On the Approximation of Quadratic Forms

The Nyström Extension & Spectral Methods in Learning

Patrick J. Wolfe (joint work with Mohamed-Ali Belabbas)

School of Engineering and Applied Sciences
Department of Statistics, Harvard University

Approximation and Learning in Higher Dimensions
Texas A&M University, 19 October 2007

Outline

Approximation of quadratic forms in learning theory

Outline

Approximation of quadratic forms in learning theory

Outline

Approximation of quadratic forms in learning theory

Spectral Methods in Learning

The discrepancy between data and information

What role do spectral methods play in statistical learning?

- Goal: get relevant "information" about very large datasets in very high dimensional spaces
  - Image segmentation, low-dimensional embeddings, ...
- What is the "relevant" information contained in the data set?
- Spectral methods reduce this question to finding a low-rank approximation to a symmetric, positive semi-definite (SPSD) kernel—equivalently, a quadratic form
- They can be quite effective, and see wide use:
  - Older methods: principal components analysis (1901), multidimensional scaling (1958), ...
  - Newer methods: isomap, Laplacian eigenmaps, Hessian eigenmaps, diffusion maps, ...

Application of Low-Rank Approximations to Learning

Inner and outer characteristics of the point cloud

Let $\{x_1, \ldots, x_n\}$ be a collection of data points in $\mathbb{R}^m$. Spectral methods can be classified according to whether they rely on:

Outer characteristics of the point cloud (PCA, discriminants).

Here we work directly in the ambient space. Require spectral analysis of a positive-definite kernel of dimension m, the extrinsic dimensionality of the data.

Inner characteristics of the point cloud (MDS, extensions).

Embedding requires the spectral analysis of a kernel of dimension n, the cardinality of the point cloud.

The spectral analysis task typically consists of finding a rank-k approximation to a symmetric, positive semi-definite matrix.

How to Approximate an SPD Matrix, in Theory?

Finding a low-rank approximation is *easy*...

- An SPSD matrix G can be written in *spectral coordinates*

$$G = U\Lambda U^T,$$

where U is orthogonal and $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n)$ is diagonal.

- The λ_i's are the eigenvalues of G, ordered such that $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0$, and the u_i's are the eigenvectors.
- For any unitarily invariant norm $\|\cdot\|$, we have that

$$\underset{\widetilde{G}:\, \text{rank}(\widetilde{G})=k}{\text{argmin}} \quad \|G - \widetilde{G}\| \quad = \quad U\Lambda_k U^T =: G_k,$$

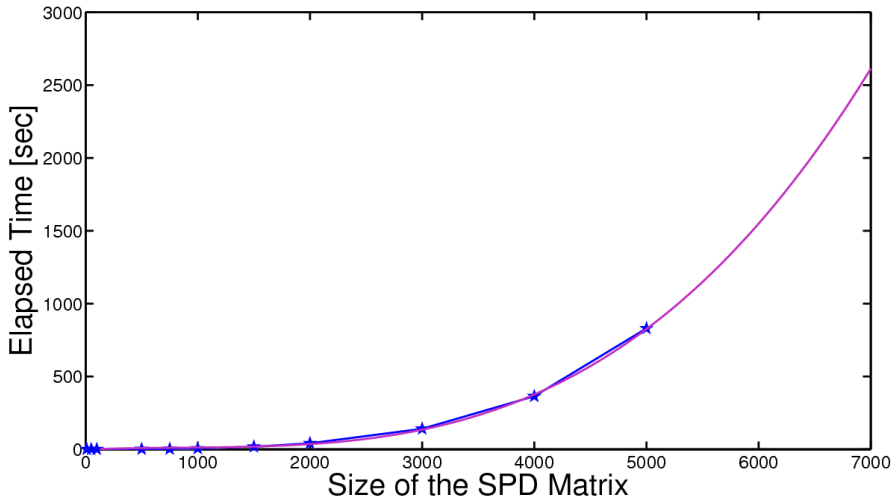
where $\Lambda_k = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_k, 0, \ldots, 0)$

How to Approximate an SPD Matrix, in Practice?

Finding a low-rank approximation is *hard*!

Changing to spectral coordinates is done using the Singular Value Decomposition of G, which requires $\mathcal{O}(n^3)$ operations

- On a Pentium IV 3GHZ desktop PC, with 1GB RAM, 512k Cache:



- Extrapolating to $n = 10^6$, factoring G takes more than 4 months.
- When n increases, G quickly becomes too large to be stored in memory

Approximating Large Kernels

How to discriminate between data and information?

This presents a practical problem for large data sets!

- A commonly used "trick" is to sparsify the kernel.
  - Fix $\varepsilon > 0$. If $G_{ij} \leq \varepsilon$, set $G_{ij} = 0$
  - Questions: How to choose ε? How accurate is the result?
- Alternative approach: discard some of the data.
  - How to construct a low-rank approximation using just some of the data? The Nyström extension provides an answer
- The basic idea is as follows:
  - Write $G = X^T X$, so that G is a Gram matrix for vectors $X_1, \ldots, X_n$.
  - Choose a subset I of vectors X_i and their correlations with *all* the other vectors to find a low-rank approximation $\widetilde{G}$.

A Provably Good Low-Rank Approximation

Our main result on approximating quadratic forms

How to choose $I : |I| = k$ so as to minimize $\|G - \tilde{G}\|$?

- This is equivalent to asking: "How to choose the *most informative* part from our dataset?"—most informative being conditioned on our reconstruction scheme
- There are $\frac{n!}{k!(n-k!)}$ multi-indices—no hope of enumerating
- We define the following distribution on multi-indices:

$$p_{G,k} = \frac{\det(G_{I\times I})}{\sum_{|I|=k} \det(G_{I\times I})}$$

- Our main result will be to show that, for spectral decomposition $G = U\Lambda U^T$, we have in Frobenius norm:

$$\mathbb{E}\|G - \tilde{G}\| \leq (k+1)(\lambda_{k+1} + \lambda_{k+2} + \ldots + \lambda_n)$$

Outline

Approximation of quadratic forms in learning theory

The Nyström Extension

Simplify the problem

- Historically, the *Nyström extension* was introduced to obtain numerical solutions to integral equations.
- Let $g : [0,1] \times [0,1] \to \mathbb{R}$ be an SPSD kernel and (u_i, λ_i^u), $i \in \mathbb{N}$, denote its pairs of eigenfunctions and eigenvalues:

$$\int_0^1 g(x,y)u_i(y)\,dy = \lambda_i^u u_i(x), \quad i \in \mathbb{N}.$$

- The Nyström extension approximates the eigenvectors of $g(x,y)$ by evaluation of the kernel at k^2 distinct points
- Let $\{(x_m, x_n)\}_{m,n=1}^k \in [0,1] \times [0,1]$.
- Define $G(m,n) \equiv G_{mn} := g(x_m, x_n)$

The Nyström Extension

Extend the solution

- We now solve a finite dimensional problem

$$\frac{1}{k}\sum_{n=1}^{k} G(m,n)v_i(n) = \lambda_i^v v_i(m), \quad i = 1,2,\ldots,k,$$

where (v_i, λ_i^v) represent the k eigenvector-eigenvalues pairs associated with G.

- What do we do with these eigenvectors? We *extend* them to approximate $\widetilde{u}_i \approxeq u_i$ as follows:

$$\widetilde{u}_i(x) = \frac{1}{\lambda_i^v k}\sum_{m=1}^{k} g(x, x_m)v_i(m).$$

- In essence: only use *partial information* about the kernel to solve a simpler eigenvalue problem, and then to *extend* the solution using complete knowledge of the kernel.

The Nyström Extension

In finite dimensions

The Nyström extension first solves a simpler eigenfunction/eigenvalue problem.

- How do we translate this to a finite dimensional setting?
- We approximate k eigenvectors of G by decomposing and then extending a $k \times k$ principal submatrix of G.
- We partition G as follows

$$G = \begin{bmatrix} A & B^T \\ B & C \end{bmatrix},$$

with $A \in \mathbb{R}^{k \times k}$; we say that this partition corresponds to the multi-index $I = \{1, 2, \ldots, k\}$.

- Define spectral decompositions $G = U \Lambda U^T$ and $A = U_A \Lambda_A U_A^T$

The Nyström Extension

The approximation error

- The Nyström extension then provides an approximation for k eigenvectors in U as

$$\widetilde{U} := \begin{bmatrix} U_A \\ BU_A\Lambda_A^{-1} \end{bmatrix}; \quad A = U_A\Lambda_A U_A^T.$$

- In turn, the approximations $\widetilde{U} \approxeq U$ and $\Lambda_A \approxeq \Lambda$ may be composed to yield an approximation $\widetilde{G} \approxeq G$ according to

$$\widetilde{G} := \widetilde{U}\Lambda_A\widetilde{U}^T = \begin{bmatrix} A & B^T \\ B & BA^{-1}B^T \end{bmatrix}.$$

- The resultant approximation error is

$$\|G - \widetilde{G}\| = \|C - BA^{-1}B^T\|,$$

the norm of the Schur complement of A in G

Adjusting Computational Load vs. Approximation Error

From eigenanalysis to partitioning

On what fronts do we gain by using the Nyström extension?

- What is required is the spectral analysis of a kernel of size $k \leq n \Rightarrow$ gain in space and time complexity.
- But we introduced another problem: how to partition G?

In other words, we have shifted the computational load from eigenanalysis to the determination a "good" partition

- The latter problem is more amenable to approximation
- We give two algorithms to solve it, along with error bounds...

The Nyström Extension

A combinatorial problem

We now introduce the problem formally with notation:

- $I, J \subset \{1, \ldots, n\}$ are multi-indices of respective cardinalities k and l, containing pairwise distinct elements in $\{1, \ldots, n\}$.
- We write $I = \{i_1, \ldots, i_k\}$, $J = \{j_1, \ldots, j_l\}$, and denote by $\bar{I}$ the complement of I in $\{1, \ldots, n\}$.
- Define $G_{I \times J}$ for the $k \times l$ matrix whose (p, q)-th entry is given by $(G_{I \times J})_{pq} = G_{i_p j_q}$. Abbreviate G_I for $G_{I \times I}$.
- The partitioning problem is equivalent to selecting a multi-index I such that the error

$$\|G - \widetilde{G}\| = \|G_{\bar{I}} - G_{\bar{I} \times I} G_I^{-1} G_{I \times \bar{I}}\| = \|S_C(G_I)\|$$

is minimized.

The Nyström Method and Exact Reconstruction

Recovery when $\text{rank}(G_I) = \text{rank}(G) = k$

When does the Nyström method admit exact reconstruction?

- If we take for I the entire set $\{1, 2, \ldots, n\}$, then the Nyström extension yields $\widetilde{G} = G$ trivially
- If G is of rank $k < n$, then there exist $I : I = |k|$ such that the Nyström method yields exact reconstruction
- These I are those such that $\text{rank}(G_I) = \text{rank}(G) = k$
  - Intuition: express G as a Gram matrix whose entries comprise the inner products of n vectors in $\mathbb{R}^k$
  - Knowing the correlation of these n vectors with a subset of k *linearly independent* vectors allows us to recover them
  - Information contained in G_I is sufficient to reconstruct G; Nyström extension performs the reconstruction
- To verify, we introduce our first lemma...

Verifying the Perfect Reconstruction Property

Characterizing Schur complements as ratios of determinants

Lemma (Crabtree-Haynsworth)

Let G_I be a nonsingular principal submatrix of some SPSD matrix G. The Schur complement of G_I in G is given element-wise by

$$(S_C(G_I))_{ij} = \frac{\det(G_{I\cup\{i\}\times I\cup\{j\}})}{\det(G_I)}. \tag{1}$$

This implies that for I such that $\text{rank}(G_I) = \text{rank}(G) = k$,

$$S_C(G_I) = G_{\bar{I}} - G_{\bar{I}\times I} G_I^{-1} G_{I\times\bar{I}} = 0.$$

- If $\text{rank}(G) = k = |I|$, then (1) implies that $\text{diag}(S_C(G_I)) = 0$
- Positive definiteness of G implies positive definiteness of $S_C(G_I)$ for any multi-index I, allowing us to conclude that $S_C(G_I)$ is identically zero.

Randomized Low-Rank Kernel Approximation

Randomized multi-index selection by weighted sampling

Whether $\text{rank}(G) = k$ or $\text{rank}(G) > k$, we are faced with the task of selecting a multi-index I from amongst a set of $\binom{n}{k}$ possibilities. This motivates our first algorithm for multi-index selection:

- Observation: Since G is positive definite, it induces a probability distribution on the set of all $I : |I| = k$ as follows:

$$p_{G,k}(I) \propto \det(G_I),$$

  with the normalizing constant being $\sum_{I,|I|=k} \det(G_I)$
- Algorithm: first sample $I \sim p_{G,k}(I)$, then perform the Nyström extension on the chosen multi-index

Recall: if $\text{rank}(G) = k$ and we seek a rank-k approximant $\widetilde{G}$, then $\widetilde{G} = G$ by our previous argument.

Randomized Multi-Index Selection by Weighted Sampling

Statement of the main result

Our randomized algorithm for multi-index selection admits the following error bound in expectation:

Theorem (Randomized Multi-Index Selection)

Let G be a real, $n \times n$, positive quadratic form with eigenvalues $\lambda_1 \geq \ldots \geq \lambda_n$. Let $\widetilde{G}$ be the Nyström approximation to G corresponding to I, with $I \sim p_{G,k}(I) \propto \det(G_I)$. Then

$$\mathbb{E}\,\|G - \widetilde{G}\| \leq (k+1) \sum_{I=k+1}^{n} \lambda_I. \tag{2}$$

Proof of the Randomized Multi-Index Result I

Randomized algorithm for multi-index selection

Proof.

We seek to bound

$$\mathbb{E}\,\|G - \widetilde{G}\| = \frac{1}{\sum_{I,|I|=k} \det(G_I)} \sum_{I,|I|=k} \det(G_I)\,\|S_C(G_I)\|.$$

Denote the eigenvalues of $S_C(G_I)$ as $\{\bar{\lambda}_j\}_{j=1}^{n-k}$; positive definiteness and subadditivity of the square root imply that

$$\|S_C(G_I)\| = \sqrt{\sum_j \bar{\lambda}_j^2} \;\leq\; \sum_j \bar{\lambda}_j = \mathrm{tr}(S_C(G_I)).$$

Proof of the Randomized Multi-Index Result II

Randomized algorithm for multi-index selection

Proof.

The Crabtree-Haynsworth Lemma yields

$$\mathrm{tr}(S_C(G_I)) = \sum_{i \notin I} \frac{\det(G_{I \cup \{i\}})}{\det(G_I)},$$

and thus

$$\mathbb{E}\,\|G - \widetilde{G}\| \leq \frac{1}{Z} \sum_{I,|I|=k} \sum_{i \notin I} \det(G_{I \cup \{i\}}), \tag{3}$$

where $Z = \sum_{I,|I|=k} \det(G_I)$ is the normalizing constant of $p_{G,k}(I)$.

Proof of the Randomized Multi-Index Result III

Randomized algorithm for multi-index selection

Proof.

Every multi-index of cardinality $k+1$ appears exactly $k+1$ times in the double sum of (3) above, whence

$$\mathbb{E}\,\|G - \widetilde{G}\| \leq \frac{(k+1)}{Z} \sum_{I,|I|=k+1} \det(G_I). \tag{4}$$

The sum of the principal $(k+1)$-minors of G can be expressed as the sum of $(k+1)$-fold products of its ordered eigenvalues (Cauchy-Binet):

$$\sum_{I,|I|=k+1} \det(G_I) = \sum_{\substack{1\leq j_1<j_2<\dots \\ <j_{k+1}\leq n}} \lambda_{j_1}\lambda_{j_2}\cdots\lambda_{j_{k+1}}.$$

Proof of the Randomized Multi-Index Result IV

Randomized algorithm for multi-index selection

Proof.

It thus follows that

$$\sum_{I,|I|=k+1} \det(G_I) \leq \sum_{\substack{1\leq j_1<j_2<\dots \\ <j_k\leq n}} \lambda_{j_1}\lambda_{j_2}\cdots\lambda_{j_k} \sum_{l=k+1}^{n} \lambda_l$$

$$= \sum_{I,|I|=k} \det(G_I) \sum_{l=k+1}^{n} \lambda_l.$$

Combining this relation with (4) above, we obtain

$$\mathbb{E}\,\|G - \widetilde{G}\| \leq \frac{(k+1)}{Z} \sum_{I,|I|=k} \det(G_I) \sum_{l=k+1}^{n} \lambda_l = (k+1) \sum_{l=k+1}^{n} \lambda_l,$$

which concludes the proof. ■

Deterministic Multi-Index Selection by Sorting

A different flavor of result

- We obtain an SPSD approximant $\widetilde{G}$ such that

$$\mathbb{E}\,\|G - \widetilde{G}\| \leq (k+1) \sum_{i=k+1}^{n} \lambda_i$$

in the Frobenius norm, as compared to the optimum

$$\|G - \widetilde{G}_{\text{opt}}\| = \left(\sum_{i=k+1}^{n} \lambda_i^2 \right)^{1/2}$$

afforded by the full spectral decomposition.

- Two practical issues:
  - Complexity of sampling from $p_{G,k}(I) \propto \det(G_I)$
  - Desire for deterministic rather than probabilistic result

Deterministic Low-Rank Kernel Approximation

Deterministic multi-index selection by sorting

- We now present a low-complexity deterministic multi-index selection algorithm and provide a bound on its worst-case error
- Let I contain the indices of the k largest diagonal elements of G and then implement the Nyström extension. Then we have:

Theorem (Deterministic Multi-Index Selection)

Let G be a real positive-definite kernel, let I contain the indices of its k largest diagonal elements, and let $\widetilde{G}$ be the corresponding Nyström approximation. Then

$$\|G - \widetilde{G}\| \leq \sum_{i \notin I} G_{ii}. \tag{5}$$

Proof of the Deterministic Multi-Index Result I

Deterministic algorithm for multi-index selection

We have sacrificied some power to obtain gains in the deterministic nature of the result and in computational efficiency:

$$\|G - \widetilde{G}\| \leq \sum_{i=k+1}^{n} G_{ii} \text{ (sorting) vs. } \mathbb{E}\,\|G - \widetilde{G}\| \leq (k+1) \sum_{i=k+1}^{n} \lambda_i \text{ (sampling)}$$

The proof of this theorem is straightforward, once we have the following generalization of the Hadamard inequality:

Lemma (Fischer's Lemma)

If G is a positive-definite matrix and G_I a nonsingular principal submatrix then

$$\det(G_{I\cup\{i\}}) < \det(G_I)G_{ii}.$$

Proof of the Deterministic Multi-Index Result II

Deterministic algorithm for multi-index selection

Proof of the Theorem.

We have from our earlier proof that $\|G - \widetilde{G}\| \leq \mathrm{tr}(S_C(G_I))$; applying Crabtree-Haynsworth in turn gives

$$\|G - \widetilde{G}\| \leq \frac{1}{\det(G_I)} \sum_{i \notin I} \det(G_{I \cup \{i\}}),$$

after which Fischer's Lemma yields $\|G - \widetilde{G}\| \leq \sum_{i \notin I} G_{ii}$. ■

- In other work (Belabbas and W., 2007), we have shown this algorith to perform well in an array signal processing context.
- Beginning with the case $k = 1$, it may be seen through repeated application of the theorem to constitute a simple stepwise-greedy approach to multi-index selection.

Remarks and Discussion

Comparison to known results

- Drineas et al. (2005) proposed to choose row/column subsets by sampling, independently and with replacement, indices in proportion to elements of $\{G_{ii}^2\}_{i=1}^n$, and were able to show:

$$\mathbb{E}\,\|G - \widetilde{G}\| \leq \|G - G_k\| + 2\sqrt{2}\sum_{i=1}^{n} G_{ii}^2,$$

- Our randomized approach yields a relative error bound
  Algorithmic complexity: $\mathcal{O}(k^3 + (n-k)k^2)$
- Our deterministic approach offers improvement if $\mathrm{tr}(G) \geq n$; complexity $\mathcal{O}(k^3 + (n-k)k^2 + n\log k)$
- Connections to the recently introduced notion of *volume sampling* in theoretical computer science

Outline

Approximation of quadratic forms in learning theory

Implementation of the Sampling Scheme

Sampling from $p_{G,k}$

- Sampling directly from $p_{G,k} \propto \det(G_I)$ is infeasable
- Simulation methods provide an appealing alternative
- We employed the Metropolis algorithm to simulate an ergodic Markov chain admitting $p_{G,k}(I)$ as its equilibrium distribution:
  - The proposal distribution is straightforward: exchange one index from I with one index from $\bar{I}$ uniformly at random
  - Distance to $p_{G,k}(\cdot)$ in total variation norm typically observed to be small after on the order of $50|I|$ iterations of the chain.
- We made no attempt to optimize this choice, as its performance in practice was observed to be satisfactory

A Metropolis Algorithm for Sampling from $p_{G,k}$

Implementation of the sampling scheme

Implementation of the Metropolis sampler is straightforward and intuitive:

- Begin with data $X = \{x_1, x_2, \ldots, x_n\} \in \mathbb{R}^m$
- Initialize (in any desired manner) a multi-index $I^{(0)}$ of cardinality k
- Compute the sub-kernel $W^{(0)}(X, I^{(0)})$
- After T iterations, return $I \sim p_{G,k}$

INPUT : Data X, $0 \leq k \leq n$, $T > 0$,
$k \times k$ sub-kernel $W^{(0)}$ with indices $I^{(0)}$

OUTPUT : Sampled k-multi-index I

for $t = 1$ to T **do**
  pick $s \in \{1, 2, \ldots, k\}$ uniformly at random
  pick $j'_s \in \{1, 2, \ldots, n\} \setminus I^{(t-1)}$ at random
  $W' \Leftarrow \text{UpdateKernel}(W^{(t-1)}, X, s, j'_s)$
  with probability $\min(1, \frac{\det(W')}{\det(W^{(t-1)})})$ **do**
    $W^{(t)} \Leftarrow W'$
    $I^{(t)} \Leftarrow \{j'_s\} \cup I^{(t-1)} \setminus \{j_s\}$
  otherwise
    $W^{(t)} \Leftarrow W^{(t-1)}$
    $I^{(t)} \Leftarrow I^{(t-1)}$
  end do
end for

HARVARD ENGINEERING AND APPLIED SCIENCES

Simulation Results

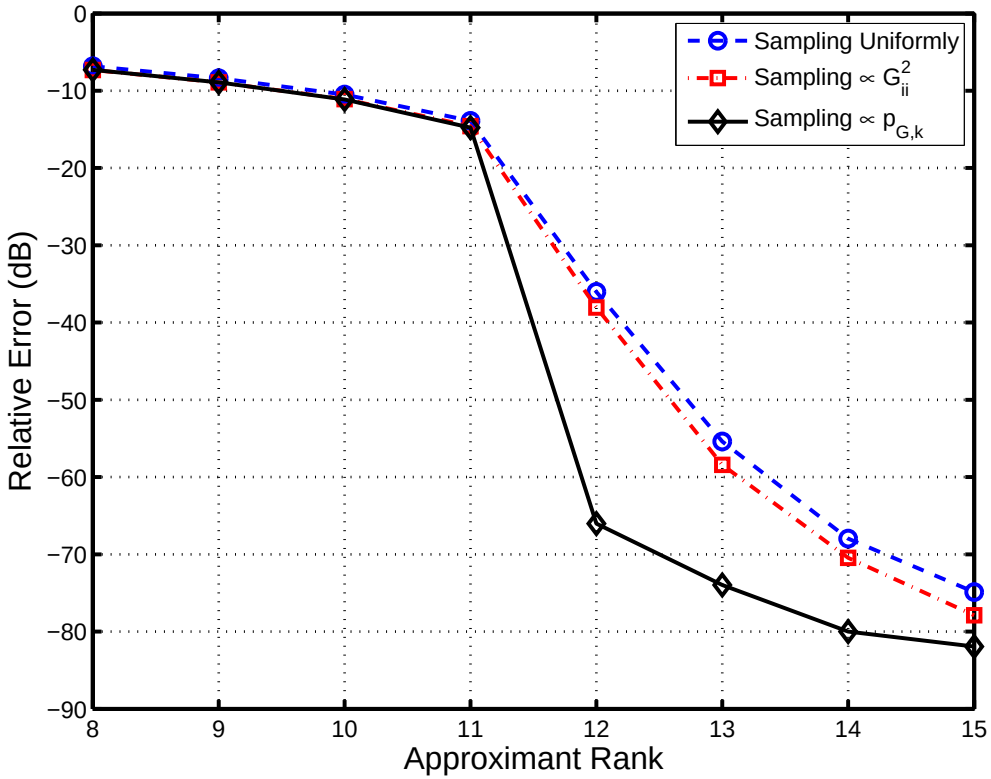
Experimental setup

First, we compare the different randomized algorithms for multi-index selection with one another, and with the method of Drineas et al. (2005):

- Three different settings for approximation error evaluation: Principal components, Diffusion maps, and Laplacian eigenmaps.
- We draw kernels at random from ensembles relevant to the test setting, and then average (though results do not imply a measure on the input space)
- For each kernel drawn, we further average over many runs of the randomized algorithm.

Approximate Principal Components Analysis

Randomized multi-index selection

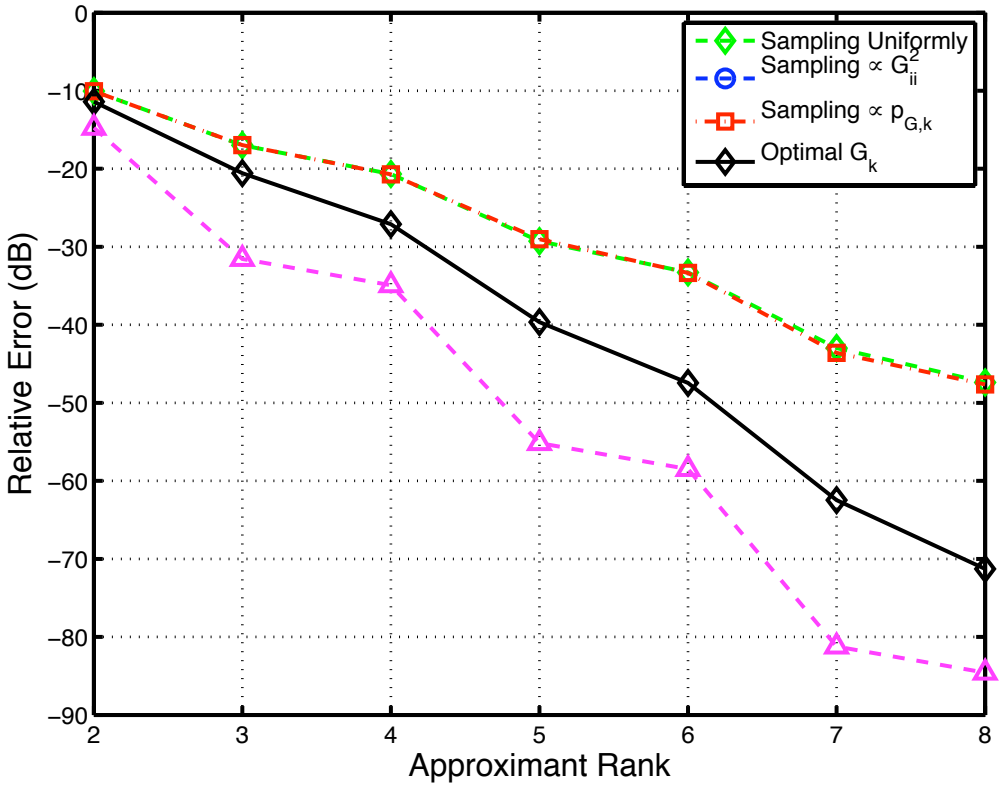


- We drew 1000 50×50 SPD matrices of rank 12 from a Wishart ensemble.
- We show the error of several algorithms used to perform a low rank approximation (outputs averaged over 250 trials)

Harvard Engineering and Applied Sciences

Approximate Diffusion Maps

Randomized multi-index selection

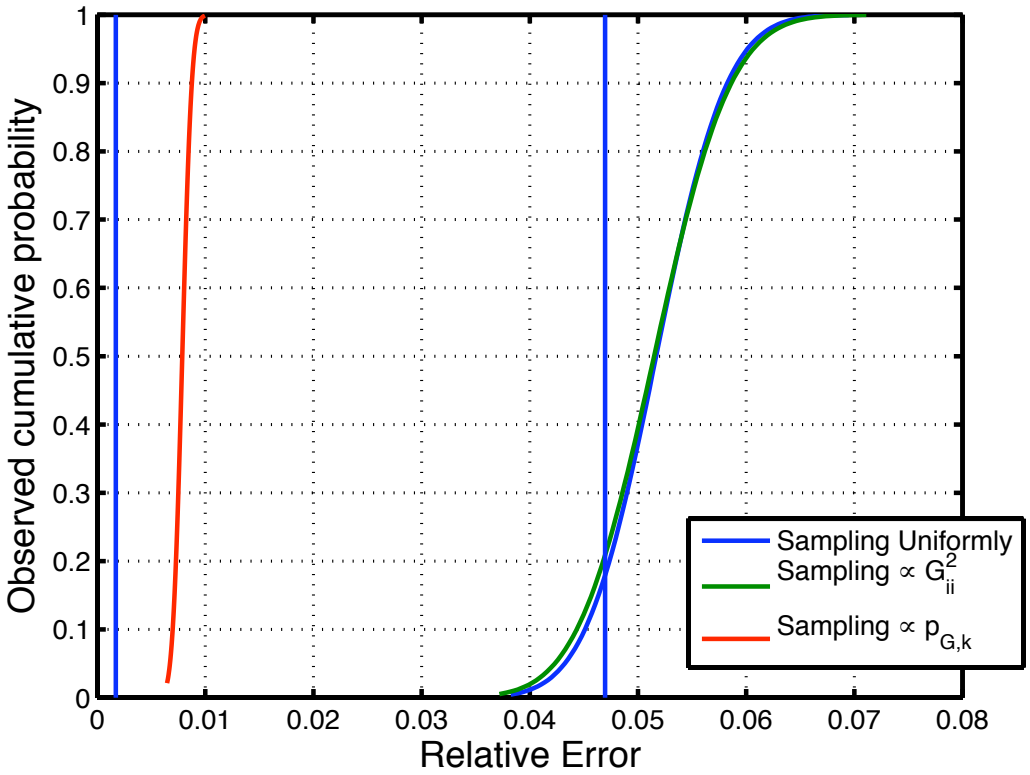


- We sample 500 points uniformly on a circle, and use the Diffusion maps algorithm to define an appropriate kernel for embedding
- We measured the resultant approximation error, averaged over 100 datasets and over 100 trials per set

HARVARD ENGINEERING AND APPLIED SCIENCES

Approximate Diffusion Maps

Deterministic multi-index selection



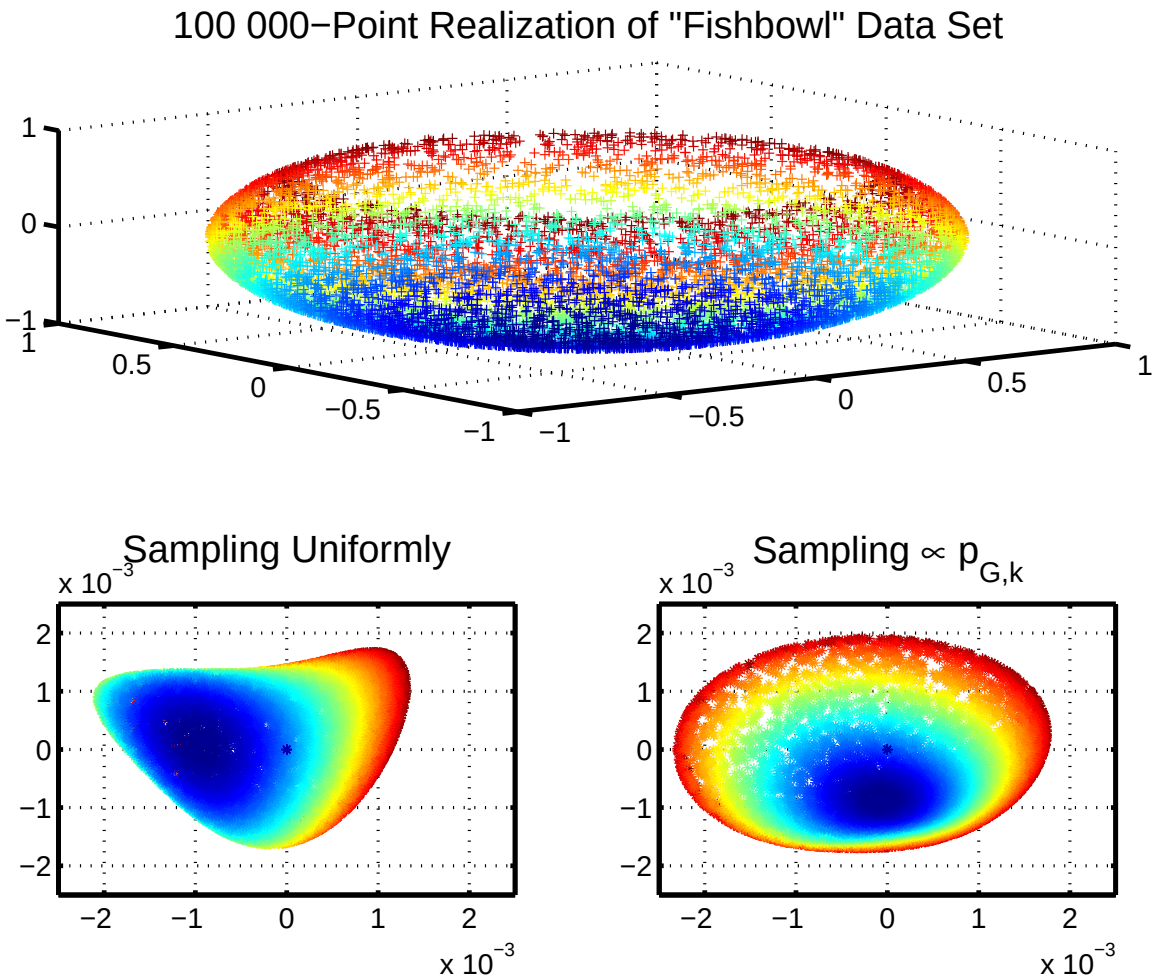
- At left, we plot the distribution of approximation error for fixed rank $k = 8$.
- The worst-case error bound of our deterministic algorithm can be clearly seen

HARVARD ENGINEERING AND APPLIED SCIENCES

Laplacian Eigenmaps Example

Embedding a massive dataset

We used the Laplacian eigenmaps algorithm embed the *fishbowl* dataset



Summary

Approximation of quadratic forms in learning theory

- Two alternative strategies for the approximate spectral decomposition of large kernels were presented, both coupled with the Nyström method:
  - Randomized multi-index selection (sampling)
  - Deterministic multi-index selection (sorting)
- Simulation studies demonstrated applicability to machine learning tasks, with measurable improvements in performance
  - Low-rank kernel approximation
  - Methods for nonlinear embeddings
- Work supported by NSF-DMS and DARPA. Related activities in our Statistics & Informations Sciences Laboratory include:
  - Exploiting variability in the space of speech sounds (DARPA)
  - Color image acquisition, processing, and display (Sony Corp.)
  - Statistical inference and algorithms for graphs and networks (NSF-DMS/MSBS, NSF-CISE/DHS)