Fast Distribution To Real Regression

Junier B. Oliva **Willie Neiswanger** **Barnabás Póczos** **Jeff Schneider** **Eric Xing**
Carnegie Mellon University

Abstract

We study the problem of distribution to real regression, where one aims to regress a mapping f that takes in a distribution input covariate $P \in \mathcal{I}$ (for a non-parametric family of distributions $\mathcal{I}$) and outputs a real-valued response $Y = f(P) + \epsilon$. This setting was recently studied in [15], where the "Kernel-Kernel" estimator was introduced and shown to have a polynomial rate of convergence. However, evaluating a new prediction with the Kernel-Kernel estimator scales as $\Omega(N)$. This causes the difficult situation where a large amount of data may be necessary for a low estimation risk, but the computation cost of estimation becomes infeasible when the data-set is too large. To this end, we propose the Double-Basis estimator, which looks to alleviate this big data problem in two ways: first, the Double-Basis estimator is shown to have a computation complexity that is independent of the number of of instances N when evaluating new predictions after training; secondly, the Double-Basis estimator is shown to have a fast rate of convergence for a general class of mappings $f \in \mathcal{F}$.

1 Introduction

A great deal of attention has been applied to studying new and better ways to perform learning tasks involving static finite vectors. Indeed, over the past century the fields of statistics and machine learning have amassed a vast understanding of various learning tasks like density estimation, clustering, classification, and regression using simple real valued vectors. However, we do not live in a world of simple objects. From the contact lists we keep, the sound waves we hear, and the distribution of cells we have, complex objects such as sets, functions, and distributions are all around us. Furthermore, with ever-increasing data collection capacities at our disposal, not only are we collecting more data, but richer and more bountiful complex data are becoming the norm.

This paper aims to make learning on massive data-sets of distributions tractable; we study distribution to real regression (DRR) where input covariates are arbitrary distributions and output responses are real values. We provide an estimator that scales well with data-set size and is efficient at evaluation-time. Furthermore, we prove that the estimator achieves a fast rate of convergence for a broad class of functions.

We consider a mapping $f : \mathcal{I} \mapsto \mathbb{R}$ that takes $P \in \mathcal{I}$, an input distribution from a family of distributions $\mathcal{I}$, and produces Y a real-valued response as:

$$Y = f(P) + \epsilon, \text{where } \mathbb{E}\left[\epsilon\right] = 0,\ \mathbb{E}\left[\epsilon^2\right] \leq \sigma_\epsilon^2. \quad (1)$$

Of course, it is infeasible to directly observe a distribution in practice. Thus, we will work on a data-set of N input sample-sets/responses:

$$\mathcal{D} = \{(\mathcal{X}_i, Y_i)\}_{i=1}^N, \text{ where} \quad (2)$$

$$\mathcal{X}_i = \{X_{i1}, \ldots, X_{in_i}\},\ X_{ij} \overset{iid}{\sim} P_i \in \mathcal{I}, \quad (3)$$

and $Y_i = f(P_i) + \epsilon_i$. Further, $P_i \overset{iid}{\sim} \Phi$, where Φ is some measure over $\mathcal{I}$ (see Figure 1).

Many interesting problems across various domains fit the DRR model. For instance, one may be interested in studying the mapping that takes in the distribution of star locations in a galaxy and outputs the galaxy's age. Also, one may be consider a mapping that takes in the distribution of prices for stocks of a particular sector and outputs the future average change in stock price for that sector.

In fact, many estimation tasks in statistics can be framed as a distribution to regression problem. For instance, in parameter estimation one studies a mapping that takes in a distribution (usually restricted to be in a parametric class of distributions) and outputs

Appearing in Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS) 2014, Reykjavik, Iceland. JMLR: W&CP volume 33. Copyright 2014 by the authors.

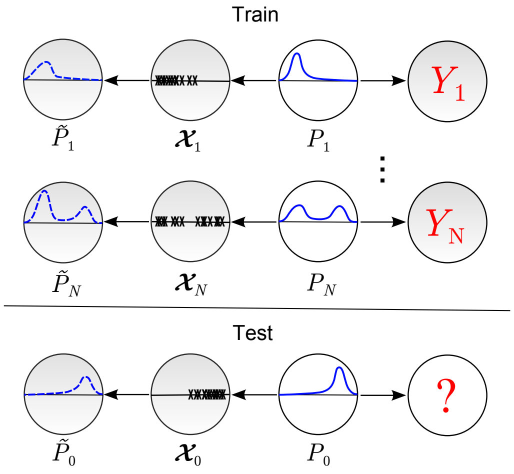


Figure 1: A graphical representation of our model. We observe a data-set of input sample-set/output response pairs $\{(\mathcal{X}_i, Y_i)\}_{i=1}^N$, where $\mathcal{X}_i = \{X_{i1}, \ldots, X_{in_i}\}$, $X_{ij} \sim P_i$ and $Y_i = f(P_i) + \epsilon_i$, for some noise ϵ_i. From these sample sets $\{\mathcal{X}_i\}_{i=1}^N$ we build density estimates $\{\tilde{P}_i\}_{i=1}^N$ using projection series estimates (8). These estimates will then be used in our response estimator (13).

a corresponding parameter. We will see that our estimator can be used to leverage previously seen sample sets to outperform standard estimation procedures, to perform model selection when cross validation is expensive, or to perform parameter estimation when no analytical sample estimate is available. In effect, we shall show that this estimator, and the concept of distribution to real regression, is powerful enough to itself learn how to perform general statistical procedures.

At its core, the problem of distribution to real value regression is a learning task over infinite dimensional objects (distributions) and would benefit greatly from learning on data-sets with a large number of input/output pairs. Hence, this paper focuses on the case where one has a massive data-set in terms of instances, i.e. $n_i = o(N)$. DRR for the case of general input distributions in a Hölder class and a smooth class of mappings has been previously studied in [15]. There, an estimator—the Kernel-Kernel estimator—analogous to the Nadaraya-Watson estimator [20] for functional distribution inputs was shown to have a polynomial rate of convergence. This rate is dependent on the dimensionality of the domains of the distributions, sample sizes, and a doubling dimension on the measure Φ over distributions, which, roughly speaking, controls the degrees of freedom of the input distributions. However, evaluating the estimator in [15] for new predictions scales as $\Omega(N)$ in the number of input/output instances in a data-set. Thus, the Kernel-Kernel estimator is not feasible for data-sets where the number of distributions, N, is in the high-thousands, millions, or even billions. Furthermore, the doubling dimension of Φ may be rather large, producing a slow convergence rate. In this paper we shall introduce an estimator for DRR, the Double-Basis estimator, which does not depend on N for evaluating an estimate for a new input distribution. Furthermore, we shall show that this estimator achieves a better rate of convergence that does not depend on the doubling dimension over a broad class of distribution to real mappings.

2 Related Work

As previously mentioned, the problem of DRR was studied in [15], where the Kernel-Kernel estimator was introduced. Since the data-set one works with is (2), first one uses kernel density estimation (KDE) [20] on $\{\mathcal{X}_1, \ldots, \mathcal{X}_N\}$ to make density estimates $\{\tilde{P}_1, \ldots, \tilde{P}_N\}$. Similarly for an unseen query input sample set $\mathcal{X}_0 \sim P_0$, one makes a KDE $\tilde{P}_0$. Then, the Kernel-Kernel estimator works as follows:

$$\hat{f}(\tilde{P}) = \sum_{i=1}^{N} W(\tilde{P}_i, \tilde{P}_0) Y_i, \text{ where} \tag{4}$$

$$W(\tilde{P}_i, \tilde{P}_0) = \begin{cases} \frac{K(D(\tilde{P}_i, \tilde{P}_0))}{\sum_j K(D(\tilde{P}_j, \tilde{P}_0))} & \text{if } \sum_j K(D(\tilde{P}_j, \tilde{P}_0)) > 0 \\ 0 & \text{otherwise .} \end{cases} \tag{5}$$

Here K is taken to be a symmetric Kernel with bounded support, and D is some metric over functions. Clearly, (4) scales as $\Omega(N)$ in terms of the number of input distributions in ones data-set. Furthermore, if one uses a Gaussian KDE, and takes $D(\tilde{P}_i, \tilde{P}_0) = \|\tilde{P}_i - \tilde{P}_0\|_2 = \sqrt{\int (\tilde{p}_i - \tilde{p}_0)^2}$ (where $\tilde{p}_i$ is the pdf of $\tilde{P}_i$) and $n_i \asymp n$, then the computation required for evaluating (4) is $\Omega(Nn^2)$.

DRR is related to the functional analysis, where one regresses a mapping whose input domain are functions [1]. However, the objects DRR works over–distributions and their pdfs–are inferred through sets of samples drawn from the objects, with finite sizes. In functional analysis, the functions are inferred through observations of (X, Y) pairs that are often taken to be an arbitrarily dense grid in the domain of the functions. For a comprehensive survey in functional analysis see [1, 18]. Also, recently [13] studied the problem of distribution to distribution regression, where both input and output covariates are distributions.

A common approach to performing ML tasks with distributions is to embed the distributions in a Hilbert space, then solve the tasks using kernel machines. Perhaps the most clear-cut of these methods is to fit a parametric model to distributions for estimating kernels [5, 4, 10]. Nonparametric methods over distributions have also been developed using kernels. For example, since we only observe distributions through finite sets, set kernels may be used [19]. Futhermore, the representer theorem was recently generalized for the space of distributions [11]. Also, kernels based on nonparametric estimators of divergences have been explored [16, 14].

3 Double-Basis Estimator

We introduce the Double-Basis Estimator for DRR. First, we shall use orthonormal basis projection estimators [20] for estimating the densities of P_i from $\mathcal{X}_i$. Suppose that $\Lambda^l \subseteq \mathbb{R}^l$, the domain of input densities is compact s.t. $\Lambda = [a, b]$. Let $\{\varphi_i\}_{i\in\mathbb{Z}}$ be an orthonormal basis for $L_2(\Lambda)$. Then, the tensor product of $\{\varphi_i\}_{i\in\mathbb{Z}}$ serves as an orthonormal basis for $L_2(\Lambda^l)$; that is,

$$\{\varphi_\alpha\}_{\alpha\in\mathbb{Z}^l} \quad \text{where} \quad \varphi_\alpha(x) = \prod_{i=1}^{l} \varphi_{\alpha_i}(x_i),\ x \in \Lambda^l$$

serves as an orthonormal basis (so we have $\forall \alpha, \rho \in \mathbb{Z}^l,\ \langle \varphi_\alpha, \varphi_\rho \rangle = I_{\{\alpha=\rho\}}$).

Let $P \in \mathcal{I} \subseteq L_2(\Lambda^l)$, then

$$p(x) = \sum_{\alpha\in\mathbb{Z}^l} a_\alpha(P)\varphi_\alpha(x) \text{ where} \tag{6}$$

$$a_\alpha(P) = \langle \varphi_\alpha, p \rangle = \int_{\Lambda^l} \varphi_\alpha(z)\mathrm{d}P(z) \ \in \mathbb{R}.$$

where $p(x)$ denotes the probability density function of the distribution P.

Suppose that the projection coefficients $a(P) = \{a_\alpha(P)\}_{\alpha\in\mathbb{Z}^l}$ are as follows for $P \in \mathcal{I}$:

$$\mathcal{I} = \{P : a(P) \in \Theta_l(\nu, \gamma, A),\ \|P\|_2^2 \le A\} \quad \text{where} \tag{7}$$

$$\Theta_l(\nu, \gamma, A) = \left\{ \{a_\alpha\}_{\alpha\in\mathbb{Z}^l} : \sum_{\alpha\in\mathbb{Z}^l} a_\alpha^2 \kappa_\alpha^2(\nu,\gamma) < A \right\},$$

$$\kappa_\alpha^2(\nu,\gamma) = \sum_{i=1}^{l} (\nu_i|\alpha_i|)^{2\gamma_i} \text{ for } \nu_i, \gamma_i, A > 0.$$

See [3, 6] for other analyses with this type of assumption. The assumption in (7) will control the tail-behavior of projection coefficients and allow us to effectively estimate $P \in \mathcal{I}$ using a finite number of projection coefficients on the empirical distribution of a sample.

Given a sample $\mathcal{X}_i = \{X_{i1}, \ldots, X_{in_i}\}$ where $X_{ij} \overset{iid}{\sim} P_i \in \mathcal{I}$, let $\widehat{P}_i$ be the empirical distribution of $\mathcal{X}_i$; i.e. $\widehat{P}_i(X = X_{ij}) = \frac{1}{n_i}$. Our estimator for p_i will be:

$$\tilde{p}_i(x) = \sum_{\alpha\,:\,\kappa_\alpha(\nu,\gamma)\le t} a_\alpha(\widehat{P}_i)\varphi_\alpha(x) \quad \text{where} \tag{8}$$

$$a_\alpha(\widehat{P}_i) = \int_{\Lambda^l} \varphi_\alpha(z)\mathrm{d}\widehat{P}_i(z) = \frac{1}{n_i}\sum_{j=1}^{n_i} \varphi_\alpha(X_{ij}). \tag{9}$$

Choosing t optimally[1] can be shown to lead to $\mathbb{E}[\|\tilde{p}_i - p_i\|_2^2] = O(n_i^{-\frac{2}{2+\gamma^{-1}}})$, where $\gamma^{-1} = \sum_{j=1}^{l} \gamma_j^{-1}$, $n_i \to \infty$ [12].

Next, we shall use random basis functions from Random Kitchen Sinks (RKS) [17] to compute our estimate of the response. [17] shows that if one has a shift-invariant kernel K (in particular we consider the RBF kernel $K(x) = \exp(-x^2/2)$) then for $x, y \in \mathbb{R}^d$:

$$K(\|x-y\|_2/\sigma) \approx z(x)^T z(y), \text{ where} \tag{10}$$

$$z(x) \equiv$$

$$\sqrt{\tfrac{2}{D}}\left[\cos(\omega_1^T x + b_1) \cdots \cos(\omega_D^T x + b_D)\right]^T \tag{11}$$

with $\omega_i \overset{iid}{\sim} \mathcal{N}(0, \sigma^{-2}I_d)$, $b_i \overset{iid}{\sim} \mathrm{Unif}[0, 2\pi]$ Let $M_t = \{\alpha : \kappa_\alpha(\nu,\gamma) \le t\} = \{\alpha_1, \ldots, \alpha_S\}$. First note that:

$$\begin{aligned}
\langle \tilde{p}_i, \tilde{p}_j \rangle &= \left\langle \sum_{\alpha\in M_t} a_\alpha(\widehat{P}_i)\varphi_\alpha, \sum_{\alpha\in M_t} a_\alpha(\widehat{P}_j)\varphi_\alpha \right\rangle \\
&= \sum_{\alpha\in M_t}\sum_{\beta\in M_t} a_\alpha(\widehat{P}_i)a_\beta(\widehat{P}_j)\langle \varphi_\alpha, \varphi_\beta \rangle \\
&= \sum_{\alpha\in M_t} a_\alpha(\widehat{P}_i)a_\alpha(\widehat{P}_j) \\
&= \left\langle \vec{a}_t(\widehat{P}_i), \vec{a}_t(\widehat{P}_j) \right\rangle,
\end{aligned}$$

where $\vec{a}_t(\widehat{P}_i) = (a_{\alpha_1}, \ldots, a_{\alpha_s})$, $M_t = \{\alpha_1, \ldots, \alpha_s\}$, and the last inner product is the vector dot product. Thus,

$$\|\tilde{p}_i - \tilde{p}_j\|_2 = \left\|\vec{a}_t(\widehat{P}_i) - \vec{a}_t(\widehat{P}_j)\right\|_2,$$

where the norm on the LHS is the L_2 norm and the ℓ_2 on the RHS.

Consider a fixed σ. Let $\omega_i \overset{iid}{\sim} \mathcal{N}(0, \sigma^{-2}I_s)$, $b_i \overset{iid}{\sim} \mathrm{Unif}[0, 2\pi]$, be fixed. Let $K_\sigma(x) = K(x/\sigma)$. Then,

$$\begin{aligned}
\sum_{i=1}^{N} \theta_i K_\sigma(\|\tilde{p}_i - \tilde{p}_0\|_2) &\approx \sum_{i=1}^{N} \theta_i z(\vec{a}_t(\widehat{P}_i))^T z(\vec{a}_t(\widehat{P}_0)) \\
&= \left(\sum_{i=1}^{N} \theta_i z(\vec{a}_t(\widehat{P}_i))\right)^T z(\vec{a}_t(\widehat{P}_0)) \\
&= \psi^T z(\vec{a}_t(\widehat{P}_0))
\end{aligned} \tag{12}$$

where $\psi = \sum_{i=1}^{N} \theta_i z(\vec{a}_t(\widehat{P}_i)) \in \mathbb{R}^s$. Hence, we consider estimators of the form (12); that is, we consider linear estimators in the non-linear space induced by $z(\vec{a}_t(\cdot))$.

[1] See appendix for details.

In particular, we consider the OLS estimator using the data-set $\{(z(\vec{a}_t(\widehat{P}_i)), Y_i)\}_{i=1}^N$:

$$\hat{f}(\tilde{P}_0) \equiv \hat{\psi}^T z(\vec{a}_t(\widehat{P}_0)) \text{ where} \tag{13}$$

$$\hat{\psi} \equiv \arg\min_\beta \|\vec{Y} - \mathbf{Z}\beta\|_2^2 \tag{14}$$

$$= (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\vec{Y} \tag{15}$$

for $\vec{Y} = (Y_1, \ldots, Y_N)^T$, and with $\mathbf{Z}$ being the $N \times D$ matrix: $\mathbf{Z} = [z(\vec{a}_t(\widehat{P}_1)) \cdots z(\vec{a}_t(\widehat{P}_N))]^T$.

3.1 Evaluation Computational Complexity

We see that after computing $\hat{\psi}$, evaluating our estimator on a new distribution P_0 amounts to taking an inner product with a $D \times 1$ vector. Including the time required for computing $z(\vec{a}_t(\widehat{P}_0))$, the computation required for the evaluation, $\hat{f}(\tilde{P}_0) = \hat{\psi}^T z(\vec{a}_t(\widehat{P}_0))$, is: one, the time for evaluating the projection coefficients $\vec{a}_t(\widehat{P}_1)$, $O(sn)$; two, the time to compute the RKS features $z(\cdot)$, $O(Ds)$; three, the time to compute the inner product, $\langle\hat{\psi}, \cdot\rangle$, $O(D)$. Hence, the total time is $O(D + Ds + sn)$. We'll see that $D = O(n\log(n))$ and $s = O(n)$ hence the total run-time for evaluating $\hat{f}(\tilde{P}_0)$ is $O(n^2\log(n))$. Since we are considering data-sets where the number of instances N far outnumbers the number of points per sample set n, $O(n^2\log(n))$ is a substantial improvement over $O(Nn^2)$.

3.2 Ridge Double-Basis Estimator

We note that a straightforward extension to the Double-Basis estimator is to use a ridge regression estimate on features $z(\vec{a}_t(\cdot))$ rather than a OLS estimate. That is, for $\lambda \geq 0$ let

$$\hat{\psi}_\lambda^T \equiv \arg\min_\beta \|\vec{Y} - \mathbf{Z}\beta\|_2^2 + \lambda\|\beta\|_2 \tag{16}$$

$$= (\mathbf{Z}^T\mathbf{Z} + \lambda I)^{-1}\mathbf{Z}^T\vec{Y}. \tag{17}$$

Clearly the Ridge Double-Basis estimator is still evaluated via a dot product with $\hat{\psi}_\lambda^T$, and our above complexity analysis holds. Furthermore, we note that the Double-Basis estimator is a special case of the Ridge Double-Basis estimator with $\lambda = 0$.

4 Theory

4.1 Assumptions

We shall assume the following:

A.1 *Sobolev Input Distributions.* Suppose that (7) holds.

A.2 *RKHS Mapping.* We shall assume that $f \in \mathcal{F}(\sigma, B)$ for $f : \mathcal{I} \mapsto \mathbb{R}$, where $\sigma, B \in \mathbb{R}$ and

$$\mathcal{F}(\sigma, B) = \Big\{ f : f(P) = \sum_{i=1}^\infty \theta_i K_\sigma(G_i, P), \tag{18}$$

$$\text{where } G_i \in \mathcal{I},\ \|\theta\|_1 \leq B \Big\}. \tag{19}$$

Here we take $K_\sigma(G_i, P) = K_\sigma(\|g_i - p\|_2) = K(\|g_i - p\|_2/\sigma)$ to be a shift-invariant kernel. In particular, we take K to be the RBF kernel: $K(x) = \exp(-x^2/2)$. Note further that:

$$|K(x) - K(x')| \leq e^{-\frac{1}{2}}|x - x'|. \tag{20}$$

A.3 *Input Sample Set Sizes.* Suppose that $\forall i\ |\mathcal{X}_i| \asymp n$.

4.2 Convergence Rate

Since by **A.2** we have that $|f(P)| \leq B$, we consider an upperbound for the risk of a truncated version of our estimator (13). Let $T_B(x) \equiv \operatorname{sign}(x)\min(|x|, B)$. For readability, let $Z(P) = z(\vec{a}_t(P))$. Let a small real $\delta > 0$ be fixed. We look to show that:

Theorem 4.1.

$$\mathbb{E}\left[\left(T_B\left(\hat{\psi}^T Z(\widehat{P}_0)\right) - f(P_0)\right)^2\right] = O\left(n^{-1/(2+\gamma^{-1})}\right) + O\left(\frac{n\log(n)\log(N)}{N}\right)$$

with probability at least $1 - \delta$.

Roughly speaking, our proof will work as follows: first, we show that a population optimal linear model in the non-linear features $Z(\cdot)$ is close to the function f; then we will show that a population optimal linear model is close to the OLS (sample optimal) linear model.

Thus, we proceed to show that predictions from the optimal linear model using $Z(P_0)$ is close to $f(P_0)$, that is:

$$\frac{1}{2}\mathbb{E}_{P_0}\left[\left(f(P_0) - \beta^T Z(\widehat{P}_0)\right)^2\right]$$

is small, where β is an optimal weight vector. Note that β minimizes:

$$\mathbb{E}\left[\left(Y_0 - \beta^T Z(\widehat{P}_0)\right)^2\right] = \tag{21}$$

$$\mathbb{E}\left[Y_0^2\right] - 2\mathbb{E}\left[Y_0 Z(\widehat{P}_0)\right]^T \beta + \beta^T \mathbb{E}\left[Z(\widehat{P}_0) Z(\widehat{P}_0)^T\right]\beta.$$

Let

$$\varsigma_i \equiv \sum_{j=1}^\infty \theta_j \left(K_\sigma(g_j, p_i) - Z(G_j)^T Z(\widehat{P}_i)\right). \tag{22}$$

Furthermore, note that:

$$Y_i = f(p_i) + \epsilon_i = \sum_{j=1}^{\infty} \theta_j K_\sigma\left(g_j, p_i\right) + \epsilon_i. \tag{23}$$

Let

$$\bar{g}_i = \sum_{\alpha\in M_t} a_\alpha(G_i)\varphi_\alpha(x).$$

Also, let $\vec{a}_t(G_j) = (a_{\alpha_1}(G_j),\dots,a_{\alpha_S}(G_j))^T$. When using kitchen sinks, we will see that Y is approximately a linear model. Precisely,

$$\begin{aligned}Y_i =& \sum_{j=1}^{\infty} \theta_j Z(G_j)^T Z(\widehat{P}_i) + \varsigma_i + \epsilon_i \\ =& \psi^T Z(\widehat{P}_i) + \varsigma_i + \epsilon_i,\end{aligned}$$

where $\psi = \sum_{i=1}^{\infty} \theta_i Z(G_i)$. First we prove the following bound for the error using the optimal linear model β:

Lemma 4.2.

$$\mathbb{E}_{P_0}\left[\left(f(P_0) - \beta^T Z(\widehat{P}_0)\right)^2\right] \le \mathbb{E}_{P_0}\left[\varsigma_0^2\right] + 4B\sqrt{\mathbb{E}_{P_0}\left[\varsigma_0^2\right]}$$

Proof. Since (21) is a quadratic function bounded below, an optimal β may be found by satisfying stationarity (taking the gradient (21) and setting to zero). We take $\beta = \Sigma^+\Sigma_Y$ where $\Sigma = \mathbb{E}[Z(\widehat{P}_0)Z(\widehat{P}_0)^T]$ is the uncentered covariance matrix, Σ^+ is its Moore-Penrose inverse, and $\Sigma_Y = \mathbb{E}[Y_0 Z(\widehat{P}_0)]$ is the vector of uncentered covariances to the response[2]. Hence,

$$\begin{aligned}
&\frac{1}{2}\mathbb{E}_{P_0}\left[\left(f(P_0) - \beta^T Z(\widehat{P}_0)\right)^2\right] \\
&= \frac{1}{2}\mathbb{E}_{P_0}\left[\left(f(P_0) - \Sigma_Y^T\Sigma^+ Z(\widehat{P}_0)\right)^2\right] \\
&= \frac{1}{2}\mathbb{E}_{P_0}\left[(f(P_0))^2\right] - \Sigma_Y^T\Sigma^+\mathbb{E}_{P_0}\left[f(P_0)Z(\widehat{P}_0)\right] \\
&\quad + \frac{1}{2}\Sigma_Y^T\Sigma^+\mathbb{E}_{P_0}\left[Z(\widehat{P}_0)Z(\widehat{P}_0)^T\right]\Sigma^+\Sigma_Y \\
&= \frac{1}{2}\mathbb{E}_{P_0}\left[\left(\psi^T Z(\widehat{P}_0) + \varsigma_0\right)^2\right] \\
&\quad - \Sigma_Y^T\Sigma^+\mathbb{E}_{P_0,\epsilon_0}\left[(f(P_0)+\epsilon_0)Z(\widehat{P}_0)\right] \\
&\quad + \frac{1}{2}\Sigma_Y^T\Sigma^+\Sigma\Sigma^+\Sigma_Y \\
&= \frac{1}{2}\psi^T\Sigma\psi + \mathbb{E}_{P_0}\left[\varsigma_0 z(\widehat{P}_0)^T\right]\psi + \frac{1}{2}\mathbb{E}_{P_0}\left[\varsigma_0^2\right] \\
&\quad - \frac{1}{2}\Sigma_Y^T\Sigma^+\Sigma_Y.
\end{aligned}$$

Also,

$$\begin{aligned}\Sigma_Y &= \mathbb{E}_{P_0,\epsilon_0}\left[(\psi^T Z(\widehat{P}_0))Z(\widehat{P}_0) + \varsigma_0 Z(\widehat{P}_0) + \epsilon_0 Z(\widehat{P}_0)\right] \\
&= \Sigma\psi + \mathbb{E}_{P_0}\left[\varsigma_0 Z(\widehat{P}_0)\right].\end{aligned}$$

Thus,

$$\begin{aligned}
&\Sigma_Y^T\Sigma^+\Sigma_Y \\
=&(\psi^T\Sigma + \mathbb{E}_{P_0}\left[\varsigma_0 Z(\widehat{P}_0)^T\right])\Sigma^+(\Sigma\psi + \mathbb{E}_{P_0}\left[\varsigma_0 Z(P_0)\right]) \\
=&\psi^T\Sigma\Sigma^+\Sigma\psi + \mathbb{E}_{P_0}\left[\varsigma_0 Z(\widehat{P}_0)^T\right]\Sigma^+\Sigma\psi \\
&+ \psi^T\Sigma\Sigma^+\mathbb{E}_{P_0}\left[\varsigma_0 Z(\widehat{P}_0)\right] \\
&+ \mathbb{E}_{P_0}\left[\varsigma_0 Z(\widehat{P}_0)^T\right]\Sigma^+\mathbb{E}_{P_0}\left[\varsigma_0 Z(\widehat{P}_0)\right] \\
=&\psi^T\Sigma\psi + 2\mathbb{E}_{P_0}\left[\varsigma_0 Z(\widehat{P}_0)^T\right]\Sigma^+\Sigma\psi \\
&+ \mathbb{E}_{P_0}\left[\varsigma_0 Z(\widehat{P}_0)^T\right]\Sigma^+\mathbb{E}_{P_0}\left[\varsigma_0 Z(\widehat{P}_0)\right].
\end{aligned}$$

Hence,

$$\begin{aligned}
&\frac{1}{2}\mathbb{E}_{P_0}\left[\left(f(P_0) - \beta^T Z(\widehat{P}_0)\right)^2\right] \\
&= \frac{1}{2}\psi^T\Sigma\psi + \mathbb{E}_{P_0}\left[\varsigma_0 Z(\widehat{P}_0)^T\right]\psi + \frac{1}{2}\mathbb{E}_{P_0}\left[\varsigma_0^2\right] \\
&\quad - \frac{1}{2}\psi^T\Sigma\psi - \mathbb{E}_{P_0}\left[\varsigma_0 Z(\widehat{P}_0)^T\right]\Sigma^+\Sigma\psi \\
&\quad - \frac{1}{2}\mathbb{E}_{P_0}\left[\varsigma_0 Z(\widehat{P}_0)^T\right]\Sigma^+\mathbb{E}_{P_0}\left[\varsigma_0 Z(\widehat{P}_0)\right] \\
&\le \frac{1}{2}\mathbb{E}_{P_0}\left[\varsigma_0^2\right] + 4B\sqrt{\mathbb{E}_{P_0}\left[\varsigma_0^2\right]},
\end{aligned} \tag{24}$$

see Appendix for details on the last bound. □

Lemma 4.3.

$$\mathbb{E}_{P_0}\left[\varsigma_0^2\right] = O\left(n^{\frac{-2}{2+\gamma^{-1}}}\right)$$

with probability at least $1-\delta$.

Proof. $|\varsigma_i| \le \sum_{j=1}^{\infty}|\theta_j|\left|K_\sigma\left(g_j,p_i\right) - Z(G_j)^T z(\widehat{P}_i)\right|$ and

$$\begin{aligned}
&\left|K_\sigma\left(g_j,p_i\right) - Z(G_j)^T Z(\widehat{P}_i)\right| \\
&\le \left|K_\sigma\left(g_j,p_i\right) - K_\sigma\left(\bar{g}_j,\tilde{p}_i\right)\right| \\
&+ \left|K_\sigma\left(\bar{g}_j,\tilde{p}_i\right) - Z(G_j)^T Z(\widehat{P}_i)\right|.
\end{aligned}$$

Also, using (20):

$$\begin{aligned}
&\left|K_\sigma\left(g_j,p_i\right) - K_\sigma\left(\bar{g}_j,\tilde{p}_i\right)\right| \\
&\le \frac{e^{-\frac{1}{2}}}{\sigma}\left|\|g_j - p_i\|_2 - \|\bar{g}_j - \tilde{p}_i\|_2\right|.
\end{aligned}$$

Moreover, using the triangle inequality:

$$\left|\|g_j - p_i\|_2 - \|\bar{g}_j - \tilde{p}_i\|_2\right| \le \|g_j - \bar{g}_j\|_2 + \|\tilde{p}_i - p_i\|_2.$$

Thus,

$$\mathbb{E}\left[\left|K_\sigma\left(g_j,p_i\right) - K_\sigma\left(\bar{g}_j,\tilde{p}_i\right)\right|\right]$$

[2] Note that if Σ is nonsingular, $\Sigma^+ = \Sigma^{-1}$ and β is unique.

$$\leq \mathbb{E}\left[\frac{e^{-\frac{1}{2}}}{\sigma}\left(\|g_j - \bar{g}_j\|_2 + \|\tilde{p}_i - p_i\|_2\right)\right]$$
$$= O\left(n^{\frac{1}{2+\gamma^{-1}}}\right),$$

where the last line follows[3] by choosing $t \asymp n^{\frac{1}{2+\gamma^{-1}}}$, and the expectation is w.r.t. $\mathcal{X}_i \sim P_i$, $P_i \sim \Phi$.

Also, note that the dimensionality of $\vec{a}_t(G_i)$ and $\vec{a}_t(\widehat{P}_i)$ is[3] $S = |M(t)| = O(n^{\gamma^{-1}/(2+\gamma^{-1})})$. Let $\mathcal{M} = \{v \in \mathbb{R}^S : \|v\|_2^2 \leq A\}$. Then, $\vec{a}_t(G_j)$, $\vec{a}_t(\widehat{P}_i) \in \mathcal{M}$. Hence, by *Claim 1* in [17]:

$$\mathbb{P}\left[\sup_{u,v\in\mathcal{M}} |K(u,v) - z(u)^T z(v)| \geq \xi\right]$$
$$\leq 2^8 \left(\frac{\sqrt{S}\,\mathrm{diam}(\mathcal{M})}{\sigma\xi}\right)^2 \exp\left(-\frac{D\xi^2}{4(S+2)}\right).$$

Thus, with probability at least $1-\delta$:

$$\sup_{u,v\in\mathcal{M}} |K(u,v) - z(u)^T z(v)| < n^{-\frac{1}{2+\gamma^{-1}}},$$

if we choose D such that:

$$D = \Omega\left(4(S+4)n^{\frac{2}{2+\gamma^{-1}}} \log\left(\delta^{-1}2^{10}ASn^{\frac{2}{2+\gamma^{-1}}}/\sigma^2\right)\right),$$

which is satisfied setting $D \asymp n\log(n)$.

Hence, probability at least $1-\delta$:

$$\frac{1}{2}\mathbb{E}_{P_0}\left[\varsigma_0^2\right]$$
$$\leq \mathbb{E}_{P_0}\left[\left(\sum_{j=1}^{\infty}|\theta_j|\Big(\tfrac{e^{-\frac{1}{2}}}{\sigma}\|g_j - \bar{g}_j\|_2 + \tfrac{e^{-\frac{1}{2}}}{\sigma}\|\tilde{p}_0 - p_0\|_2 + \left|K_\sigma\left(\bar{g}_j, \tilde{p}_0\right) - Z(G_j)^T Z(\widehat{P}_0)\right|\Big)\right)^2\right]$$
$$= \mathbb{E}_{P_0}\left[\left(\sum_{j=1}^{\infty}|\theta_j|\left(\|\tilde{p}_0 - p_0\|_2 + O\left(n^{\frac{1}{2+\gamma^{-1}}}\right)\right)\right)^2\right]$$
$$= \left(\sum_{j=1}^{\infty}|\theta_j|\right)^2 \mathbb{E}_{P_0}\left[\left(\|\tilde{p}_0 - p_0\|_2 + O\left(n^{\frac{1}{2+\gamma^{-1}}}\right)\right)^2\right]$$
$$= O\left(n^{\frac{-2}{2+\gamma^{-1}}}\right).$$

□

Thus, we see that $f(P)$ is close to the linear model in the non-linear spaced induced by the $O(n\log(n))$ features $Z(\cdot)$:

Then, *Theorem 11.3 of* [2] states that the estimated linear predictor $\hat{\beta} \in \mathbb{R}^d$ has an error to the mean conditional response (when truncated) relative an optimal linear predictor β as follows:

$$\mathbb{E}\left[\left(T_B\left(\hat{\beta}^T x\right) - \mathbb{E}\left[y|x\right]\right)^2\right] \leq$$
$$8\mathbb{E}\left[\left(\beta^T x - \mathbb{E}\left[y|x\right]\right)^2\right] + O(\max\{\sigma_\epsilon^2, B^2\}d\log(N)/N). \quad (25)$$

Using our notation we have that:

$$\mathbb{E}\left[\left(T_B\left(\hat{\psi}^T Z(\widehat{P}_0)\right) - f(P_0)\right)^2\right]$$
$$= O\left(n^{-1/(2+\gamma^{-1})}\right) + O\left(\frac{n\log(n)\log(N)}{N}\right), \quad (26)$$

where we have bounded $\mathbb{E}\left[\left(T_B\left(\hat{\beta}^T x\right) - \mathbb{E}\left[y|x\right]\right)^2\right]$ using Lemmas 4.2 and 4.3, giving us our desired rate.

5 Experiments

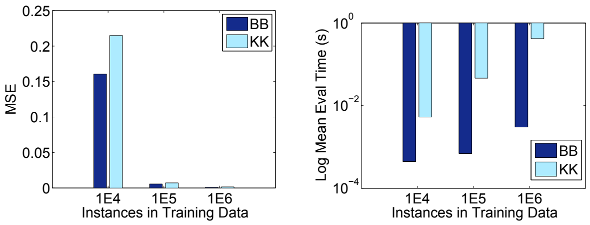
We perform experiments that demonstrate the ability of the Double-Basis estimator to learn distribution-to-real mappings from large training datasets, which can be applied to yield fast, accurate, and useful predictions. We illustrate this on a few statistical estimation tasks, which aim to take a set of samples from a distribution as input and yield some estimated quantity as output. For many such tasks, we can generate large amounts of relevant output quantities and associated input samples synthetically, and can train the Double-Basis estimator on these big datasets, giving us an automated procedure to learn a mapping for these statistical estimation tasks. We will show that, in some cases, this mapping can be more accurate, faster, and more robust than existing statistical procedures.

In all of the following experiments, we train on data of the form $\mathcal{D} = \{(\mathcal{X}_i, Y_i)\}_{i=1}^N$.

5.1 Synthetic Mapping

First, we look to emphasize the computational improvement in evaluation time of the Double-Basis estimator over the Kernel-Kernel estimator using experiments with synthetic data. Our experiments are as follows. We first set $N \in \{1\text{E}4, 1\text{E}5, 1\text{E}6\}$. Then, we generate a random mapping f such that $f(P) = \sum_{i=1}^{10}\theta_i K_\sigma(G_i, P)$. We took $\sigma = 1$, $\theta_i \sim \text{Unif}[-5,5]$, and G_i to be the pdf of a mixture of two truncated Gaussians (each with weight .5) on the interval $[0,1]$, whose mean locations are chosen uniformly at random in $[0,1]$, and whose variance parameters are taken uniformly at random in $[.05, .1]$. For $j = \{1,\ldots,N\}$ we

[3] See Appendix for details.



(a) Estimation Error (b) Estimation Time

Figure 2: Results on predicting synthetic mapping f.

also set P_j to be a randomly generated mixture of two truncated Gaussians as previously described. We then generate Y_i under the the noiseless case, i.e $Y_i = f(P_i)$ (kernel values were computed numerically). Then, we generated $\mathcal{X}_i = \{X_{i1}, \ldots, X_{in}\}$ where $n \propto N^{3/5}$ and $X_{i1} \overset{iid}{\sim} P_i$. $\tilde{P}_i$ was then estimated using the samples $\mathcal{X}_i$.

We compared the performance of both the Double-Basis (BB), and the Kernel-Kernel (KK) estimator on a separate test set of $\mathcal{D}_t = \{(\mathcal{X}_j, Y_j)\}_{j=1}^{N_t}$ where $N_t = 1\text{E}5$, that was generated as $\mathcal{D}$ was. We measured performance in terms of mean squared error (MSE) and mean evaluation time per new query $\mathcal{X}_0$ (Figures 2(a) and 2(b) respectively). One can see that in this case both estimators have similar MSEs, with the BB estimator doing somewhat better in each configuration of the data-set size. However, one can observe a striking difference in the average time to evaluate a new estimate $\hat{f}(\tilde{P})$. Figure 2(b) is presented in a log scale, and illustrates the Kernel-Kernel estimator's lack of scaling on data-set size, N. On the other hand, the Double-Basis estimator is considerbly efficient even at large N and has a speed-up of about ×12, ×67, and ×139 over the Kernel-Kernel estimate for $N = 1\text{E}4, 1\text{E}5, 1\text{E}6$ respectively.

5.2 Choosing k: model selection for Gaussian mixtures

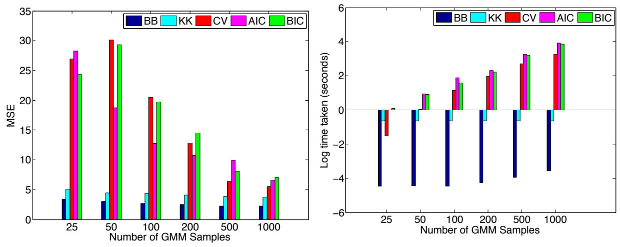
Many common statistical tasks involve producing a mapping from a distribution to a real value, and may be tackled using DRR. One such task is that of model selection, where one is given a set $\mathcal{X}_0 = \{X_{01}, \ldots, X_{0n_0}\}$ drawn from an unknown distribution P and wants to find some parameter that is indicative of the complexity of the true distribution. In other words, the mapping of interest takes in a distribution and outputs a hyperparameter of the distribution that is often illustrative of the distribution's complexity.

In particular, we shall consider the model selection problem of selecting k, the number of components in a Gaussian mixture model (GMM). GMMs are often used in modeling data, however the selection of how many components to use is often a difficult choice. Attempting an MLE fit to training data will lead to choosing $k = n_0$ with one mixture component corresponding to each data-point. Hence, in order to effectively select k, one must fit a GMM for each potential choice of k using an algorithm such as the expectation maximization algorithm (EM) [9], then select the choice of k that optimizes some score. In practice this often becomes computationally expensive. Typically scores used include Akaike information criterion (AIC), Bayesian information criterion (BIC), or a cross-validated data-fitting score on a holdout set (CV). We note that often GMMs are used to cluster data, where each data-point X_{0i} is a assigned to a cluster based on which mixture component most likely generated it. Hence, the problem of selecting the number of mixture components in a GMM is closely related to the problem of selecting the number of clusters to use, which is itself a difficult problem.

Since selecting k in GMMs is a DRR problem, and it is a relatively smooth mapping (that is, similar distributions should have a similar number of components), we hypothesize that one may learn to perform model selection in GMMs using the Double-Basis estimator. Particularly, by using a supervised dataset of {sample-set, k} pairs, the Double-Basis estimator will be able to leverage previously seen data to perform model selection for a new unseen input sample set.

Our experiment proceeds as follows. We can generate our own training data for this task by randomly drawing a value for k (over some bounded range), then drawing 2-dimensional Gaussian mixture parameters for each of the k components[4], and finally drawing samples from each Gaussian. That is, we generate $N = 28,000$ input sample set/k response pairs: $\mathcal{D} = \{(\mathcal{X}_i, k_i)\}_{i=1}^{N}$, where $\mathcal{X}_i = \{X_{i1}, \ldots, X_{in}\}$, $X_{ij} \in \mathbb{R}^2$, $X_{ij} \overset{iid}{\sim} \text{GMM}(k_i)$, $k_i \sim \text{Unif}\{1, \ldots, 10\}$, and $\text{GMM}(k_i)$ is a random GMM generated as follows, for $j = 1, \ldots, k_i$: the prior weights for each component is taken to be $\pi_j = 1/k_i$; the means are $\mu_j \sim \text{Unif}[-5, 5]^2$; and covariances are $\Sigma_j = a^2 A A^T + B$, where $a \sim \text{Unif}[1, 2]$ $A_{uv} \sim \text{Unif}[-1, 1]$, and B is a diagonal 2×2 matrix with $B_{uu} \sim \text{Unif}[0, 1]$. We train and get results using n in the following range: $n \in 10, 25, 50, 200, 500, 1000$. We perform model selection using the mapping learned by the Ridge Double-Basis estimator (16) (denoted BB in experiments), and compare it with model selection via AIC, BIC, and CV. We also compare agasint the Kernel-Kernel (KK) smoother. For all methods we computed the mean squared error between the true and predicted value for k over 2000 test sample sets (Figure 3). We see that the Double-Basis estimator has both the lowest MSE and the lowest average evaluation time for computing a new prediction. In fact, the Double-Basis estimator can carry out the model

[4] See Appendix for figures of typical GMMs.



(a) Estimation Error (b) Estimation Time

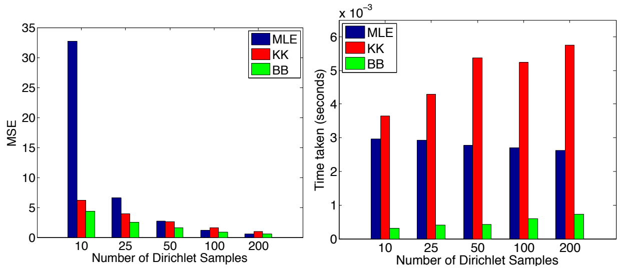
Figure 3: Results on predicting the number of GMM components.

selection prediction orders of magnitude faster than the CV, AIC, or BIC procedures.

5.3 Low Sample Dirichlet Parameter Estimation

Similar to model selection, general parameter point estimation is a statistical task that may be posed as a DRR problem. That is, in parameter estimation one considers a set $\mathcal{X}_0 = \{X_{01}, \ldots, X_{0n_0}\}$ where points are drawn from some distribution $P(\eta_0)$ that is parameterized by η_0, and attempts to estimate η_0. In particular, we use DRR and the Double-Basis estimator to perform parameter estimation for Dirichlet distributions. The Dirichlet distribution is a family of continuous, multivariate distributions parameterized by a vector $\alpha \in \mathbb{R}^d_+$, with support over the d-simplex. Since every element of the support sums to one, the Dirichlet is often used to model distributions over proportion data. As before, we hypothesize that the Double-Basis estimator will serve as a way to leverage previously seen sample sets to help perform parameter estimation for new unseen sets. Effectively, our estimator will be able to "boost" the sample-size of a new input sample set by making use of what was learned on previously seen labeled sample sets.

Maximum likelihood parameter estimation for α, given a set of Dirichlet samples, is often performed via iterative optimization algorithms, such as gradient ascent or Newton's method [8], as a closed form solution for the MLE does not appear to exist in the literature. In this experiment, we aim to use DDR as a new method for Dirichlet parameter estimation. In particular, we generate samples from Dirichlet distributions with parameter values in a prespecified range, and use these as training data to learn a mapping from data samples to Dirichlet α parameter values.



(a) Estimation Error (b) Estimation Time

Figure 4: Results predicting Dirichlet parameters.

In our experiments, we first fix the range of α values to be constrained such that the i^{th} component $\alpha_i \in [0.1, 10]$. For each 28,000 training instances, we uniformly sample a new α parameter vector within this range, and then generate n points from the associated Dirichlet(α) distribution, where $n \in \{10, 25, 50, 200, 500, 1000\}$. We compare the Ridge Double-Basis estimator (16) against a Newtons-method procedure for maximum likelihood estimation (MLE) from the fastfit toolbox [7], and again against the Kernel-Kernel smoother. For all methods, for each n, we compute the mean squared error between the true and the estimated α parameter. We also record the time taken to perform the parameter estimation in each case. Results are shown in Figure 4. We see that the Double-Basis estimator achieves the lowest MSE in all cases, and has the lowest average compute time. It is worth noting that the Double-Basis estimator performs particularly well relative to the MLE in cases where the sample size is low. We envision that Double-Basis estimator is particularly well suited for cases where one hopes to quickly, and in an automatic fashion, construct an estimator that can achieve highly accurate results for a statistical estimation problem for which an optimal estimator might be hard to derive analytically.

6 Conclusion

In conclusion, this paper presents a new estimator, the Double-Basis (BB) estimator, for performing distribution to real regression. In particular, this estimator scales independently of N (the number input sample-set/response pairs) in a large dataset for performing evaluations for response predictions. This is a great improvement over the linear scaling with N that the Kernel-Kernel (KK) estimator has and allows one to explore DRR in new domains with large collections of distributions, such as astronomy and finance. Furthermore, we prove an efficient upper bound on the risk for the BB estimator. Also, we empirically showed the improved scaling of the Double-Basis estimator, as well improvements in risk over the KK estimator. It is worth noting that while the BB estimator regresses a mapping in a nonlinear space (induced by RKS features), the KK estimator is outputs only a weighted average of training set responses.

Acknowledgements

This work is supported in part by NSF grants IIS1247658 and IIS1250350.

References

[1] F. Ferraty and P. Vieu, *Nonparametric functional data analysis: theory and practice*, Springer, 2006.

[2] László Györfi, *A distribution-free theory of nonparametric regression*, Springer, 2002.

[3] Y. Ingster and N. Stepanova, *Estimation and detection of functions from anisotropic sobolev classes*, Electronic Journal of Statistics **5** (2011), 484–506.

[4] T.S. Jaakkola, D. Haussler, et al., *Exploiting generative models in discriminative classifiers*, Advances in neural information processing systems (1999), 487–493.

[5] T. Jebara, R. Kondor, and A. Howard, *Probability product kernels*, The Journal of Machine Learning Research **5** (2004), 819–844.

[6] B. Laurent, *Efficient estimation of integral functionals of a density*, The Annals of Statistics **24** (1996), no. 2, 659–681.

[7] Thomas Minka, *The fastfit matlab toolbox*.

[8] Thomas Minka, *Estimating a dirichlet distribution*, Technical report, MIT, 2000.

[9] Todd K Moon, *The expectation-maximization algorithm*, Signal processing magazine, IEEE **13** (1996), no. 6, 47–60.

[10] P.J. Moreno, P. Ho, and N. Vasconcelos, *A kullback-leibler divergence based kernel for svm classification in multimedia applications*, Advances in Neural Information Processing Systems **16** (2003), 1385–1393.

[11] K. Muandet, B. Schölkopf, K. Fukumizu, and F. Dinuzzo, *Learning from distributions via support measure machines*, arXiv preprint arXiv:1202.6504 (2012).

[12] M. Nussbaum, *On optimal filtering of a function of many variables in white gaussian noise*, Problemy Peredachi Informatsii **19** (1983), no. 2, 23–29.

[13] Junier B Oliva, Barnabás Póczos, and Jeff Schneider, *Distribution to distribution regression*.

[14] B. Póczos, L. Xiong, D.J. Sutherland, and J. Schneider, *Nonparametric kernel estimators for image classification*, Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE, 2012, pp. 2989–2996.

[15] Barnabás Póczos, Alessandro Rinaldo, Aarti Singh, and Larry Wasserman, *Distribution-free distribution regression*, AISTATS (2013).

[16] Barnabás Póczos, Liang Xiong, and Jeff Schneider, *Nonparametric divergence estimation with applications to machine learning on distributions*, arXiv preprint arXiv:1202.3758 (2012).

[17] Ali Rahimi and Benjamin Recht, *Random features for large-scale kernel machines*, Advances in neural information processing systems, 2007, pp. 1177–1184.

[18] J.O. Ramsay and B.W. Silverman, *Applied functional data analysis: methods and case studies*, vol. 77, Springer New York:, 2002.

[19] A. Smola, A. Gretton, L. Song, and B. Schölkopf, *A hilbert space embedding for distributions*, Algorithmic Learning Theory, Springer, 2007, pp. 13–31.

[20] Alexandre B Tsybakov, *Introduction to nonparametric estimation*, Springer, 2008.