# Large-Scale Data-Dependent Kernel Approximation

**Catalin Ionescu**[3,2*]
Google DeepMind[3]
cdi@google.com

**Alin-Ionut Popa**[2*]
Institute of Mathematics
of the Romanian Academy[2]
alin.popa@imar.ro

**Cristian Sminchisescu**[1,2]
Department of Mathematics
Faculty of Engineering, Lund University[1]
cristian.sminchisescu@math.lth.se

## Abstract

Learning a computationally efficient kernel from data is an important machine learning problem. The majority of kernels in the literature do not leverage the geometry of the data, and those that do are computationally infeasible for contemporary datasets. Recent advances in approximation techniques have expanded the applicability of the kernel methodology to scale linearly with the data size. Data-dependent kernels, which could leverage this computational advantage, have however not yet seen the benefit. Here we derive an *approximate large-scale learning procedure for data-dependent kernels* that is efficient and performs well in practice. We provide a Lemma that can be used to derive the asymptotic convergence of the approximation in the limit of infinite random features, and, under certain conditions, an estimate of the convergence speed. We empirically prove that our construction represents a valid, yet efficient approximation of the data-dependent kernel. For large-scale datasets of millions of datapoints, where the proposed method is now applicable *for the first time*, we notice a significant performance boost over both baselines consisting of data independent kernels and of kernel approximations, at comparable computational cost.

## 1 Introduction

Kernel methods offer a rigorous methodology to construct high performance learning machines with strong theoretical guarantees. However, as the data size increases, limitations become apparent. Scaling the kernel methods has not been straightforward, but the machinery of reproducing kernel Hilbert spaces offers a framework for approximation. One scalable approach has been to 'linearize' the kernel by designing explicit feature maps whose inner products can be shown to approximate the kernel everywhere [1, 2, 3, 4, 5, 6]. The methods described in [1, 2, 3] give approximations for different types of kernels, [4, 5] offer theoretical convergence analysis for certain classes of kernels and [6] proposes a linear approximation methodology for kernel learning. This allows the application of fast linear learning algorithms while, at the same time, offering the non-linear properties, and the asymptotic performance guarantees that make kernel machines attractive.

The explicit feature map approach has shown its effectiveness for a variety of practically useful kernels (*e.g.* the exponentiated chi-square used to compare histograms [3, 7]), and learning problems. However, it remains open how such a methodology can be applied in a weakly or semi-supervised setup. Such frameworks are appealing and could benefit most from scalability, as unlabeled data is plentiful and easy to collect. Many approaches [8, 9, 10, 11] use the data geometry to propagate information and improve performance, by constructing data-dependent kernels. In many cases, data can also be intrinsically low-dimensional, therefore significant efforts were invested in discovering low-dimensional representations that preserve relevant properties.

In this paper we present a scalable approach to learning data-dependent kernels, which avoids dealing with Gram (or kernel) matrices directly, in the way kernel methods do. We show that the successful data-dependent kernel of [8] can be approximated very efficiently using random Fourier features. We propose an approximation for the data-dependent kernel [8], in a formulation that multiplies the random features of the data-independent kernel, obtained with [2], by a weighted covariance matrix built using both labeled and unlabeled data. This effectively warps the distances between the Fourier features and therefore, we sometimes refer to it as the 'warping' matrix. Our resulting data-dependent kernel approximation has the same properties as the one in [8], but no longer suffers from the memory and time constraints associated with building the Gram matrix linked with the kernel function. The construction is made possible by an astute application of a Woodbury identity which moves the

Proceedings of the $20^{th}$ International Conference on Artificial Intelligence and Statistics (AISTATS) 2017, Fort Lauderdale, Florida, USA. JMLR: W&CP volume 54. 
($^*$)Authors contributed equally.

learning problem from a reproducing kernel Hilbert space (RKHS) to the Fourier space of the kernel, reducing the computational load from $O(N^3)$ to $O(N)$.

### 1.1 Related Work

The most successful applications of kernel methods [12, 13] have been in the supervised case. As dependencies on matrices of size $N^2$ (with $N$ the dataset size) makes the standard methodology hard to apply for large datasets, scaling up kernel methods is of great interest. Initial approximations were based on Nyström schemes [14]. More recently it was observed that data independent kernel approximations are possible in certain cases. Rahimi and Recht [1] showed how an approximation can be obtained in the case of Euclidean translation-invariant kernels by exploiting Bochner's theorem [15]. This was extended by [2] to translation invariance over general groups. Inspired by these ideas, different authors developed approximations for other kernel classes, supported by additional theoretical results [3, 7, 16, 17, 18, 19].

Data-dependent kernels have a history going back at least to the seminal paper of [20] as well as [21] who use conformal maps to warp distances around support vectors. This relies on two observations: **(1)** unlabeled data is cheaper (and more readily available) than labeled data, and **(2)** with enough unlabeled samples, the geometry of the data can be recovered and used to propagate supervisory information. A thorough review is beyond the scope of this paper, but an excellent summary can be found in [22]. The understanding is still incomplete but theoretical work [23] suggests that an important factor associated with improved performance is the sampling of unlabeled data which should be denser (in terms of distances between points) than the margin. Scalability is once again, key.

A more recent approach to obtaining data-dependent kernels is based on manifold regularization – a semi-supervised learning framework that performs well in practice [24]. The idea is to augment a given learning algorithm with a regularization term based on the graph Laplacian computed over the all data samples, both labeled and unlabeled. This brings in information about data geometry and can offer important performance gains. The work of [8] shows that the geometric regularization term can be embedded in the representation in order to obtain data-dependent kernels. That methodology is promising, but as presented, does not scale, as it requires inverting a large matrix of the size of the *labeled and unlabeled data* – an $O(N^3)$ operation. An attempt to overcome this difficulty was presented in [9], where landmark points are used to approximate the geometric component of the kernel in [8]. The authors demonstrated practical applicability for up to 60,000 datapoints.

In this work we develop novel theoretical and practical methodology in order to make data-dependent kernels applicable to large scale datasets. We show that by relying on approximate kernel feature maps, we can effectively handle sets of over a million datapoints.

### 1.2 Outline and Contributions

In the next section, §2.1, we briefly introduce the random feature methodology to approximate a RKHS kernel function as the inner product of two non-linear feature maps. In section §2.2, the data-dependent kernels introduced in [8] are presented as a sum of an original (base) kernel term and a warping term depending on a geometric operator, like the graph Laplacian computed over all data. These two ingredients are put together in the language of random matrices in section §3, and this is one of our main contributions. In particular, in section §3.1 we show that the kernel of [8] can be approximated efficiently using random feature maps. This is illustrated in Proposition 2 by using the Sherman-Morrison-Woodbury (SMW) identity. Lemma 3 then shows that the approximation error is bounded by the approximation error of the original kernel. In section §3.2 we review computational complexity aspects that can improve the calculation of locally weighted models like a Laplacian matrix. In section §3.3 we provide an algorithmic representation of the proposed method. In the experiments section §4, we empirically validate the proof of Lemma 3 by showing that the approximation is both valid and accurate. We give competitive results in approximating data-dependent kernels [8] and illustrate that our methodology scales to sets of millions of unlabeled datapoints without difficulty.

## 2 Kernel Methods

In this section we briefly review the kernel approximation methodology that we will rely upon when introducing our contributions in §3.

### 2.1 Kernel Approximation with Fourier Bases

Many recent efforts have been concentrated on developing feasible large-scale approximations for kernel methods. The main idea is to approximate the non-linear kernel using a feature map embedding. The feature map should be able to support linear computations.

Some of the most popular approaches are based on approximations of the form $k(x, y) = \mathbb{E}_\mu \phi_\omega(x)\phi_\omega(y) = \int_\omega \phi_\omega(x)\phi_\omega(y)\mu(\omega)d\omega$. A finite dimensional approximation can easily be obtained by Monte Carlo sampling. An example of an approximation in this vein is proposed in [1]. Let $k \in \mathcal{H}$ be a translation-invariant positive definite kernel, where $\mathcal{H}$ is a unique RKHS of functions. Then, Bochner's theorem, a fundamental result from functional analysis, states that there exists $\mu(\omega)$, a positive measure, that relates to $k$ by means of the usual direct and inverse Fourier operators. An immediate consequence is an explicit embedding in $\mathbb{R}^d$, whose inner product approximates the

original $\mathcal{H}$ inner product. The kernel can be written as

$$\begin{aligned} k(x,y) &= \int_\omega e^{j\omega(x-y)}\mu(\omega)d\omega \\ &= \int_\omega \phi_\omega(x)\phi_\omega(y)\mu(\omega)d\omega \\ &\simeq \frac{1}{d}\sum_i \phi_{\omega_i}(x)\phi_{\omega_i}(y) \\ &\simeq \frac{1}{d}\phi(x)\phi(y)^\top \end{aligned}$$

with $\omega_i \sim \mu(\omega)$ and some abuse of the previous notation to write $\phi : \mathcal{X}(\subset \mathbb{R}^p) \rightarrow \mathbb{R}^d$ with $\phi_\omega(x) = \left(\frac{\phi_{\omega_1}(x)}{\sqrt{d}}, \ldots, \frac{\phi_{\omega_d}(x)}{\sqrt{d}}\right)^\top = \phi(x)$, where we make $\omega$ implicit. The form of $\phi$ and $\mu$ are independent of the data and only dependent on the kernel. We will denote the $N \times p$ data matrix as $X = (x_1^\top, \ldots, x_N^\top)$ and the $N \times d$ matrix that encodes the kernel approximation features $\phi(X) = (\phi(x_i)^\top, \ldots, \phi(x_N)^\top) = (\phi_{\omega_1}(X), \ldots, \phi_{\omega_d}(X)) = \Phi$, where $\Phi_i = \phi_{\omega_i}(X)$, $N$ is the number of data points and $d$ the number of random features used for the kernel approximation.

Let the kernel matrix associated to the dataset $X$ be $K(i,j) = k(x_i, x_j)$. Given this notation, we can define the Fourier approximation $\widehat{K}$ of $K$

$$\widehat{K} = \sum_i^d \widehat{K}_i = \sum_i^d \Phi_i \Phi_i^\top = \Phi\Phi^\top \tag{1}$$

We notice that because $\widehat{K}_i$ are i.i.d, matrix concentration results like [25] apply.

The goal of this paper is to derive an approximation that is 'data-aware', yet scales favorably in the dataset size $N$ and the representation dimension $d$ (in our methodology, $d$ will not depend on $N$).

### 2.2 Data-Dependent Kernels

In [8], the authors propose a new kernel $\tilde{k}$ by modifying an existing one, $k$, using a data-dependent norm. Let $\mathcal{H}$ be a RKHS and $\mathcal{V}$ be a linear space with positive semidefinite inner product (quadratic form). $\mathcal{S} : \mathcal{H} \rightarrow \mathcal{V}$ is a bounded linear operator. Let $\tilde{\mathcal{H}}$ be the RKHS over $\mathcal{X} \rightarrow \mathbb{R}$, with the associated inner product defined as:

$$\langle f, g\rangle_{\tilde{\mathcal{H}}} \triangleq \langle f, g\rangle_{\mathcal{H}} + \langle \mathcal{S}f, \mathcal{S}g\rangle_{\mathcal{V}} \tag{2}$$

The authors of [8] consider the case when $\mathcal{S}$ is a weighted *point cloud operator i.e.* if $f(X) = (f(x_1), \ldots, f(x_N))$, then $\langle \mathcal{S}f, \mathcal{S}f\rangle_{\mathcal{V}} = f(X)Mf(X)^\top$ with the condition that $M_{i,j} = M(x_i, x_j)$ is positive semidefinite due to [15]. Positive semidefiniteness is required for $\langle \mathcal{S}f, \mathcal{S}f\rangle_{\mathcal{V}}$ to be a (semi-)norm. This is interesting because the inner product for $\tilde{\mathcal{H}}$ depends on the data through $M$ which is built using the data points.

**Proposition 1** *Let $\tilde{\mathcal{H}}$ be as above. Then*

1. *$\tilde{\mathcal{H}}$ is a RKHS and $\mathcal{H} = \tilde{\mathcal{H}}$.*
2. *If $\tilde{k}(x,y)$ is the reproducing kernel of $\tilde{\mathcal{H}}$ with $k_x^\top = (k(x,x_1), \ldots, k(x,x_N))$ then*

$$\tilde{k}(x,y) = k(x,y) - k_x^\top (I + MK)^{-1} M k_y, \tag{3}$$

The proposition only assumes that $M$ is positive semidefinite, so we choose to use a Laplacian matrix in our experiments. The choice of the Laplacian is popular in the semi-supervised literature because of theoretical work connecting it to the continuous Laplace-Beltrami operator representing the manifold structure of the data. Our objective, described in the next section, is to provide an efficient large-scale approximation to $\tilde{k}$ based on (1). Note that in practice $\tilde{k}$ is a formulation of the data-dependent kernel. It uses the original kernel formulation $k$, constructed from labeled data, and the matrix $M$, which is constructed from both labeled and unlabeled data. In matrix form, it can be written as

$$\widetilde{K} = K - K(I + MK)^{-1}MK \tag{4}$$

However, this is non-trivial, as directly implementing (3) requires inverting an $N \times N$ matrix, which is large for big datasets.

## 3 Large-Scale Data-Dependent Kernel Learning

In this section we describe our contributions to the large-scale kernel learning methodology, including the derivation of a novel Fourier approximation for the data-dependent kernel (§3.1), as well as a locally weighted function construction for efficient computation (§3.2). We prove that our construction represents a valid approximation of the data-dependent kernel (3). Our empirical evaluation from §4.1 supports the claim of asymptotic convergence of the approximation in the limit of infinite random features.

### 3.1 Data-Dependent Kernel Approximation

The Fourier approximation $\overline{K}$ of the original data-dependent kernel (4) can be derived as

$$\overline{K} = \widehat{K} - \widehat{K}(I + M\widehat{K})^{-1}M\widehat{K} \tag{5}$$

Since $K \simeq \widehat{K}$, it is clear that $\overline{K} \simeq \widetilde{K}$, but computationally there is no gain, as it still requires inverting a potentially large $N \times N$ matrix. Instead, we propose and focus on a new **efficient formulation** $\breve{K}$ so that only a lower-dimensional $d \times d$ matrix inversion is necessary.

$$\breve{K} = \Phi(I + \Phi^\top M\Phi)^{-1}\Phi^\top \tag{6}$$

**Proposition 2** *With the above definitions*

$$\overline{K} = \breve{K} \tag{7}$$

**Proof**

$$\overline{K} = \widehat{K} - \widehat{K}(I + M\widehat{K})^{-1}M\widehat{K} \quad (8)$$
$$= \Phi\Phi^\top - \Phi\Phi^\top(I + M\Phi\Phi^\top)^{-1}M\Phi\Phi^\top \quad \text{by (1)} \quad (9)$$
$$= \Phi(I - \Phi^\top(I + M\Phi\Phi^\top)^{-1}M\Phi)\Phi^\top \quad (10)$$
$$= \Phi(I + \Phi^\top M\Phi)^{-1}\Phi^\top \quad (11)$$
$$= \breve{K} \quad (12)$$

where (11) comes by applying the Sherman-Morrison-Woodbury (SMW) identity

$$(I + AB^\top)^{-1} = I - A(I + B^\top A)^{-1}B^\top \quad (13)$$

with $A = \Phi^\top$ and $B = \Phi^\top M$ and using the symmetry of $M$. SMW requires that $I + \Phi^\top M\Phi$ is invertible. This is true since $\Phi^\top M\Phi$ is positive semidefinite. Adding the identity term guarantees positive definiteness, therefore also invertibility. ∎

This result allows us to use the efficient $\breve{K}$ in our experiments, but study it using $\overline{K}$, the Fourier approximation of the original data-dependent kernel $\widetilde{K}$. The form of $\overline{K}$ makes it easier to relate to $\widetilde{K}$. The efficient $\breve{K}$ suggests the following random feature approximation for $\widetilde{K}$

$$\widetilde{\Phi} = \Phi(I + \Phi^\top M\Phi)^{-1/2} \quad (14)$$

In order to obtain a linear approximation of a kernel method using (4), all we need is to calculate (14) and use the corresponding linear method. We do not need to compute the Gram matrix associated with the kernel. Also, this alone implies very significant computational gains in both learning and testing if $d \ll N$, since we need to compute the inverse of $I + \Phi^\top M\Phi$ (see (6)) which is $d \times d$, instead of the inverse of $I + M\widehat{K}$ (see (5)) which is $N \times N$.

We should note that SMW is often used as a method to perform low-rank updates of matrix inverses in iterative methods, sometimes leading to numerical inaccuracy. This is not the way we use SMW. Here, its use has the effect of transferring the learning problem from the warped RKHS to the warped random feature space, while preserving the effect of warping. A main insight of our paper is that this process is as simple as (14). Also, please note that the implication of SMW in our problem is of great impact in applications, especially in large-scale problems. The original method [8] is a reinterpretation of manifold regularization which works for the out-of-sample case: since the kernel approximation works everywhere [1], the out-of-sample properties should hold. Notice that (7) holds for any pair of points since the proof is independent of the outer $\Phi$.

Knowing that the mapping is simple and $\overline{K} = \breve{K}$, we can bridge the gap between $\overline{K}$ and $\widetilde{K}$. The following Lemma bounds the deviation between the two using the deviation between $\widehat{K}$ and $K$.

**Lemma 3** *Let $\overline{K}$ and $\widetilde{K}$ defined as above and denoting $\mathbb{E}\|\widehat{K}M(I + \widehat{K}M)^{-1}\| \leq R$ and $\mathbb{E}\|(I + MK)^{-1}MK\| \leq T$, with $R$, $T$ constants we have that*

$$\mathbb{E}\|\overline{K} - \widetilde{K}\| \leq \mathbb{E}\|K - \widehat{K}\|(1 + T + RT + R) \quad (15)$$

In the Appendix we provide a detailed proof.

In [25] it is shown that $\mathbb{E}\|\widehat{K} - K\| \leq \frac{C}{d}$, where $C$ is a constant. This has as consequence the convergence of $\widehat{K}$ to $K$ in the limit of infinite samples. Using our Lemma above, in conjunction with this result, we can obtain an asymptotic convergence guarantee for $\overline{K}$ to $\widetilde{K}$. Also, given the quantitative nature of both results and using a finite sample bound on $\mathbb{E}\|\widehat{K}M(I + \widehat{K}M)^{-1}\|$ we can immediately obtain the convergence rate.

This theoretical upper bound is supported by extensive empirical results, as we try to focus on the practical aspects of this data-dependent kernel approximation. More details can be seen in section §4.1, with direct reference to fig. 1 (*middle*), where we provide an empirical convergence guarantee for $\overline{K}$ to $\widetilde{K}$. Also, in §4.3 we want to emphasize the scalability of the method by applying it on Human3.6M [26] which contains millions of data points.

### 3.2 Local Weighting Function for Efficient Computation

As stated in section §3.1 we choose to use the Laplacian matrix for $M$, as it better captures the data geometry. This gives us a similarity measure between all data points defining the input space, both labeled and unlabeled. We consider $M = \alpha L^c$ where $L$ is the symmetric normalized Laplacian matrix *i.e.* $L = I - D^{-1/2}WD^{1/2}$ with $D$ and $W$ being the degree and the adjacency matrices, respectively. This fulfills the positive semidefiniteness requirement and has the additional computational benefit that $L$ can be viewed as a local weighting function. $\alpha$ and $c$ are additional parameters representing the Laplacian regularizer and the Laplacian degree, respectively.

Due to our approximation choice in eq. (6), we only need to compute a 'warping' matrix, which modifies the distance associated to the Fourier embedding, *i.e.* $U = \phi(X)^\top M \phi(X)$, and invert it. This is a square $d \times d$ matrix of the same dimensionality as the Fourier approximation i.e. $d$. The small size of the matrix makes the inversion easy to perform but computing the matrix remains difficult if $M$ is dense.

We can leverage the sparsity of $M$ computationally, in the following way. If $X_i$ is the set of nearest neighbors of datapoint $x_i$ and $M_i$ the Laplacian matrix around $x_i$, $U$ can

be written as

$$\Phi^\top M \Phi \simeq \Phi^\top \sum_i M_i \Phi \tag{16}$$

$$= \sum_i \phi(X_i)^\top \, M_i \, \phi(X_i) = U \tag{17}$$

The locality (block diagonal structure) of $M$ thus turns the problem of computing $U$ into a local procedure which can be scaled up reliably. Obtaining $U$, once $M$ is computed, requires $O(Nd^2)$ operations. When $M$ is modeled as a Laplacian matrix, standard practice [27] requires $O(Nk)$ space complexity for a sparse encoding. Computing it exactly can be expensive for large datasets as it scales at $O(N^2 \log N)$, due to $N \log N$ steps required for exact nearest neighbor calculations. However, more efficient approximate nearest neighbor (ANN) techniques like locality sensitive hashing (LSH) can be employed without adverse effects provided they have high recall. Precision is not critical: if ANN returns points that are far away (not neighbors) they will get weighted down by $M$ when computing $U$. With ANN, considering the access $O(1)$, the overall complexity of computing $M$ also becomes $O(Nr)$, where $r$ is the number of retrieved items. In practice $r$ should be more than $k$ but with a good hashing function $k \ll N$.

### 3.3 Data-Dependent Kernel Approximation Algorithm

As we mentioned previously, we aim to shift the focus towards the practical aspect of the kernel approximation methodology. Thus, we provide an algorithmic approach in order to obtain the kernel approximation. The steps of our proposed method are given in Algorithm 1. Given a data matrix $X$ of size $N \times p$, which corresponds to $N$ data points in $\mathbb{R}^p$, we construct a Fourier approximation for the data-dependent kernel $\tilde{\Phi}$. First, we build the random Fourier approximation $\Phi$ for $X$ using (1). Second, we compute the Laplacian matrix $L$ by building the adjacency matrix $W \in \mathbb{R}^{N \times N}$ and the degree matrix $D \in \mathbb{R}^{N \times N}$ (which is diagonal) associated to the data matrix $X$. Nodes $i$ and $j$, corresponding to data points $x_i$ and $x_j$, are connected by an edge if $i$ is among the $q$ nearest neighbors of $j$, or $j$ is among the $q$ nearest neighbors of $i$. Parameters $q$, $\alpha$, $c$ and $\sigma$, as well as the number of Random Features (RF) dimensions used for kernel approximation, are determined by validation. Finally, we construct the data-dependent kernel approximation $\tilde{\Phi}$ using (14). We obtain a mapping of our data into a data-dependent kernel feature space where feature learning is performed using matrix inversion. The obtained features can be incorporated within any desired learning model, *e.g.* SVM, KRR.

**Algorithm 1** Calculate $\tilde{\Phi}$ (Data-Dependent Kernel Approximation)

**Require:**
$X \in \mathbb{R}^{N \times p}$ - data matrix corresponding to $N$ data points in $\mathbb{R}^p$
$\alpha \in \mathbb{R}$ - Laplacian regularizer
$c \in \mathbb{N}$ - Laplacian degree
$q \in \mathbb{N}$ - number of neighbors required for Laplacian graph

**Ensure:** $\tilde{\Phi}$
$\widehat{K} = \phi(X)\phi(X)^\top = \Phi\Phi^\top$
$L = I - D^{-1/2} W D^{-1/2}$ with $W, D \in \mathbb{R}^{N \times N}$ where

$$W_{ij} = \begin{cases} e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}} & \text{if } x_i \in q\text{-NN}(x_j, X) \\ 0 & \text{otherwise} \end{cases}$$

$D_{ii} = \sum_j W_{ij}$
$M = \alpha L^c$
$U = \Phi^\top M \Phi$
$\tilde{\Phi} = \Phi(I + U)^{-1/2}$

## 4 Experiments

We performe a diverse set of experiments in order to validate our models. We consider two learning methods: support vector machines (SVM) and kernel ridge regression (KRR). For the classification task, the KRR model is used by predicting the score of each class and selecting the one with the highest score as output. For these models, we consider both the kernel (SVM and KRR) and the Random Features (RF) approximation variants (RFSVM and RFKRR) as purely supervised baselines. Their exact semi-supervised extensions, LapKRR and LapSVM, serve as baselines for our proposed models, LapRFSVM and LapRFKRR. The kernel function used is radial basis function (RBF). We use the RBF kernel as it is standard for Euclidean spaces. Other types of kernel functions can be used with the proposed method, according to the nature of the tackled problem. In all our semi-supervised models we use the symmetric normalized graph Laplacian with a Gaussian kernel. We first analyze a toy dataset in section §4.1 to check that basic intuitions are correct. In §4.2 we experiment with a number of medium size datasets to compare the performance of our models to the ones of [8] and [9]. We then move to larger datasets (see §4.3), where the kernel versions are no longer applicable, and evaluate the accuracy and the computational and performance aspects. In order to assess the performance of our method, we follow the experimental procedure of [8], so we make 10 different splits of the data into train/test and report the average performance. Finally, in §4.4 we study the dimensionality reduction component of our framework, and assess its impact on performance and computation time.

### 4.1 Two Moons Dataset

This dataset for semi-supervised problems consists of unlabeled datapoints (500 in our case) as well as 2 labeled ones. Figure 1 (left) shows this data: labeled points (larger filled dots) and unlabeled points (light blue are from the positive class and yellow from the negative class), and the

test data (darker blue for the positive class and orange for the negative class).

*Empirical Approximation of $\overline{K}$ by $\widetilde{K}$:* We first look into the accuracy of the approximation. In the first experiment we measure the empirical approximation of $\widehat{K}$ by $K$ (original kernel) as well as the approximation of $\overline{K}$ by $\widetilde{K}$ (data-dependent or warped kernel). We vary the dimensionality of the Fourier approximation and take 1000 samples for $\widehat{K}$ and $\overline{K}$. We measure the average and max values for $\|\widehat{K} - K\|$ and $\|\overline{K} - \widetilde{K}\|$ for each dimensionality (figure 1 middle). It can be seen that as we increase the dimensionality of the Fourier approximation, the bound becomes tighter. Also, note that the max value of the approximation error is not much larger than the average error. Thus, we are able to obtain an empirical guarantee for the convergence of $\overline{K}$ by $\widetilde{K}$. The second plot (fig. 1 right) accentuates the trade-off between the number of random features used for the kernel approximation and the accuracy of the model. Notice that not only the warping works very well on this dataset, achieving near perfect performance, but the approximate model converges well to it. It also provides a regularization that the standard RFKRR model does not achieve.

*Approximation of the Warping Matrix $U$:* Since this data is 2 dimensional, it is easy to visualize the landscape of the different classifiers (fig. 2). The second and third plots (b & c) show the landscapes of the LapKRR and its corresponding RF approximation. LapKRR (b) looks somewhat fainter because of the normalization of the scale of the color plots. Note that the approximate model is much more diffuse than LapKRR especially at the borders. The fourth plot (d) shows the low-dimensional approximation of the warping. Although the low rank approximation generates some artifacts in the shape of the border, it also renders it much sharper. The rightmost plot (e) shows the landscapes of the LapKRR model obtained using the kernel approximation of [9]. You can notice that the landscape of this method is more sharp around the two clusters defined by the dataset.

### 4.2 Small Scale Experiments

Next, we tested our models on a number of standard small datasets, where in some cases both exact and approximation methods can be obtained. TEXT is a simple text classification dataset, which contains 1,946 data samples. USPS (test) is a digit recognition dataset (10 classes) with 1,607 data samples. These two experiments are performed in a transductive setting similar to the one used in [24]. We sample 50 labeled examples from the training set (the rest being unlabeled) for each of the 10 runs. USPS (out-of-sample) is the separate test set for the USPS dataset with 400 data samples.

Please notice (table 1) that the performance of the kernel and the approximated methods, respectively, are on par on each of these experiments. The small datasets reveal how well the approximation works, which we found to be satisfactory when the dimensionality of the RF features is large enough. Also, please note the additional performance gained by the methods that use unlabeled data.

| Model | MNIST | | CIFAR10 | |
|---|---|---|---|---|
| Tasks | Feature/ Kernel | Training | Feature/ Kernel | Training |
| SVM | 7.03 | 381.04 | 30.2 | 1600.87 |
| KRR | 7.03 | 440.75 | 30.2 | 1843.07 |
| RFSVM | 16.68 | 331.95 | 21.31 | 2306.43 |
| RFKRR | 16.68 | 212.42 | 21.31 | 477.33 |
| LapRFSVM | 769.89 | 382.03 | 770.50 | 1220.03 |
| LapRFKRR | 769.89 | 208.19 | 770.50 | 394.30 |

Table 2: Computation time in seconds on MNIST and CIFAR10, for RF approximations with $d = 10,000$ on a system with Intel Xeon (3.2 GHz) and 32 GB of RAM memory. Training time includes all 10 classifiers. For kernel methods, computation includes the kernel matrix. For semi-supervised models the Laplacian is considered precomputed. Feature computation for LapRFSVM and LapRFKRR includes the inversion which dominates the computation time. Without it, the computation time is similar to RFSVM and RFKRR.

### 4.3 Large-Scale Experiments

For large-scale experiments we consider datasets from computer vision: MNIST (60,000 examples), CIFAR10 (50,000 examples) and Human3.6M (1,055,424 examples).

MNIST contains 60,000 training examples and 10,000 testing examples for digit recognition. CIFAR10 contains 50,000 training and 10,000 testing examples (32x32 pixel RGB images) grouped in 10 classes. For this data we extract GIST features [28], instead of using directly raw pixel values. For MNIST, we sample 10,000/50,000 labeled/unlabeled examples from the training set for each of the 10 runs. For CIFAR10, we use 20,000/30,000 labeled/unlabeled examples as the dataset is more challenging.

Unlike the small-scale experiment setup where we used the transductive setting, in this test setup we used the separate test set provided. For computational considerations, we are not able to compute the exact LapSVM or LapKRR on the larger datasets. The classical kernel approach is no longer applicable in these model setups due to the size of the kernel matrices.

We also compare our method with the one proposed in [9] whenever possible. They build the Laplacian regularizer component for the data-dependent kernel out of a small subsample of landmark points, formed from both labeled and unlabeled data. The comparison can only be performed for datasets of up to 64,000 examples, as [9] requires to build the kernel matrix, thus we are not able to compare it on the Human 3.6M test case. The results can be visualized in table 1. We note that our approximation to the original data-

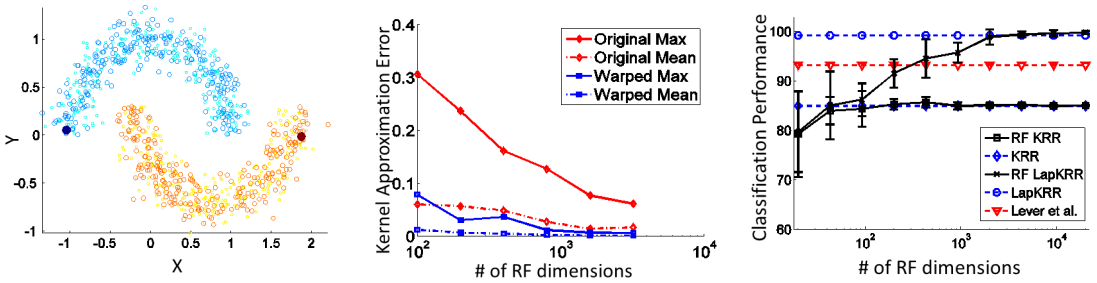
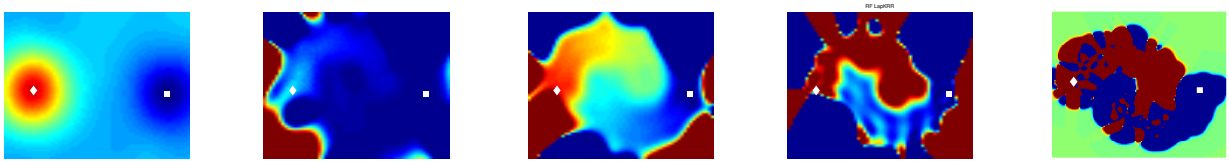



Figure 1: (Left) 'Two moons" dataset. (Middle) kernel approximation errors (max and average absolute error) for the original kernel, $K$, and warped (data-dependent) kernel, $\widetilde{K}$. (Right) classification performance on the two moons dataset, with 1 example per class, plotted against the dimensionality of the Fourier features. Note that the semi-supervised extension delivers some performance boost even with a very poor approximation (500 dimensions). With 2,000 RF features the performance is the same as the exact LapKRR model. We also include a comparison with the method of [9]. For their approximation method we use 500 landmark points.



(a) (b) (c) (d) (e)

Figure 2: Two moons dataset classifier landscape (same color scale). The labels are shown with square and rhombus. (a) Simple classification model, without warping, approximated with random Fourier features; (b) LapKRR model from [8]; (c) our RF approximation with full warping matrix; (d) approximate model with low rank warping obtained by spectral decomposition (rank 10); (e) LapKRR with kernel approximated by [9].

| Model | TEXT | USPS (t) | USPS (oos) | MNIST | CIFAR10 |
|---|---|---|---|---|---|
| SVM | 19.24 (6.08) | 23.78 (2.88) | 24.90 (2.90) | 3.99 (0.09) | 48.11 (0.41) |
| RFSVM | 24.32 (2.93) | 23.91 (2.87) | 24.75 (3.02) | 3.89 (0.08) | 48.17 (0.25) |
| LapSVM | 10.40 (1.06) | **13.26 (2.76)** | 15.27 (2.67) | N/A | N/A |
| [9] | 21.72 (7.00) | 18.23 (3.14) | 19.95 (4.10) | 3.03 (0.06) | 38.49 (0.18) |
| LapRFSVM | **9.96 (1.24)** | 13.42 (2.91) | 15.60 (2.69) | **2.93 (0.04)** | 37.34 (0.21) |
| KRR | 19.22 (6.06) | 24.29 (2.68) | 25.07 (2.54) | 6.43 (0.29) | 47.64 (0.13) |
| RFKRR | 24.28 (3.09) | 24.52 (2.86) | 25.82 (2.65) | 8.78 (0.27) | 47.84 (0.15) |
| LapKRR | 10.01 (1.26) | 13.49 (2.76) | **14.55 (2.82)** | N/A | N/A |
| [9] | 21.99 (4.58) | 17.70 (2.96) | 19.25 (3.09) | 3.54 (0.07) | 35.49 (0.17) |
| LapRFKRR | 10.32 (1.08) | 13.46 (2.79) | 15.52 (2.93) | 3.13 (0.05) | **35.25 (0.28)** |

Table 1: Average classification error on several datasets comparing the approximated (LapRFKRR, LapRFSVM) and non-approximated versions of the model (LapKRR, LapSVM), as well as non semi-supervised baselines (KRR, SVM) and their approximated counterparts (RFKRR, RFSVM) trained with the same labeled data. Also shown is the performance of [9]. We use the kernel matrices approximated by [9] and introduce them in our framework, reproducing the same experiments. The comparison can only be performed for datasets of up to 64,000 examples, as [9] requires to build the kernel matrix. The RF features for TEXT and USPS have 4,000 dimensions and 10,000 dimensions for MNIST and CIFAR10. The training data is randomly sampled and results are averages over 10 runs (variance in parentheses).

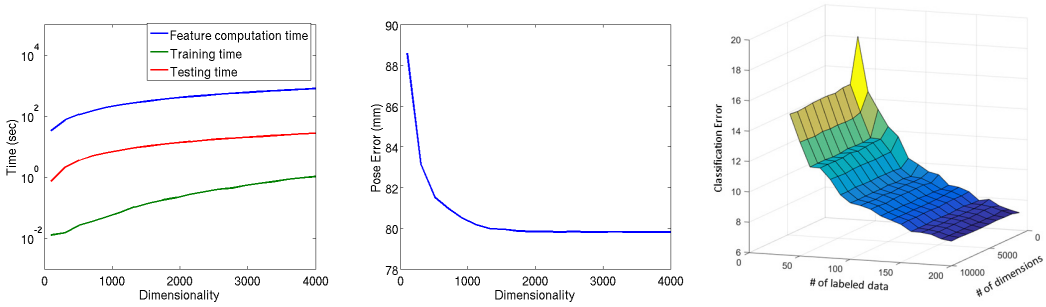



Figure 3: Dimensionality reduction (left and middle) performed on Human3.6M makes us gain an order of magnitude in computation time (up) while maintaining good performance (down). Thus we can efficiently handle very large datasets. In the right most plot we measure the trade-off between the number of labeled data and the number of used RF dimensions for kernel approximation on USPS (t). As the number of RF dimensions becomes larger than 2,000, the performance of the model becomes stable and it improves only if the size of the label set is increased. The size of the labeled set has a larger impact on performance than the number of RF dimensions.

| Pose error (mm) | | Labeled vs. unlabeled split | |
|---|---|---|---|
| RFKRR | LapRFKRR | # of Labeled | # of Unlabeled |
| 57.83 | 57.72 | 105,543 | 949,881 |
| 61.6 | 60.83 | 10,555 | 1,044,869 |
| 77.99 | 71.95 | 1,056 | 1,054,368 |
| 87.41 | 79.81 | 528 | 1,054,896 |
| 89.48 | 84.85 | 352 | 1,055,072 |
| 94.88 | 91.68 | 264 | 1,055,160 |
| 97.37 | 93.2 | 212 | 1,055,212 |

Table 3: Performance evaluation for Human3.6M, with different splits of labeled and unlabeled data. The RFKRR column gives the performance of models trained using only labeled data while LapRFKRR uses both labeled and unlabeled data within the data-dependent kernel approximation setup. As the size of labeled data decreases, the performance of RFKRR decreases as well. However, we obtain improvements in the semi-supervised learning setting, thus demonstrating both the scalability and the advantage of using data-dependent kernel approximations.

dependent kernel achieves very high performance and seems to outperform the approximation of [9] in most situations. Our intuition is that [9] performs better when the unlabeled data characterizes well the input space.

We study the computational times of our different models on these datasets (table 2). Note that the supervised kernel models, SVM and KRR, can be applied since the kernel matrices for training are 10,000 and 20,000 for these datasets. But LapSVM and LapKRR require the full 60,000 dataset which is infeasible. Our proposed method however has no problem scaling to 60,000 and beyond.

We also consider a 3D human pose estimation problem based on 2D image information. We run our experiments on the very large Human 3.6M dataset [26], where we sample a subset of 1,055,424 poses for training and 56,860 poses for testing. We examine the following learning task: given the 2D pose, learn a model which is able to estimate the corresponding 3D pose. Thus, our input data consists of 2D human body joint positions and the target data of the corresponding 3D joint positions. We normalize the 2D pose data by setting the origin of the coordinate system in the pelvis joint. Also, we rotate each 2D pose such that the neck pelvis axis would align with the $OY$ axis and scale it such that the average limb size would be 1. Our learning model is kernel ridge regression as it is simple and demonstrates the use of kernel methods. We choose the radial basis function kernel approximation for our problem due to the nature of the data. The random features approximation is based on $d = 4,000$ dimensions. By default, the entire dataset is fully labeled with both 2D and 3D information. For the semi-supervised problem, we consider 3D pose to be missing for some of the data, according to different splits. The performance of the model is illustrated in table 3. Please note, that during this experiment we varied the ratio between labeled and unlabeled data, keeping the total number of data points used. The reason behind this is that we want to see the impact of the labeled data, given that the quantity of unlabeled data is dense ($\simeq 1,000,000$). The purpose of this experiment is to empirically illustrate that the proposed data-dependent kernel approximation improves 3D pose estimation in a non-trivial, semi supervised learning scenario, where we work with large-scale datasets of over 1 million elements.

### 4.4 Dimensionality Reduction

In this section we study the effect of the dimensionality reduction on the warping $U$. In a classification problem we have seen this to beneficially make the separation sharper. We performed a similar analysis on the sampled data from Human3.6M [26], as shown in figure 3. We use $1,055,424$ data points with a split of $528$ labeled data and $1,054,896$ unlabeled data. The Fourier approximation has $d = 4,000$ dimensions. Following the eigen-decomposition of the warping matrix we choose $20$ subsets of dimensions (with linear spacing between $100$ and $4,000$ for the dimension of the subset), with the highest corresponding eigenvalues. The subset of $100$ dimensions contains the most significant eigenvectors and we increase the size of the subset by adding the remaining dimensions based on the value of their corresponding eigenvalues (the higher ones). We observe that with a subset of nearly $1,000$ most significant dimensions we obtain a performance similar with the one of the model containing all dimensions.

## 5 Conclusions

In this paper we derive an approximate learning procedure for data-dependent kernels, that performs well in practice. Our methodology relies on low-dimensional kernel approximations, thus overcoming the computational challenges of applying semi-supervised frameworks like manifold regularization to large datasets. We prove that our construction represents a valid approximation of the data-dependent kernel and provide a Lemma for asymptotic convergence. Our experiments show that the method performs on par with exact kernel based equivalents in small datasets. For large datasets, we show that our methodology can now take full advantage of unlabeled data, being superior to kernel approximations that use only labeled data, at comparable computational cost. The method is demonstrated to effectively handle datasets of millions of items in practice.

**Acknowledgments:** This work was supported in part by the European Research Council Consolidator grant SEED, CNCS-UEFISCDI under JRP-RO-FR-2014-16 and the EU Horizon 2020 Grant #688835 (DE-ENIGMA). We would also like to thank Andrei Zanfir for his assistance with the proof of Lemma 3.

## References


[1] A. Rahimi and B. Recht, "Random features for large-scale kernel machines," in *NIPS*, 2007.

[2] F. Li, C. Ionescu, and C. Sminchisescu, "Random Fourier approximations for skewed multiplicative histogram kernels," in *LNCS (DAGM)*, September 2010.

[3] A. Vedaldi and A. Zisserman, "Efficient additive kernels via explicit feature maps," in *PAMI*, 2012.

[4] D. J. Sutherland and J. Schneider, "On the error of random fourier features," *UAI*, 2015.

[5] B. Sriperumbudur and Z. Szabo, "Optimal rates for random fourier features," *NIPS*, 2015.

[6] E. G. Băzăvan, F. Li, and C. Sminchisescu, "Fourier kernel learning," in *ECCV*, 2012.

[7] F. Li, G. Lebanon, and C. Sminchisescu, "Chebyshev approximations to the histogram $\chi^2$ kernel," in *CVPR*, 2012.

[8] V. Sindhwani, P. Niyogi, and M. Belkin, "Beyond the point cloud: from transductive to semi-supervised learning," in *ICML*, 2005.

[9] G. Lever, T. Diethe, and J. Shawe-Taylor, "Data dependent kernels in nearly-linear time," *AISTATS*, 2012.

[10] Q. Que and M. Belkin, "Back to the future: Radial basis function networks revisited," in *AISTATS*, 2016.

[11] J. Oliva, A. Dubey, B. Poczos, J. Schneider, and E. P. Xing, "Bayesian nonparametric kernel-learning," in *AISTATS*, 2016.

[12] T. Hofmann, B. Schölkopf, and A. J. Smola, "Kernel methods in machine learning," *The Annals of Statistics*, vol. 36, no. 3, pp. pp. 1171–1220, 2008.

[13] J. Shawe-Taylor and N. Cristianini, *Kernel methods for pattern analysis*. Cambridge university press, 2004.

[14] C. Williams and M. Seeger, "Using the nyström method to speed up kernel machines," in *NIPS*, 2000.

[15] W. Rudin, *Fourier analysis on groups*. Interscience tracts in pure and applied mathematics, Wiley, 1990.

[16] P. Kar and H. Karnick, "Random feature maps for dot product kernels.," *AISTATS*, 2012.

[17] A. Cotter, J. Keshet, and N. Srebro, "Explicit approximations of the gaussian kernel," *CoRR*, vol. abs/1109.4603, 2011.

[18] V. Sreekanth, A. Vedaldi, C. V. Jawahar, and A. Zisserman, "Generalized RBF feature maps for efficient detection," in *BMVC*, 2010.

[19] J. Yang, V. Sindhwani, Q. Fan, H. Avron, and M. Mahoney, "Random laplace feature maps for semigroup kernels on histograms," in *CVPR*, 2014.

[20] X. Zhu, G. Zoubin, and J. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *ICML*, 2003.

[21] S. Wu and S. Amari, "Conformal transformation of kernel functions: A data-dependent way to improve support vector machine classifiers," *Neural Process. Lett.*, vol. 15, Feb. 2002.

[22] O. Chapelle, B. Schölkopf, A. Zien, *et al.*, *Semi-supervised learning*. MIT pres, 2006.

[23] A. Singh, R. Nowak, and X. Zhu, "Unlabeled data: Now it helps, now it doesn"t," in *Advances in Neural Information Processing Systems 21* (D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, eds.), pp. 1513–1520, Curran Associates, Inc., 2009.

[24] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples.," *JMLR*, 2006.

[25] D. Lopez-Paz, S. Sra, A. Smola, Z. Ghahramani, and B. Schölkopf, "Randomized nonlinear component analysis," *arXiv preprint arXiv:1402.0119*, 2014.

[26] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments," *PAMI*, 2014.

[27] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, pp. 395–416, Dec. 2007.

[28] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *IJCV*, 2001.

# Large-Scale Data-Dependent Kernel Approximation
*Appendix*

This appendix presents the additional detail and proofs associated with the main paper [1].

## 1 Introduction

Let $k : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}_+$ be a positive definite translation invariant function e.g. a Gaussian kernel $k(x, y) = \exp(-\gamma\|x-y\|^2)$. By Bochner's theorem there exists $\mu$ a positive function such that

$$k(x, y) = \int_\omega e^{i\omega^\top (x-y)}\mu(\omega)$$

Since $\mu$ is positive we can use it to draw i.i.d. samples $\omega_i \sim \mu$ which allows us to define a random feature map such that $\phi(x) = [\phi_1(x) \dots \phi_d(x)]$, where $\phi_i(x) = \cos(\omega_i^\top x + b_i)$ (where $b_i \sim \texttt{Uniform}[0, 2\pi]$). Let $\widehat{k}(x, y) = \sum_i^d \widehat{k}_i(x, y) = \frac{1}{d}\sum_i^d \phi_i(x)\phi_i(y)^\top = \frac{1}{d}\phi(x)\phi(y)^\top$. This is a standard construction; see [2, 3] for more details.

Let $X$ be a fixed data matrix $N \times p$ corresponding to $N$ data points in $\mathbb{R}^p$ and let the matrix counterparts of the above notation applied to $X$ be $K(i, j) = k(X(i, :), X(j, :))$, as well as $\widehat{K}$, $\widehat{K}_i$, $\Phi_i(= \phi_i(X))$ and $\Phi(= \phi(X))$.

With this notation we have

$$\widehat{K} = \sum_i^d \widehat{K}_i = \sum_i^d \Phi_i\Phi_i^\top = \Phi\Phi^\top \tag{1}$$

We notice that $\widehat{K}_i$ are i.i.d. thus matrix concentration results apply to it.

To this end we want to use

**Theorem 1 (Matrix Bernstein [4])** *Let $Z_1 \dots Z_m$ be independent $n \times n$ Hermitian random matrices with $\mathbb{E}[Z_i] = 0$ and $\|Z_i\| \le R$. Let $\sigma^2 = \max\{\|\sum_i \mathbb{E}[Z_i^\top Z_i]\|, \|\sum_i \mathbb{E}[Z_i Z_i^\top]\|\}$, where $\|.\|$ is the operator norm. Then*

$$\mathbb{E}\|\sum_i Z_i\| \le \sigma\sqrt{3\log(2n)} + R\log(2n) \tag{2}$$

**Theorem 2 ($\widehat{K}$ convergence [3])** *Let $\widehat{K}$ be an $d$ term random feature approximation of the kernel matrix $K \in \mathbb{R}^{N\times N}$*

$$\mathbb{E}\|\widehat{K} - K\| \le \sqrt{\frac{3N^2\log N}{d}} + \frac{2N\log N}{d} \tag{3}$$

**Proof** [1] Then $\widehat{K}_i$ are independent and we know that $\mathbb{E}[\widehat{K}] = K$.

$$E = \widehat{K} - K = \sum_i^d E_i, \;\; E_i = \frac{1}{d}(\widehat{K}_i - K) \tag{4}$$

Thus $\mathbb{E}[E_i] = 0$ and $E_i$ are i.i.d. as well.

First we must show that each are bounded

$$\|E_i\| = \frac{1}{d}\|\Phi_i\Phi_i^\top - \mathbb{E}[\Phi\Phi^\top]\| \le \frac{1}{d}(\|\Phi_i\|^2 + \mathbb{E}[\|\Phi\|^2] \le \frac{1}{d}(\|\Phi_i\|^2 + \|\mathbb{E}[\Phi]\|^2) \le \frac{2B}{d} \tag{5}$$

[1] This is from [3] reproduced for a self-contained understanding of our main results.

where we used first the definitions of $\widehat{K}_i$ and $K$, followed by the triangle inequality, then Jensen for the expected value. $B$ is a finite bound for $\|\phi\|$ ($\|\phi\|^2 \leq B$). We know that such a bound exists, by the way $\phi$ is constructed.

Then the variance of $E_i$ is

$$\mathbb{E}[E_i^2] = \frac{1}{d^2}\mathbb{E}[(\Phi_i\Phi^\top - K)^2] \tag{6}$$

$$= \frac{1}{d^2}\mathbb{E}[(\|\Phi_i\|^2\Phi_i\Phi_i^\top - \Phi_i\Phi_i^\top K - K\Phi_i\Phi_i^\top + K^2)] \tag{7}$$

$$\preccurlyeq \frac{1}{d^2}[BK - 2K^2 + K^2] \preccurlyeq \frac{BK}{d^2} \tag{8}$$

where we unravel the square, then use $\mathbb{E}[\widehat{K}_i] = \mathbb{E}[\Phi_i\Phi_i^\top] = K$. The second $\preccurlyeq$ is due to $K$ being positive definite.

$$\|\mathbb{E}[E^2]\| \leq \|\sum_i^d \mathbb{E}[E_i^2]\| \leq \frac{1}{d}B\|K\| \tag{9}$$

where we first used Jensen's inequality, then the semi-definite bound above with $d$ terms.

Given these bounds on the variance and the norm of the random variables, we can apply (2) to get

$$\mathbb{E}\|\widehat{K} - K\| \leq \sqrt{\frac{3B\|K\|\log N}{d}} + \frac{2B\log N}{d} \tag{10}$$

## 2 Data-Dependent Kernel

Let $L$ be the normalized Laplacian i.e. $L = I - D^{-1/2}WD^{-1/2}$ with $W$ again some fixed positive definite function of the data and $D$ a diagonal matrix with the sum of each row of $W$. Let $M = L$ or some positive power of the Laplacian $M = \alpha L^c$. Then we define

$$\widetilde{K} = K - K(I + MK)^{-1}MK \tag{11}$$

as a new kernel, similarly to the one defined in [5].

So the goal is to obtain $\widetilde{\Phi}$ with both some guarantees of consistency and a large deviation bound, in order to characterize the speed of convergence.

To this end we define

$$\overline{K} = \widehat{K} - \widehat{K}(I + M\widehat{K})^{-1}M\widehat{K} \tag{12}$$

and

$$\breve{K} = \Phi(I + \Phi^\top M\Phi)^{-1}\Phi^\top \tag{13}$$

The Sherman-Morrison-Woodbury (SMW) identity in its simplest form states that if both $I + UV^\top$ and $I + V^\top U$ are invertible then

$$(I + UV^\top)^{-1} = I - U(I + V^\top U)^{-1}V^\top \tag{14}$$

**Proposition 2** *With the definitions above*

$$\overline{K} = \breve{K} \tag{15}$$

**Proof**

$$\overline{K} = \widehat{K} - \widehat{K}(I + M\widehat{K})^{-1}M\widehat{K} \tag{16}$$

$$= \Phi\Phi^\top - \Phi\Phi^\top(I + M\Phi\Phi^\top)^{-1}M\Phi\Phi^\top \qquad \text{by (1)} \tag{17}$$

$$= \Phi(I - \Phi^\top(I + M\Phi\Phi^\top)^{-1}M\Phi)\Phi^\top \tag{18}$$

$$= \Phi(I + \Phi^\top M\Phi)^{-1}\Phi^\top \tag{19}$$

$$= \breve{K} \qquad \text{by (13)} \tag{20}$$

Where (19) comes by applying (14) with $U = \Phi^\top$ and $V = \Phi^\top M$ and using the symmetry of $M$.

So $\widetilde{\Phi} = \Phi(I + \Phi^\top M\Phi)^{-1/2}$ but given (15) we can use $\overline{K}$ instead of $\breve{K}$ for the convergence proofs. Now the goal is to obtain a bound on $\mathbb{E}\|\overline{K} - \widetilde{K}\|$.

**Lemma 3** *Let $\overline{K}$ and $\widetilde{K}$ defined as above and denoting $\mathbb{E}\|\widehat{K}M(I + \widehat{K}M)^{-1}\| \leq R$ and $\mathbb{E}\|(I + MK)^{-1}MK\| \leq T$, with $R$, $T$ constants we have that*

$$\mathbb{E}\|\overline{K} - \widetilde{K}\| \leq \mathbb{E}\|K - \widehat{K}\|(1 + T + RT + R) \tag{21}$$

**Proof**

$$\|\overline{K} - \widetilde{K}\| = \|\widehat{K} - \widehat{K}(I + M\widehat{K})^{-1}M\widehat{K} - K + K(I + MK)^{-1}MK\| \tag{22}$$

$$\leq \|\widehat{K} - K\| + \|\widehat{K}(I + M\widehat{K})^{-1}M\widehat{K} - K(I + MK)^{-1}MK\| \tag{23}$$

If we apply the triangle inequality for the second term in the right side of inequality (23) in the form of $\|A + B + C\| \leq \|A\| + \|B\| + \|C\|$ with,

$$A = \widehat{K}(I + MK)^{-1}MK - K(I + MK)^{-1}MK \tag{24}$$

$$B = \widehat{K}(I + M\widehat{K})^{-1}MK - \widehat{K}(I + MK)^{-1}MK \tag{25}$$

$$C = \widehat{K}(I + M\widehat{K})^{-1}M\widehat{K} - \widehat{K}(I + M\widehat{K})^{-1}MK \tag{26}$$

we obtain the following,

$$\|\widehat{K}(I + M\widehat{K})^{-1}M\widehat{K} - K(I + MK)^{-1}MK\| \leq \|\widehat{K}(I + MK)^{-1}MK - K(I + MK)^{-1}MK\| \tag{27}$$

$$+ \|\widehat{K}(I + M\widehat{K})^{-1}MK - \widehat{K}(I + MK)^{-1}MK\| \tag{28}$$

$$+ \|\widehat{K}(I + M\widehat{K})^{-1}M\widehat{K} - \widehat{K}(I + M\widehat{K})^{-1}MK\| \tag{29}$$

For $\|A\|$ we obtain the following bound,

$$\|\widehat{K}(I + MK)^{-1}MK - K(I + MK)^{-1}MK\| \leq \|\widehat{K} - K\|\|(I + MK)^{-1}MK\| \tag{30}$$

For $\|B\|$ we obtain the following bound,

$$\|\widehat{K}(I + M\widehat{K})^{-1}MK - \widehat{K}(I + MK)^{-1}MK\| = \|\widehat{K}(I + M\widehat{K})^{-1}M(\widehat{K} - K)(I + MK)^{-1}MK\| \tag{31}$$

$$\leq \|\widehat{K}(I + M\widehat{K})^{-1}M\|\|\widehat{K} - K\|\|(I + MK)^{-1}MK\| \tag{32}$$

$$= \|\widehat{K}M - \widehat{K}M(I + \widehat{K}M)^{-1}\widehat{K}M\|\|\widehat{K} - K\|\|(I + MK)^{-1}MK\| \tag{33}$$

$$= \|\widehat{K}M(I + \widehat{K}M)^{-1}\|\|\widehat{K} - K\|\|(I + MK)^{-1}MK\| \tag{34}$$

In order to obtain eq. (31) we apply the identity $XZ^{-1}Y - XW^{-1}Y = XZ^{-1}(W - Z)W^{-1}Y$ with $W = I + MK$, $X = \widehat{K}$, $Y = MK$ and $Z = I + M\widehat{K}$. To reach (33) we apply the SMW identity; for eq. (34) we apply the identity $Q - Q(I + Q)^{-1}Q = Q(I + Q)^{-1}$ with $Q = \widehat{K}M$.

For $\|C\|$ we have the following bound,

$$\|\widehat{K}(I + M\widehat{K})^{-1}M\widehat{K} - \widehat{K}(I + M\widehat{K})^{-1}MK\| \leq \|\widehat{K}(I + M\widehat{K})^{-1}M\|\|K - \widehat{K}\| \tag{35}$$

$$= \|\widehat{K}M - \widehat{K}M(I + \widehat{K}M)^{-1}\widehat{K}M\|\|K - \widehat{K}\| \tag{36}$$

$$= \|\widehat{K}M(I + \widehat{K}M)^{-1}\|\|K - \widehat{K}\| \tag{37}$$

For eqs. (36) and (37) we follow the same proof as for eqs. (33) and (34).

We will focus on the first term of the right side of (37).

$$\|\widehat{K}M(I+\widehat{K}M)^{-1}\| \leq \|\widehat{K}\|\|M\|\|(I+\widehat{K}M)^{-1}\| \tag{38}$$

We seek to provide a bound for $\|(I+\widehat{K}M)^{-1}\|$. We know that $\sigma_{max}((I+\widehat{K}M)^{-1}) = \frac{1}{\sigma_{min}(I+\widehat{K}M)}$, with $\sigma_{max}(.)$ and $\sigma_{min}(.)$ being the maximum and minimum singular values, respectively. From [6] (with direct reference to their eq. 3.12) we can write the following inequality (which is valid for any non-singular complex matrix of order $N$, in our case $I+\widehat{K}M$), with $\|.\|_F$ being the Frobenius norm

$$\sigma_{min}(I+\widehat{K}M) \geq \big|\det(I+\widehat{K}M)\big| \left(\frac{\sqrt{N-1}}{\|I+\widehat{K}M\|_F}\right)^{N-1} \tag{39}$$

For $|\det(I+\widehat{K}M)|$ we have the following bound, where $\lambda_i(.)$ is the $i^{th}$ eigenvalue

$$\begin{aligned}
\big|\det(I+\widehat{K}M)\big| &= \big|\prod_i \lambda_i(I+\widehat{K}M)\big| && (40)\\
&= \big|\prod_i (1+\lambda_i(\widehat{K}M))\big| && (41)\\
&\geq 1 && (42)
\end{aligned}$$

The last inequality results due to the fact that $\widehat{K}M$ is positive semi-definite. Thus, (39) becomes

$$\begin{aligned}
\sigma_{min}(I+\widehat{K}M) &\geq \left(\frac{\sqrt{N-1}}{\|I+\widehat{K}M\|_F}\right)^{N-1} && (43)\\
\sigma_{max}((I+\widehat{K}M)^{-1}) &\leq \left(\frac{\|I+\widehat{K}M\|_F}{\sqrt{N-1}}\right)^{N-1} && (44)
\end{aligned}$$

We know that the right hand side of (44) is bounded, as $N$ is the number of data samples, and $\widehat{K}M$ is positive semi-definite.

Given the bounds of $\|A\|$, $\|B\|$ and $\|C\|$, we substitute them in (23). Applying the expectations on both sides, leads to the claim.

**Proposition 3** *Given the results before we can claim* $\mathbb{E}\|\overline{K}-\widetilde{K}\| \leq \left(\sqrt{\frac{3N^2\log N}{d}}+\frac{2N\log N}{d}\right)(1+T+RT+R)$

**Proof** Given the bound for $\mathbb{E}\|\widehat{K}-K\|$, the claim for deviation is

$$\begin{aligned}
\mathbb{E}\|\overline{K}-\widetilde{K}\| &\leq \mathbb{E}\|\widehat{K}-K\|(1+T+RT+R) && (45)\\
&\leq \left(\sqrt{\frac{3N^2\log N}{d}}+\frac{2N\log N}{d}\right)(1+T+RT+R) \qquad \text{by (3)} && (46)
\end{aligned}$$

Finally note that a convergence rate immediately follows once $T$ and $R$ are determined. However, these will depend on the explicit forms of $K$ and $M$, which is beyond the scope of this analysis.

## References


[1] C. Ionescu, A.-I. Popa, and C. Sminchisescu, "Large-scale data-dependent kernel approximation," in *AISTATS*, 2017.

[2] A. Rahimi and B. Recht, "Random features for large-scale kernel machines," in *NIPS*, 2007.

[3] D. Lopez-Paz, S. Sra, A. Smola, Z. Ghahramani, and B. Schölkopf, "Randomized nonlinear component analysis," *arXiv preprint arXiv:1402.0119*, 2014.

[4] L. Mackey, M. I. Jordan, R. Y. Chen, B. Farrell, J. A. Tropp, *et al.*, "Matrix concentration inequalities via the method of exchangeable pairs," *The Annals of Probability*, vol. 42, no. 3, pp. 906–945, 2014.

[5] V. Sindhwani, P. Niyogi, and M. Belkin, "Beyond the point cloud: from transductive to semi-supervised learning," in *ICML*, 2005.

[6] H.-B. Li, T.-Z. Huang, and H. Li, "Some new results on determinantal inequalities and applications," *Journal of Inequalities and Applications*, vol. 2010, no. 1, p. 1, 2010.