Sampling Circulant Matrix Approach: A Comparison of Recent Kernel Matrix Approximation Techniques in Ridge Kernel Regression

NP Slagle
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332
npslagle@gmail.com

Abstract

As part of a survey of state-of-the-art kernel approximation algorithms, we present a new sampling algorithm for circulant matrix construction to perform fast kernel matrix inversion in kernel ridge regression, comparing theoretical and experimental performance of that of multilevel circulant kernel approximation, incomplete Cholesky decomposition, and random features, all recent advances in the literature. In particular, the new circulant approach rivals the remaining three algorithms in accuracy, executing in time complexity of mixed competitiveness, warranting further study.

1 Survey of the Problem: Ridge Regression

Ridge regression, also known as Tychonoff regularization, appears as early as [1], though [2] standardized the approach in statistics. Formally, given a $D \times N$ data matrix X and corresponding vector of labels $\mathbf{y}$, ridge regression estimates $f(\mathbf{x}) = y$ with $\hat{f}(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$, where $\mathbf{w}$ is a weight vector minimizing

$$||\mathbf{w}^T X - \mathbf{y}||^2 + ||\mathbf{w}^T\Gamma_D||^2, \tag{1}$$

where Γ is a regularization matrix. The exact solution is

$$\hat{\mathbf{w}} = (XX^T + \Gamma_D\Gamma_D^T)^{-1}X\mathbf{y}. \tag{2}$$

Typically, we choose Γ to be a diagonal matrix with positive entries. Herein, we replace $\Gamma_D\Gamma_D^T$ with λI_D, where I_D is the identity matrix.

2 Kernelization

Kernelization boasts a long history in pattern recognition and density estimation [3],[4]. Recent efforts in kernelization include kernelized SVMs [4], kernelized HMMs [5], kernel regression [4], to name a few. Kernelization affords a cheap approach to transforming the space of data points to highly nonlinear features.

Kernel ridge regression substitutes a kernel matrix $K = \phi(X)^T\phi(X)$ for the inner product form X^TX after applying the matrix inversion lemma to equation 2, replacing equation 2 with

$$\hat{\mathbf{w}} = (\phi(X)\phi(X)^T + \lambda I_D)\phi(X)\mathbf{y} = \phi(X)(K + \lambda I_N)^{-1}\mathbf{y}. \tag{3}$$

Since $\phi(X)$ can be infinite-dimensional, we specify the prediction of a test point $\mathbf{x}$ as

$$\hat{f}(\mathbf{x}) = \phi(\mathbf{x})^T\phi(X)(K + \lambda I_N)^{-1}\mathbf{y} = K_{\mathbf{x}}(K + \lambda I_N)^{-1}\mathbf{y}, \tag{4}$$

where $K_{\mathbf{x}}$ is the matrix of kernel entries of $\mathbf{x}$ and the training data.

3 The Problem: Complexity Challenges in Kernel Matrix Inversion

Clearly, the computational bottlenecks in kernel ridge regression are specification of the kernel matrix ($O(N^2)$) and the associated inversion ($O(N^3)$). Even for reasonably-sized data sets, specifying the kernel matrix, to say nothing of performing the inversion, can be beyond the reach of modern hardware platforms. Given T test points, the matrix multiplications cost $O(TN^2)$ for a total of $O(TN^2 + N^3)$.

3.1 State-of-the-Art

To mitigate the explosive growth of kernel matrices in problems with large associated data sets, we present a brief survey of the recent approximation techniques incomplete Cholesky decomposition [6], random features [7], and multilevel circulant kernel matrix approximation [8], [9].

3.1.1 Incomplete Cholesky Decomposition

Given an $N \times N$ kernel matrix K, incomplete Cholesky decomposition seeks an $R_{N\times M}$ matrix with $M << N$ such that

$$K \approx RR^T. \tag{5}$$

Obtaining R requires partial orthonormalization of the kernel matrix at complexity $O(NM^2)$. Substituting RR^T for K in equation 4, we have 6

$$\hat{f}(\mathbf{x}) = K_{\mathbf{x}}(RR^T + \lambda I_N)^{-1}\mathbf{y}. \tag{6}$$

Leveraging the matrix inversion lemma, we obtain

$$\hat{f}(\mathbf{x}) = r_{\mathbf{x}}(R^T R + \lambda I_M)^{-1}R\mathbf{y}, \tag{7}$$

where $r_{\mathbf{x}}$ is a variant of $K_{\mathbf{x}}$ reduced to features in R. Notice, the matrix inversion in equation 7 ($O(M^3)$) is less costly than that of equation 4 ($O(N^3)$). We typically call M the number of pivots in obtaining the R matrix. Again, given T test points, the total calculation requires $O((T+N)M^2 + M^3)$. Typically, we specify $\eta > 0$, a thresholding parameter guiding the level of approximation. A small η increases the number of pivots and thus improves the approximation.

3.1.2 Random Features

Random features selects random basis vectors of the feature space by characterizing a radial basis kernel as

$$k(\mathbf{x}-\mathbf{y}) = \int_{\mathbb{R}^D} p(\omega)e^{j\omega'(\mathbf{x}-\mathbf{y})}d\omega = E_\omega\left[\zeta(\mathbf{x})\zeta(\mathbf{y})\right], \tag{8}$$

using Bochner's theorem [7]. Since $p(\omega)$ and $k(||t||)$ are real-valued functions, we can substitute $\cos(\omega'(\mathbf{x}-\mathbf{y}))$ for the exponential factor, obtaining

$$k(\mathbf{x}-\mathbf{y}) = E_\omega\left[\cos(\omega'(\mathbf{x}-\mathbf{y}))\right] = E_\omega\left[(\cos(\omega'\mathbf{x}), \sin(\omega'\mathbf{x}))(\cos(\omega'\mathbf{y}), \sin(\omega'\mathbf{y}))^T\right], \tag{9}$$

so $\zeta(\mathbf{x}) = (\cos(\omega'\mathbf{x}), \sin(\omega'\mathbf{x}))$. Thus, we sample ω from $p(\omega)$, then calculate the random features. The functions $p(\omega)$ and $k(\mathbf{t})$ form a conjugate pair; for instance, if p is Gaussian, then k is the Gaussian RBF. The Laplace and Cauchy distributions also form a conjugate pair. If R is a $N \times D$ matrix such that $K \approx RR^T$, then once again we can apply the matrix inversion lemma to obtain

$$\hat{f}(\mathbf{x}) = r_{\mathbf{x}}(R^T R + \lambda I_D)^{-1}R\mathbf{y}, \tag{10}$$

where $r_{\mathbf{x}}$ represents test points transformed to the random features space. Again, inversion of the $D \times D$ matrix $R^T R$ requires $O(D^3)$. Thus, the total complexity is $O((T+N)D^2 + D^3)$.

3.2 Multilevel Circulant Matrices

An $N \times N$ matrix C is *circulant* if there exist $c_0, \ldots, c_{N-1} \in \mathbb{C}$ such that $C_{i,j} = c_{j-i \bmod N}$. That is,

$$C = \begin{bmatrix} c_0 & c_{N-1} & \cdots & c_{N-1} \\ c_1 & c_0 & \cdots & c_{N-2} \\ \vdots & \vdots & \ddots & \vdots \\ c_{N-1} & c_{N-2} & \cdots & c_0 \end{bmatrix} \tag{11}$$

3.2.1 Circulant Matrix Inversion and Multiplication

Setting aside briefly an explanation of circumstances under which we can substitute a circulant matrix for a kernel matrix, let's consider properties of circulant matrices useful in analyzing complexity if we perform such a substitution in kernel ridge regression. In particular, if C is an $N \times N$ circulant matrix, then the eigenvalues $\lambda_0, \ldots, \lambda_N$ of C are precisely the discrete Fourier transform of the defining column vector $[c_0, \ldots, c_N]^T$ [10], computable in $O(N \log N)$ time complexity using the fast Fourier transform. Furthermore, if Φ is the matrix obtained by performing a discrete Fourier transform on each of the columns of the $N \times N$ identity matrix, then

$$C = \frac{1}{N}\Phi^* diag(\lambda_0, \ldots, \lambda_{N-1})\Phi. \tag{12}$$

Since Φ and $\frac{1}{N}\Phi^*$ are inverses of one another, we can characterize the inverse

$$(C + \lambda I_N)^{-1} = \frac{1}{N}\Phi^* diag\left(\frac{1}{\lambda_0 + \lambda}, \ldots, \frac{1}{\lambda_{N-1} + \lambda}\right)\Phi. \tag{13}$$

Thus, we can compute the inverse of the regularized $C+\lambda I_N$ in $O(N^2 \log N)$ time. Notice, however, that solving the kernel ridge regression problem for a given training set of size N and a given test set of size T doesn't require the explicit inverse, but rather the product $K_{\mathbf{x}}(K + \lambda I_N)^{-1}\mathbf{y}$. Substituting a suitable circulant C for K gives

$$K_{\mathbf{x}}(K + \lambda I_N)^{-1}\mathbf{y} \approx K_{\mathbf{x}}(C + \lambda I_N)^{-1}\mathbf{y} = K_{\mathbf{x}}\frac{1}{N}\Phi^* diag\left(\frac{1}{\lambda_0 + \lambda}, \ldots, \frac{1}{\lambda_{N-1} + \lambda}\right)\Phi\mathbf{y}. \tag{14}$$

Since $K_{\mathbf{x}}\frac{1}{N}\Phi^*$ is simply the inverse discrete Fourier transform applied to the column vector $K_{\mathbf{x}}$, we can compute the product in $O(N \log N)$ time complexity using the FFT. Similarly, we can compute $\Phi\mathbf{y}$ in time complexity $O(N \log N)$. Thus, given T test points and a suitable vector $[c_0, \ldots, c_N]$, we can compute the right-hand side of equation 18 in $O(TN \log N)$, a substantial improvement over naive inversion.

3.2.2 Multilevel Circulant Matrices

We can generalize circulant matrices with a recursive definition as follows [1]; a zero-level circulant matrix is an ordinary circulant matrix as defined above. A p-level circulant matrix M of order $\mathbf{n}_p \in \mathbb{N}$ is a matrix comprised of $\mathbf{n}_p$ block matrices, each a $p-1$-level circulant matrix, in circulant arrangement. So if these $\mathbf{n}_p$ block matrices are $M_{p,0}, \ldots, M_{p,\mathbf{n}_p-1}$, then

$$M = \begin{bmatrix} M_0 & M_{\mathbf{n}_p-1} & \cdots & M_{\mathbf{n}_p-1} \\ M_1 & M_0 & \cdots & M_{\mathbf{n}_p-2} \\ \vdots & \vdots & \ddots & \vdots \\ M_{\mathbf{n}_p-1} & M_{\mathbf{n}_p-2} & \cdots & M_0 \end{bmatrix} \tag{15}$$

We characterize the total ordering of an $N \times N$ p-level circulant matrix M with a p-tuple of positive integers $\mathbf{n}_0, \ldots, \mathbf{n}_p$ such that $N = \prod_{t=0}^{p} \mathbf{n}_t$. As with the zero-level circulant matrix, we can characterize an $N \times N$ p-level circulant matrix with N scalar entries indexed with multidimensional indices whose qth entry is a member of the set $\{0, \ldots, \mathbf{n}_q - 1\}$. If $\mathbf{j}$ and $\mathbf{l}$ are multidimensional indices, then the entry $M_{\mathbf{j},\mathbf{l}} = m_{\mathbf{l}_p - \mathbf{j}_p \bmod \mathbf{n}_p, \ldots, \mathbf{l}_0 - \mathbf{j}_0 \bmod \mathbf{n}_0}$.

For example, suppose $p = 1$, $\mathbf{n}_1 = 3$, and $\mathbf{n}_0 = 2$, and M contains scalars $m_{0,0}$, $m_{0,1}$, $m_{1,0}$, $m_{1,1}$, $m_{2,0}$,and $m_{2,1}$. Then

$$M = \begin{bmatrix} m_{0,0} & m_{0,1} & m_{2,0} & m_{2,1} & m_{1,0} & m_{1,1} \\ m_{0,1} & m_{0,0} & m_{2,1} & m_{2,0} & m_{1,1} & m_{1,0} \\ m_{1,0} & m_{1,1} & m_{0,0} & m_{0,1} & m_{2,0} & m_{2,1} \\ m_{1,1} & m_{1,0} & m_{0,1} & m_{0,0} & m_{2,1} & m_{2,0} \\ m_{2,0} & m_{2,1} & m_{1,0} & m_{1,1} & m_{0,0} & m_{0,1} \\ m_{2,1} & m_{2,0} & m_{1,1} & m_{1,0} & m_{0,1} & m_{0,0} \end{bmatrix} \tag{16}$$

[1] We present the generalization herein for completeness; our experiments apply a single level circulant matrix

Similar to a zero-level circulant matrix, we can characterize the eigenvalues $\lambda_0, \ldots, \lambda_{N-1}$ of an $N \times N$ p-level circulant matrix M by applying the multidimensional discrete Fourier transform to the N scalars [10]. Again,

$$M = \frac{1}{N}\Phi^* diag(\lambda_0, \ldots, \lambda_{N-1})\Phi, \tag{17}$$

where Φ is the multidimensional discrete Fourier applied to the columns of the $N \times N$ identity matrix, each of whose indices follow $\mathbf{n}$, the total ordering of M. Thus, as before, given the scalars in M, we can compute

$$K_{\mathbf{x}}(M + \lambda I_N)^{-1}\mathbf{y} = K_{\mathbf{x}}\frac{1}{N}\Phi^* diag\left(\frac{1}{\lambda_0 + \lambda}, \ldots, \frac{1}{\lambda_{N-1} + \lambda}\right)\Phi\mathbf{y}. \tag{18}$$

in $O(N \log N)$ time complexity per test point, for a total time complexity of $O(NT \log N)$.

3.2.3 Kernel Approximation with Multilevel Circulant Matrices: G. Song's Algorithm

G. Song, et al. [9] gives an algorithm for multilevel circulant matrix construction; required for the algorithm is the kernel function $k : \mathbb{R} \to \mathbb{R}$ and a *magic* sequence of positive numbers $\{h_{N,k}\}$, indexed by the data size and the various levels $k = 0, \ldots, p$ of the p-level circulant matrix.

Algorithm:
For all multi-indices $\mathbf{j}$, define $e_{\mathbf{j}} =< \mathbf{j}_0 h_{N,0}, \ldots, \mathbf{j}_p h_{N,p} >$.
Let $t_{\mathbf{j}} = k(||e_{\mathbf{j}}||_2)$.
Let $D_{\mathbf{j}} = \{\mathbf{j}_0, \mathbf{n}_0 - \mathbf{j}_0 \bmod \mathbf{n}_0\} \times \cdots \times \{\mathbf{j}_p, \mathbf{n}_{p-1} - \mathbf{j}_{p-1} \bmod \mathbf{n}_{p-1}\}$, defined such that we exclude duplicates in each set.
Let $u_{\mathbf{j}} = \sum_{\mathbf{l} \in D_{\mathbf{j}}} t_{\mathbf{j}}$.
Let U be the p-level matrix defined using $\{u_{\mathbf{j}}\}$. Substitute U for K, the kernel matrix.

Figure 1: Multilevel circulant matrix construction

3.2.4 Analysis of G. Song's Algorithm

The time complexity of the algorithm in equation 1, discussed in [11], is $O((p+1)N2^{p+1})$. Clearly, the more factors of N we incorporate, the more closely the complexity approaches $O(N^2)$. Notice, interestingly, the algorithm doesn't explicitly mention the particular data on hand, but rather depends on the kernel function k and the *magic* sequence $\{h_{N,k}\}_{k=0}^{p}$. G. Song [9] gives the following theorem relating the data and the magic sequence, applying a *weighted* norm $||\cdot||_{W_r}$.

Definition 1. *Suppose multidimensional index* $\mathbf{j}$ *belongs to the multidimensional vector* $\mathbf{n} = (\mathbf{n}_0, \ldots, \mathbf{n}_p)$ *as defined above. For* $r > 0$*, define* $W_{r,\mathbf{n}}(\mathbf{j}) = e^{r\left|\left|\frac{\mathbf{n}}{2} - \mathbf{j}\right|\right|_2}$*. For a multilevel vector* $\mathbf{a}$ *indexed by* $\mathbf{n}$*, define the weighted norm* $||\mathbf{a}||_{W_r} = \sup_{\mathbf{j}} W_{r,\mathbf{n}}(\mathbf{j})\,|\mathbf{a}_{\mathbf{j}}|$*. For multilevel matrix* M *indexed by* $\mathbf{n}$*, define the induced multilevel matrix norm* $||M||_{W_r} = \sup\{||M\mathbf{a}||_\infty : ||\mathbf{a}||_{W_r} = 1\}$.

Theorem 1. *Circulant Convergence*
Given a population from which we extract a data set $X = \{x_{\mathbf{j}}\}$ *of size* N*, a (radial basis) kernel matrix* K_N*, a magic sequence of positive numbers* h_N*, circulant* U_N *constructed in the algorithm above, and*

- *there exist* $a, b > 0$ *where* $|k(s)| \leq ae^{-bs}$*,*
- *(Hölder* β*) there exist* $d, \beta > 0$ *such that* $|k(s) - k(t)| \leq d|s - t|^\beta$*,*
- *there exist* $h_0 > 0$ *and* $c \in \mathbb{R}$ *where* $||x_{\mathbf{j}} - x_{\mathbf{l}}||_2 \geq h_0||\mathbf{j} - \mathbf{l}||_2 + c$ *for* $N \in \mathbb{N}$*, and*
- *there exists* $h > 0$ *such that* $h_{N,k} \geq h$ *for all* $N \in \mathbb{N}$ *and* $k = 0, \ldots, p-1$*,*

then for each $r > 0$*, there exists* $r_0 > 0$ *such that for each* r' *where* $0 < r' < \frac{r_0}{4}$*, we have* $||U_N^{-1} - K_N^{-1}||_{W_{r_0}} \leq \left(||U_N - K_N||_{W_{r_0}} + 1\right) e^{-r' \min\{\mathbf{n}_0, \ldots, \mathbf{n}_{p-1}\}}$.

The first two conditions impose exponential decay and uniform smoothness on the radial basis kernel. Obvious affirmative examples include the Gaussian RBF and any finite extent kernel. The latter conditions constrain the distribution of the data and relate the data to the *magic* sequence of numbers. Essentially, the conditions require that as we pile more points into the sample from the population, the distances between points in the ordering remain bounded above a positive constant proportional to the distance between their indices. That is, points with sufficient distance between their indices cannot arbitrarily approach one another as we sample more points. These conditions together impose a decay condition on the kernel matrices $\{K_N\}_{N=1}^{\infty}$, guaranteeing a good approximation using a (multilevel) circulant matrix.

3.2.5 The *Magic* Sequence Partially Revealed

Curiously, the existing literature fails to exhibit a precise procedure by which we choose this *magic* sequence given the data. An application of the approximation algorithm, [8] selects data points in $\mathbb{R}$ uniformly spaced in some interval. Obviously, if the sorted set of data points exhibits pairwise distances of Δx, then setting $h_N = \Delta x$ satisfies the latter two conditions of the theorem. More generally, if we apply a zero-level circulant approximation, then given data vectors $\mathbf{x_1}, \ldots, \mathbf{x_N} \in \mathbb{R}^D$, we can sphere the data, sort by magnitude in ascending order, and let $h_0 = \min_j\{||\mathbf{x}_{j+1}||_2 - ||\mathbf{x}_j||_2\} > 0$ in $O(N \log N)$ time complexity.

Proposition 1. *The h_0, if it exists, obtained above satisfies the third condition of theorem 1.*

Proof. Notice, $||\mathbf{x}_{j+1} - \mathbf{x}_j||_2 \geq ||\mathbf{x}_{j+1}||_2 - ||\mathbf{x}_j||_2 \geq h_0$. We can induct on k to obtain $||\mathbf{x}_{j+k} - \mathbf{x}_j||_2 \geq h_0 k$. □

Assuming the number of pairs of data points sharing the same $L2$ norm is small, we can discard all but one for each unique magnitude. If most or all of the data points share the same magnitude, we can discard features until the magnitudes begin to differ. In our experiments, we leverage the h_0 above as a starting value in search of the optimal, letting $h = h_0$. In the experiments herein, the optimal h_N in the experiments tends to be much larger than the given h_0[2]

3.2.6 Some Intuition into the Algorithm

The algorithm relies on a particular layout of the data points. For instance, in the zero-level circulant matrix case, the algorithm assumes that points $\mathbf{x}_i$ and $\mathbf{x}_j$ are arranged such that $||\mathbf{x}_i - \mathbf{x}_j||_2 \approx h_N|i - j| \bmod N$. To enforce symmetry in the circulant matrix, for each $k \neq \frac{N}{2}$, we assign $u_k = u_{N-k} = k\,(h_N k) + k\,(h_N(N-k))$. Thus, the circulant matrix is symmetric, as is any kernel matrix. The circulant approximation effectively smooths the two values together. Interestingly, the existing literature, to our knowledge, fails to demonstrate how to prove whether the fourth condition in G. Song's theorem holds, except in trivial cases in which the data adhere to a growing grid; necessary and sufficient conditions for convergence are currently unknown.

3.3 Kernel Approximation with Circulant Matrices: Sampling Circulant Algorithm

We present a new algorithm similar in spirit to that of G. Song, requiring slightly more execution time but empirically better error rates in kernel ridge regression. Essentially, we sample a small number, say $O(\log N)$ of entries from the true kernel matrix to calculate the vector characterizing the circulant matrix.

[2]The h_0 values for the data sets tend to be near $1e-8$, whereas optimal h_N values are greater than 0.01.

Algorithm:
Input a kernel function k, data columns $\mathbf{x}_0, \ldots, \mathbf{x}_N$ and $M \in \mathbb{N}$; output a vector $\mathbf{u}$ defining the circulant matrix.
Let $\mathbf{u}_0 = k(0)$.
Let $\mathbf{u}_j = 0$ for $j = 1, \ldots, N-1$.
Repeat M times:
Sample $l_0, \ldots, l_{\lceil \log N \rceil}$ indices from $\{0, \ldots, N-1\}$.
For $j = 1, \ldots, \lfloor \frac{N}{2} \rfloor$, add $\frac{1}{2} \sum_{k=0}^{\lceil \log N \rceil} k\left(||x_j - x_{j-l_k \bmod N}||_2\right) + k\left(||x_j - x_{N-j-l_k \bmod N}||_2\right)$ to both $\mathbf{u}_j$ and $\mathbf{u}_{N-j}$.
End Repeat
Rescale $\mathbf{u}_1, \ldots, \mathbf{u}_{N-1}$ by $M\lceil \log N \rceil$.

Figure 2: Sampling Circulant Algorithm

3.3.1 Analysis of Our New Algorithm

Clearly, algorithm 2 requires $O(MN \log N)$ time complexity, where M is the number of sample iterations and N is the number of data points. As before, we can compute the eigenvalues of the circulant matrix defined by $\mathbf{u}$ in $O(N \log N)$ time complexity leveraging the FFT. As with the multilevel circulant algorithm with $p = 0$, given T test points, the total kernel ridge regression computation requires $O(TN \log N)$, for a total $O((T + M)N \log N)$ time complexity.

3.3.2 Convergence Characteristics

Since the entries of the circulant matrix obtained by the sampling circulant algorithm represent sample means of corresponding entries in the true kernel matrix, we have the following.

Proposition 2. *Convergence Characteristics of the Sampling Circulant Matrix*
Given an $N \times N$ kernel matrix K and the circulant matrix U_M obtained in algorithm 2 after M iterations,

$$U = \lim_{M \to \frac{N}{\log N}} U_M = \underset{C \text{ is circulant}}{\operatorname{argmin}} ||K - C||_{fro}. \tag{19}$$

That is, the circulant matrix we obtain converges to the optimal circulant approximation for K under the Frobenius norm. In some sense, this suggests that our approximation approaches the best possible circulant match for K for a given N. G. Song's theorem, by contrast, suggests convergence only as N becomes arbitrarily large.

3.4 Advantages of the Circulant Approaches

3.4.1 Argument by Complexity

G. Song's multilevel circulant algorithm with $p = 0$ and the sampling circulant algorithm require time complexities $O(TN \log N)$ and $O((T + M)N \log N)$, respectively. Random features and incomplete Cholesky decomposition require time complexity asymptotic with naive inversion if we press the error tolerance too much. The experiments below illustrate this, as incomplete Cholesky decomposition's applicability varies dramatically from data set to data set.

3.4.2 Argument by Parameter Manipulation

G. Song's multilevel circulant algorithm features a running time invariant with respect to applicable parameters (the $\mathbf{h}_N$ vector, bandwidths, and regularizers), desirable when we must search with fine granularity for the optimal parameter set. Our sampling circulant algorithm features a single parameter which affects run time in an obvious way; increasing M increases linearly the runtime of a portion of the algorithm. By contrast, incomplete Cholesky decomposition features a parameter whose runtime effects are difficult to predict; a candidate η can result in intractable runtimes if the number of pivots approaches N. Random features suffers a similar, albeit more manageable drawback; the runtime asymptotically follows the cube of the number of features. Finally, like random features, the sampling circulant algorithm exhibits a parameter space simple to search; M

Table 1: Optimal parameter errors I, CT

	Random Features			Incomplete Cholesky Decomposition			
N	D^*	error	time (s)	η^*	pivots	error	time (s)
1K	178	521.68	0.591	0.0667	122	205	19.383
5K	595	1173	14.84	0.075	143	389.4	106.19

is a positive integer, so we need only increase it to search for a better test error. Empirically, as we'll see below, we need not increase M too far from zero.

3.5 Disadvantages of the Circulant Approaches

The unpredictable applicability of incomplete Cholesky decomposition is a double-edged sword; a kernel matrix very easily approximated with a relatively low number of pivots can rival the circulant methods in accuracy with a fraction of the runtime.

4 Experiments

4.1 Data Sets

We apply the above algorithms to selections from CT slices on an axial axis [13] and the Sloane Digital Sky Survey 2006 (SDSS) [12]. Our experiments with SDSS attempt to estimate the *zSpec* attribute, a measure of red shift, given various position attributes. The comparable experiments with CT attempt to estimate the position of a slice given various visual attributes. Our selection of SDSS data contains 11 attributes, while the CT data contains 385 attributes.

4.2 Design

4.2.1 Regularizers and Bandwidths

We sphere the data set (subtract the sample mean and divide by the sample standard deviation of the features.) To select the optimal regularizer and bandwidth, we randomly select 1000 records from each of the two data sets, and search for the optimal pair in the rectangle $[0, 100] \times [0, 100]$. At each stage of the iteration, we search ten slices along each axis, pick the best point, then perform ten additional slices around the optimal pick, until the size of a slice is 0.5. For completeness, we search $[0, 10] \times [0, 10]$ and $[0, 1] \times [0, 1]$ similarly, though in both data sets, the optimal pair was much larger. For each regularizer, bandwidth pair, we perform ten cross validation passes, training on 90% of the subset randomly selected, and testing on the 10% remaining. The optimal pair for CT seems to be $\sigma^* = 50, \lambda^* = 75$. For SDSS, we found $\sigma^* = 25, \lambda^* = 40$.

4.2.2 Parameter Selection

In applying random features, we vary the number of features from $\left[1, N^{\frac{3}{4}}\right]$. In applying incomplete Cholesky decomposition, we vary the parameter η within $[0, 1]$ to obtain an optimal pivot count. In applying the multilevel circulant algorithm, we sort the data by $L2$ norms, then vary h_N between $[0, 1]$, typically choosing quite small values. For the sampling circulant algorithm, we vary M, the number of repeats, from 1 to 25 for small N (1K to 5K) and 1 to 10 for larger values of N. For each value of each of the aforementioned parameters, we perform ten cross-validation passes, randomly selecting 90% of the current data subset for training and the remaining 10% for testing, averaging the error rates and execution times. Tables 1, 2, 3, and 4 exhibit experiment results, where each reported error is the $L2$ norm of the difference between the reported label vector of test points and the measured label vector of test points. Thus, the errors mostly increase as N increases.

Table 2: Optimal parameter errors II, CT

	Multilevel Circulant			Sampling Circulant		
N	h_N^*	error	time (s)	M^*	error	time (s)
1K	0.1	200.05	0.92	3	189.53	2.77
5K	0.00333	359.43	14.37	1	356.5	22.7

Table 3: Optimal parameter errors I, SDSS

	Random Features			Incomplete Cholesky Decomposition			
N	D^*	error	time (s)	η^*	pivots	error	time (s)
500	21	1.2292	0.000131	0.1	3	0.68096	0.257
1K	8	1.728	0.0066591	0.04	3	0.97043	0.4529358
5K	500	3.8858	3.55909	0.04	5.8	1.7509	4.33
7.5K	58	4.788	0.44742	0.01667	9	1.9789	10.09
10K	1000	5.5283	59.9	0.05	4.8	2.208	7.9

Table 4: Optimal parameter errors II, SDSS

	Multilevel Circulant			Sampling Circulant		
N	h_N^*	error	time (s)	M^*	error	time (s)
500	0.05	0.82251	0.50	1	0.66958	0.97
1K	0.072	0.91605	1.1054	1	0.88463	2.7
5K	0.012	1.4062	19.951	3	1.4302	60.908
7.5K	0.01667	3.3033	79.84	2	1.7853	101.26
10K	0.00333	2.74	85.79	3	2.5111	

4.3 Analysis

The fastest of the four algorithms, clearly, is random features. The most accurate of the four is marginally the sampling circulant algorithm, followed closely by the multilevel circulant algorithm. Both random features and incomplete Cholesky decomposition ostensibly can match the accuracy of the circulant methods, given enough processing space and time, though their operating overhead asymptotically follows that of naive inversion if we press for better accuracy. Incomplete cholesky decomposition performs poorly on CT but predicts well on SDSS with relatively few pivots, outperforming the circulant methods in execution time while rivaling accuracy.

5 Concluding Remarks

We have presented a new kernel approximation technique using circulant matrices, similar to work by G. Song, et al. [8], [9]. Applied in kernel ridge regression, our algorithm offers competitive accuracy compared to random features, incomplete Cholesky decomposition, and the multilevel circulant matrix approximation. As expected, our algorithm is marginally slower than the multilevel circulant approach and much slower than random features. Our experiments and those of Ding, et al. [11] suggest that circulant matrices offer a good approximation to kernel matrices even when convergence fails to occur[3], or that both necessary and sufficient conditions are not currently known to guarantee convergence. In either case, the circulant matrix approach seems applicable in a wider range of data sets than is obvious from the sufficient conditions of G. Song's theorem or the relatively weak statement of Frobenius norm optimality in proposition 2. We plan to explore the theoretical underpinnings of the sampling circulant algorithm and pursue techniques to expand and speed the pointwise summations to improve the approximation. On the question of convergence, we suspect that a line of reasoning similar to that of G. Song, et al. [9] can account for the empirically good approximation of our algorithm.

[3]Divergent series in real analysis sometimes offer reasonable approximations to their associated functions; consider the Stirling series for $\log \Gamma(x)$.

References

1. A.N. Tychonoff. On the stability of inverse problems, *Doklady Akademii Nauk SSSR* 39 (5): 195198, 1943.
2. A.E. Hoerl AE. Application of ridge analysis to regression problems, *Chemical Engineering Progress*, 58, 5459, 1962.
3. B.W. Silverman. *Density Estimation for Statistics and Data Analysis*, Chapman and Hall/CRC, 1986.
4. J. Shawe-Taylor, N. Cristianini. *Kernel Methods for Pattern Analysis,*Cambridge University Press, 2004.
5. L. Song, B. Boots, S. Siddiqi, G. Gordon and A. Smola. Hilbert space embedding of hidden Markov model. *ICML*, 2010.
6. G. Golub and C. Van Loan. *Matrix Computations (3rd ed.)*, Johns Hopkins, 1996.
7. A. Rahimi and B. Recht. Random Features for Large-Scale Kernel Machines, *NIPS*, 2007.
8. G. Song. Approximation of kernel matrices in machine learning. PhD thesis, Syracuse University, Syracuse, NY, USA, 2010.
9. G. Song, Y. Xu. Approximation of high-dimensional kernel matrices by multilevel circulant matrices. *Journal of Complexity*, 26(4):375405, 2010.
10. E.E. Tyrtyshnikov. A unifying approach to some old and new theorems on distribution and clustering. *Linear Algebra and its Applications*, 232:143, 1996.
11. L.Z. Ding, S.Z. Liao. Approximate model selection for large scale LSSVM. *Journal of Machine Learning Research Proceedings Track*, 20:165180, 2011.
12. The Sloan Digital Sky Survey [http://sdss.org], 2006.
13. A. Frank, A. Asuncion. UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science, 2010.