (9) is satisfied.*

Proof For any $v, w \in \mathcal{B}_{k'_X}(B_{k'})$:

$$\|\Phi_{\mathfrak{K}}(v) - \Phi_{\mathfrak{K}}(w)\| = (\mathfrak{K}(v,v) + \mathfrak{K}(w,w) - 2\mathfrak{K}(v,w))^{\frac{1}{2}} \leq L_{\mathfrak{K}} \|v - w\|^{\frac{\alpha}{2}}$$

■

The above type of regularity only leads to a Hölder feature mapping of order at most $\frac{1}{2}$ (when the kernel function is Lipschitz continuous in each variable). Since this order plays an important role in the rate of convergence of the upper bound in the main error control theorem, it is desirable to study conditions ensuring more regularity, in particular a feature mapping which has at least Lipschitz continuity. For this, we consider the following stronger condition, namely that the kernel function is twice differentiable in a specific sense:

Lemma 12 *Assume that, for any $u, v \in \mathcal{B}_{k'_X}(B_{k'})$ and unit norm vector e of $\mathcal{H}_{k'_X}$, the function $h_{u,v,e} : (\lambda, \mu) \in \mathbb{R}^2 \mapsto \mathfrak{K}(u + \lambda e, v + \mu e)$ admits a mixed partial derivative $\partial_1 \partial_2 h_{u,v,e}$ at the point $(\lambda, \mu) = (0, 0)$ which is bounded in absolute value by a constant $C^2_{\mathfrak{K}}$ independently of (u, v, e).*

Then (9) is satisfied with $\alpha = 1$ and $L_{\mathfrak{K}} = C_{\mathfrak{K}}$, that is, the canonical feature mapping of $\mathfrak{K}$ is Lipschitz continuous on $\mathcal{B}_{k'_X}(B_{k'})$.

Proof The argument is along the same lines as (Steinwart and Christmann, 2008), Lemma 4.34. Observe that, since $h_{u,v,e}(\lambda + \lambda', \mu + \mu') = h_{u+\lambda e, v+\mu e, e}(\lambda', \mu')$, the function $h_{u,v,e}$ actually admits a uniformly bounded mixed partial derivative in any point $(\lambda, \mu) \in \mathbb{R}^2$ such that $(u + \lambda e, v + \mu e) \in \mathcal{B}_{k'_X}(B_{k'})$. Let us denote $\Delta_1 h_{u,v,e}(\lambda, \mu) := h_{u,v,e}(\lambda, \mu) - h_{u,v,e}(0, \mu)$. For any $u, v \in \mathcal{B}_{k'_X}(B_{k'})$, $u \neq v$, let us denote $\lambda := \|v - u\|$ and the unit vector $e := \lambda^{-1}(v - u)$; we have

$$\begin{aligned}
\|\Phi_{\mathfrak{K}}(u) - \Phi_{\mathfrak{K}}(v)\|^2 &= \mathfrak{K}(u,u) + \mathfrak{K}(u + \lambda e, u + \lambda e) - \mathfrak{K}(u, u + \lambda e) - \mathfrak{K}(u + \lambda e, u) \\
&= \Delta_1 h_{u,v,e}(\lambda, \lambda) - \Delta_1 h_{u,v,e}(\lambda, 0) \\
&= \lambda \partial_2 \Delta_1 h_{u,v,e}(\lambda, \lambda'),
\end{aligned}$$

where we have used the mean value theorem, yielding existence of $\lambda' \in [0, \lambda]$ such that the last equality holds. Furthermore,

$$\begin{aligned}
\partial_2 \Delta_1 h_{u,v,e}(\lambda, \lambda') &= \partial_2 h_{u,v,e}(\lambda, \lambda') - \partial_2 h_{u,v,e}(0, \lambda') \\
&= \lambda \partial_1 \partial_2 h_{u,v,e}(\lambda'', \lambda'),
\end{aligned}$$

using again the mean value theorem, yielding existence of $\lambda'' \in [0, \lambda]$ in the last equality. Finally, we get

$$\|\Phi_{\mathfrak{K}}(u) - \Phi_{\mathfrak{K}}(v)\|^2 = \lambda^2 \partial_1 \partial_2 h_{u,v,e}(\lambda', \lambda'') \leq C^2_{\mathfrak{K}} \|v - u\|^2 .$$

■

Lemma 13 *Assume that the kernel $\mathfrak{K}$ takes the form of either (a) $\mathfrak{K}(u, v) = g(\|u - v\|^2)$ or (b) $\mathfrak{K}(u, v) = g(\langle u, v \rangle)$, where g is a twice differentiable real function of real variable defined on $[0, 4B^2_{k'}]$ in case (a), and on $[-B^2_{k'}, B^2_{k'}]$ in case (b). Assume $\|g'\|_\infty \leq C_1$ and $\|g''\|_\infty \leq C_2$. Then $\mathfrak{K}$ satisfies the assumption of Lemma 12 with $C_{\mathfrak{K}} := 2C_1 + 16C_2 B^2_{k'}$ in case (a), and $C_{\mathfrak{K}} := C_1 + B^2_{k'} C_2$ for case (b).*

Proof In case (a), we have $h_{u,v,e}(\lambda,\mu) = g(\|u-v+(\lambda-\mu)e\|^2)$. It follows

$$\begin{aligned}|\partial_1\partial_2 h_{u,v,e}(0,0)| &= \left|-2g'(\|u-v\|^2)\,\|e\|^2 - 4g''(\|u-v\|^2)\,\langle u-v,e\rangle^2\right| \\ &\leq 2C_1 + 16B_{k'}^2 C_2\,.\end{aligned}$$

In case (b), we have $h_{u,v,e}(\lambda,\mu) = g(\langle u+\lambda e, v+\mu e\rangle)$. It follows

$$\begin{aligned}|\partial_1\partial_2 h_{u,v,e}(0,0)| &= \left|g'(\langle u,v\rangle)\,\|e\|^2 + g''(\langle u,v\rangle)\,\langle u,e\rangle\,\langle v,e\rangle\right| \\ &\leq C_1 + B_{k'}^2 C_2\,.\end{aligned}$$

■

A.3 Proof of Lemma 3

Proof Let $\mathcal{H},\mathcal{H}'$ the RKHS associated to k,k' with the associated feature mappings Φ,Φ'. Then it can be checked that $(x,x') \in \mathcal{X}\times\mathcal{X}' \mapsto \Phi(x)\otimes\Phi'(x')$ is a feature mapping for $\overline{k}$ into the Hilbert space $\mathcal{H}\otimes\mathcal{H}'$. Using (Steinwart and Christmann, 2008), Th. 4.21, we deduce that the RKHS $\overline{H}$ of $\overline{k}$ contains precisely all functions of the form $(x,x') \in \mathcal{X}\times\mathcal{X}' \mapsto F_w(x,x') = \langle w, \Phi(x)\otimes\Phi(x')\rangle$, where w ranges over $\mathcal{H}\otimes\mathcal{H}'$. Taking w of the form $w = g\otimes g'$, $g\in\mathcal{H}, g\in\mathcal{H}'$, we deduce that $\overline{H}$ contains in particular all functions of the form $f(x,x') = g(x)g(x')$, and further

$$\widetilde{\mathcal{H}} := \operatorname{span}\left\{(x,x') \in \mathcal{X}\times\mathcal{X}' \mapsto g(x)g(x'); g\in\mathcal{H}, g'\in\mathcal{H}'\right\} \subset \overline{H}.$$

Denote $\mathcal{C}(\mathcal{X}), \mathcal{C}(\mathcal{X}'), \mathcal{C}(\mathcal{X}\times\mathcal{X}')$ the set of real-valued continuous functions on the respective spaces. Let

$$\mathcal{C}(\mathcal{X})\otimes\mathcal{C}(\mathcal{X}') := \operatorname{span}\left\{(x,x') \in \mathcal{X}\times\mathcal{X}' \mapsto f(x)f'(x'); f\in\mathcal{C}(\mathcal{X}), f'\in\mathcal{C}(\mathcal{X}')\right\}.$$

Let $G(x,x')$ be an arbitrary element of $\mathcal{C}(\mathcal{X})\otimes\mathcal{C}(\mathcal{X}')$, $G(x,x') = \sum_{i=1}^k \lambda_i g_i(x) g_i'(x')$ with $g_i \in \mathcal{C}(\mathcal{X}), g_i' \in \mathcal{C}(\mathcal{X}')$ for $i=1,\ldots,k$. For $\varepsilon>0$, by universality of k and k', there exist $f_i\in\mathcal{H}, f_i'\in\mathcal{H}'$ so that $\|f_i-g_i\|_\infty \leq \varepsilon$, $\|f_i'-g_i'\|_\infty \leq \varepsilon$ for $i=1,\ldots,k$. Let $F(x,x') := \sum_{i=1}^k \lambda_i f_i(x) f_i'(x') \in \widetilde{\mathcal{H}}$. We have

$$\begin{aligned}\left\|F(x,x') - G(x,x')\right\|_\infty &\leq \left\|\sum_{i=1}^k \lambda_i (g_i(x)g_i'(x) - f_i(x)f_i'(x))\right\|_\infty \\ &= \Bigg\|\sum_{i=1}^k \lambda_i \Big[(f_i(x)-g_i(x))(g_i'(x')-f_i'(x')) \\ &\qquad + g_i(x)(g_i'(x)-f_i'(x')) + (g_i(x)-f_i(x))g_i'(x')\Big]\Bigg\|_\infty \\ &\leq \varepsilon \sum_{i=1}^k |\lambda_i|\left(\varepsilon + \|g_i\|_\infty + \left\|g_i'\right\|_\infty\right).\end{aligned}$$

This establishes that $\widetilde{\mathcal{H}}$ is dense in $\mathcal{C}(\mathcal{X}) \otimes \mathcal{C}(\mathcal{X}')$ for the supremum norm. It can be easily checked that $\mathcal{C}(\mathcal{X}) \otimes \mathcal{C}(\mathcal{X}')$ is an algebra of functions which does not vanish and separates points on $\mathcal{X} \times \mathcal{X}'$. By the Stone-Weiertrass theorem, it is therefore dense in $\mathcal{C}(\mathcal{X} \times \mathcal{X}')$ for the supremum norm. We deduce that $\widetilde{\mathcal{H}}$ (and thus also $\overline{\mathcal{H}}$) is dense in $\mathcal{C}(\mathcal{X} \times \mathcal{X}')$, so that $\overline{k}$ is universal. ■

A.4 Proof of Corollary 5

Proof Denote $\mathcal{E}^* = \inf_{f:\mathfrak{P}_X \times \mathcal{X} \to \mathbb{R}} \mathcal{E}(f, \infty)$. Let $\varepsilon > 0$. Since $\overline{k}$ is a universal kernel on $\mathfrak{P}_X \times \mathcal{X}$ and ℓ is Lipschitz, there exists $f_0 \in \mathcal{H}_{\overline{k}}$ such that $\mathcal{E}(f_0, \infty) \leq \mathcal{E}^* + \frac{\varepsilon}{2}$ (Steinwart and Christmann, 2008).

Let us introduce the shorthand notation

$$\widehat{\mathcal{E}}_\ell(f, N, n) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{n_i} \sum_{j=1}^{n_i} \ell(f(\tilde{X}_{ij}), Y_{ij}).$$

We will also write $\mathcal{E}(f, \infty) = \mathcal{E}_\ell(f, \infty)$ to indicate the dependence of the risk on the loss.

By comparing the objective function in (3) at the minimizer $\widehat{f}_\lambda$ and at the null function, we deduce that we must have $\|\widehat{f}_\lambda\| \leq \sqrt{\ell_0/\lambda}$. Let us denote $R_\lambda = \sqrt{\frac{\ell_0}{\lambda}}$ and $M_\lambda = B_{\mathfrak{K}} B_k R_\lambda$.

Define the truncated loss

$$\ell_\lambda(t, y) := \begin{cases} \ell(M_\lambda, y), & t > M_\lambda, \\ \ell(t, y), & -M_\lambda \leq t \leq M_\lambda, \\ \ell(-M_\lambda, y), & t < -M_\lambda. \end{cases}$$

We will write $\mathcal{E}_{\ell_\lambda}$ and $\widehat{\mathcal{E}}_{\ell_\lambda}(f, N, n)$ to denote the modified true and empirical risks when the loss is ℓ_λ. This loss is easily seen to be L_ℓ-Lipschitz. It is also bounded, which can be seen by noting that it suffices to bound $\ell_\lambda(t, y)$ for $(t, y) \in [-M_\lambda, M_\lambda] \times \mathcal{Y}$, in which case we have $\ell_\lambda(t, y) = \ell(t, y) \leq \ell_0 + M_\lambda L_\ell$ by the definition of ℓ_0 and the fact that ℓ is L_ℓ-Lipschitz. Applying Theorem 5.1 for $R = R_\lambda = \sqrt{\ell_0/\lambda}$, $B_{\ell_\lambda} = \ell_0 + L_\ell B_{\mathfrak{K}} B_k R_\lambda$, and $\delta = 1/(Nn)^2$, gives that with probability at least $1 - 1/(Nn)^2$,

$$\sup_{f \in B_{\overline{k}(R_\lambda)}} \left| \widehat{\mathcal{E}}_{\ell_\lambda}(f, N, n) - \mathcal{E}_{\ell_\lambda}(f, \infty) \right| \leq \varepsilon(N, n)$$

where

$$\varepsilon(N, n) := \frac{C_1}{\sqrt{\lambda}} \left(\frac{\log N + \log n}{n} \right)^{\frac{\alpha}{2}} + \frac{C_2}{\sqrt{\lambda N}} + \left(C_3 + \frac{C_4}{\sqrt{\lambda}} \right) \sqrt{\frac{\log N + \log n}{N}}.$$

The following result captures the key property that on the ball $\mathcal{B}_{\overline{k}}(R_\lambda)$, ℓ and ℓ_λ yield the same true and empirical risks.

Lemma 14 *$\forall f \in \mathcal{B}_{\overline{k}}(R_\lambda)$, $\mathcal{E}_\ell(f, \infty) = \mathcal{E}_{\ell_\lambda}(f, \infty)$ and $\widehat{\mathcal{E}}_\ell(f, N, n) = \widehat{\mathcal{E}}_{\ell_\lambda}(f, N, n)$.*

Proof If $f \in \mathcal{B}_{\overline{k}}(R_\lambda)$, then the reproducing property and Cauchy-Schwarz imply that for an arbitrary $\tilde{x}$, $|f(\tilde{x})| \leq B_{\mathfrak{K}} B_k R_\lambda = M_\lambda$. The result now follows from the definitions of ℓ_λ and of the true and empirical risks. ■

Let N, n be large enough so that $\|f_0\| \leq R_\lambda$. We can now deduce that with probability at least $1 - 1/(Nn)^2$,

$$\begin{aligned}
\mathcal{E}_\ell(\widehat{f}_\lambda, \infty) &= \mathcal{E}_{\ell_\lambda}(\widehat{f}_\lambda, \infty) \\
&\leq \widehat{\mathcal{E}}_{\ell_\lambda}(\widehat{f}_\lambda, N, n) + \varepsilon(N, n) \\
&= \widehat{\mathcal{E}}_\ell(\widehat{f}_\lambda, N, n) + \varepsilon(N, n) \\
&= \widehat{\mathcal{E}}_\ell(\widehat{f}_\lambda, N, n) + \lambda \|\widehat{f}_\lambda\|^2 - \lambda \|\widehat{f}_\lambda\|^2 + \varepsilon(N, n) \\
&\leq \widehat{\mathcal{E}}_\ell(f_0, N, n) + \lambda \|f_0\|^2 - \lambda \|\widehat{f}_\lambda\|^2 + \varepsilon(N, n) \\
&\leq \widehat{\mathcal{E}}_\ell(f_0, N, n) + \lambda \|f_0\|^2 + \varepsilon(N, n) \\
&= \widehat{\mathcal{E}}_{\ell_\lambda}(f_0, N, n) + \lambda \|f_0\|^2 + \varepsilon(N, n) \\
&\leq \mathcal{E}_{\ell_\lambda}(f_0, \infty) + \lambda \|f_0\|^2 + 2\varepsilon(N, n) \\
&= \mathcal{E}_\ell(f_0, \infty) + \lambda \|f_0\|^2 + 2\varepsilon(N, n) \\
&\leq \mathcal{E}^* + \frac{\varepsilon}{2} + \lambda \|f_0\|^2 + 2\varepsilon(N, n).
\end{aligned}$$

The last two terms become less than $\frac{\varepsilon}{2}$ for N, n sufficiently large by the assumptions on the growth of N, n, and $\lambda = \lambda(N, n)$. This establishes that for any $\varepsilon > 0$, there exist N_0 and n_0 such that

$$\sum_{N \geq N_0} \sum_{n \geq n_0} \Pr(\mathcal{E}(\widehat{f}_\lambda, \infty) \geq \mathcal{E}^* + \varepsilon) \leq \sum_{N \geq N_0} \sum_{n \geq n_0} \frac{1}{N^2 n^2} < \infty,$$

and so the result follows by the Borel-Cantelli lemma. ■

A.5 Proof of Theorem 7

Proof Observe:

$$\bar{k}(\tilde{x}, \tilde{x}') = \exp\left\{ \frac{-1}{2\sigma_P^2} \|\Psi(\widehat{P}_X) - \Psi(\widehat{P}'_X)\|^2 \right\} \exp\left\{ \frac{-1}{2\sigma_X^2} \|x - x'\|^2 \right\},$$

and denote:

$$\tilde{k}(\tilde{x}, \tilde{x}') = \exp\left\{ \frac{-1}{2\sigma_P^2} \|Z_P(\widehat{P}_X) - Z_P(\widehat{P}'_X)\|^2 \right\} \exp\left\{ \frac{-1}{2\sigma_X^2} \|x - x'\|^2 \right\},$$

We omit the arguments of $\bar{k}, \tilde{k}$ for brevity. Let k_q be the final approximation ($k_q = \bar{z}(\tilde{x})^T \bar{z}(\tilde{x}')$) and then we have

$$|\bar{k} - k_q| = |\bar{k} - \tilde{k} + \tilde{k} - k_q| \leq |\bar{k} - \tilde{k}| + |\tilde{k} - k_q|. \tag{23}$$

From Eqn. (23) it follows that,

$$P(|\bar{k} - k_q| \geq \epsilon_l + \epsilon_q) \leq P(|\bar{k} - \tilde{k}| \geq \epsilon_l) + P(|\tilde{k} - k_q| \geq \epsilon_q). \tag{24}$$

By a direct application of Hoeffding's inequality,

$$P(|\tilde{k} - k_q| \geq \epsilon_q) \leq 2\exp(-\frac{Q\epsilon_q^2}{2}). \tag{25}$$

Recall that $\langle \Psi(\widehat{P}_X), \Psi(\widehat{P}'_X) \rangle = \frac{1}{n_1 n_2} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} k'_X(X_i, X'_j)$. For a pair X_i, X'_j, we have again by Hoeffding

$$P(|z'_X(X_i)^T z'_X(X'_j) - k'_X(X_i, X'_j)| \geq \epsilon) \leq 2\exp(-\frac{L\epsilon^2}{2}).$$

Let Ω_{ij} be the event $|z'_X(X_i)^T z'_X(X'_j) - k'_X(X_i, X'_j)| \geq \epsilon$, for particular i, j. Using the union bound we have

$$P(\Omega_{11} \cup \Omega_{12} \cup \ldots \cup \Omega_{n_1 n_2}) \leq 2n_1 n_2 \exp(-\frac{L\epsilon^2}{2})$$

This implies

$$P(|Z_P(\widehat{P}_X)^T Z_P(\widehat{P}'_X) - \langle \Psi(\widehat{P}_X), \Psi(\widehat{P}'_X) \rangle| \geq \epsilon) \leq 2n_1 n_2 \exp(-\frac{L\epsilon^2}{2}). \tag{26}$$

Therefore,

$$\begin{aligned}
\left|\bar{k} - \tilde{k}\right| &= \left| \exp\left\{\frac{-1}{2\sigma_X^2}\|x - x'\|^2\right\} \left[\exp\left\{\frac{-1}{2\sigma_P^2}\|\Psi(\widehat{P}_X) - \Psi(\widehat{P}'_X)\|^2\right\} \right.\right. \\
&\quad \left.\left. - \exp\left\{\frac{-1}{2\sigma_P^2}\|Z_P(\widehat{P}_X) - Z_P(\widehat{P}'_X)\|^2\right\} \right] \right| \\
&\leq \left| \left[\exp\left\{\frac{-1}{2\sigma_P^2}\|\Psi(\widehat{P}_X) - \Psi(\widehat{P}'_X)\|^2\right\} - \exp\left\{\frac{-1}{2\sigma_P^2}\|Z_P(\widehat{P}_X) - Z_P(\widehat{P}'_X)\|^2\right\} \right] \right|
\end{aligned}$$

$$
\begin{aligned}
&= \left| \exp\left\{ \frac{-1}{2\sigma_P^2} \|\Psi(\widehat{P}_X) - \Psi(\widehat{P}'_X)\|^2 \right\} \left[1 - \exp\left\{ \frac{-1}{2\sigma_P^2} \Big(\|Z_P(\widehat{P}_X) - Z_P(\widehat{P}'_X)\|^2 \right.\right.\right. \\
&\qquad \left.\left.\left. - \|\Psi(\widehat{P}_X) - \Psi(\widehat{P}'_X)\|^2 \Big) \right\} \right] \right| \\
&\leq \left| \left[1 - \exp\left\{ \frac{-1}{2\sigma_P^2} \Big(\|Z_P(\widehat{P}_X) - Z_P(\widehat{P}'_X)\|^2 - \|\Psi(\widehat{P}_X) - \Psi(\widehat{P}'_X)\|^2 \Big) \right\} \right] \right| \\
&= \left| 1 - \exp\left\{ \frac{-1}{2\sigma_P^2} \Big(Z_P(\widehat{P}_X)^T Z_P(\widehat{P}_X) - \langle \Psi(\widehat{P}_X), \Psi(\widehat{P}_X) \rangle + Z_P(\widehat{P}'_X)^T Z_P(\widehat{P}'_X) \right.\right. \\
&\qquad \left.\left. - \langle \Psi(\widehat{P}'_X), \Psi(\widehat{P}'_X) \rangle - 2\big(Z_P(\widehat{P}_X)^T Z_P(\widehat{P}'_X) - \langle \Psi(\widehat{P}_X), \Psi(\widehat{P}'_X) \rangle \big) \Big) \right\} \right| \\
&\leq \left| 1 - \exp\left\{ \frac{1}{2\sigma_P^2} \Big(|Z_P(\widehat{P}_X)^T Z_P(\widehat{P}_X) - \langle \Psi(\widehat{P}_X), \Psi(\widehat{P}_X) \rangle| + |Z_P(\widehat{P}'_X)^T Z_P(\widehat{P}'_X) \right.\right. \\
&\qquad \left.\left. - \langle \Psi(\widehat{P}'_X), \Psi(\widehat{P}'_X) \rangle| + 2|\big(Z_P(\widehat{P}_X)^T Z_P(\widehat{P}'_X) - \langle \Psi(\widehat{P}_X), \Psi(\widehat{P}'_X) \rangle \big)| \Big) \right\} \right|
\end{aligned}
$$

The result now follows by applying the bound of Eqn. (26) to each of the three terms in the exponent of the preceding expression, together with the stated formula for ϵ in terms of ϵ_ℓ.

■

A.6 Proof of Theorem 9

Proof The proof is very similar to the proof of Theorem 7. We use Lemma 8 to replace bound (25) with:

$$
P\Big(\sup_{x,x' \in \mathcal{M}} |\tilde{k} - k_q| \geq \epsilon_q \Big) \leq 2^8 \Big(\frac{\sigma'_X r}{\epsilon_q} \Big)^2 \exp\Big(\frac{-Q\epsilon_q^2}{2(d+2)} \Big). \tag{27}
$$

Similarly, Eqn. (26) is replaced by

$$
\begin{aligned}
P\Big(\sup_{x,x' \in \mathcal{M}} &|Z_P(\widehat{P}_X)^T Z_P(\widehat{P}'_X) - \langle \Psi(\widehat{P}_X), \Psi(\widehat{P}'_X) \rangle| \geq \epsilon \Big) \\
&\leq 2^9 n_1 n_2 \Big(\frac{\sigma_P \sigma_X r}{\epsilon_l} \Big)^2 \exp\Big(\frac{-L\epsilon_l^2}{2(d+2)} \Big).
\end{aligned} \tag{28}
$$

The remainder of the proof now proceeds as in the previous proof.

■

A.7 Results in Tabular Format

Table 1: Average Classification Error of Marginal Transfer Learning on Synthetic Dataset

Examples per Task \ Tasks	16	64	256
8	36.01	33.08	31.69
16	31.55	31.03	30.96
32	30.44	29.31	23.87
256	23.78	7.22	1.27

Table 2: Average Classification Error of Pooling on Synthetic Dataset

Examples per Task \ Tasks	16	64	256
8	49.14	49.11	50.04
16	49.89	50.04	49.68
32	50.32	50.21	49.61
256	50.01	50.43	49.93

Table 3: RMSE of Marginal Transfer Learning on Parkinson's Disease Dataset

Examples per Task \ Tasks	10	15	20	25	30	35
20	13.78	12.37	11.93	10.74	10.08	11.17
24	14.18	11.89	11.51	10.90	10.55	10.18
28	14.95	13.29	12.00	10.21	10.59	9.52
34	13.27	11.66	11.79	9.16	9.34	10.50
41	12.89	11.27	11.17	9.91	9.10	10.05
49	13.15	11.70	13.81	10.12	9.01	8.69
58	12.16	9.59	9.85	9.28	8.44	7.62
70	13.03	9.16	8.80	9.03	8.16	7.88
84	11.98	9.18	9.74	9.03	7.30	7.01
100	12.69	8.48	9.52	8.01	7.14	7.5

Table 4: RMSE of Pooling on Parkinson's Disease Dataset

Examples per Task \ Tasks	10	15	20	25	30	35
20	13.64	11.93	11.95	11.06	11.91	12.08
24	13.80	11.83	11.70	11.98	11.68	11.48
28	13.78	11.70	11.72	11.18	11.58	11.73
34	13.71	12.20	12.04	11.17	11.67	11.92
41	13.69	11.73	12.08	11.28	11.55	12.59
49	13.75	11.85	11.79	11.17	11.34	11.82
58	13.70	11.89	12.06	11.06	11.82	11.65
70	13.54	11.86	12.14	11.21	11.40	11.96
84	13.55	11.98	12.03	11.25	11.54	12.22
100	13.53	11.85	11.92	11.12	11.96	11.84

Table 5: Average Classification Error of Marginal Transfer Learning on Satellite Dataset

Examples per Task \ Tasks	10	20	30	40
5	8.62	7.61	8.25	7.17
15	6.21	5.90	5.85	5.43
30	6.61	5.33	5.37	5.35
45	5.61	5.19	4.71	4.70
all training data	5.36	4.91	3.86	4.08

Table 6: Average Classification Error of Pooling on Satellite Dataset

Examples per Task \ Tasks	10	20	30	40
5	8.13	7.54	7.94	6.96
15	6.55	5.81	5.79	5.57
30	6.06	5.36	5.56	5.31
45	5.58	5.12	5.30	4.99
all training data	5.37	4.98	5.32	5.14

Table 7: Average Classification Error of Marginal Transfer Learning on Flow Cytometry Dataset

Examples per Task \ Tasks	5	10	15	20
1024	9	9.03	9.03	8.70
2048	9.12	9.56	9.07	8.62
4096	8.96	8.91	9.01	8.66
8192	9.18	9.20	9.04	8.74
16384	9.05	9.08	9.04	8.63

Table 8: Average Classification Error of Pooling on Flow Cytometry Dataset

Examples per Task \ Tasks	5	10	15	20
1024	9.41	9.48	9.32	9.52
2048	9.92	9.57	9.45	9.54
4096	9.72	9.56	9.36	9.40
8192	9.43	9.53	9.38	9.50
16384	9.42	9.56	9.40	9.33

Acknowledgments

C. Scott and A. Deshmukh where supported in part by NSF Grants No. 1422157, 1217880, and 1047871. G. Blanchard acknowledges support by the DFG via Research Unit 1735 *Structural Inference in Statistics*.

References

N. Aghaeepour, G. Finak, H. Hoos, T. R. Mosmann, R. Brinkman, R. Gottardo, R. H. Scheuermann, FlowCAP Consortium, DREAM Consortium, et al. Critical assessment of automated flow cytometry data analysis techniques. *Nature methods*, 10(3):228–238, 2013.

R. K. Ando and T. Zhang. A high-performance semi-supervised learning method for text chunking. *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (ACL 05)*, pages 1–9, 2005.

A. Arnold, R. Nallapati, and W.W. Cohen. A comparative study of methods for transductive transfer learning. *Seventh IEEE International Conference on Data Mining Workshops*, pages 77–82, 2007.

P. Bartlett and S. Mendelson. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.

P. Bartlett, M. Jordan, and J. McAuliffe. Convexity, classification, and risk bounds. *J. Amer. Stat. Assoc.*, 101(473):138–156, 2006.

J. Baxter. A Bayesian/information theoretic model of learning to learn via multiple task sampling. *Machine Learning*, 28(1):7–39, 1997.

J. Baxter. A model of inductive bias learning. *Journal of Artificial Intelligence Research*, 12:149–198, 2000.

S. Bickel, M. Brückner, and T. Scheffer. Discriminative learning under covariate shift. *J. Machine Learning Research*, pages 2137–2155, 2009.

G. Blanchard, G. Lee, and C. Scott. Generalizing from several related classification tasks to a new unlabeled sample. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 2178–2186. 2011.

J. Carbonell, S. Hanneke, and L. Yang. A theory of transfer learning with applications to active learning. *Machine Learning*, 90(2):161–189, 2013.

R. Caruana. Multitask learning. *Machine Learning*, 28:41–75, 1997.

C. Chang and C. Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

A. Christmann and I. Steinwart. Universal kernels on non-standard input spaces. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 406–414, 2010.

N. Courty, R. Flamary, D. Tuia, and A. Rakotomamonjy. Optimal transport for domain adaptation. *IEEE Trans. Patt. Anal. Mach. Intell.*, 39(9):1853–1865, 2016.

P. Drineas and M. W. Mahoney. On the nyström method for approximating a gram matrix for improved kernel-based learning. *The Journal of Machine Learning Research*, 6:2153–2175, 2005.

T. Evgeniou, C. A. Michelli, and M. Pontil. Learning multiple tasks with kernel methods. *J. Machine Learning Research*, pages 615–637, 2005.

R. Fan, K. Chang, C. Hsieh, X. Wang, and C. Lin. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874, 2008.

M. Ghifary, D. Balduzzi, B. Kleijn, and M. Zhang. Scatter component analysis: A unified framework for domain adaptation and domain generalization. *IEEE Trans. Patt. Anal. Mach. Intell.*, 39(7):1411–1430, 2017.

A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. Smola. A kernel approach to comparing distributions. In R. Holte and A. Howe, editors, *Proceedings of the 22nd AAAI Conference on Artificial Intelligence*, pages 1637–1641, 2007a.

A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. Smola. A kernel method for the two-sample-problem. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 513–520, 2007b.

T. Grubinger, A. Birlutiu, H. Schöner, T. Natschläger, and T. Heskes. Domain generalization based on transfer component analysis. In I. Rojas, G. Joya, and A. Catala, editors, *Advances in Computational Intelligence. IWANN 2015*, volume 9094 of *Lecture Notes in Computer Science*, pages 325–334. Springer, 2015.

C. Hsieh, K. Chang, C. Lin, S. S. Keerthi, and S. Sundararajan. A dual coordinate descent method for large-scale linear svm. In *Proceedings of the 25th international conference on Machine learning*, pages 408–415. ACM, 2008.

T. Joachims. Making large scale svm learning practical. Technical report, Universität Dortmund, 1999.

O. Kallenberg. *Foundations of Modern Probability*. Springer, 2002.

Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *IJCAI*, volume 14, pages 1137–1145, 1995.

V. Koltchinskii. Rademacher penalties and structural risk minimization. *IEEE Transactions on Information Theory*, 47(5):1902 – 1914, 2001.

A. Maurer. Transfer bounds for linear feature learning. *Machine Learning*, 75(3):327–350, 2009.

A. Maurer, M. Pontil, and B. Romera-Paredes. Sparse coding for multitask and transfer learning. In Sanjoy Dasgupta and David McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 343–351, 2013.

C. McDiarmid. On the method of bounded differences. *Surveys in Combinatorics*, 141: 148–188, 1989.

K. Muandet, D. Balduzzi, and B. Schölkopf. Domain generalization via invariant feature representation. In *Proceedings of the 30th International Conference on International Conference on Machine Learning (ICML'13)*, volume 28 of *Proceedings of Machine Learning Research*, pages I–10–I–18, 2013.

S. J. Pan, I. Tsang, J. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2011.

K. R. Parthasarathy. *Probability Measures on Metric Spaces*. Academic Press, 1967.

A. Pentina and S. Ben-David. Multi-task and lifelong learning of kernels. In K. Chaudhuri, C. Gentile, and S. Zilles, editors, *Algorithmic Learning Theory: 26th International Conference (ALT'15)*, volume 9355 of *Lecture Notes in Computer Science*, pages 194–208. Springer, 2015.

A. Pentina and C. Lampert. A pac-bayesian bound for lifelong learning. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 991–999, 2014.

I.F. Pinelis and A.I. Sakhanenko. Remarks on inequalities for probabilities of large deviations. *Theory Probab. Appl.*, 30(1):143–148, 1985.

J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence. *Dataset Shift in Machine Learning*. The MIT Press, 2009.

A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *Advances in neural information processing systems*, pages 1177–1184, 2007.

A. Rettinger, M. Zinkevich, and M. Bowling. Boosting expert ensembles for rapid concept recall. *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI 06)*, 1:464–469, 2006.

S. Sharma and J. W. Cutler. Robust orbit determination and classification: A learning theoretic approach. *Interplanetary Network Progress Report*, 203:1, 2015.

B. Sriperumbudur, A. Gretton, K. Fukumizu, B. Schölkopf, and G. Lanckriet. Hilbert space embeddings and metrics on probability measures. *Journal of Machine Learning Research*, 11:1517–1561, 2010.

I. Steinwart and A. Christmann. *Support Vector Machines*. Springer, 2008.

Z. Szabo, A. Gretton, B. Poczos, and B. Sriperumbudur. Two-stage sampled learning theory on distributions. In G. Lebanon and S.V.N. Vishwanathan, editors, *Proc. 18th Int. Conf. on Artificial Intelligence and Statistics*, pages 948–957, 2015.

S. Thrun. Is learning the n-th thing any easier than learning the first? *Advances in Neural Information Processing Systems*, pages 640–646, 1996.

J. Toedling, P. Rhein, R. Ratei, L. Karawajew, and R. Spang. Automated in-silico detection of cell populations in flow cytometry readouts and its application to leukemia disease monitoring. *BMC Bioinformatics*, 7:282, 2006.

A. Tsanas, M. A. Little, P. E. McSharry, and L. O. Ramig. Accurate telemonitoring of parkinson's disease progression by noninvasive speech tests. *IEEE transactions on Biomedical Engineering*, 57(4):884–893, 2010.

J. Wiens. *Machine Learning for Patient-Adaptive Ectopic Beat Classication*. Masters Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 2010.

C. Williams and M. Seeger. Using the nyström method to speed up kernel machines. In *Proceedings of the 14th Annual Conference on Neural Information Processing Systems*, number EPFL-CONF-161322, pages 682–688, 2001.