



(On the Nyström Method for)
Approximating a Gram Matrix for Improved
Kernel-Based Learning

Michael W. Mahoney
(joint work with P. Drineas;
thanks to R. Kannan)

Yale University
Dept. of Mathematics
<http://cs-www.cs.yale.edu/homes/mmahoney>

COLT June 2005



Motivation (1 of 3)

Methods to extract **linear** structure from the data:

- Support Vector Machines (SVMs).
- Gaussian Processes (GPs).
- Singular Value Decomposition (SVD) and the related PCA.

Kernel-based learning methods to extract **non-linear** structure:

- Choose **features** to define a (dot product) space F .
- **Map** the data, X , to F by $\phi: X \rightarrow F$.
- Do classification, regression, and clustering in F with linear methods.



Motivation (2 of 3)

- Use **dot products** for information about mutual positions.
- Define the **kernel** or **Gram matrix**: $G_{ij} = k_{ij} = (\phi(X^{(i)}), \phi(X^{(j)}))$.
- Algorithms that are expressed in terms of dot products can be given the **Gram matrix** G instead of the **data covariance matrix** $X^T X$.
- Note: Isomap, LLE, graph Laplacian eigenmaps, Hessian eigenmaps, SDE (**dimensionality reduction methods** for **nonlinear manifolds**) are kernel-PCA for particular Gram matrices.
- Note: for **Mercer kernels**, G is **SPSD**.



Motivation (3 of 3)

If the *Gram matrix* G -- $G_{ij} = k_{ij} = (\phi(X^{(i)}), \phi(X^{(j)}))$ -- is dense but (nearly) low-rank, then *calculations of interest* still need $O(n^2)$ space and $O(n^3)$ time:

- *matrix inversion* in GP prediction,
- *quadratic programming* problems in SVMs,
- computation of *eigendecomposition* of G .

Relevant *recent work* using *low-rank methods*:

- Achlioptas, McSherry, and Schölkopf, 2002, ``*randomized kernels*''.
- Williams and Seeger, 2001, the ``*Nystrom method*''.



Overview

Our **main algorithm**:

- Randomized algorithm to approximate a Gram matrix.
- Low-rank approximation in terms of columns (and rows) of $G=X^T X$.

Our **main quality-of-approximation theorem**:

- Provably good approximation if nonuniform probabilities are used.

Discussion of the **Nystrom method**:

- Nystrom method for integral equations and matrix problems.
- Relationship to randomized SVD and CUR algorithms.



Review of Linear Algebra

F-norm: $\|A\|_F^2 = \sum_{ij} A_{ij}^2$

2-norm: $\|A\|_2 = \sup_{x \in R^n, x \neq 0} \frac{\|Ax\|}{\|x\|}$

SVD: $A = U\Sigma V^T$

$$A_k = U_k \Sigma_k V_k^T$$

SPSD: $x^T A x \geq 0 \quad \forall x \neq 0$

MPGI: $A^+ = V\Sigma^{-1}U^T$

$$\max_{t:1 \leq t \leq n} |\sigma_t(A + E) - \sigma_t(A)| \leq \|E\|_2$$

$$\sum_{k=1}^n (\sigma_k(A + E) - \sigma_k(A))^2 \leq \|E\|_F^2$$



Our Main Algorithm

Input: $n \times n$ SPSD matrix G , probabilities $\{p_i, 1=1,\dots,n\}$, $c \leq n$, and $k \leq c$.

Output: $n \times c$ matrix C , and $c \times c$ matrix W_k^+ (s.t. $CW_k^+C^T \approx G$).

Algorithm:

- Pick c columns of G in i.i.d. trials, with replacement and with respect to the probabilities $\{p_i\}$; let \mathcal{I} be the set of indices of the sampled columns.
- Scale each sampled column (with index $i \in \mathcal{I}$) by dividing its by $\sqrt{cp_i}$.
- Let C be the $n \times c$ matrix containing the rescaled sampled columns.
- Let W be the $c \times c$ matrix of G with entries $G_{ij}/c\sqrt{p_i p_j}$, $i, j \in \mathcal{I}$.
- Compute W_k^+ .



Our Main Theorem

Let $\varepsilon > 0$ and $\eta = 1 + \sqrt{8 \log(1/\delta)}$.

Construct an approximation $CW_k + C^T$ with our Main Algorithm by sampling c columns of G with probabilities $p_i = G_{ii}^2 / \sum_i G_{ii}^2$.

If $c \geq 64k\eta^2/\varepsilon^4$, then w.h.p.:

$$\|G - CW_k + C^T\|_F \leq \|G - G_k\|_F + \varepsilon \sum_i G_{ii}^2.$$

If $c \geq 4\eta^2/\varepsilon^2$, then w.h.p.:

$$\|G - CW_k + C^T\|_2 \leq \|G - G_k\|_2 + \varepsilon \sum_i G_{ii}^2.$$



Notes About Our Main Result (1 of 2)

Note: the **structural simplicity** of our main result:

- C consists of a small number of **representative data points**.
- W consists of the **induced subgraph** defined by those points.

$$\begin{pmatrix} G \end{pmatrix} \approx \begin{pmatrix} \tilde{G} \end{pmatrix} = \begin{pmatrix} C \end{pmatrix} \begin{pmatrix} W \end{pmatrix}^+ \begin{pmatrix} C^T \end{pmatrix}$$

Computational resource requirements:

- Assume the **data** X (or Gram matrix G) are **stored externally**.
- Algorithm performs **two passes** over the data.
- Algorithm uses $O(n)$ **additional scratch space** and **additional computation time**.



Notes About Our Main Result (2 of 2)

How to interpret the **sampling probabilities**?

If the **sampling probabilities** were:

$$p_i = ||G^{(i)}||^2 / ||G||_F^2$$

- they would provide a **bias** towards data points that are more “important” - **longer** and/or **more representative**.
- the additional error would be $\varepsilon ||G||_F$ and not $\varepsilon \sum_i G_{ii}^2 = \varepsilon ||X||_F^2$ (where $G=X^T X$).

Our **sampling probabilities** ignore correlations:

$$p_i = G_{ii}^2 / \sum_i G_{ii}^2 = ||X^{(i)}||^2 / ||X||_F^2$$



Proof of Our Main Theorem (1 of 4)

Let $G = X^T X$ and let:

$$\begin{aligned} C &= GSD = X^T XSD \\ C_X &= XSD \\ &= \tilde{U}\tilde{\Sigma}\tilde{V}^T \\ W &= (SD)^T GSD = C_X^T C_X = \tilde{V}\tilde{\Sigma}^2\tilde{V}^T. \end{aligned}$$

Then:

$$\begin{aligned} CW_k^+ C^T &= GSD(W_k)^+ (GSD)^T \\ &= X^T \hat{U}_k \hat{U}_k^T X. \end{aligned}$$

The error matrix is:

$$\begin{aligned} G - \tilde{G}_k &= X^T X - X^T \hat{U}_k \hat{U}_k^T X \\ &= \left(X - \hat{U}_k \hat{U}_k^T X \right)^T \left(X - \hat{U}_k \hat{U}_k^T X \right). \end{aligned}$$



Proof of Our Main Theorem (2 of 4)

First, bound the spectral norm:

$$\begin{aligned}\|G - \tilde{G}_k\|_2 &= \left\| \left(X - \hat{U}_k \hat{U}_k^T X \right)^T \left(X - \hat{U}_k \hat{U}_k^T X \right) \right\|_2 \\ &= \|X - \hat{U}_k \hat{U}_k^T X\|_2^2 \\ &\leq \|X - X_k\|_2^2 + 2\|XX^T - C_X C_X^T\|_2 \\ &\leq \|G - G_k\|_2 + 2\|E_{XX^T}\|_2.\end{aligned}$$

Note: If $k \geq r = \text{rank}(W)$, then:

$$\|G - \tilde{G}_r\|_2 \leq \|E_{XX^T}\|_2$$



Proof of Our Main Theorem (3 of 4)

Next, bound the Frobenius norm:

Let $E = XX^TXX^T - C_X C_X^T C_X C_X^T$. Then:

$$\begin{aligned}\|G - \tilde{G}_k\|_F^2 &= \|X^T X - X^T \hat{U}_k \hat{U}_k^T X\|_F^2 \\ &= \|X^T X\|_F^2 - 2 \|XX^T \hat{U}_k\|_F^2 + \|\hat{U}_k^T X X^T \hat{U}_k\|_F^2 \\ &\leq \|X^T X\|_F^2 - 2 \left(\sum_{t=1}^k \sigma_t^4(C_X) - \sqrt{k} \|E\|_F \right) + \sum_{t=1}^k \sigma_t^4(C_X) + \sqrt{k} \|E\|_F \\ &= \|X^T X\|_F^2 - \sum_{t=1}^k \sigma_t^4(C_X) + 3\sqrt{k} \|E\|_F \\ &\leq \|X^T X\|_F^2 - \sum_{t=1}^k \sigma_t^2(X^T X) + 4\sqrt{k} \|E\|_F \\ &\leq \|G - G_k\|_F^2 + 4\sqrt{k} \|E\|_F,\end{aligned}$$



Proof of Our Main Theorem (4 of 4)

Goal: Approximate the product of two (or more) matrices. (DK,DKM,DM)

Input: $m \times n$ matrix A , number $c \leq n$, and probabilities $\{p_i, i=1, \dots, n\}$

Output: $m \times c$ matrix C (s.t. $CC^T \approx AA^T$)

Algorithm:

- Randomly sample c columns from A according to $\{p_i\}$
- Rescale each column by $1/\sqrt{cp_{i_t}}$ to form C

Theorem: Let $\eta = 1 + \sqrt{8 \log(1/\delta)}$. If $p_i = |A^{(i)}|^2 / \|A\|_F^2$ and $c \geq 4 \eta^2 / \varepsilon^2$:

- $\|AA^T - CC^T\| \leq \varepsilon \|A\|_F^2$
- $\|AA^T AA^T - CC^T CC^T\| \leq \varepsilon \|A\|_F^4$



The Nystrom Method (1 of 3)

Consider the eigenfunction problem:

$$\int_D K(t, s)\Phi(s)ds = \lambda\Phi(t) \quad t \in D. \quad (1)$$

Discretize with a quadrature rule ($\{w_j\}$ are the weights and $\{s_j\}$ are the quadrature points):

$$\sum_{j=1}^n w_j k(x, s_j)\tilde{\phi}(s_j) = \tilde{\lambda}\tilde{\phi}(x). \quad (2)$$

Choose a set $\{x_i\}$ of Nyström points (often the same as $\{s_i\}$):

$$\sum_{j=1}^n w_j k(x_i, s_j)\tilde{\phi}(s_j) = \tilde{\lambda}\tilde{\phi}(x_i). \quad (3)$$

Extend the eigenvectors $\tilde{\phi}_m$ on the Nyström points to $\bar{\phi}_m(x)$ on D :

$$\bar{\phi}_m(x) = \frac{1}{\tilde{\lambda}_m} \sum_{j=1}^n w_j k(x, s_j)\tilde{\phi}_m(s_j). \quad (4)$$

$\bar{\phi}_m(x)$ is the *Nyström extension* of $\tilde{\phi}_m$ and it approximates Φ_m .



The Nystrom Method (2 of 3)

Partition an $m \times n$ matrix A as:

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}. \quad (1)$$

Let the SVD of A_{11} be $A_{11} = \tilde{U}\tilde{\Sigma}\tilde{V}^T$. The Nyström extension of \tilde{U} and \tilde{V} gives:

$$\bar{U} = \begin{bmatrix} \tilde{U} \\ A_{21}\tilde{V}\tilde{\Sigma}^{-1} \end{bmatrix} \text{ and } \bar{V} = \begin{bmatrix} \tilde{V} \\ A_{12}^T\tilde{U}\tilde{\Sigma}^{-1} \end{bmatrix}. \quad (2)$$

Then, to approximate A (via matrix completion), set $\tilde{A} = \bar{U}\tilde{\Sigma}\bar{V}^T$:

$$\tilde{A} = \begin{bmatrix} \tilde{U} \\ A_{21}\tilde{V}\tilde{\Sigma}^{-1} \end{bmatrix} \tilde{\Sigma} \begin{bmatrix} \tilde{V}^T & \tilde{\Sigma}^{-1}\tilde{U}^T A_{12} \end{bmatrix} \quad (3)$$

$$= \begin{bmatrix} A_{11} \\ A_{21} \end{bmatrix} A_{11}^+ \begin{bmatrix} A_{11} & A_{12} \end{bmatrix}. \quad (4)$$



The Nystrom Method (3 of 3)

Randomized SVD Algorithms (of Frieze, Kannan, and Vempala, and Drineas, Kannan, and Mahoney)

- Randomly sample columns (xor rows).
- Compute/approximate low-dimensional singular vectors.
- Nystrom-extend to approximate H_k , the high-dim. sing. vect.
- Bound $\|A - H_k H_k^T A\|_{2,F} \leq \|A - A_k\|_{2,F} + \varepsilon \|A\|_F$.

Randomized CUR Algorithms (of Drineas, Kannan, and Mahoney)

- Randomly sample columns and rows
- Bound $\|A - CUR\|_{2,F} \leq \|A - A_k\|_{2,F} + \varepsilon \|A\|_F$.
- Does not need or use the SPSD property



Conclusion

Main Result: We randomly sample columns (biased towards longer columns) of a Gram matrix G to get an approximation s.t.:

$$\|G - CW_k^+ C^T\|_{2,F} \leq \|G - G_k\|_{2,F} + \varepsilon \|X\|_F^2.$$

Open problem: Sample with respect to probabilities that include correlations, preserve the SPSD property, and obtain bounds with an additional error of $\varepsilon \|G\|_F$. (Probably a corollary of general CUR.)