SAND20XX-XXXXR

LDRD PROJECT NUMBER: 181061

LDRD PROJECT TITLE

(Active) Learning on Groups of Data with Information-Theoretic Estimators

PROJECT TEAM MEMBERS

Dougal Sutherland (CMU)
Philip Kegelmeyer (Sandia PI, 8900)
Robert Hutchinson (Sandia PM, 0500)

ABSTRACT

A wide range of machine learning problems, including astronomical inference about galaxy clusters, scene classification, parametric statistical inference, and predictions of public opinion, can be well-modeled as learning a function on (samples from) distributions. This project explores problems in learning such functions via kernel methods, particularly for large-scale problems.

When learning from large numbers of distributions, the computation of typical methods scales between quadratically and cubically, and so they are not amenable to large datasets. We investigate the approach of approximate embeddings into Euclidean spaces such that inner products in the embedding space approximate kernel values between the source distributions. We first improve the understanding of the workhorse methods of random Fourier features: we show that of the two approaches in common usage, one is strictly superior. We then present a new embedding for a class of information-theoretic distribution distances, and evaluate it and existing embeddings on several real-world applications.

INTRODUCTION

Traditional machine learning approaches focus on learning problems defined on vectors, mapping whatever kind of object we wish to model to a fixed number of real-valued attributes. Though this approach has been very successful in a variety of application areas, choosing natural and effective representations can be quite difficult.

In many settings, we wish to perform machine learning tasks on objects that can be viewed as a collection of lower-level objects or more directly as samples from a distribution. For example:

- Images can be thought of as a collection of local patches (Póczos et al. 2012); similarly, videos are collections of frames.
- The total mass of a galaxy cluster can be predicted based on the positions and velocities of individual galaxies (Ntampaka et al. 2015, Ntampaka et al. (2016))
- Support for a political candidate among various demographic groups can be estimated by learning a regression model from electoral districts of individual voters to district-level support for political candidates (Flaxman, Wang, and Smola 2015).
- Documents are made of sentences, which are themselves composed of words, which themselves can be seen as being represented by sets of the contexts in which they appear.
- Parametric statistical inference problems learn a function from sample sets to model parameters (discussed later in this report).
- Expectation propagation techniques relay on maps from sample sets to messages normally computed via expensive numerical integration (Jitkrittum et al. 2015).
- Causal arrows between distributions can be estimated from samples (Lopez-Paz et al. 2015).

In order to use traditional techniques on these collective objects, we must create a single vector that represents the entire set. Though there are various ways to summarize a set as a vector, we can often discard less information and require less effort in feature engineering by operating directly on sets of feature vectors.

One method for machine learning on sets is to consider them as samples from some unknown underlying probability distribution over feature vectors. Each example then has its own distribution: if we are classifying images as sets of patches, each image is defined as a distribution over patch features, and each class of clusters is a set of patch-level feature distributions. We can then define a kernel based on statistical estimates of a distance between probability distributions. Letting $X \subseteq \mathbb{R}^d$ denote the set of possible feature vectors, we thus define a kernel $k: 2^X \times 2^X \to \mathbb{R}$. This lets us perform classification, regression, anomaly detection, clustering, low-dimensional embedding, and any of many other applications with the well-developed suite of kernel methods. We will shortly discuss several such kernels, and estimators of them.

When used for a learning problem with N training items, however, typical kernel methods require operating on an $N \times N$ kernel matrix, which requires far too much computation to scale to datasets with a large number of instances. We discuss here one way to avoid this problem:

approximate embeddings $z: X \to \mathbb{R}^D$, à la Rahimi and Recht (2007), such that $z(x)^\mathsf{T} z(y) \approx k(x, y)$. These embeddings are available for several distributional kernels, and are also evaluated empirically later.

We approximate kernels between densities p_i, p_j with random features of sample sets $\chi_i \sim p_i, \chi_j \sim p_j$.

This report overviews two major contributions:

- Improvements in the understanding of random Fourier features, in particular the result that one commonly-used version of their implementation is strictly superior to the other. Proofs and a more complete analysis are given in Sutherland and Schneider (2015).
- A novel approach for embedding kernels based on the total variation, Jensen-Shannon, and Hellinger distances, developed in Sutherland et al. (2016).

The following related work was also conducted during this fellowship, but is not discussed in detail here:

- The application of distribution learning approaches to determining the mass of galaxy clusters from velocity information. See our papers Ntampaka et al. (2015), Ntampaka et al. (2016).
- The use of region classifiers in a system for actively seeking out instances of regional patterns based on point-level observations. See our paper Ma et al. (2015).

DETAILED DESCRIPTION OF METHOD

LEARNING ON DISTRIBUTIONS

Our previous work (Póczos et al. 2012), along with the simultaneous related paper (Muandet et al. 2012), helped establish as empirically quite effective the following technique for learning on distributions. Let P be the set of probability distributions under consideration.

1. Choose a distance $\rho: P \times P \to \mathbb{R}$.
2. Define a Mercer kernel $k: P \times P \to \mathbb{R}$ based on ρ.

3. Estimate k based on observed samples with $\hat{k} : 2^X \times 2^X \to \mathbb{R}$, which should itself be a kernel on 2^X.

4. Use $\hat{k}$ in a standard kernel method, such as an SVM or Gaussian Process, to perform classification, regression, collective anomaly detection, or other machine learning tasks.

The typical choice in step 2 is that of a "generalized" Gaussian RBF kernel: $k(P,Q) = \exp\left(-\frac{1}{2\sigma^2}\rho^2(P,Q)\right)$. Sometimes a linear kernel with origin O, $k(P,Q) = \frac{1}{2}(\rho^2(P,O) + \rho^2(Q,O) - \rho^2(P,Q))$, is preferred. These kernels are positive semidefinite (for all O and σ) precisely when ρ is *Hilbertian*, i.e. isometric to an L_2 norm (Haasdonk and Bahlmann 2004).

We will consider for now the following distances on distributions:

- L_2 distance, given by $L_2(p,q) = \sqrt{\int (p(x) - q(x))^2 \mathrm{d}x}$.
- Hellinger distance, $H(p,q) = \sqrt{\frac{1}{2}\int \left(\sqrt{p(x)} - \sqrt{q(x)}\right)^2 \mathrm{d}x}$.
- Total variation distance, $TV(p,q) = \frac{1}{2}\int |p(x) - q(x)| \mathrm{d}x$.
- Jensen-Shannon distance, $JS(p,q) = \frac{1}{2}KL\left(p \parallel \frac{p+q}{2}\right) + \frac{1}{2}KL\left(q \parallel \frac{p+q}{2}\right)$, where KL is the well-known Kullback-Liebler divergence $KL(p,q) = \int p(x) \log \frac{p(x)}{q(x)} \mathrm{d}x$.
- The maximum mean discrepancy (MMD) (Gretton, Borgwardt, et al. 2012), which is defined in terms of a *base kernel* κ as

$$MMD(P,Q) = \sqrt{\mathbb{E}_{X,X' \sim P}\kappa(X,X') - 2\mathbb{E}_{X \sim P, Y \sim Q}\kappa(X,Y) + \mathbb{E}_{Y,Y' \sim Q}\kappa(Y,Y')}.$$

- This corresponds to the difference between the *mean embeddings* in the reproducing kernel Hilbert space corresponding to κ. We refer to the corresponding linear kernel as $MMK(P,Q) = \mathbb{E}_{X \sim P, Y \sim Q}\kappa(X,Y)$.

Póczos et al. (2012) and Sutherland et al. (2012) considered particular nonparametric estimators of these and similar distances, and found that approaches similar to the Jensen-Shannon distance performed best on at least some practical problems.

The approaches considered there can be powerful, but usually require computing an $N \times N$ matrix of kernel evaluations, which can be infeasible for large datasets. The use of divergences 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 in fact do not induce a PSD kernel. In general this must be remedied by altering the Gram matrix a ``nearby" PSD one. Typical approaches involve eigendecomposing the Gram matrix, which usually 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).

Instead, we will consider approximate embedding methods, in which we find an embedding $z: 2^X \to \mathbb{R}^D$ such that $z(X)^\top z(Y) \approx k(P, Q)$. Learning primal models in $\mathbb{R}^D$ using the z features can then usually be accomplished in time linear in n, with the models on z approximating the models on k.

RANDOM FOURIER FEATURES

Recent interest in this type of embedding method was spurred by Rahimi and Recht (2007). Their approach, known as random Fourier features, assumes a continuous shift-invariant kernel on $\mathbb{R}^d$, i.e. those that can be written $k(x, y) = k(\Delta)$, where we will use $\Delta := x - y$ throughout. In this case, Bochner's theorem (1959) guarantees that the Fourier transform $\Omega(\cdot)$ of k will be a nonnegative measure; if $k(0) = 1$, it will be properly normalized. Thus if we define

$$\tilde{z}(x) := \sqrt{\frac{2}{D}}\left[\sin(\omega_1^\top x) \quad \cos(\omega_1^\top x) \quad \ldots \quad \sin(\omega_{D/2}^\top x) \quad \cos(\omega_{D/2}^\top x)\right]^\top, \ \{\omega_i\}_{i=1}^{D/2} \sim \Omega^{D/2}$$

and let $\tilde{s}(x, y) := \tilde{z}(x)^\top \tilde{z}(y)$, we have that

$$\tilde{s}(x, y) = \frac{2}{D}\sum_{i=1}^{D/2} \sin(\omega_i^\top x)\sin(\omega_i^\top y) + \cos(\omega_i^\top x)\cos(\omega_i^\top y) = \frac{1}{D/2}\sum_{i=1}^{D/2} \cos(\omega_i^\top \Delta).$$

Noting that $\mathbb{E}\cos(\omega^\top \Delta) = \int \Re e^{\omega^\top \Delta i} d\Omega(\omega) = \Re k(\Delta)$, we therefore have $\mathbb{E}\,\tilde{s}(x, y) = k(x, y)$.

Note k is the characteristic function of Ω, and $\tilde{s}$ the empirical characteristic function corresponding to the samples $\{\omega_i\}$.

Rahimi and Recht (2007) alternatively proposed

$$\check{z}(x) := \sqrt{\frac{2}{D}}[\cos(\omega_1^\top x + b_1) \quad \dots \quad \cos(\omega_D^\top x + b_D)]^\top,\ \{\omega_i\}_{i=1}^D \sim \Omega^D,\ \{b_i\}_{i=1}^D \sim \mathrm{Unif}_{[0,2\pi]}^D.$$

Letting $\check{s}(x,y) := \check{z}(x)^\top \check{z}(y)$, we have

$$\check{s}(x,y) = \frac{1}{D}\sum_{i=1}^{D} \cos(\omega_i^\top x + b_i)\cos(\omega_i^\top y + b_i)$$

$$= \frac{1}{D}\sum_{i=1}^{D} \cos(\omega_i^\top (x-y)) + \cos(\omega_i^\top (x+y) + 2b_i).$$

Let $t := x + y$ throughout. Since $\mathbb{E}\cos(\omega^\top t + 2b) = \mathbb{E}_\omega[\mathbb{E}_b \cos(\omega^\top t + 2b)] = 0$, we also have $\mathbb{E}\,\check{s}(x,y) = k(x,y)$.

Thus, in expectation, both $\tilde{z}$ and $\check{z}$ work; they are each the average of bounded, independent terms with the correct mean. For a given embedding dimension, $\tilde{z}$ has half as many terms as $\check{z}$, but each of those terms has lower variance; which embedding is superior is, therefore, not obvious. We will answer this question, as well as giving uniform convergence bounds for each embedding.

We can in fact find the covariance of the reconstructions:

$$\mathrm{Cov}\left(\tilde{s}(\Delta), \tilde{s}(\Delta')\right) = \frac{2}{D}\mathrm{Cov}(\cos(\omega^\top \Delta), \cos(\omega^\top \Delta'))$$

$$= \frac{1}{D}\left[\mathbb{E}[\cos(\omega^\top(\Delta - \Delta')) + \cos(\omega^\top(\Delta + \Delta'))] - 2\mathbb{E}[\cos(\omega^\top \Delta)]\mathbb{E}[\cos(\omega^\top \Delta)]\right]$$

$$= \frac{1}{D}\left[k(\Delta - \Delta') + k(\Delta + \Delta') - 2k(\Delta)k(\Delta')\right],$$

so that

$$\mathrm{Var}\,\tilde{s}(\Delta) = \frac{1}{D}\left[1 + k(2\Delta) - 2k(\Delta)^2\right].$$

Similarly,

$$\begin{aligned}\mathrm{Cov}(\check{s}(x,y),\check{s}(x',y')) &= \frac{1}{D}\mathrm{Cov}(\cos(\omega^\mathsf{T}\Delta)+\cos(\omega^\mathsf{T}t+2b),\cos(\omega^\mathsf{T}\Delta')+\cos(\omega^\mathsf{T}t'+2b))\\ &= \frac{1}{D}[\mathrm{Cov}(\cos(\omega^\mathsf{T}\Delta),\cos(\omega^\mathsf{T}\Delta'))+\mathrm{Cov}(\cos(\omega^\mathsf{T}t+2b),\cos(\omega^\mathsf{T}t'+2b))\\ &\quad +\underbrace{\mathrm{Cov}(\cos(\omega^\mathsf{T}\Delta),\cos(\omega^\mathsf{T}t'+2b))}_{0}+\underbrace{\mathrm{Cov}(\cos(\omega^\mathsf{T}t+2b),\cos(\omega^\mathsf{T}\Delta'))}_{0}]\\ &= \frac{1}{D}\left[\frac{1}{2}\underline{k}(\Delta-\Delta')+\frac{1}{2}\underline{k}(\Delta+\Delta')-\underline{k}(\Delta)\underline{k}(\Delta')+\frac{1}{2}\underline{k}(t-t')\right],\end{aligned}$$

and so

$$\mathrm{Var}\,\check{s}(x,y)=\frac{1}{D}\left[1+\frac{1}{2}\underline{k}(2\Delta)-\underline{k}(\Delta)^2\right].$$

Thus $\tilde{s}$ has lower variance than $\check{s}$ when $\underline{k}(2\Delta) < 2\underline{k}(\Delta)^2$, i.e.

$$\mathrm{Var}\cos(\omega^\mathsf{T}\Delta)=\frac{1}{2}+\frac{1}{2}\underline{k}(2\Delta)-\underline{k}(\Delta)^2\le\frac{1}{2}.$$

In this case, the L_2 approximation of $\tilde{s}$ is strictly better than that of $\check{s}$, since the bias is zero in both cases.

Consider a kernel of the form $\underline{k}(\Delta)=\exp(-\gamma\|\Delta\|^\beta)$ for any norm and some $\beta\ge 1$. For example, the Gaussian kernel uses $\|\cdot\|_2$ and $\beta=2$, and the Laplacian kernel uses $\|\cdot\|_1$ and $\beta=1$. Then

$$\begin{aligned}2\underline{k}(\Delta)^2-\underline{k}(2\Delta) &= 2\exp\left(-\gamma\|\Delta\|^\beta\right)^2-\exp\left(-\gamma\|2\Delta\|^\beta\right)\\ &= 2\exp\left(-2\gamma\|\Delta\|^\beta\right)-\exp\left(-2^\beta\gamma\|\Delta\|^\beta\right)\\ &\ge 2\exp\left(-2\gamma\|\Delta\|^\beta\right)-\exp\left(-2\gamma\|\Delta\|^\beta\right)=\exp\left(-2\gamma\|\Delta\|^\beta\right)>0,\end{aligned}$$

and so for these kernels $\tilde{s}$ has lower variance than $\check{s}$.

This property can also be shown for the Matérn kernel of half-integer order, as can be seen by rewriting it using equation 4.16 of Rasmussen and Williams (2006) and gather terms appropriately.

Sutherland and Schneider (2015) shows furthermore that bounds on the kernel approximation in terms of L_∞ error are tighter for $\tilde{s}$ than for $\check{s}$, as well as various other results about the quality of the approximation.

MAXIMUM MEAN DISCREPANCY EMBEDDING

Armed with an approximate embedding for shift-invariant kernels on $\mathbb{R}^d$, we now develop our first embedding for a distributional kernel, . Recall that, given samples $\{X_i\}_{i=1}^n \sim P^n$ and $\{Y_j\}_{j=1}^m \sim Q^m$, $MMK(P,Q)$ can be estimated as

$$MMK(X,Y) = \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m} k\,(X_i, Y_j).$$

Simply plugging in an approximate embedding $z(x)^\mathsf{T} z(y) \approx k(x,y)$ yields

$$MMK(X,Y) \approx \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m} z\,(X_i)^\mathsf{T} z(Y_j) = \left[\frac{1}{n}\sum_{i=1}^{n} z(X_i)\right]^\mathsf{T}\left[\frac{1}{m}\sum_{j=1}^{m} z(Y_j)\right] = \overline{z}(X)^\mathsf{T}\overline{z}(Y),$$

where we defined $\overline{z}(X) := \frac{1}{n}\sum_{i=1}^{n} z\,(X_i)$. This additionally has a natural interpretation as the direct estimate of in the Hilbert space induced by the feature map z, which approximates the Hilbert space associated with k.

Note that $e^{-\gamma MMD^2}$ can be approximately embedded with $z(\overline{z}(\cdot))$.

This natural approximation, or its equivalents, have been considered many times quite recently (Mehta and Gray 2010, Li and Tsang (2011), Zhao and Meng (2014), Flaxman, Wang, and Smola (2015), Jitkrittum et al. (2015), Lopez-Paz et al. (2015), Chwialkowski et al. (2015), Sutherland and Schneider (2015)).

L_2 DISTANCE EMBEDDING

Oliva et al. (2014) gave an embedding for $e^{-\gamma L_2^2}$, by first embedding L_2 with orthonormal projections and then applying random Fourier features. We omit the details here for the sake of brevity; it is in fact similar to the MMD embedding when using a Gaussian RBF kernel for κ, but using a fixed design instead of random sampling for the Fourier frequencies.

HOMOGENEOUS DENSITY DISTANCE EMBEDDING

We will now show how to extend this general approach to a class of information theoretic distances that includes total variation, Jenson-Shannon divergence, and squared Hellinger.

We consider a class of metrics that we term (HDDs):

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

where $\kappa: \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}_+$ is a 1-homogenous negative-type kernel. That is, $\kappa(tx, ty) = t\kappa(x,y)$ for all $t > 0$, and there exists some Hilbert space with $\|x - y\|^2 = \kappa(x,y)$. Some important squared HDDs include:

- Jensen-Shannon distance, where $\kappa(p(x), q(x)) = \frac{p(x)}{2}\log\left(\frac{2p(x)}{p(x)+q(x)}\right) + \frac{q(x)}{2}\log\left(\frac{2q(x)}{p(x)+q(x)}\right)$ and $\mathrm{d}\mu(\lambda) = \frac{\mathrm{d}\lambda}{\cosh(\pi\lambda)(1+\lambda^2)}$.
- Squared Hellinger distance, where $\kappa(p(x), q(x)) = \frac{1}{2}\left(\sqrt{p(x)} - \sqrt{q(x)}\right)^2$ and $\mathrm{d}\mu(\lambda) = \frac{1}{2}\delta(\lambda = 1)\mathrm{d}\lambda$.
- Total Variation distance, with $\kappa(p(x), q(x)) = |p(x) - q(x)|$ and $\mathrm{d}\mu(\lambda) = \frac{2}{\pi}\frac{\mathrm{d}\lambda}{1+4\lambda^2}$.

Vedaldi and Zisserman (2012) studied embeddings of a similar class of kernels, also using the key result of Fuglede (2005) we employ, but for discrete distributions only.

Fuglede (2005) shows that κ corresponds to a bounded measure $\mu(\lambda)$ by

$$\kappa(x,y) = \int_{\mathbb{R}_{\geq 0}} \left| x^{\frac{1}{2}+\mathbb{i}\lambda} - y^{\frac{1}{2}+\mathbb{i}\lambda} \right|^2 \mathrm{d}\mu(\lambda).$$

Let $Z := \mu(\mathbb{R}_{\geq 0})$ and $c_\lambda := (-\frac{1}{2} + \mathbb{i}\lambda)/(\frac{1}{2} + \mathbb{i}\lambda)$; then

$$\kappa(x,y) = \mathbb{E}_{\lambda \sim \frac{\mu}{Z}} |g_\lambda(x) - g_\lambda(y)|^2 \qquad \text{where } g_\lambda(x) := \sqrt{Z} c_\lambda \left(x^{\frac{1}{2}+\mathbb{i}\lambda} - 1 \right).$$

We approximate the expectation with an empirical mean. Let $\lambda_j \sim \frac{\mu}{Z}$ for $j \in \{1, \ldots, M\}$; then

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

Hence, the squared hdd is

$$\begin{aligned}
\rho^2(p,q) &= \int_{[0,1]^d} \kappa\left(p(x), q(x)\right) \mathrm{d}x \\
&= \int_{[0,1]^d} \mathbb{E}_{\lambda\sim\frac{\mu}{Z}} \left|g_\lambda(p(x)) - g_\lambda(q(x))\right|^2 \mathrm{d}x \\
&\approx \frac{1}{M}\sum_{j=1}^{M} \int_{[0,1]^d} \left(\left(\Re(g_{\lambda_j}(p(x))) - \Re(g_{\lambda_j}(q(x)))\right)^2 + \left(\Im(g_{\lambda_j}(p(x))) - \Im(g_{\lambda_j}(q(x)))\right)^2\right) \mathrm{d}x \\
&= \frac{1}{M}\sum_{j=1}^{M} \left\|p^R_{\lambda_j} - q^R_{\lambda_j}\right\|^2 + \left\|p^I_{\lambda_j} - q^I_{\lambda_j}\right\|^2,
\end{aligned}$$

where

$$p^R_\lambda(x) := \Re(g_\lambda(p(x))), \qquad p^I_\lambda(x) := \Im(g_\lambda(p(x))).$$

Each p_λ function is in $L_2([0,1]^d)$, so we can approximate $e^{-\gamma\rho^2(p,q)}$ with the $\vec{a}$ embedding of Oliva et al. (2014); let

$$A(P) := \frac{1}{\sqrt{M}}\left(\vec{a}\left(p^R_{\lambda_1}\right)^\mathsf{T}, \vec{a}\left(p^I_{\lambda_1}\right)^\mathsf{T}, \ldots, \vec{a}\left(p^R_{\lambda_M}\right)^\mathsf{T}, \vec{a}\left(p^I_{\lambda_M}\right)^\mathsf{T}\right)^\mathsf{T}$$

so that the kernel is estimated by $z(A(P))$.

However, the projection coefficients of the p_λ functions do not have simple forms as before; instead, we directly estimate the density as $\hat{p}$ using a technique such as kernel density estimation (KDE), and then estimate $\vec{a}\left(\hat{p}_\lambda\right)$ for each λ with numerical integration. Denote the estimated features as $\hat{A}\left(\hat{p}\right)$.

For small d, simple Monte Carlo integration is sufficient.

In higher dimensions, three problems arise: (i) density estimation becomes statistically difficult, (ii) the embedding dimension increases exponentially, and (iii) accurate numerical integration

becomes expensive. We can attempt to address (i) and (ii) with sparse nonparametric graphical models (Lafferty, Liu, and Wasserman 2012), and (iii) with MCMC integration. High-dimensional multimodal integrals remain particularly challenging to current MCMC techniques, though some progress is being made (Betancourt 2015); (Lan, Streets, and Shahbaba 2014) give a heuristic algorithm.

Sutherland et al. (2016) bound the error probability for this estimator for a pair of distributions P, Q satisfying certain smoothness properties.

RESULTS

We now turn to case studies of the application of distributional kernels to real machine learning tasks.

MIXTURE ESTIMATION

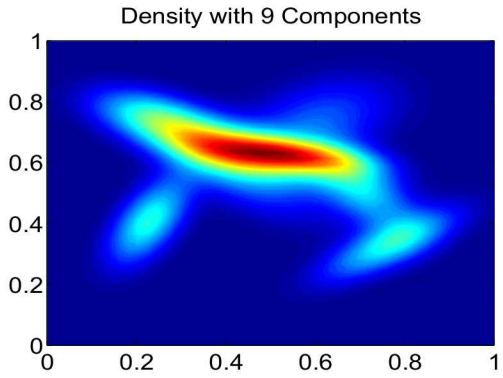
Statistical inference procedures can be viewed as functions from distributions to the reals; we can therefore consider learning such procedures. Jitkrittum et al. (2015) trained -based GP regression for the messages computed by numerical integration in an expectation propagation system, and saw substantial speedups by doing so. We, inspired by Oliva et al. (2014), consider a problem where we not only obtain speedups over traditional algorithms, but actually see far superior results. Specifically, we consider predicting the number of components in a Gaussian mixture. We generate mixtures as follows:

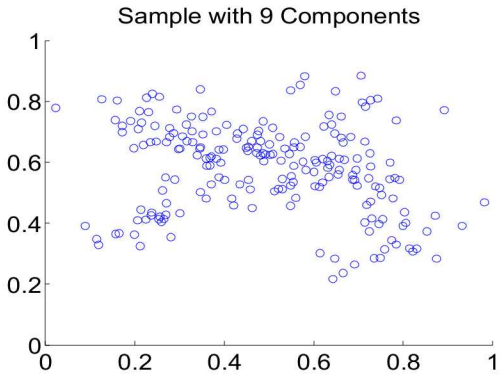
1. Draw the number of components Y_i for the ith distribution as $Y_i \sim \text{Unif}\{1, \dots, 10\}$.

2. For each component, select a mean $\mu_k^{(i)} \sim \text{Unif}[-5,5]^2$ and covariance $\Sigma_k^{(i)} = a_k^{(i)} A_k^{(i)} A_k^{(i)\mathsf{T}} + B_k^{(i)}$, where $a \sim \text{Unif}[1,4]$, $A_k^{(i)}(u,v) \sim \text{Unif}[-1,1]$, and $B_k^{(i)}$ is a diagonal 2×2 matrix with $B_k^{(i)}(u,u) \sim \text{Unif}[0,1]$.

3. Draw a sample $X^{(i)}$ from the equally-weighted mixture of these components.

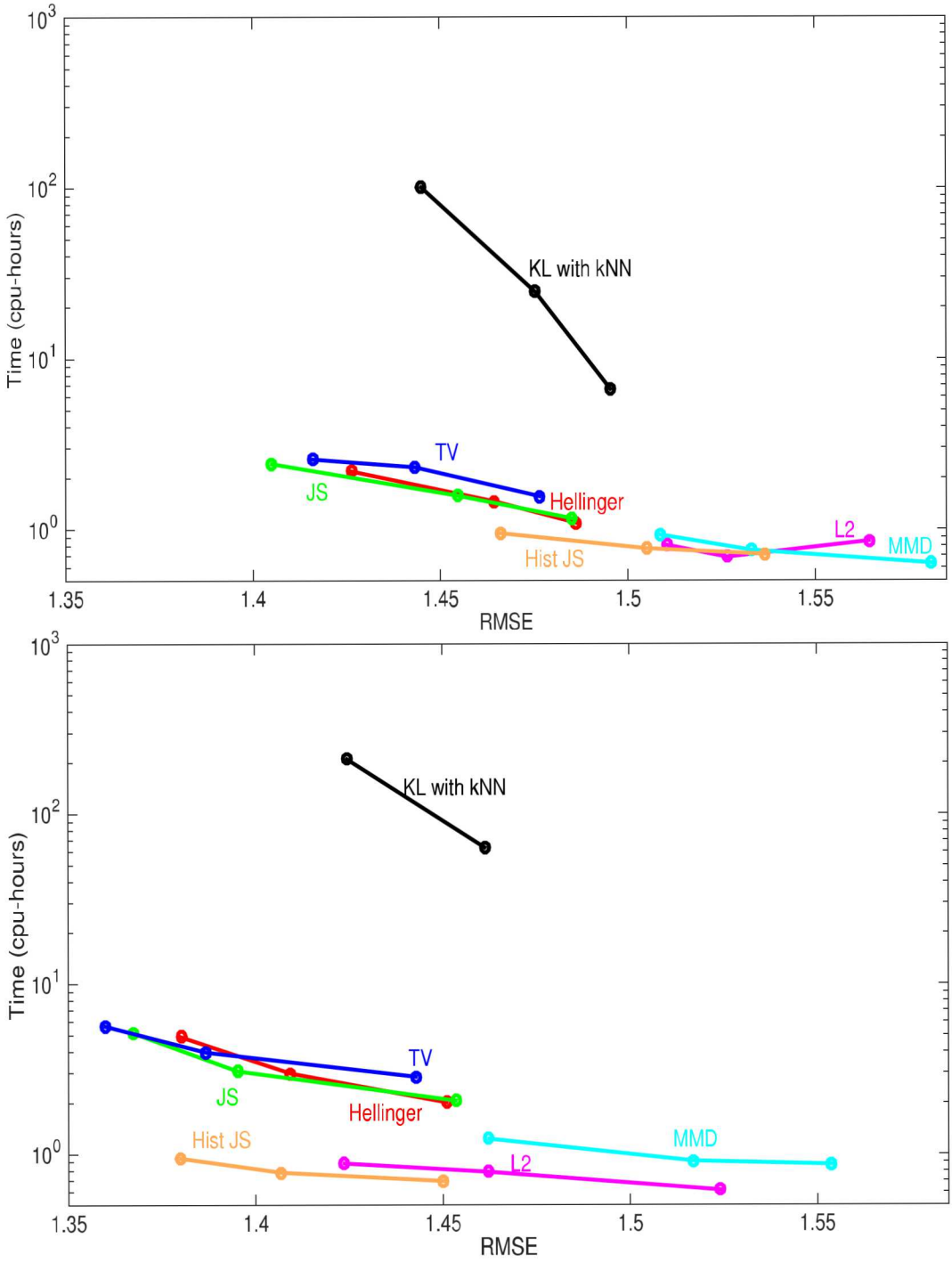
Shown below is an example of the density function of a 9-component mixture distribution, along with a sample of size $n = 200$ drawn from it. Predicting the number of components is difficult even for humans.





We compare generalized RBF kernels based on the , L_2, and HDD embeddings, as well as the embedding of Vedaldi and Zisserman (2012) and the full Gram matrix techniques of Póczos et al. (2012) applied to the KL divergence estimator of Wang, Kulkarni, and Verdú (2009), as in Ntampaka et al. (2015).

We now 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 in both cases. They also outperform the histogram kernels, 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 the best constant predictor. Linear versions of the L_2 and MMD kernels were also no better than the constant predictor.


Time (cpu-hours)
RMSE
KL with kNN
TV
JS
Hellinger
Hist JS
L2
MMD
1.35
1.4
1.45
1.5
1.55

The three points on each line correspond to training set sizes of 4K, 8K, and 16K; error is on the fixed test set of size 2K. The first image shows results for sample sets of size 200, the second for 800. Note the logarithmic scale on the time axis. The kernel for sets of size 800 and 16K training sets was too slow to run. AIC-based predictions achieved RMSEs of 2.7 (for 200 samples) and 2.3 (for 800); BIC errors were 3.8 and 2.7; a constant predictor of 5.5 had RMSE of 2.8.

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 the number of distributions. Note that the histogram embeddings used an optimized C implementation by the paper's authors (Vedaldi and Fulkerson 2008), and the KL kernel used the fairly optimized implementation of skl-groups, whereas the HDD embeddings used a simple Matlab implementation.

SCENE CLASSIFICATION

For the last several years, modern computer vision has become overwhelmingly based on deep neural networks. Image classification networks typically broadly follow the architecture of Krizhevsky, Sutskever, and Hinton (2012), i.e. several convolutional and pooling layers to extract complex features of input images followed by one or two fully-connected layers to classify the images.

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 therefore treat the activations as a sample of size hw from an n-dimensional distribution. Wu, Gao, and Liu (2016) set accuracy records on several scene classification datasets with a particular method of extracting features from distributions. That method, however, resorts to ad-hoc statistics; we compare to our more principled alternatives here.

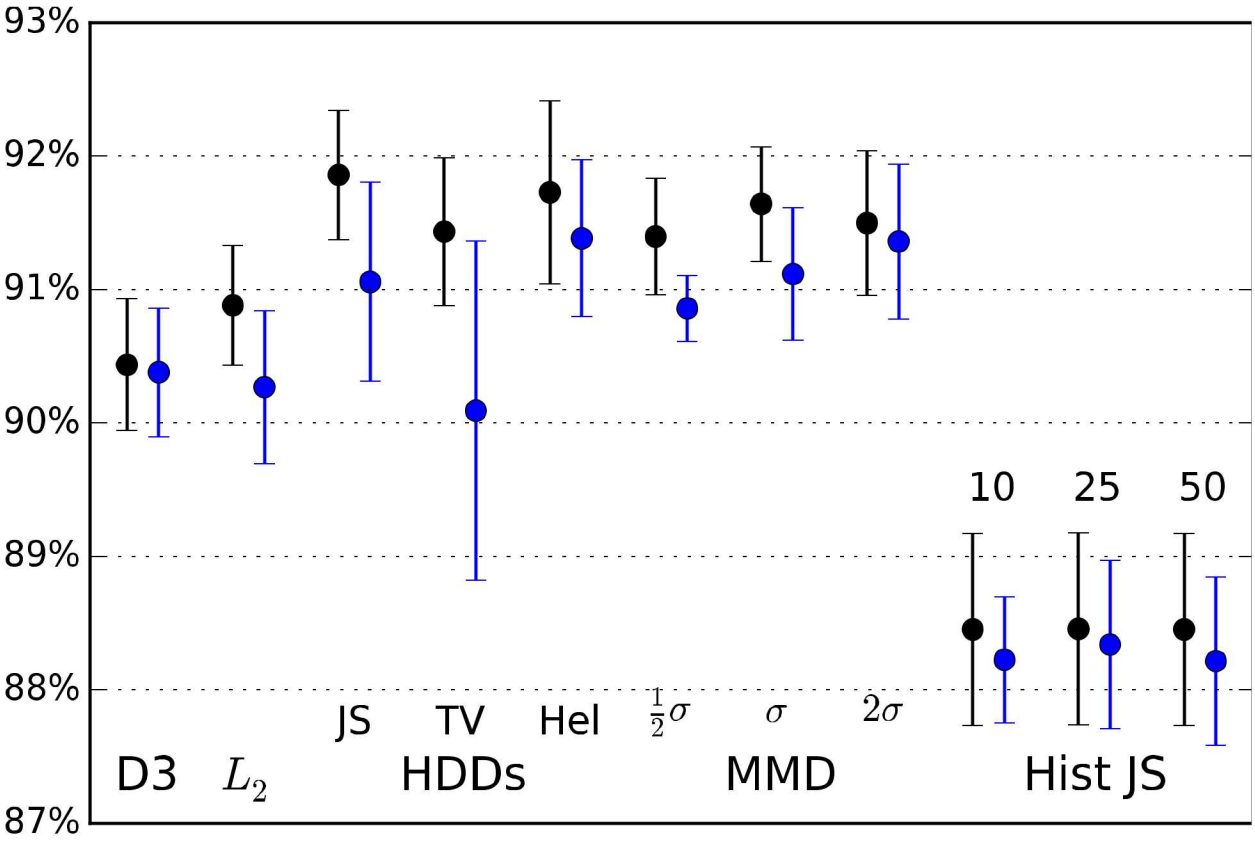
We consider here the Scene-15 dataset (Lazebnik, Schmid, and Ponce 2006), which contains 4 485 natural images in 15 categories based on location. We follow Wu, Gao, and Liu (2016) in extracting features from the last convolutional layer of the model . We replace that layer's rectified linear activations with sigmoid squashing to [0,1].1 After resizing the images so the shortest edge is at least 314 pixels, and the longest at most 1120, as did Wu, Gao, and Liu

---

1 We used piecewise-linear weights 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. Ideally, we would use a model with sigmoid or similar scaling, to avoid this change.

(2016), *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 select 100 images from each class for training, and test on the remainder; the figure below shows the results of 10 random splits. We do not add any spatial information to the model, unlike Wu, Gao, and Liu (2016); still, we match the best prior published performance of 91.59 ± 0.48, using a deep network trained on a large scene classification dataset Zhou et al. (2014). Adding spatial information brought the D3 method of Wu, Gao, and Liu (2016) 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.



The figure shows mean and standard deviation accuracies on the Scene-15 dataset. The left, black lines show performance with linear features; the right, blue lines show generalized RBF embedding features. D3 refers to the method of Wu, Gao, and Liu (2016). bandwidths are relative to σ, the median of pairwise distances; histogram methods use varying numbers of bins.

DISCUSSION

ADVANTAGES OF EMBEDDINGS

It is worth emphasizing here that, in modern large-scale distributed systems, embeddings such as those discussed here have an additional advantage over traditional pairwise metheds: the embedding can be computed with $O(1)$ communication between nodes storing the data, merely by transmitting a single random seed to each node. Learning operations would then, of course, require more data, but in that case only simple and well-understood learning algorithms are needed, whose distributed variants have been thoroughly studied and efficiently implemented.

RANDOM FOURIER FEATURES

Our full paper presents substantially more analysis of random Fourier features (Sutherland and Schneider 2015), showing the superiority of the $\tilde{z}$ embedding to $\check{z}$ as well as evaluating the various bounds shown in that paper. The practical gap in performance depends on the problem: for some problems, the difference is insignificant, but in some cases the gap between the two embeddings is nontrivial.

In terms of the theoretical analysis, Sriperumbudur and Szabó (2015) later gave a more powerful analysis which they showed to be rate-optimal in terms of both the dependence on n and the dependence on the diameter of the input space. In practice, though, the values of the simpler bound given by Rahimi and Recht (2007) and tightened in Sutherland and Schneider (2015) are sometimes tighter.

HDD EMBEDDINGS

The results section demonstrated that on two practical problems, the HDD embedding obtained superior learning performance to alternative embeddings at somewhat increased computational cost. The amount of the improvement in performance seems to vary from problem to problem, but it seems that on some practical problems the tradeoff between computational limits and performance requirements will favor HDD embeddings over histogram-based approaches, the L2 embedding of Oliva et al. (2014), or the MMD embedding using a Gaussian RBF base kernel.

Theoretically, the reason for the gap in performance between HDDs and other approaches is not clear. Some learning theoretic results have been established for learning based on L2 (Oliva et al. 2014) and MMD (Szabó et al. 2014), but neither allows one to easily establish a relationship between the rates of one technique to the rates of another in any particular problem. Results similar to those of (Oliva et al. 2014) could surely be obtained for learning with HDD embedding, but this would not necessarily improve the state of understanding of this difference: such rates assume that the learning problem is representable in the class of distributions distinguishable by the distance function at hand. Although some aspects of the relationship

between these distance functions can be understood, the way that those properties connect to learning in practice -- and indeed which properties matter for learning rates -- is unclear.

In their current form, however, HDD embeddings are limited to low-dimensional problems. Their requirement of both explicit density estimation and numerical integration is a substantial drawback in certain circumstances. We have begun to explore some approaches along these lines, based on ideas related to random projections explaiting the central limit theorem as well as integration with high-dimensional sparse nonparametric density estimators (Lafferty, Liu, and Wasserman 2012).

Compared to these approaches, however, another allows for more flexibility in terms of operating on different (non-vectorial) input structures and allowing for the modeling of complex relationships among inputs, without requiring explicit density estimation and (sometimes) having a very simple embedding: the maximum mean discrepancy. Its performance was sometimes lackluster in the results discussed above and related problems, but those results considered only base kernels from the Gaussian RBF family, and indeed did not even fully optimize the bandwidth of the kernel there, since doing so is relatively expensive. Given a more complex embeddable kernel class and an effective method for choosing a kernel from it, it seems highly likely that embeddings based on the MMD would match or outperform the results of HDDs in the problems considered here, and perhaps more importantly allow for the application of distribution embeddings to settings where directly-defined distances are less meaningful. This would improve the applicability of distribution learning methods and reduce the amount of required hand-tuning on the part of a machine learning practitioner. Though there are challenges in doing so, this seems a more fruitful path for future work than to extend the applicability of HDD embeddings to higher-dimensional spaces.

ANTICIPATED IMPACT

The immediate continuation of this work is to, as just discussed, extend the MMD embeddings to more complex kernel classes. In particular, one can view the representations learned through deep learning as a form of kernel learning, and indeed one that provides a direct embedding for the learned kernel and so avoids the need for Fourier-type features.

The most immediate place of application is as follows: In the scene classification experiment considered above, we used the features learned by a standard convolutional deep network as samples from an image-level distribution of local features, and classified images based on those sets of features. Here features are trained using fully-connected final layers as the learning model, but then used in a separate distributional kernel model.

We can instead make a coherent model which combines feature extraction with a learning model based on a distributional kernel, by treating the approximate distributional embedding as a layer in the network. With the mean map-based embedding, gradients propagate through this layer easily, and so standard stochastic gradient algorithms can be used either to fine-tune features trained on a different task or to learn features well-suited to distributional kernel models from the start.

In doing so, we learn features for an MMD kernel based on a convolutional network through standard deep learning techniques. This line of inquiry was begun in Oliva et al. (2015); it showed some initial promise, but it seems that substantial improvements to the empirical results of image classification networks (which are of course extremely widely studied in recent years) will require a somewhat different approach to the features used in earlier layers of the network as well. Perhaps simply larger filter sizes and other such minor changes would help, but evaluating that question requires a substantial amount of effort and computer time as it is trained on large image datasets.

A simpler architecture in which empirical work is ongoing is the setting of Flaxman, Wang, and Smola (2015), where person-level demographic features are combined into regional features and used to model voting behavior. In this case, more complex demographic similarities than the simple equally-weighted Gaussian RBF used in that paper could both improve the quality of the modeling of the paper and help political scientists in their interpretation of the resulting model, by (depending on the structure of the network used) flagging important variables and interactions and dropping ones unimportant to the final prediction.

NATURAL LANGUAGE PROCESSING

This framework could also prove applicable to natural language processing. Until recently, much work treated words as unique symbols, e.g. with ``one-hot'' vectors, where the ith word from a vocabulary of size V is represented as a vector with ith component 1 and all other components 0. It has recently become widely accepted that applications can benefit from richer word embeddings which take into account the similarity between distinct words, and much work has been done on dense *word embeddings* so that distances or inner products between word embeddings represent word similarity in some way (e.g. Collobert and Weston 2008, Turian, Ratinov, and Bengio (2010), Mikolov et al. (2013)). These embeddings can be learned in various ways, but often involve optimizing the representation's performance in some supervised learning task.

First, it is worth noting that although this breaks the traditional ``bag of words'' text model (where documents can be represented simply by the sum of the words' one-hot encodings), we can represent documents by viewing them as sample sets of word vectors.

Kusner et al. (2015) recently adopted this model, using *k*NN classifiers based on the Earth Mover's Distance (EMD) between documents, and obtained excellent empirical results. EMD, however, is expensive to compute even for each pair of documents when the vocabulary is large, and additionally must be computed pairwise between documents; an approximate embedding is not known.

Yoshikawa, Iwata, and Sawada (2014), in their empirical results, considered this model with MMD-based kernels (but computing pairwise kernel values rather than approximate embeddings). Their main contribution, however, is to optimize the word embedding vectors for final classification performance; by doing so with random initializations, they saw mild performance improvements over MMD kernels using substantially less training data for the embeddings but at much higher computational cost. Yoshikawa, Iwata, and Sawada (2015) extend the approach to Gaussian process regression models, but do not compare to separately-learned word embeddings.

Future work that empirically compares these embedding methods, particularly on larger datasets, could establish whether the bag-of-words document representation is best implemented with these techniques. Also, fine-tuning word embeddings learned on a standard dataset simultaneously with learning the model for a particular application, as is common in deep learning models for computer vision, could see advances in the state of the art of document topic classification and similar tasks.

In fact, embedding words as a single vector does not allow for as rich a word representation as we might wish. Vilnis and McCallum (2015) embed words instead as Gaussian distributions, and use the KL divergence between word embeddings to measure asymmetric hypernym relationships: for example, their embedding for the word *Bach* is ``included'' in their embeddings for *famous* and *man*, and mostly included in *composer*. Gaussian distributions, of course, are still fairly limiting; for example, a multimodal embedding might be able to capture word sense ambiguity, whereas a Gaussian embedding would be forced to attempt to combine both senses in a single broad embedding.

We can thus consider richer, nonparametric classes of word embeddings: perhaps by representing a word as a (possibly weighted) set of latent vectors. Comparisons could then be performed either with an -based kernel, when symmetry is desired, or with KL estimators (or similar) when not.

One approach would be to choose these vectors arbitrarily, optimizing them for the output of some learning problem: this would be implemantionally similar to the approach of Yoshikawa, Iwata, and Sawada (2014) and Yoshikawa, Iwata, and Sawada (2015) for MMD distances, or

somewhat like that of Vilnis and McCallum (2015) but with greater computational cost, and greater flexibility, for KL distances.

Another approach is inspired by the classic distributional hypothesis of Harris (1954), that the semantics of words are characterized by the contexts in which it appears. Many word embedding approaches can be viewed as matrix factorizations of a matrix M with rows corresponding to words, columns to some notion of context, and entries containing some measure of association between the two; the factorization $M = WC^{\mathsf{T}}$ then typically discards the matrix C and uses the rows of W as word vectors. This approach is sometimes taken explicitly; interestingly, the popular method of Mikolov et al. (2013) can be seen as approximating this form as well (Levy and Goldberg 2014). This view inspires a natural alternative: treat each word as the sample set of contexts in which it appears, representing each context via the learned context vectors. This is perhaps the most direct instantiation of the distributional hypothesis: compare words by comparing the distribution of contexts in which they appear.

TWO-SAMPLE TESTING

Another, related area under progress now is in the choice of kernels for two-sample tests, the more traditional use of MMD (Gretton, Borgwardt, et al. 2012). Such tests are, like classification, reliant on a good based kernel to get good results at reasonable sample sizes. Current work is extending the kernel selection criterion of Gretton, Sriperumbudur, et al. (2012) to quadratic-time tests (which are much more powerful than the linear-time tests considered in that paper in the limited-data regime) and applying it to optimizing deep kernels. Initial results along those lines are promising.

CONCLUSION

This work substantially furthers the understanding of approximate embeddings for kernels on distributions. After establishing a framework for learning on distributions, we improved our understanding of the random Fourier features key to most approximate kernel embeddings on any domain, including on distributions. We then presented 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. 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 show their quality on several synthetic and real-world datasets.

REFERENCES

Akiake, Hirotugu. 1973. “Information Theory and an Extension of the Maximum Likelihood Principle.” In *2nd International Symposium on Information Theory*.

Betancourt, Michael. 2015. “Adiabatic Monte Carlo.”

Bochner, Salomon. 1959. *Lectures on Fourier Integrals*. Princeton University Press.

Chen, Yihua, Eric K Garcia, Maya R Gupta, Ali Rahimi, and Luca Cazzanti. 2009. “Similarity-Based Classification: Concepts and Algorithms.” *Journal of Machine Learning Research* 10: 747–76.

Chwialkowski, Kacper, Aaditya Ramdas, Dino Sejdinovic, and Arthur Gretton. 2015. “Fast Two-Sample Testing with Analytic Representations of Probability Measures.”

Collobert, Ronan, and Jason Weston. 2008. “A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning.” In *ICML*. doi:10.1145/1390156.1390177.

Flaxman, Seth R., Yu-Xiang Wang, and Alexander J. Smola. 2015. “Who Supported Obama in 2012? Ecological Inference Through Distribution Regression.” In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD ’15*, 289–98. ACM Press. doi:10.1145/2783258.2783300.

Fuglede, Bent. 2005. “Spirals in Hilbert Space: With an Application in Information Theory.” *Expositiones Mathematicae* 23 (1): 23–45.

Gretton, Arthur, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alex J Smola. 2012. “A Kernel Two-Sample Test.” *The Journal of Machine Learning Research* 13. JMLR.org.

Gretton, Arthur, Bharath K. Sriperumbudur, Dino Sejdinovic, Heiko Strathmann, and Massimiliano Pontil. 2012. “Optimal Kernel Choice for Large-Scale Two-Sample Tests.” In *Advances in Neural Information Processing Systems*, 25:1214–22.

Haasdonk, Bernard, and Claus Bahlmann. 2004. “Learning with Distance Substitution Kernels.” In *Pattern Recognition: 26th DAGM Symposium*, 220–27.

Harris, Z. 1954. “Distributional Structure.” *Word* 10 (23): 146–62.

Jitkrittum, Wittawat, Arthur Gretton, Nicolas Heess, S M Ali Eslami, Balaji Lakshminarayanan, Dino Sejdinovic, and Zoltán Szabó. 2015. “Kernel-Based Just-in-Time Learning for Passing Expectation Propagation Messages.” In *Uncertainty in Artificial Intelligence*.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton. 2012. "ImageNet Classification with Deep Convolutional Neural Networks." *Advances In Neural Information Processing Systems*.

Kusner, Matt, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. "From Word Embeddings to Document Distances." In *Proceedings of the 32nd International Conference on Machine Learning*, 957–66.

Lafferty, John, Han Liu, and Larry Wasserman. 2012. "Sparse Nonparametric Graphical Models." *Statistical Science* 27 (4): 519–37. doi:10.1214/12-STS391.

Lan, Shiwei, Jeffrey Streets, and Babak Shahbaba. 2014. "Wormhole Hamiltonian Monte Carlo." In *AAAI*.

Lazebnik, Svetlana, Cordelia Schmid, and Jean Ponce. 2006. "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories." In *CVPR*.

Levy, Omer, and Yoav Goldberg. 2014. "Neural Word Embedding as Implicit Matrix Factorization." In *Advances in Neural Information Processing Systems*, 2177–85.

Li, Shukai, and Ivor W Tsang. 2011. "Learning to Locate Relative Outliers." In *Asian Conference on Machine Learning*, 20:47–62. JMLR: Workshop and Conference Proceedings.

Lopez-Paz, David, Krikamol Muandet, Bernhard Schölkopf, and Ilya Tolstikhin. 2015. "Towards a Learning Theory of Cause-Effect Inference." In *ICML*.

Ma, Yifei, Dougal J. Sutherland, Roman Garnett, and Jeff Schneider. 2015. "Active Pointillistic Pattern Search." In *Eighteenth International Conference on Artificial Intelligence and Statistics*. AISTATS.

Mehta, Nishant A., and Alexander G. Gray. 2010. "Generative and Latent Mean Map Kernels."

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. "Distributed Representations of Words and Phrases and Their Compositionality." In *Advances in Neural Information Processing Systems*.

Muandet, Krikamol, Bernhard Schölkopf, Kenji Fukumizu, and Francesco Dinuzzo. 2012. "Learning from Distributions via Support Measure Machines." In *Advances in Neural Information Processing Systems*.

Ntampaka, Michelle, Hy Trac, Dougal J. Sutherland, Nicholas Battaglia, Barnabás Póczos, and Jeff Schneider. 2015. "A Machine Learning Approach for Dynamical Mass Measurements of Galaxy Clusters." *The Astrophysical Journal* 803 (2): 50.

Ntampaka, Michelle, Hy Trac, Dougal J. Sutherland, Sebastian Fromenteau, Barnabás Póczos, and Jeff Schneider. 2016. "Dynamical Mass Measurements of Contaminated Galaxy Clusters Using Machine Learning." *The Astrophysical Journal*.

Oliva, Junier B., Willie Neiswanger, Barnabás Póczos, Jeff Schneider, and Eric Xing. 2014. "Fast Distribution to Real Regression." In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS)*.

Oliva, Junier B., Dougal J. Sutherland, Barnabás Póczos, and Jeff Schneider. 2015. "Deep Mean Maps."

Póczos, Barnabás, Liang Xiong, Dougal J. Sutherland, and Jeff Schneider. 2012. "Nonparametric Kernel Estimators for Image Classification." In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2989–96. doi:10.1109/CVPR.2012.6248028.

Rahimi, Ali, and Benjamin Recht. 2007. "Random Features for Large-Scale Kernel Machines." In *Advances in Neural Information Processing Systems*. MIT Press.

Rasmussen, Carl Edward, and Christopher K. I. Williams. 2006. *Gaussian Processes for Machine Learning*. MIT Press.

Schwarz, Gideon. 1978. "Estimating the Dimension of a Model." *Ann. Statist.* 6 (2). The Institute of Mathematical Statistics: 461–64.

Sriperumbudur, Bharath K., and Zoltán Szabó. 2015. "Optimal Rates for Random Fourier Features."

Sutherland, Dougal J., and Jeff Schneider. 2015. "On the Error of Random Fourier Features." In *Uncertainty in Artificial Intelligence*.

Sutherland, Dougal J., Junier B. Oliva, Barnabás Póczos, and Jeff Schneider. 2016. "Linear-Time Learning on Distributions with Approximate Kernel Embeddings." In *Association for the Advancement of Artificial Intelligence*.

Sutherland, Dougal J., Liang Xiong, Barnabás Póczos, and Jeff Schneider. 2012. "Kernels on Sample Sets via Nonparametric Divergence Estimates."

Szabó, Zoltán, Bharath Sriperumbudur, Barnabás Póczos, and Arthur Gretton. 2014. "Learning Theory for Distribution Regression." In *Artificial Intelligence and Statistics*. AISTATS.

Turian, Joseph, Lev Ratinov, and Yoshua Bengio. 2010. "Word Representations: A Simple and General Method for Semi-Supervised Learning." In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*.

Vedaldi, Andrea, and Brian Fulkerson. 2008. "VLFeat: An Open and Portable Library of Computer Vision Algorithms." http://www.vlfeat.org/.

Vedaldi, Andrea, and Andrew Zisserman. 2012. "Efficient Additive Kernels via Explicit Feature Maps." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34 (3): 480–92. doi:10.1109/TPAMI.2011.153.

Vilnis, Luke, and Andrew McCallum. 2015. "Word Representations via Gaussian Embedding." In *International Conference on Learning Representations*.

Wang, Qing, Sanjeev R Kulkarni, and Sergio Verdú. 2009. "Divergence Estimation for Multidimensional Densities via K-Nearest-Neighbor Distances." *IEEE Transactions on Information Theory* 55 (5): 2392–2405.

Wu, Jianxin, Bin-Bin Gao, and Guoqing Liu. 2016. "Visual Recognition Using Directional Distribution Distance." In *Association for the Advancement of Artificial Intelligence*.

Yoshikawa, Yuya, Tomoharu Iwata, and Hiroshi Sawada. 2014. "Latent Support Measure Machines for Bag-of-Words Data Classification." In *Advances in Neural Information Processing Systems*, 1961–9.

———. 2015. "Non-Linear Regression for Bag-of-Words Data via Gaussian Process Latent Variable Set Model." In *AAAI*, 3129–35.

Zhao, Ji, and Deyu Meng. 2014. "FastMMD: Ensemble of Circular Discrepancy for Efficient Two-Sample Test."

Zhou, Bolei, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. 2014. "Learning Deep Features for Scene Recognition Using Places Database." In *NIPS*.