Linear-time Learning on Distributions with Approximate Kernel Embeddings

Dougal J. Sutherland[*] DSUTHERL@CS.CMU.EDU
Carnegie Mellon University

Junier B. Oliva[*] JOLIVA@CS.CMU.EDU
Carnegie Mellon University

Barnabás Póczos BAPOCZOS@CS.CMU.EDU
Carnegie Mellon University

Jeff Schneider SCHNEIDE@CS.CMU.EDU
Carnegie Mellon University

Editor: Dmitry Storcheus

Abstract

Many interesting machine learning problems are best posed by considering instances that are distributions, or sample sets drawn from distributions. Previous work devoted to machine learning tasks with distributional inputs has done so through pairwise kernel evaluations between pdfs (or sample sets). While such an approach is fine for smaller datasets, the computation of an $N \times N$ Gram matrix is prohibitive in large datasets. Recent scalable estimators that work over pdfs have done so only with kernels that use Euclidean metrics, like the L_2 distance. However, there are a myriad of other useful metrics available, such as total variation, Hellinger distance, and the Jensen-Shannon divergence. This work develops the first random features for pdfs whose dot product approximates kernels using these non-Euclidean metrics, allowing estimators using such kernels to scale to large datasets by working in a primal space, without computing large Gram matrices. We provide an analysis of the approximation error in using our proposed random features and show empirically the quality of our approximation both in estimating a Gram matrix and in solving learning tasks in real-world and synthetic data.

Keywords: Learning on Distributions, Approximate Kernel Embeddings, Nonparametric Statistics

1. Introduction

As machine learning matures, focus has shifted towards datasets with richer, more complex instances. For example, a great deal of effort has been devoted to learning functions on vectors of a large fixed dimension. While complex static vector instances are useful in a myriad of applications, many machine learning problems are more naturally posed by considering instances that are distributions, or sets drawn from distributions. Political scientists can learn a function from community demographics to vote percentages to understand who supports a candidate (Flaxman et al., 2015). The mass of dark matter halos can be inferred from the velocity of galaxies in a cluster (Ntampaka et al., 2015). Expensive expectation propagation messages can be sped up by learning a "just-in-time" regression model (Jitkrittum et al., 2015). All of these applications are aided by

∗. These two authors contributed equally.

working directly over sets drawn from the distribution of interest, rather than having to develop a per-problem ad-hoc set of summary statistics.

Distributions are inherently infinite-dimensional objects, since in general they require an infinite number of parameters for their exact representation. Hence, it is not immediate how to extend traditional finite vector technique machine learning techniques to distributional instances. However, recent work has provided various approaches for dealing with distributional data in a nonparametric fashion. For example, regression from distributional covariates to real or distributional responses is possible via kernel smoothing (Póczos et al., 2012a; Oliva et al., 2013), and many learning tasks can be solved with RKHS approaches (Muandet et al., 2012; Póczos et al., 2012b). A major shortcoming of both approaches is that they require computing N kernel evaluations per prediction, where N is the number of training instances in a dataset. Often, this implies that one must compute a $N \times N$ Gram matrix of pairwise kernel evaluations. Such approaches fail to scale to datasets where the number of instances N is very large. Another shortcoming of these approaches is that they are often based on Euclidean metrics, either working over a linear kernel, or one based on the L_2 distance over distributions. While such kernels are useful in certain applications, better performance can sometimes be obtained by considering non-Euclidean based kernels. To this end, Póczos et al. (2012b) use a kernel based on Rényi divergences; however, this kernel is not positive semi-definite (PSD), leading to even higher computational cost and other practical issues.

This work addresses these major shortcomings by developing an embedding of random features for distributions. The dot product of the random features for two distributions will approximate kernels based on various distances between densities (see Figure 1). With this technique, we can approximate kernels based on total variation, Hellinger, and Jensen-Shannon divergences, among others. Since there is then no need to compute a Gram matrix, one will be able to use these kernels while still scaling to datasets with a large number of instances using primal-space techniques. We provide an approximation bound for the embeddings, and demonstrate the efficacy of the embeddings on both real-world and synthetic data. To the best of our knowledge, this work provides the first non-discretized embedding for non-L_2 kernels for probability density functions.

2. Related Work

The two main lines of relevant research are the development of kernels on probability distributions and explicit approximate embeddings for scalable kernel learning.

Learning on distributions In computer vision, the popular "bag of words" model (Leung and Malik, 2001) represents a distribution by quantizing it onto codewords (usually by running k-means on all, or many, of the input points from all sets), then compares those histograms with some kernel (often exponentiated χ^2).

Another approach estimates a distance between distributions, often the L_2 distance or Kullback-Leibler (KL) divergence, parametrically (Jaakkola and Haussler, 1998; Moreno et al., 2003; Jebara et al., 2004) or nonparametrically (Sricharan et al., 2013; Krishnamurthy et al., 2014). The distance

Figure 1: We approximate kernels of densities p_i, p_j with features of samples $\chi_i \overset{iid}{\sim} p_i, \chi_j \overset{iid}{\sim} p_j$.

can then be used in kernel smoothing Póczos et al. (2012a); Oliva et al. (2013) or Mercer kernels Moreno et al. (2003); Kondor and Jebara (2003); Jebara et al. (2004); Póczos et al. (2012b).

These approaches can be powerful, but usually require computing an $N \times N$ matrix of kernel evaluations, which can be infeasible for large datasets. Using these distances in Mercer kernels faces an additional challenge, which is that the estimated Gram matrix may not be PSD, due to estimation error or because some divergences do not induce a PSD kernel. In general this is remedied by replacing the Gram matrix with a "nearby" PSD one. Typical approaches involve eigendecomposing the Gram matrix, which costs $O(N^3)$ computation and also presents challenges for traditional inductive learning, where the test points are not known at training time (Chen et al., 2009).

One way to alleviate the scaling problem is the Nyström extension (Williams and Seeger, 2001), in which some columns of the Gram matrix are used to estimate the remainder. In practice, one frequently must compute many columns, and methods to make the result PSD are known only for mildly-indefinite kernels (Belongie et al., 2002).

Another approach is to represent a distribution by its mean RKHS embedding under some kernel k. The RKHS inner product is known as the *mean map kernel* (MMK), and the distance the *maximum mean discrepancy* (MMD) (Gretton et al., 2009; Muandet et al., 2012; Szabó et al., 2015). When k is the common RBF kernel, the MMK estimate is proportional to an L_2 inner product between Gaussian kernel density estimates.

Approximate embeddings Recent interest in approximate kernel embeddings was spurred by the "random kitchen sink" (RKS) embedding (Rahimi and Recht, 2007), which approximates shift-invariant kernels K on $\mathbb{R}^\ell$ by sampling their Fourier transform. Le et al. (2013) gave an approximation which is faster for large ℓ, which Yang et al. (2014) use for kernel learning; Yu et al. (2015) also consider kernel learning based on embeddings. Storcheus et al. (2015) give learning theory for this setting.

A related line of work considers additive kernels, of the form $K(x, y) = \sum_{j=1}^{\ell} \kappa(x_j, y_j)$, usually defined on $\mathbb{R}_{\geq 0}^{\ell}$ (e.g. histograms). Maji and Berg (2009) construct an embedding for the intersection kernel $\sum_{j=1}^{\ell} \min(x_j, y_j)$ via step functions. Vedaldi and Zisserman (2010) consider any homogeneous κ, so that $\kappa(tx, ty) = t\,\kappa(x, y)$, which also allows them to embed histogram kernels such as the additive χ^2 kernel and Jensen-Shannon divergence. Their embedding uses the same fundamental result of Fuglede (2005) as ours; we expand to the continuous rather than the discrete case. Vempati et al. (2010) later apply RKS embeddings to obtain generalized RBF kernels (1).

For embeddings of kernels on input spaces other than $\mathbb{R}^\ell$, the RKS embedding extends naturally to locally compact abelian groups (Li et al., 2010). Oliva et al. (2014) embedded an estimate of the L_2 distance between continuous densities via orthonormal basis functions. An embedding for the base kernel k also gives a simple embedding for the mean map kernel (Flaxman et al., 2015; Jitkrittum et al., 2015; Lopez-Paz et al., 2015; Sutherland and Schneider, 2015).

3. Embedding Information Theoretic Kernels

For a broad class of distributional distances d, including many common and useful information theoretic divergences, we consider generalized RBF kernels of the form

$$K(p, q) = \exp\left(-\frac{1}{2\sigma^2} d^2(p, q)\right). \tag{1}$$

We will construct features $z(A(\cdot))$ such that $K(p,q) \approx z(A(p))^\mathsf{T} z(A(q))$ as follows:

1. We define a random function ψ such that $d(p,q) \approx \|\psi(p) - \psi(q)\|$, where $\psi(p)$ is a function from $[0,1]^\ell$ to $\mathbb{R}^{2M}$. Thus the metric space of densities with distance d is approximately embedded into the metric space of $2M$-dimensional L_2 functions.

2. We use orthonormal basis functions to approximately embed smooth L_2 functions into finite vectors in $\mathbb{R}^{|V|}$. Combined with the previous step, we obtain features $A(p) \in \mathbb{R}^{2M|V|}$ such that d is approximated by Euclidean distances between the A features.

3. We use the RKS embedding $z(\cdot)$ so that inner products between $z(A(\cdot))$ features, in $\mathbb{R}^D$, approximate $K(p,q)$.

We can thus approximate the powerful kernel K without computing an expensive Gram matrix.

3.1 Homogeneous Density Distances (HDDs)

We consider kernels based on metrics which we term homogeneous density distances (HDDs):

$$d^2(p,q) = \int_{[0,1]^\ell} \kappa(p(x), q(x))\,\mathrm{d}x, \tag{2}$$

where $\kappa(x,y) : \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}_+$ is a negative-type kernel, i.e. a squared Hilbertian metric, and $\kappa(tx,ty) = t\kappa(x,y)$ for all $t > 0$. Table 1 shows a few important instances. Note that we assume the support of the distributions is contained within $[0,1]^\ell$.

Name	$\kappa(p(x), q(x))$	$\mathrm{d}\mu(\lambda)$
Jensen-Shannon (JS)	$\frac{p(x)}{2}\log\left(\frac{2p(x)}{p(x)+q(x)}\right) + \frac{q(x)}{2}\log\left(\frac{2q(x)}{p(x)+q(x)}\right)$	$\frac{\mathrm{d}\lambda}{\cosh(\pi\lambda)(1+\lambda^2)}$
Squared Hellinger (H^2)	$\frac{1}{2}\left(\sqrt{p(x)} - \sqrt{q(x)}\right)^2$	$\frac{1}{2}\,\delta(\lambda = 0)\,\mathrm{d}\lambda$
Total Variation (TV)	$\lvert p(x) - q(x)\rvert$	$\frac{2}{\pi}\frac{1}{1+4\lambda^2}\,\mathrm{d}\lambda$

Table 1: Squared HDDs.

We then use these distances in a generalized RBF kernel (1). d is a Hilbertian metric (Fuglede, 2005), so K is positive definite (Haasdonk and Bahlmann, 2004). Note that we use the $\sqrt{\mathrm{TV}}$ metric, even though TV is itself a metric.

We can approximate the expectation with an empirical mean. Letting $\lambda_j \overset{iid}{\sim} \frac{\mu}{Z}$ for $j \in \{1,\ldots,M\}$,

$$\kappa(x,y) \approx \frac{1}{M}\sum_{j=1}^{M} |g_{\lambda_j}(x) - g_{\lambda_j}(y)|^2.$$

Hence, using $\mathfrak{R}, \mathfrak{I}$ to denote the real and imaginary parts:

$$\begin{aligned} d^2(p,q) &= \int_{[0,1]^\ell} \kappa(p(x), q(x))\,\mathrm{d}x \\ &= \int_{[0,1]^\ell} \mathbb{E}_{\lambda\sim\frac{\mu}{Z}} |g_\lambda(p(x)) - g_\lambda(q(x))|^2\,\mathrm{d}x \\ &\approx \frac{1}{M}\sum_{j=1}^{M} \int_{[0,1]^\ell} \big(\big(\mathfrak{R}(g_{\lambda_j}(p(x))) - \mathfrak{R}(g_{\lambda_j}(q(x)))\big)^2 + \big(\mathfrak{I}(g_{\lambda_j}(p(x))) - \mathfrak{I}(g_{\lambda_j}(q(x)))\big)^2 \big)\mathrm{d}x \\ &= \|\psi(p) - \psi(q)\|^2, \end{aligned} \tag{3}$$

where we have defined $p^R_{\lambda_j}(x) = \mathfrak{R}(g_{\lambda_j}(p(x)))$, $p^I_{\lambda_j}(x) = \mathfrak{I}(g_{\lambda_j}(p(x)))$ and

$$[\psi(p)](x) = \frac{1}{\sqrt{M}}\big(p^R_{\lambda_1}(x), \ldots, p^R_{\lambda_M}(x), p^I_{\lambda_1}(x), \ldots, p^I_{\lambda_M}(x)\big).$$

Hence, the HDD between densities p and q is approximately the L_2 distance from $\psi(p)$ to $\psi(q)$, where ψ maps a function $f : [0,1]^\ell \mapsto \mathbb{R}$ to a vector-valued function $\psi(f) : [0,1]^\ell \mapsto \mathbb{R}^{2M}$ of λ functions. M can typically be quite small, since the kernel it approximates is one-dimensional.

3.2 Finite Embeddings of L_2

If densities p and q are smooth, then the L_2 metric between the p_λ and q_λ functions may be well approximated using projections to basis functions. Suppose that $\{\varphi_i\}_{i\in\mathbb{Z}}$ is an orthonormal basis for $L_2([0,1])$; then we can construct an orthonormal basis for $L_2([0,1]^\ell)$ by the tensor product:

$$\{\varphi_\alpha\}_{\alpha\in\mathbb{Z}^\ell} \qquad \text{where} \quad \varphi_\alpha(x) = \prod_{i=1}^{\ell} \varphi_{\alpha_i}(x_i),\ x \in [0,1]^\ell,$$
$$\forall f \in L_2([0,1]^\ell),\ f(x) = \sum_{\alpha\in\mathbb{Z}^\ell} a_\alpha(f)\,\varphi_\alpha(x)$$

and $a_\alpha(f) = \langle \varphi_\alpha, f\rangle = \int_{[0,1]^\ell} \varphi_\alpha(t)\,f(t)\,\mathrm{d}t \in \mathbb{R}$. Let $V \subset \mathbb{Z}^\ell$ be an appropriately chosen finite set of indices. If $f, f' \in L_2([0,1]^\ell)$ are smooth and $\vec{a}(f) = (a_{\alpha_1}(f), \ldots, a_{\alpha_{|V|}}(f))$, then $\|f - f'\|^2 \approx \|\vec{a}(f) - \vec{a}(f')\|^2$. Thus we can approximate d^2 as the squared distance between finite vectors:

$$d^2(p,q) \approx \|\psi(p) - \psi(q)\|^2 \approx \|A(p) - A(q)\|^2 \tag{4}$$

where $A : L_2([0,1]^\ell) \to \mathbb{R}^{2M|V|}$ concatenates the $\vec{a}$ features for each λ function:

$$A(p) = \frac{1}{\sqrt{M}}\left(\vec{a}(p^R_{\lambda_1}), \ldots, \vec{a}(p^R_{\lambda_M}), \vec{a}(p^I_{\lambda_1}), \ldots, \vec{a}(p^I_{\lambda_M})\right). \tag{5}$$

We will discuss how to estimate $\vec{a}(p^R_\lambda)$, $\vec{a}(p^I_\lambda)$ shortly.

3.3 Embedding RBF Kernels into $\mathbb{R}^D$

The A features approximate the HDD (2) in $\mathbb{R}^{2M|V|}$; thus applying the RKS embedding (Rahimi and Recht, 2007) to the A features will approximate our generalized RBF kernel (1). The RKS embedding is[1] $z : \mathbb{R}^m \to \mathbb{R}^D$ such that for fixed $\{\omega_i\}_{i=1}^{D/2} \overset{iid}{\sim} \mathcal{N}(0, \sigma^{-2} I_m)$ and for each $x, y \in \mathbb{R}^m$:

$$z(x)^\mathsf{T} z(y) \approx \exp\left(-\tfrac{1}{2\sigma^2} \|x - y\|^2\right), \text{ where } z(x) = \sqrt{\tfrac{2}{D}} \left(\sin(\omega_1^\mathsf{T} x), \cos(\omega_1^\mathsf{T} x), \ldots\right). \tag{6}$$

Thus we can approximate the HDD kernel (1) as:

$$K(p, q) = \exp\left(-\frac{1}{2\sigma^2} d^2(p, q)\right) \approx \exp\left(-\frac{1}{2\sigma^2} \|A(p) - A(q)\|^2\right) \approx z(A(p))^\mathsf{T} z(A(q)). \tag{7}$$

3.4 Finite Sample Estimates

Our final approximation for HDD kernels (7) depends on integrals of densities p and q. In practice, we are unlikely to directly observe an input density, but even given a pdf p, the integrals that make up $A(p)$ are not readily computable. We thus first estimate the density as $\hat{p}$, e.g. with kernel density estimation (KDE), and estimate $A(p)$ as $A(\hat{p})$. Recall that the elements of $A(\hat{p})$ are:

$$a_\alpha(\hat{p}^S_{\lambda_j}) = \int_{[0,1]^\ell} \varphi_\alpha(t)\, \hat{p}^S_{\lambda_j}(t)\, \mathrm{d}t \tag{8}$$

where $j \in \{1, \ldots, M\}, S \in \{R, I\}, \alpha \in V$. In lower dimensions, we can approximate (8) with simple Monte Carlo numerical integration. Choosing $\{u_i\}_{i=1}^{n_e} \overset{iid}{\sim} \mathrm{Unif}([0,1]^\ell)$, we can get $\hat{A}(\hat{p})$ by

$$\hat{a}_\alpha(\hat{p}^S_{\lambda_j}) = \frac{1}{n_e} \sum_{i=1}^{n_e} \varphi_\alpha(u_i)\, \hat{p}^S_{\lambda_j}(u_i). \tag{9}$$

We note that in high dimensions, one may use any high-dimensional density estimation scheme (e.g. Lafferty et al. 2012) and estimate (8) with MCMC techniques (e.g. Hoffman and Gelman 2014).

3.5 Summary and Complexity

The algorithm for computing features $\{z(A(p_i))\}_{i=1}^N$ for a set of distributions $\{p_i\}_{i=1}^N$, given sample sets $\{\chi_i\}_{i=1}^N$ where $\chi_i = \{X_j^{(i)} \in [0,1]^\ell\}_{j=1}^{n_i} \overset{iid}{\sim} p_i$, is thus:

1. Draw M scalars $\lambda_j \overset{iid}{\sim} \frac{\mu}{Z}$ and $D/2$ vectors $\omega_r \overset{iid}{\sim} \mathcal{N}(0, \sigma^{-2} I_{2M|V|})$, in $O(M\,|V|\,D)$ time.
2. For each of the N input distributions i:
   (a) Compute a kernel density estimate from χ_i, $\hat{p}_i(u_j)$ for each u_j in (9), in $O(n_i n_e)$ time.
   (b) Compute $\hat{A}(\hat{p}_i)$ using a numerical integration estimate as in (9), in $O(M\,|V|\,n_e)$ time.
   (c) Get the RKS features, $z(\hat{A}(\hat{p}_i))$, in $O(M\,|V|\,D)$ time.

Supposing each $n_i \asymp n$, this process takes a total of $O\left(Nnn_e + NM\,|V|\,n_e + NM\,|V|\,D\right)$ time. Taking $|V|$ to be asymptotically $O(n)$, $n_e = O(D)$, and $M = O(1)$, this is $O(NnD)$ time, compared to $O(N^2 n \log n + N^3)$ for Póczos et al. (2012b) and $O(N^2 n^2)$ for Muandet et al. (2012).

1. There are two versions of the embedding in common use, but this one is preferred (Sutherland and Schneider, 2015).

4. Theory

We bound $\Pr\left(\left|K(p,q) - z(\hat{A}(\hat{p}))^\mathsf{T} z(\hat{A}(\hat{q}))\right| \geq \varepsilon\right)$ for two fixed densities p and q in each source of error: kernel density estimation (ε_{KDE}); approximating $\mu(\lambda)$ with M samples (ε_λ); truncating the projection coefficient series ($\varepsilon_{\text{tail}}$); Monte Carlo integration (ε_{int}); and the RKS embedding (ε_{RKS}).

We need some smoothness assumptions on p and q: that they are members of a periodic Hölder class $\Sigma_{\text{per}}(\beta, L_\beta)$, that they are bounded below by ρ_* and above by ρ^*, and that their kernel density estimates are in $\Sigma_{\text{per}}(\hat{\gamma}, \widehat{L})$ with probability at least $1-\delta$. We use a suitable form of kernel density estimation, to obtain a uniform error bound with a rate based on the function C^{-1} (Giné and Guillou, 2002). We use the Fourier basis and choose $V = \{\alpha \in \mathbb{Z}^\ell \mid \sum_{j=1}^{\ell} |\alpha_j|^{2s} \leq t\}$ for parameters $0 < s < \hat{\gamma}, t > 0$.

Then, for any $\varepsilon_{\text{RKS}} + \frac{1}{\sigma_k\sqrt{e}}\left(\varepsilon_{\text{KDE}} + \varepsilon_\lambda + \varepsilon_{\text{tail}} + \varepsilon_{\text{int}}\right) \leq \varepsilon$,

$$\begin{aligned}
\Pr\left(\left|K(p,q) - z(\hat{A}(\hat{p}))^\mathsf{T} z(\hat{A}(\hat{q}))\right| \geq \varepsilon\right) \leq \\
2\exp\left(-D\varepsilon_{\text{RKS}}^2\right) + 2\exp\left(-M\varepsilon_\lambda^4/(8Z^2)\right) + \delta + 2M\left(1 - \mu\big([0, u_{\text{tail}})\big)\right) \\
+ 2C^{-1}\left(\frac{\varepsilon_{\text{KDE}}^4 n^{2\beta/(2\beta+\ell)}}{4\log n}\right) + 8M\,|V| \exp\left(-\tfrac{1}{2}n_e\left(\frac{\sqrt{1 + \varepsilon_{\text{int}}^2/(8\,|V|\,Z)} - 1}{\sqrt{\rho^*} + 1}\right)^2\right)
\end{aligned}$$

where $u_{\text{tail}} = \sqrt{\max\left(0, \frac{\rho_* t}{8M\ell\widehat{L}^2}\frac{4^{\hat{\gamma}} - 4^s}{4^{\hat{\gamma}}}\varepsilon_{\text{tail}}^2 - \frac{1}{4}\right)}$.

The bound decreases when the function is smoother (larger $\beta, \hat{\gamma}$; smaller $\widehat{L}$) or lower-dimensional (ℓ), or when we observe more samples (n). Using more projection coefficients (higher t or smaller s, giving higher $|V|$) improves the approximation but makes numerical integration more difficult. Likewise, taking more samples from μ (higher M) improves that approximation, but increases the number of functions to be approximated and numerically integrated.

For the proof and further details, see the appendix.

5. Numerical Experiments

We evaluate RBF kernels based on various distances. First, we try our JS, Hellinger, and TV embeddings. We compare to L_2 kernels as in Oliva et al. (2014): $\exp\left(-\frac{1}{2\sigma^2}\|p-q\|_2^2\right) \approx z(\vec{a}(\hat{p}))^\mathsf{T} z(\vec{a}(\hat{q}))$ (L2). We also try the MMD distance (Muandet et al., 2012) with approximate kernel embeddings: $\exp\left(-\frac{1}{2\sigma^2}\widehat{\text{MMD}}(p,q)\right) \approx z\left(\bar{z}(\hat{p})\right)^\mathsf{T} z\left(\bar{z}(\hat{q})\right)$, where $\bar{z}$ is the mean embedding $\bar{z}(\hat{p}) = \frac{1}{n}\sum_{i=1}^n z(X_i)$ (MMD). We further compare to RKS with histogram JS embeddings (Vempati et al., 2010) (Hist JS); we also tried χ^2 embeddings, but their performance was quite similar. We finally try the full Gram matrix approach of Póczos et al. (2012b) with the KL estimator of Wang et al. (2009) in an RBF kernel (KL), as did Ntampaka et al. (2015).

Throughout these experiments we use $M = 5$, $|V| = 10^\ell$ (selected as rules of thumb; larger values did not improve performance), and use a validation set (10% of the training set) to choose bandwidths for KDE and the RBF kernel as well as model regularization parameters. Except in the scene classification experiments, the histogram methods used 10 bins per dimension; performance with other values was not better. The KL estimator used the fourth nearest neighbor.

5.1 Gram Matrix Estimation

We first illustrate that our embedding, using the parameter selections as above, can approximate the Jensen-Shanon kernel well. We compare three different approaches to estimating $K(p_i, p_j) = \exp(-\frac{1}{2\sigma^2}\,\mathrm{JS}(p_i, p_j))$. Each approach uses kernel density estimates $\hat{p}_i$. The estimates are compared on a dataset of $N = 50$ random GMM distributions $\{p_i\}_{i=1}^N$ and samples of size $n = 2\,500$: $\chi_i = \{X_j^{(i)} \in [0,1]^2\}_{j=1}^n \overset{iid}{\sim} p_i$. See the appendix for more details.

The first approach approximates JS based on empirical estimates of entropies $\mathbb{E}\log\hat{p}_i$. The second approach estimates JS as the Euclidean distance of vectors of projection coefficients (4): $\mathrm{JS}_{\mathrm{pc}}(p_i, p_j) = \|\hat{A}(\hat{p}_i) - \hat{A}(\hat{p}_j)\|^2$. For these first two approaches we compute the Gram matrix entries as $G_{ij}^{\mathrm{ent}} = \exp(-\frac{1}{2\sigma^2}\,\mathrm{JS}_{\mathrm{ent}}(p_i, p_j))$, and $G_{ij}^{\mathrm{pc}} = \exp(-\frac{1}{2\sigma^2}\,\mathrm{JS}_{\mathrm{pc}}(p_i, p_j))$. Lastly, we directly estimate the JS kernel with our random features (7): $G_{ij}^{\mathrm{rks}} = z(\hat{A}(\hat{p}_i))^{\mathsf{T}} z(\hat{A}(\hat{p}_j))$, with $D = 7\,000$.

The appendix plots the true pairwise kernel values versus the aforementioned estimates. Quantitatively, the entropy method obtained a squared correlation to the true kernel value of $R^2_{\mathrm{ent}} = 0.981$; using the A features with an exact kernel yielded $R^2_{\mathrm{pc}} = 0.974$; adding RKS embeddings gave $R^2_{\mathrm{rks}} = 0.966$. Thus our method's estimates are nearly as good as direct estimation via entropies.

5.2 Estimating the Number of Mixture Components

We will now illustrate the efficacy of HDD random features in a regression task, following Oliva et al. (2014): estimate the number of components from a mixture of truncated Gaussians. We generate the distributions as follows: Draw the number of components Y_i for the ith distribution as $Y_i \sim \mathrm{Unif}\{1, \ldots, 10\}$. For each component select a mean $\mu_k^{(i)} \sim \mathrm{Unif}[-5, 5]^2$ and covariance $\Sigma_k^{(i)} = a_k^{(i)} A_k^{(i)} A_k^{(i)\mathsf{T}} + B_k^{(i)}$, where $a \sim \mathrm{Unif}[1, 4]$, $A_k^{(i)}(u, v) \sim \mathrm{Unif}[-1, 1]$, and $B_k^{(i)}$ is a diagonal 2×2 matrix with $B_k^{(i)}(u, u) \sim \mathrm{Unif}[0, 1]$. Then weight each component equally in the mixture. Given a sample χ_i, we predict the number of components Y_i. An example distribution and sample are shown in Figure 2; predicting the number of components is difficult even for humans.

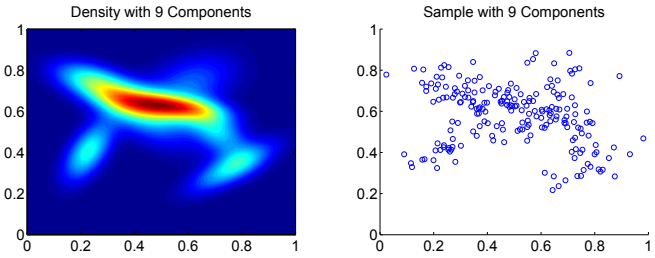
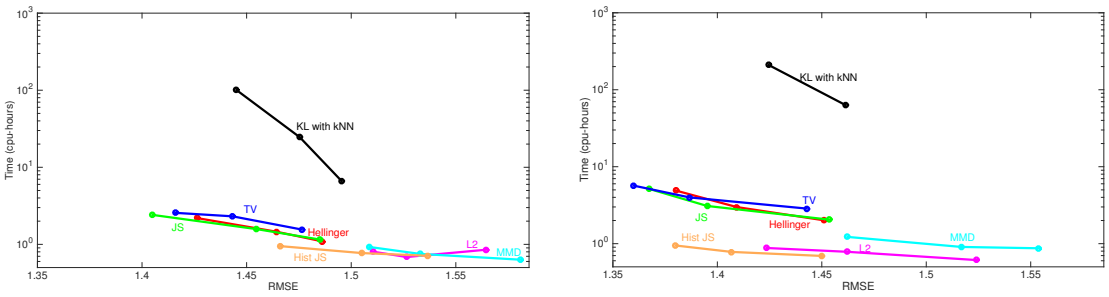


Figure 2: A GMM and 200 points drawn from it.

Figure 3 presents results for predicting with ridge regression the number of mixture components Y_i, given a varying number of sample sets χ_i, with $|\chi_i| \in \{200, 800\}$; we use $D = 5\,000$. The HDD-based kernels achieve substantially lower error than the L_2 and MMD kernels. They also outperform the histogram kernel, especially with $|\chi_i| = 200$, and the KL kernel. Note that fitting mixtures with EM and selecting a number of components using AIC (Akiake, 1973) or BIC (Schwarz, 1978) performed much worse than regression; only AIC with $|\chi_i| = 800$ outperformed a constant predictor of 5.5. Linear versions of the L_2 and MMD kernels were also no better than the constant predictor.



(a) Samples of size 200.

(b) Samples of size 800.

Figure 3: Error and computation time for estimating mixture components. The points on each line show training set sizes 4K, 8K, and 16K; the test set is of size 2K. Note the logarithmic time scale. The KL kernel for $|\chi_i| = 800$ with 16K training sets was too slow to run. AIC-based predictions achieved RMSEs of 2.7 and 2.3; BIC errors were 3.8 and 2.7; constant predictor RMSE was 2.8.

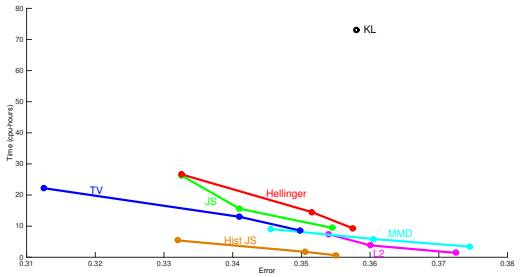


Figure 4: Error rate and computation time for classifying CIFAR-10 cats versus dogs. The three points on each line show training set sizes 2.5K, 5K, and 10K; the test set is fixed of size 2K. Note the linear time scale. The KL kernel was too slow to run for 5K or 10K training points.

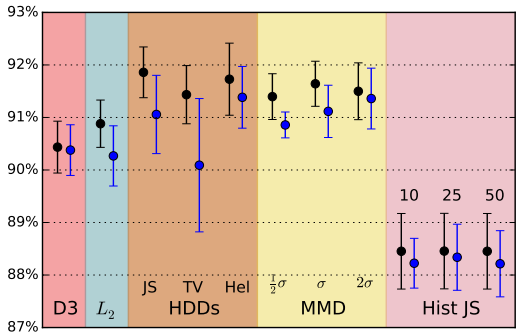


Figure 5: Mean and standard deviation of accuracies on the Scene-15 dataset in 10 random splits. Left, black lines use $\hat{A}(\cdot)$ features; right, blue lines, $z(\hat{A}(\cdot))$ features. MMD methods vary bandwidth relative to the median pairwise distance; histogram methods vary number of bins.

The HDD embeddings were more computationally expensive than the other embeddings, but much less expensive than the KL kernel, which grows at least quadratically in N. Note that the histogram embeddings used an optimized C implementation (Vedaldi and Fulkerson, 2008), as did the KL kernel[2], while the HDD embeddings used a simple Matlab implementation.

5.3 Image Classification

As another example of the performance of our embeddings, we now attempt to classify images based on their distributions of pixel values. We took the "cat" and "dog" classes from the CIFAR-10 dataset (Krizhevsky and Hinton, 2009), and represented each 32×32 image by a set of triples (x, y, v), where x and y are the position of each pixel in the image and v the pixel value after converting to grayscale. The horizontal reflection of the image was also included, so each sample set $\chi_i \subset \mathbb{R}^3$ had $|\chi_i| = 2\,048$. This is not the best representation for these images; rather, we show that given this simple representation, our HDD kernels perform well relative to the other options.

2. https://github.com/dougalsutherland/skl-groups/

We used the kernel estimates in an SVM classifier from LIBLINEAR (Fan et al. 2008, for the embeddings) or LIBSVM (Chang and Lin 2011, for the KL kernel), with $D = 7\,000$. Figure 4 shows results on the standard test set (of size 2K) with 2.5K, 5K, and 10K training images. Our JS and Hellinger embedding match the histogram JS embedding in accuracy, while our TV embedding beats histogram JS; all outperform L_2 and MMD. We could only run the KL kernel for the smallest size; its accuracy was comparable to the HDD and histogram embeddings, at far higher computational cost.

5.4 Scene Classification

Modern computer vision classification systems typically consist of a deep network with several convolutional and pooling layers to extract complex features of input images, followed by one or two fully-connected classification layers. The activations are of shape $n \times h \times w$, where n is the number of filters; each unit corresponds to an overlapping patch of the original image. We can thus treat the final pooled activations as a sample of size hw from an n-dimensional distribution, similarly to how Póczos et al. (2012b) and Muandet et al. (2012) used SIFT features from image patches. Wu et al. (2015) set accuracy records on several scene classification datasets with a particular ad-hoc method of extracting features from distributions (D3); we compare to our more principled alternatives.

We consider the Scene-15 dataset (Lazebnik et al., 2006), which contains $4\,485$ natural images in 15 location categories, and follow Wu et al. in extracting features from the last convolutional layer of the `imagenet-vgg-verydeep-16` model (Simonyan and Zisserman, 2015). We replace that layer's rectified linear activations with sigmoid squashing to $[0, 1]$.[3] hw ranges from 400 to $1\,000$. There are 512 filter dimensions; we concatenate features $\hat{A}(\hat{p}_i)$ extracted from each independently.

We train on the standard for this dataset of 100 images from each class (1500 total) and test on the remainder; Figure 5 shows results. We do not add any spatial information to the model; still, we match the best prior published performance of 91.59 ± 0.48, which trained on over 2 million external images (Zhou et al., 2014). Adding spatial information brought the D3 method slightly above 92% accuracy; their best hybrid method obtained 92.9%. Using these features, however, our methods match or beat MMD and substantially outperform D3, L_2, and the histogram embeddings.

6. Discussion

This work presents the first nonlinear embedding of density functions for quickly computing HDD-based kernels, including kernels based on the popular total variation, Hellinger and Jensen-Shanon divergences. While such divergences have shown good empirical results in the comparison of densities, nonparametric uses of kernels with these divergences previously necessitated the computation of a large $N \times N$ Gram matrix, prohibiting their use in large datasets. Our embeddings allow one to work in a primal space while using information theoretic kernels. We analyze the approximation error of our embeddings, and illustrate their quality on several synthetic and real-world datasets.

Acknowledgments

This work was funded in part by NSF grant IIS1247658 and by DARPA grant FA87501220324. DJS is also supported by a Sandia Campus Executive Program fellowship.

3. We used piecewise-linear weights before the sigmoid function such that 0 maps to 0.5, the 90th percentile of the positive observations maps to 0.9, and the 10th percentile of the negative observations to 0.1, for each filter.

References

Hirotugu Akiake. Information theory and an extension of the maximum likelihood principle. In *2nd Int. Symp. on Inf. Theory*, 1973.

Serge Belongie, Charless Fowlkes, Fan Chung, and Jitendra Malik. Spectral partitioning with indefinite kernels using the Nyström extension. In *ECCV*, 2002.

Chih-Chung Chang and Chih-Jen Lin. LIBSVM: a library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):1–27, 2011.

Yihua Chen, Eric K Garcia, Maya R Gupta, Ali Rahimi, and Luca Cazzanti. Similarity-based classification: Concepts and algorithms. *JMLR*, 10:747–776, 2009.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. LIBLINEAR: A library for large linear classification. *JMLR*, 9:1871–1874, 2008.

Seth R. Flaxman, Yu-xiang Wang, and Alexander J. Smola. Who supported Obama in 2012? Ecological inference through distribution regression. In *KDD*, pages 289–298, 2015. ISBN 9781450336642. doi: 10.1145/2783258.2783300.

Bent Fuglede. Spirals in Hilbert space: With an application in information theory. *Exposition. Math.*, 23(1):23–45, April 2005.

Evarist Giné and Armelle Guillou. Rates of strong uniform consistency for multivariate kernel density estimators. *Ann. Inst. H. Poincaré Probab. Statist.*, 38(6):907–921, 2002.

A Gretton, K Fukumizu, Z Harchaoui, and B K Sriperumbudur. A fast, consistent kernel two-sample test. In *NIPS*, 2009.

Bernard Haasdonk and Claus Bahlmann. Learning with distance substitution kernels. In *Pattern Recognition: 26th DAGM Symposium*, pages 220–227, 2004.

Matthew D Hoffman and Andrew Gelman. The No-U-Turn Sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *JMLR*, 15(1):1593–1623, 2014. URL `http://arxiv.org/abs/1111.4246`.

T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In *NIPS*, 1998.

Tony Jebara, Risi Kondor, and Andrew Howard. Probability product kernels. *JMLR*, 5:819–844, 2004.

Wittawat Jitkrittum, Arthur Gretton, Nicolas Heess, SM Eslami, Balaji Lakshminarayanan, Dino Sejdinovic, and Zoltán Szabó. Kernel-based just-in-time learning for passing expectation propagation messages. *UAI*, 2015.

Risi Kondor and Tony Jebara. A kernel between sets of vectors. In *ICML*, 2003.

Akshay Krishnamurthy, Kirthevasan Kandasamy, Barnabas Poczos, and Larry Wasserman. Nonparametric estimation of Rényi divergence and friends. In *ICML*, 2014.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. *University of Toronto, Tech. Rep*, 2009.

John Lafferty, Han Liu, and Larry Wasserman. Sparse nonparametric graphical models. *Statistical Science*, 27(4):519–537, 2012. ISSN 0883-4237. doi: 10.1214/12-STS391.

Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

Quoc Le, Tamás Sarlós, and Alex J Smola. Fastfood - approximating kernel expansions in loglinear time. In *ICML*, 2013.

Thomas Leung and Jitendra Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *IJCV*, 43, 2001.

Fuxin Li, Catalin Ionescu, and Cristian Sminchisescu. Random Fourier approximations for skewed multiplicative histogram kernels. In *Pattern Recognition: DAGM*, pages 262–271, 2010.

Jianhua Lin. Divergence measures based on the Shannon entropy. *IEEE Trans. Inf. Theory*, 37(1): 145–151, 1991.

David Lopez-Paz, Krikamol Muandet, Bernhard Schölkopf, and Ilya Tolstikhin. Towards a learning theory of causation. *ICML*, 2015.

Subhransu Maji and Alexander C Berg. Max-margin additive classifiers for detection. In *ICCV*, 2009.

Pedro J Moreno, Purdy P Ho, and Nuno Vasconcelos. A Kullback-Leibler divergence based kernel for SVM classification in multimedia applications. In *NIPS*, 2003.

Krikamol Muandet, Kenji Fukumizu, Francesco Dinuzzo, and Bernhard Schölkopf. Learning from distributions via support measure machines. In *NIPS*, 2012.

Michelle Ntampaka, Hy Trac, Dougal J Sutherland, Nicholas Battaglia, Barnabás Póczos, and Jeff Schneider. A machine learning approach for dynamical mass measurements of galaxy clusters. *The Astrophysical Journal*, 803(2):50, October 2015. doi: 10.1088/0004-637X/803/2/50.

Junier B Oliva, Barnabás Póczos, and Jeff Schneider. Distribution to distribution regression. In *ICML*, 2013.

Junier B Oliva, Willie Neiswanger, Barnabás Póczos, Jeff Schneider, and Eric Xing. Fast distribution to real regression. In *AISTATS*, 2014.

Barnabás Póczos, Alessandro Rinaldo, Aarti Singh, and Larry Wasserman. Distribution-free distribution regression. *AISTATS*, 2012a.

Barnabás Póczos, Liang Xiong, Dougal J Sutherland, and Jeff Schneider. Nonparametric kernel estimators for image classification. In *CVPR*, 2012b.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *NIPS*, 2007.

Gideon Schwarz. Estimating the dimension of a model. *Ann. Statist.*, 6(2):461–464, 03 1978.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

Kumar Sricharan, Dennis Wei, and Alfred O. Hero, III. Ensemble estimators for multivariate entropy estimation. *IEEE Trans. Inf. Theory*, 59:4374–4388, 2013.

Dmitry Storcheus, Mehryar Mohri, and Afshin Rostamizadeh. Foundations of coupled nonlinear dimensionality reduction. 2015.

Dougal J. Sutherland and Jeff Schneider. On the error of random Fourier features. In *UAI*, 2015.

Zoltán Szabó, Arthur Gretton, Barnabás Póczos, and Bharath Sriperumbudur. Two-stage sampled learning theory on distributions. *AISTATS*, 2015.

A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/`, 2008.

Andrea Vedaldi and Andrew Zisserman. Efficient additive kernels via explicit feature maps. In *CVPR*, 2010.

Sreekanth Vempati, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Generalized RBF feature maps for efficient detection. In *British Machine Vision Conference*, 2010.

Qing Wang, Sanjeev R Kulkarni, and Sergio Verdú. Divergence estimation for multidimensional densities via k-nearest-neighbor distances. *IEEE Trans. Inf. Theory*, 55(5):2392–2405, 2009.

Christopher K I Williams and Matthias Seeger. Using the Nyström method to speed up kernel machines. In *NIPS*, 2001.

Jianxin Wu, Bin-Bin Gao, and Guoqing Liu. Visual recognition using directional distribution distance, 2015.

Zichao Yang, Alexander J Smola, Le Song, and Andrew Gordon Wilson. la carte - learning fast kernels. *AISTATS*, 2014.

Felix X. Yu, Sanjiv Kumar, Henry Rowley, and Shih-Fu Chang. Compact nonlinear maps and circulant extensions. 2015.

Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using Places database. In *NIPS*, 2014.

Appendix A. Gram Matrix Estimation

We illustrate our embedding's ability to approximate the Jensen-Shanon divergence. In the examples below the densities considered are mixtures of five equally weighted truncated spherical Gaussians on $[0,1]^2$. That is,

$$p_i(x) = \frac{1}{5}\sum_{j=1}^{5} \mathcal{N}_t(m_{ij}, \operatorname{diag}(s_{ij}^2))$$

where $m_{ij} \overset{iid}{\sim} \operatorname{Unif}([0,1]^2)$, $s_{ij} \overset{iid}{\sim} \operatorname{Unif}([0.05, 0.15]^2)$ and $\mathcal{N}_t(m,s)$ is the distribution of a Gaussian truncated on $[0,1]^2$ with mean parameter m and covariance matrix parameter s. We work over the sample set $\{\chi_i\}_{i=1}^N$, where $\chi_i = \{X_j^{(i)} \in [0,1]^2\}_{j=1}^n \overset{iid}{\sim} p_i$, $n = 2500$, $N = 50$.

We compare three different approaches to estimating $K(p_i, p_j) = \exp(-\frac{1}{2\sigma^2}\operatorname{JS}(p_i, p_j))$. Each approach uses density estimates $\hat{p}_i$, which are computed using kernel density estimation. The first approach is based on estimating JS using empirical estimates of entropies:

$$\begin{aligned}
\operatorname{JS}(p_i, p_j) &= -\tfrac{1}{2}\mathbb{E}_{p_i}\left[\log\left(\frac{1}{p_i(x)}\right)\right] - \tfrac{1}{2}\mathbb{E}_{p_j}\left[\log\left(\frac{1}{p_j(x)}\right)\right] + \mathbb{E}_{\frac{1}{2}p_i + \frac{1}{2}p_j}\left[\log\left(\frac{2}{p_i(x)+p_j(x)}\right)\right] \\
&\approx -\tfrac{1}{2}\sum_{m=1}^{\lceil n/2\rceil} \log\left(\frac{1}{\hat{p}_i(X_m^{(i)})}\right) - \tfrac{1}{2}\sum_{m=1}^{\lceil n/2\rceil} \log\left(\frac{1}{\hat{p}_j(X_m^{(j)})}\right) \\
&\quad + \tfrac{1}{2}\sum_{m=1}^{\lceil n/2\rceil} \log\left(\frac{2}{\hat{p}_i(X_m^{(i)}) + \hat{p}_j(X_m^{(i)})}\right) + \tfrac{1}{2}\sum_{m=1}^{\lceil n/2\rceil} \log\left(\frac{2}{\hat{p}_i(X_m^{(j)}) + \hat{p}_j(X_m^{(k)})}\right) \\
&= \operatorname{JS}_{\text{ent}}(p_i, p_j),
\end{aligned}$$

where density estimates $\hat{p}_i$ above are based on points $\{X_m^{(i)}\}_{m=\lceil n/2\rceil+1}^n$ to avoid biasing the empirical means. The second approach estimates JS as the Euclidean distance of vectors of projection coefficients:

$$\operatorname{JS}(p_i, p_j) \approx \|\hat{A}(\hat{p}_i) - \hat{A}(\hat{p}_j)\|^2 = \operatorname{JS}_{\text{pc}}(p_i, p_j),$$

where here the density estimates $\hat{p}_i$ are based on the entire set of points χ_i. We build Gram matrices for each approach by setting $G_{ij}^{\text{ent}} = \exp(-\frac{1}{2\sigma^2}\operatorname{JS}_{\text{ent}}(p_i, p_j))$ and $G_{ij}^{\text{pc}} = \exp(-\frac{1}{2\sigma^2}\operatorname{JS}_{\text{pc}}(p_i, p_j))$. Lastly, we directly estimate the JS kernel with random features:

$$G_{ij}^{\text{rks}} = z(\hat{A}(\hat{p}_i))^{\mathsf{T}} z(\hat{A}(\hat{p}_j)).$$

We compare the effectiveness of each approach by computing the R^2 score of the estimates produced versus a true JS kernel value computed through numerically integrating the true densities (see Figure 6 and Table 2). The RBF values estimated with our random features produce estimates that are nearly as good as directly estimating JS divergences through entropies, whilst allowing us to work over a primal space and thus avoid computing a $N \times N$ Gram matrix for learning tasks.

Appendix B. Proofs

We will now prove the bound on $\Pr\left(\left|K(p,q) - z(\hat{A}(\hat{p}))^{\mathsf{T}} z(\hat{A}(\hat{q}))\right| \geq \varepsilon\right)$ for fixed densities p, q.

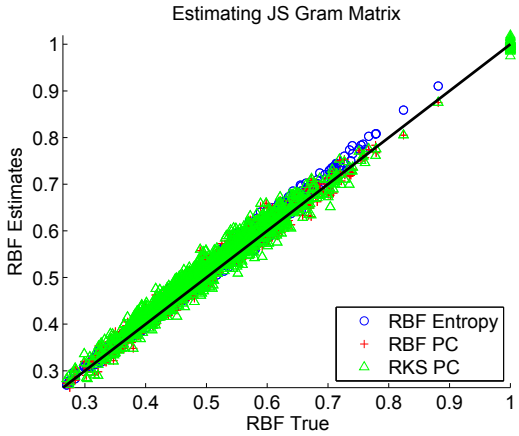


Figure 6: Estimating RBF values using the JS diveregence.

Table 2: R^2 values of estimates of JS Gram elements.

Method	R^2
Entropies	0.9812
PCs	0.9735
RKS	0.9662

Setup We will need a few assumptions on the densities:

1. p and q are bounded above and below: for $x \in [0,1]^\ell$, $0 < \rho_* \leq p(x), q(x) \leq \rho^* < \infty$.

2. $p, q \in \Sigma(\beta, L_\beta)$ for some $\beta, L_\beta > 0$. $\Sigma(\beta, L)$ refers to the Hölder class of functions f whose partial derivatives up to order $\lfloor \beta \rfloor$ are continuous and whose rth partial derivatives, where r is a multi-index of order $\lfloor \beta \rfloor$, satisfy $|D^r f(x) - D^r f(y)| \leq L\|x - y\|^\beta$. Here $\lfloor \beta \rfloor$ is the greatest integer *strictly* less than β.

3. p, q are periodic.

These are fairly standard smoothness assumptions in the nonparametric estimation literature.

Let $\gamma = \min(\beta, 1)$. If $\beta > 1$, then $p, q \in \Sigma(1, L_\gamma)$ for some L_γ; otherwise, clearly $p, q \in \Sigma(\beta, L_\beta)$. Then, from assumption 3, $p, q \in \Sigma_{\text{per}}(\gamma, L_\gamma)$, the periodic Hölder class. We'll need this to establish the Sobolev ellipsoid containing p and q.

We will use kernel density estimation with a bounded, continuous kernel so that the bound of Giné and Guillou (2002) applies, with bandwidth $h \asymp n^{-1/(2\beta+\ell)} \log n$, and truncating density estimates to $[\rho_*, \rho^*]$.

We also use the Fourier basis $\varphi_\alpha = \exp\left(2\mathbb{i}\pi\alpha^\mathsf{T} x\right)$, and define V as the set of indices α s.t. $\sum_{j=1}^{\ell} |\alpha_j|^{2s} \leq t$ for parameters $0 < s \leq 1$, $t > 0$ to be discussed later.

Decomposition Let $r_\sigma(\Delta) = \exp\left(-\Delta^2/(2\sigma^2)\right)$. Then

$$\left|K(p,q) - z(\hat{A}(\hat{p}))^\mathsf{T} z(\hat{A}(\hat{q}))\right| \leq \left|K(p,q) - r_{\sigma_k}\left(\|\hat{A}(\hat{p}) - \hat{A}(\hat{q})\|\right)\right| + \left|r_{\sigma_k}\left(\|\hat{A}(\hat{p}) - \hat{A}(\hat{q})\|\right) - z(\hat{A}(\hat{p}))^\mathsf{T} z(\hat{A}(\hat{q}))\right|.$$

The latter term was bounded by Rahimi and Recht (2007). For the former, note that r_σ is $\frac{1}{\sigma\sqrt{e}}$-Lipschitz, so the first term is at most $\frac{1}{\sigma_k\sqrt{e}}\left|d(p,q) - \|\hat{A}(\hat{p}) - \hat{A}(\hat{q})\|\right|$. Breaking this up with the triangle inequality:

$$\begin{aligned}\left|d(p,q) - \|\hat{A}(\hat{p}) - \hat{A}(\hat{q})\|\right| &\leq |d(p,q) - d(\hat{p},\hat{q})| + |d(\hat{p},\hat{q}) - \|\psi(\hat{p}) - \psi(\hat{q})\||| \\ &\quad + |\|\psi(\hat{p}) - \psi(\hat{q})\| - \|A(\hat{p}) - A(\hat{q})\|| + \left|\|A(\hat{p}) - A(\hat{q})\| - \|\hat{A}(\hat{p}) - \hat{A}(\hat{q})\|\right|. \quad (10)\end{aligned}$$

Estimation error Recall that d is a metric, so the reverse triangle inequality allows us to address the first term with

$$|d(p,q) - d(\hat{p},\hat{q})| \leq d(p,\hat{p}) + d(q,\hat{q}).$$

For d^2 the total variation, squared Hellinger, or Jensen-Shannon HDDs, we have that $d^2(p,\hat{q})$ is upper bounded by $\mathrm{TV}(p,\hat{p})$ (Lin, 1991). Moreover, as the distributions are supported on $[0,1]^\ell$, $\mathrm{TV}(p,\hat{p}) = \frac{1}{2}\|p - \hat{p}\|_1 \leq \frac{1}{2}\|p - \hat{p}\|_\infty$.

It is a consequence of Giné and Guillou (2002) that, for any $\delta > 0$, there is some C_δ depending on the kernel such that $\Pr\left(\|p - \hat{p}\|_\infty > \frac{\sqrt{C_\delta \log n}}{n^{\beta/(2\beta+\ell)}}\right) < \delta$. Thus

$$\Pr\left(|d(p,q) - d(\hat{p},\hat{q})| \geq \varepsilon\right) < 2C^{-1}\left(\frac{\varepsilon^4 n^{2\beta/(2\beta+\ell)}}{4\log n}\right),$$

where $C_{C^{-1}(x)} = x$.

λ approximation The second term of (10), the approximation error due to sampling λs, admits a simple Hoeffding bound. Note that $\left\|\hat{p}_\lambda^R - \hat{q}_\lambda^R\right\|^2 + \left\|\hat{p}_\lambda^I - \hat{q}_\lambda^I\right\|^2$, viewed as a random variable in λ only, has expectation $d^2(\hat{p},\hat{q})$ and is bounded by $[0, 4Z]$ (where $Z = \int_{\mathbb{R}_{\geq 0}} \mathrm{d}\mu(\lambda)$): write it as $Z\int|\hat{p}(x)^{\frac{1}{2}+\mathbb{i}\lambda} - \hat{q}(x)^{\frac{1}{2}+\mathbb{i}\lambda}|^2\,\mathrm{d}x$, expand the square, and use $\int\sqrt{\hat{p}(x)\hat{q}(x)}\mathrm{d}x \leq 1$ (via Cauchy-Schwarz).

For nonnegative random variables X and Y, $\Pr\left(|X - Y| \geq \varepsilon\right) \leq \Pr\left(\left|X^2 - Y^2\right| \geq \varepsilon^2\right)$, so we have that $\Pr\left(|\|\psi(\hat{p}) - \psi(\hat{q})\| - d(\hat{p},\hat{q})| \geq \varepsilon\right)$ is at most $2\exp(-M\varepsilon^4/(8Z^2))$.

Tail truncation error The third term of (10), the error due to truncating the tail projection coefficients of the p_λ^S functions, requires a little more machinery. First note that

$$\left|\|\psi(\hat{p}) - \psi(\hat{q})\|^2 - \|A(\hat{p}) - A(\hat{q})\|^2\right| \leq \sum_{j=1}^{M}\sum_{S=R,I}\sum_{\alpha\notin V}\left|a_\alpha(\hat{p}_\lambda^S - \hat{q}_\lambda^S)\right|^2. \quad (11)$$

Let $\mathcal{W}(s, L)$ be the Sobolev ellipsoid of functions $\sum_{\alpha\in\mathbb{Z}^\ell} a_\alpha\varphi_\alpha$ such that the coefficients a_α have $\sum_{\alpha\in\mathbb{Z}^\ell}\left(\sum_{j=1}^\ell|\alpha_j|^{2s}\right)|a_\alpha|^2 \leq L$, where φ is still the Fourier basis. Then Lemma 14 of Krishnamurthy et al. (2014) shows that $\Sigma_{\text{per}}(\gamma, L_\gamma) \subseteq \mathcal{W}(s, L')$ for any $0 < s < \gamma$ and $L' = \ell L_\gamma^2(2\pi)^{-2\lfloor\gamma\rfloor}\frac{4^\gamma}{4^\gamma-4^s}$.

So, suppose that $\hat{p}, \hat{q} \in \Sigma_{\text{per}}(\hat{\gamma}, \widehat{L})$ with probability at least $1-\delta$. Since $x \mapsto x^{\frac{1}{2}+\mathbb{i}\lambda}$ is $\frac{\sqrt{1+4\lambda^2}}{2\sqrt{\rho_*}}$-Lipschitz on $[\rho_*, \infty)$, $\hat{p}_\lambda^S \in \Sigma_{\text{per}}\left(\hat{\gamma}, \frac{1}{2}\sqrt{1+4\lambda^2}\,\widehat{L}\,\rho_*^{-\frac{1}{2}}\right)$ and so $\hat{p}_\lambda^S - \hat{q}_\lambda^S$ is in $\mathcal{W}(s, (1+4\lambda^2)\widehat{L}')$ for $s < \hat{\gamma}$ and $\widehat{L}' = \ell\widehat{L}^2\rho_*^{-1}/(1-4^{s-\hat{\gamma}})$.

Recall that we chose V to be the set of $\alpha \in \mathbb{Z}^\ell$ such that $\sum_{j=1}^\ell|\alpha_j|^{2s} \leq t$. Thus $\sum_{\alpha\notin V}|a_\alpha(\hat{p}_\lambda^S - \hat{q}_\lambda^S)|^2 \leq \sum_{\alpha\notin V}|a_\alpha(\hat{p}_\lambda^S - \hat{q}_\lambda^S)|^2\left(\sum_{j=1}^\ell|\alpha_j|^{2s}\right)/t \leq (1+4\lambda^2)\widehat{L}'/t$.

The tail error term therefore exceeds ε with probability no more than

$$\delta + 2\sum_{j=1}^M \Pr\left((1+4\lambda_j^2)\widehat{L}'/t \geq \varepsilon^2/(2M)\right).$$

The latter probability, of course, depends on the choice of HDD d. Letting $\zeta = t\varepsilon^2/(8M\widehat{L}') - \frac{1}{4}$, it is 1 if $\zeta < 0$ and $1 - \mu\left([0, \sqrt{\zeta}]\right)/Z$ otherwise. If $\zeta \geq 0$, squared Hellinger's probability is 0, and total variation's is $\frac{2}{\pi}\arctan(\sqrt{\zeta})$. A closed form for the cumulative distribution function for the Jensen-Shannon measure is unfortunately unknown.

Numerical integration error The final term of (10) also bears a Hoeffding bound. Define the projection coefficient difference $\Delta_{\lambda,\alpha}^S(p, q) = a_{\alpha,\lambda}(p_\lambda^S) - a_\alpha(q_\lambda^S)$, and $\hat{\Delta}$ similarly but with $\hat{a}$. Then

$$\left|\|A(\hat{p}) - A(\hat{q})\|^2 - \left\|\hat{A}(\hat{p}) - \hat{A}(\hat{q})\right\|^2\right| \leq \sum_{j=1}^M\sum_{S=R,I}\sum_{\alpha\in V}\left|\left|\Delta_{\alpha,\lambda_j}^S(\hat{p}, \hat{q})\right|^2 - \left|\hat{\Delta}_{\alpha,\lambda_j}^S(\hat{p}, \hat{q})\right|^2\right|. \quad (12)$$

Letting $\hat{\epsilon}(p) = a_\alpha(\hat{p}_\lambda^S) - \hat{a}_\alpha(\hat{p}_\lambda^S)$, each summand is at most $(\hat{\epsilon}(p) + \hat{\epsilon}(q))^2 + 2\left|\Delta_{\lambda,\alpha}^S(\hat{p}, \hat{q})\right|(\hat{\epsilon}(p) + \hat{\epsilon}(q))$. Also, $\left|\Delta_{\alpha,\lambda}^S(\hat{p}, \hat{q})\right| \leq 2\sqrt{Z}$, using Cauchy-Schwarz on the integral and $\|\varphi_\alpha\|_2 = 1$. Thus each summand in (12) can be more than ε only if one of the $\hat{\epsilon}$s is more than $\sqrt{Z + \varepsilon/4} - \sqrt{Z}$.

Now, using (9), $\hat{a}_\alpha(\hat{p}_\lambda^S)$ is an empirical mean of n_e independent terms, each with absolute value bounded by $(\sqrt{\rho^*}+1)\max_x|\varphi_\alpha(x)| = \sqrt{\rho^*}+1$. Thus, using a Hoeffding bound on the $\hat{\epsilon}$s, we get that $\Pr\left(\left|\|A(\hat{p}) - A(\hat{q})\|^2 - \|\hat{A}(\hat{p}) - \hat{A}(\hat{q})\|^2\right| \geq \varepsilon\right)$ is no more than $8MS\exp\left(-\frac{n_e\left(\sqrt{Z+\varepsilon^2/(8S)}-\sqrt{Z}\right)^2}{2Z(\sqrt{\rho^*}+1)^2}\right)$.

Final bound Combining the bounds for the decomposition (10) with the pointwise rate for RKS features, we get:

$$\begin{aligned}\Pr\left(\left|K(p,q) - z(\hat{A}(\hat{p})^{\mathsf{T}} z(\hat{A}(\hat{q}))\right| \geq \varepsilon\right) &\leq 2\exp\left(-D\varepsilon_{\text{RKS}}^2\right) + 2C^{-1}\left(\frac{\varepsilon_{\text{KDE}}^4 n^{2\beta/(2\beta+\ell)}}{4\log n}\right) \\ &+ 2\exp\left(-M\varepsilon_\lambda^4/(8Z^2)\right) + \delta + 2M\left(1 - \mu\left[0, \sqrt{\max\left(0, \frac{\rho_* t\varepsilon_{\text{tail}}^2}{8M\ell\widehat{L}^2}\frac{4^{\hat{\gamma}} - 4^s}{4^{\hat{\gamma}}} - \frac{1}{4}\right)}\right]\right) \\ &+ 8M|V|\exp\left(-\tfrac{1}{2}n_e\left(\frac{\sqrt{1 + \varepsilon_{\text{int}}^2/(8|V|Z)} - 1}{\sqrt{\rho^*} + 1}\right)^2\right) \quad (13)\end{aligned}$$

for any $\varepsilon_{\text{RKS}} + \frac{1}{\sigma_k\sqrt{e}}\left(\varepsilon_{\text{KDE}} + \varepsilon_\lambda + \varepsilon_{\text{tail}} + \varepsilon_{\text{int}}\right) \leq \varepsilon$.