Randomized Sketching for Generalized Eigenvalue Problems

Surya Appana, Ryan Schneider (advisor), and James Demmel (advisor)

University of California, Berkeley

Fall 2025

Abstract

This work investigates how random matrix sketching can accelerate algorithms for generalized eigenvalue problems (GEPs), with a particular emphasis on applications in machine learning such as Fisher Discriminant Analysis (FDA). Randomized subspace embeddings offer an efficient alternative by compressing the matrices into a lower-dimensional subspace that approximately preserves the geometry of the original problem. We develop and analyze sketched Rayleigh-Ritz methods for generalized eigenvalue problems. Our study focuses on the following key questions: how does the embedding dimension affect solution quality, and which sketching operators provide the best trade-offs between speed, accuracy, and stability? Through experiments on synthetic and real datasets, we evaluate multiple sketching strategies and quantify their performance using both speed and evaluation metrics such as classification accuracy and computational cost. The results show that randomized sketches can dramatically reduce runtime while retaining high accuracy. This work highlights the potential of randomized numerical linear algebra to make FDA and related GEP-based methods scalable to modern high-dimensional machine learning applications.

Keywords: Generalized eigenvalue problems (GEPs), Krylov Subspace Methods, Sketch-and-solve paradigm, Rayleigh-Ritz, Subspace embedding, Fisher Discriminant Analysis.

1 Background and Motivation

1.1 Generalized eigenvalue problems

In the standard nonsymmetric eigenvalue problem (EP), given an n-by-n matrix A, we aim to find (λ, x) such that

$$Ax = \lambda x.$$

This problem is already ubiquitous in numerical linear algebra and in a variety of applications, including machine learning. In the generalized eigenvalue problem (GEP), we are given a matrix pair (A, B) (with both matrices of the same size) and aim to find (λ, x) such that

$$Ax = \lambda Bx.$$

In cases where B is invertible, this can be turned into an eigenvalue problem, although B may be poorly conditioned and inverting may be computationally expensive. We call $A - \lambda B$ as a function of λ a *matrix pencil*. In cases where $\det(A - \lambda B) = 0$ for all λ, the matrix pair is called *singular*. Else, it is *regular*. The following table shows analogies between the theoretical aspects of EPs and GEPs. Assume that (A, B) of order n is regular unless otherwise stated.

Concept	$Ax = \lambda x$	$Ax = \lambda Bx$

Eigenvalues	A has n complex and finite eigenvalues which solve $\det(A-\lambda I)=0$	(A,B) has n (generalized) eigenvalues. We have infinite eigenvalues when B is singular.
Rayleigh Quotient	$R(A,x)=\frac{x^TAx}{x^Tx}$	$R(A,B,x)=\frac{x^TAx}{x^TBx}$
Similarity/Equivalence	Matrices A, B are similar if there exists nonsingular S such that $B=S^{-1}AS$. Similar matrices have the same eigenvalues.	If X and Y are nonsingular, we call the pairs (A,B) and (YAX,YBX) equivalent.
Schur Decomposition	There exists unitary U such that $U^HAU=T$ is upper triangular. The eigenvalues of A appear on the diagonal of T.	There exist untary U,V such that matrices in the equivalent pair $(S,T)=(V^HAU,V^HBU)$ are upper-triangular.
Jordan/Weierestrass Canonical form	For A, there exists nonsingular X such that $X^{-1}AX$ is block diagonal with Jordan blocks $J_{k_i}(\lambda_i)\in\mathbb{C}^{k_i\times k_i}$.	(A,B) is equivalent to a form $[(J,I),(I,N)]$, where J and N are in Jordan canonical form and N is nilpotent.
Hermitian matrices	A is Hermitian if $A=A^H$. A has real eigenvalues and an orthogonal diagonalization.	If (A,B) is Hermitian (A and B are Hermitian), it does not necessarily have real (generalized) eigenvalues. The Hermitian pair is a *definite pair* if the quantity $\gamma(A,B):=\min_{x\in\mathbb{C}^n,\|x\|_2=1}\|x^H(A+iB)x\|$ is strictly positive. A definite pair (A,B) is regular, and there exists nonsingular X such that X^HAX and X^HBX are diagonal.

1.2 Krylov subspace methods

The nonsymmetric **eigenvalue problem (EP)** $Av=\lambda v$ is ubiquitous in numerical linear algebra and appears in a variety of applications such as machine learning. For very large systems, especially when the matrix is only accessible through matrix-vector multiplications $x\mapsto Ax$, finding all eigenpairs is computationally intensive due to the $O(n^2)$ cost of storing A and the $O(n^3)$ cost of calculations, motivating iterative methods over direct methods such as the power method or QR-iteration. Central among these are **Krylov Subspace Methods (KSMs)**, which seek to find approximate eigenvectors located in a Krylov subspace:

$$K_p(A,b):=\text{span}\{b,Ab,A^2b,\dots,A^{p-1}b\}$$

This subspace often contains an excellent approximation to an eigenvector, even for a modest depth p, since it effectively takes the first few vectors from the power method. Suppose b is random and $K:=[b,Ab,\dots,A^{n-1}b]$ is invertible. Since $K^{-1}AK=:C$ is upper Hessenberg, we can approximate the eigenvalues of A with the eigenvalues of the simpler matrix C.

However, K is likely to be highly ill-conditioned since, by construction, A^kb for $k=0,1,2,\dots$ approaches the eigenvector of A associated with the largest eigenvalue. Let $K=QR$ be the QR decomposition of K. Then $Q^TAQ=RCR^{-1}=:H$, which is also upper Hessenberg. Thus, if we can find an orthogonal matrix Q whose columns span the columnspace of K, we can determine approximate eigenvalues of A. In practice,

we only care about computing the first k columns of Q and determining k eigenvalues of A (where $k \ll n$). Let $Q = [Q_k, Q_u]$ where Q_k is the first k columns of Q and Q_u contains the rest:

$$H = Q^T AQ = \begin{bmatrix} Q_k^T AQ_k & Q_k^T AQ_u \\ Q_u^T AQ_k & Q_u^T AQ_u \end{bmatrix}$$
$$\equiv \begin{bmatrix} H_k & H^{uk} \\ H_{ku} & H_u \end{bmatrix}$$

Note that H_k is upper Hessenberg and H_{ku} contains as single, possibly nonzero entry at its upper right corner. If this is zero or close to zero, then the eigenvalues of H_k are exactly or approximately equal to eigenvalues of A. To compute the first k columns of Q, the classicial approach is the *Arnoldi process* as shown in Algorithm 1.

Algorithm 1 The Arnoldi algorithm for (partial) reduction to Hessenberg form

```
q_1 ← b/‖b‖_2  // k is the number of columns of Q and H to compute
for j = 1 to k do
    z ← A q_j
    for i = 1 to j do
        h_{i,j} ← q_i^T z
        z ← z − h_{i,j} q_i
    h_{j+1,j} ← ‖z‖_2
    if h_{j+1,j} = 0 then
        quit
    q_{j+1} ← z/h_{j+1,j}
```

This process requires only $O(nk^2)$ arithmetic operations for k iterations. Once we have obtained an orthogonal basis for $K_k(A, b)$ using the Arnoldi Process, we can then approximate the eigenvalues of A with the eigenvalues of H_k and multiply the associated eigenvectors by Q to obtain approximate eigenvectors for A. This approach is known as the *Rayleigh-Ritz* method, with approximate eigenvalues and eigenvectors known as the *Ritz values* and *Ritz vectors*.

1.3 Randomized matrix sketching

Randomized embeddings are motivated by the need to develop fast, reliable, and scalable algorithms in numerical linear algebra by reducing dimensionality. One important desirable quality of a random embedding on an N-point random subset of $\mathbb{R}^n$ is to approximately preserve distances between points (within a small factor). The following well-known lemma shows that when $k = O(\log(N)/\epsilon^2)$, such an embedding exists.

Lemma (Johnson-Lindenstrauss)**.** *Let N be any integer and $\epsilon \in (0, 1)$. Suppose k is a positive integer such that $k \geq 4(\epsilon^2/2 - \epsilon^3/3)^{-1} \log(n)$. Then for any set X of N points in $\mathbb{R}^n$ there exists a map $f : \mathbb{R}^n \to \mathbb{R}^k$ such that for all $x, y \in X$, we have*

$$(1 - \epsilon)\|x - y\|^2 \leq \|f(x) - f(y)\|^2 \leq (1 + \epsilon)\|x - y\|^2,$$

where this norm is the ℓ_2 norm.

In practice, we apply the "sketch-and-solve" paradigm to reduce the dimension of a problem by multiplying a large matrix by a randomized *subspace embedding* matrix with a much smaller output dimension and solving the reduced problem. A subspace embedding is defined as follows.

Defn (Subspace Embedding)**.** *Suppose $A \in \mathbb{C}^{n \times d}$ whose columns span the subspace $V \subseteq \mathbb{C}^n$. Then $S \in \mathbb{C}^{s \times n}$ is a subspace embedding for V with distortion ϵ if for all $x \in \mathbb{C}^d$, we have*

$$(1 - \epsilon)\|Ax\|_2 \leq \|SAx\|_2 \leq (1 + \epsilon)\|Ax\|_2.$$

In other words, S approximately preserves the ℓ_2 norm of vectors in the range of A.

Since the subspace V is typically unknown, we must draw the subspace embedding S randomly to satisfy the above property with high probability. Suppose A above has full column rank (and therefore V has dimension d). To embed V with distortion ϵ, we optimally have $s \approx d/\epsilon^2$. There are a few popular ways of constructing a randomized subspace embedding that work for the eigenvalue problem.

1.3.1 SRFTs (Subsampled random Fourier transforms)

These subspace embeddings take the form

$$S = \sqrt{\frac{n}{s}} DFE \in \mathbb{C}^{s \times n},$$

where $D \in \mathbb{C}^{s \times n}$ diagonally projects onto s independently chosen random coordinates, $F \in \mathbb{C}^{n \times n}$ is the unitary discrete Fourier transform (DFT), and $E \in \mathbb{C}^{n \times n}$ is a diagonal matrix whose entries are independent Steinhaus random variables (uniform on the complex unit circle). Using the subsampled FFT algorithm, we can apply S to an $n \times d$ matrix in $O(nd \log(d))$ operations.

1.3.2 Sparse Maps

A sparse random embedding takes the form

$$S = \frac{1}{\sqrt{s}}[s_1, \ldots, s_n] \in \mathbb{C}^{s \times n},$$

where each column s_i has a fixed number of nonzero entries ζ drawn from the Steinhaus distribution whose coordinates are uniformly random. This embedding is cheaper to apply on sparse data.

1.4 Randomized sketching for eigenvalue problems

Recognizing the computational limitations inherent in traditional KSMs, Nakatsukasa and Tropp [2] introduced the integration of random matrix sketching into subspace projection methods such as Rayleigh-Ritz (RR). This approach leverages the "sketch-and-solve paradigm" to reduce the dimensionality of the underlying computations. Suppose $B \in \mathbb{C}^{n \times d}$ is such that its column span contains approximate eigenvectors for A (e.g. computed through a truncated Arnoldi process). The classic RR method solves the least-squares minimization problem:

$$\min_{M \in \mathbb{C}^{d \times d}} \|AB - BM\|_F.$$

The reduced EP is to find eigenvalues and eigenvectors for $\hat{M} = B^\dagger AB$. By drawing a random sketching matrix $S \in \mathbb{C}^{s \times n}$ (the authors suggest $s = 4d$), the RR problem is transformed into the **sketched Rayleigh-Ritz (sRR)** problem:

$$\min_{M \in \mathbb{C}^{d \times d}} \|S(AB - BM)\|_F.$$

The resulting solution $\hat{M} = (SB)^\dagger (SAB)$ is computed significantly faster than the optimal solution M^*. This randomized approach gives immense flexibility in selecting the basis B. Specifically, highly conditioned bases, where $\kappa_2(B) \lesssim u^{-1}$ (where u is the unit roundoff), can be tolerated, allowing the use of computationally cheaper, non-orthogonal basis generation techniques, such as k-truncated Arnoldi. For an unstructured basis, the complexity of solving the sRR problem is reduced from $O(nd^2)$ (classic RR) to $O(d^3 + nd \log d)$ (sRR, using a fast sketch and k-truncated Arnoldi), yielding substantial practical speedups, often exceeding $10\times$ faster than optimized standard routines in tested examples.

A variational formulation of RR is introduced again for the GEP. Suppose we want to solve $Ax = \lambda Bx$ for $A, B \in \mathbb{C}^{n \times n}$ and nonzero x. Further suppose that $V \in \mathbb{C}^{n \times d}$ contains approximate eigenvectors. Our goal is to solve:

$$\min_{M \in \mathbb{C}^{d \times d}} \|AV - BVM\|_F. \tag{1}$$

A solution is $\hat{M} = (BV)^\dagger(AV)$, for which we can pose the standard EP. For the sketched problem, given a subspace embedding S for Range($[AV, BV]$), we have the following:

$$\min_{M \in \mathbb{C}^{d \times d}} \|S(AV - BVM)\|_F. \tag{2}$$

This has a solution $\hat{M} = (SBV)^\dagger(SAV)$, yielding a reduced EP. Once again, excluding the generation of basis V, we again have cost $O(d^3 + nd\log(d))$ compared to $O(nd^2)$ in classic RR.

1.5 The Generalized Eigenvalue Problem (GEP) and applications

The GEP is crucial in various applications in machine learning such as Fisher Discriminant Analysis (FDA) and Kernel Supervised Principal Component Analysis (Kernel SPCA) [1]. Fisher Discriminant Analysis (FDA), also known in its classical form as Linear Discriminant Analysis (LDA), is a supervised classification technique used when data belong to two or more distinct classes. Given observations represented by feature vectors in a high-dimensional space, we seek a low-dimensional projection onto which the classes remain maximally separated. FDA is particularly useful when classes can be approximated as linearly separable and when dimensionality reduction is desirable for visualization or classification.

The central idea of FDA is to identify a projection direction w such that the projected class means are far apart while the variance within each class remains small. We therefore aim to maximize the Fisher discriminant ratio in w:

$$J(w) = \frac{w^T S_B w}{w^T S_W w},$$

where S_B is the between-class scatter matrix and S_W is the pooled within-class scatter matrix. In practice, we begin by computing the class means μ_k for each class C_k and the overall mean μ. The *pooled within-class scatter matrix* is given by

$$S_W = \sum_{k=1}^{K} \sum_{x \in C_k} (x - \mu_k)(x - \mu_k)^T,$$

which measures the degree of spread of each class around its own mean. The *between-class scatter matrix* is defined as

$$S_B = \sum_{k=1}^{K} n_k(\mu_k - \mu)(\mu_k - \mu)^T,$$

where n_k is the number of samples in the k-th class. This captures how distinct the class means are from the overall mean. Maximizing Fisher's discriminant reduces to solving the generalized eigenvalue problem

$$S_B w = \lambda S_W w.$$

For the special case of two classes, the optimal projection has the closed-form solution

$$w = S_W^{-1}(\mu_1 - \mu_2),$$

while for more than two classes, the discriminant subspace is spanned by the eigenvectors associated with the largest eigenvalues of $S_W^{-1} S_B$. These directions correspond to the projections that maximize the Fisher ratio and therefore provide the best linear separation among the class groups.

In high-dimensional settings where the feature dimension n is large, directly forming or inverting the within-class scatter matrix S_W is often computationally infeasible or numerically unstable. When n exceeds the number of available samples, S_W is necessarily singular, since its rank cannot exceed the total number of centered data points. Even when S_W is not exactly singular, it is frequently ill-conditioned, which makes computing $S_W^{-1} S_B$ numerically problematic. As a result, both the classical two-class solution $w = S_W^{-1}(\mu_1 - \mu_2)$ and the multi-class generalized eigenvalue problem $S_B w = \lambda S_W w$ become impractical under high-dimensional regimes.

The computational cost of operating directly on $n \times n$ scatter matrices is also prohibitive. Constructing either S_W or S_B requires processing dense matrices with $O(n^2)$ storage, and solving the generalized eigenvalue problem requires $O(n^3)$ time. These costs become intractable when n is in the tens or hundreds of thousands,

as commonly encountered in applications such as genomics, image analysis, or text processing. Moreover, in many problems the between-class scatter matrix S_B has rank at most $K-1$, where K is the number of classes. Thus, even though the discriminant subspace is intrinsically low-dimensional, the classical computation is forced to operate in the full ambient dimension n, which is highly inefficient.

To address these issues, we can take matrix pair $(S_B,\ S_W)$ and formulate a sketched Rayleigh-Ritz methods for the generalized eigenvalue problem as in (2). Applying a subspace embedding $S \in \mathbb{R}^{s\times n}$ with $s \ll n$ to between-class or within-class scatter matrices corresponds to compressing the geometry of class variability into a lower-dimensional surrogate space while attempting to preserve the essential structure. We project onto a randomly oriented s-dimensional subspace, and as a result, computations such as evaluating scatter matrices or solving generalized eigenvalue problems can be carried out in the reduced space without significantly altering the essential geometric information of the original high-dimensional data.

1.6 Objectives

We have demonstrated how generalized eigenvalue problems apply in machine learning and how randomized sketching can accelerate algorithmic solutions. We now lay out the primary research questions and methods for approaching them:

Research Questions:

1. How does the embedding dimension of the sketching operator S affect the accuracy of solutions?
2. Which choices of sketching operators perform best in practice, in terms of speed and accuracy?
3. How do formulations of sketched Rayleigh-Ritz compare for the GEP?

Methods:

1. **Vary sketch dimension s systematically** to study how embedding size affects accuracy; compare sketched FDA solutions with the full-dimensional solution using classification accuracy, subspace distance, and Fisher ratio error.
2. **Evaluate multiple sketching operators** (Gaussian, SRFT, sparse sign matrices, etc.) under identical conditions to compare speed, accuracy, and numerical stability. Apply to synthetic and real datasets to assess robustness across controlled and practical settings.
3. **Measure computational cost** of forming sketched scatter matrices and solving the sketched generalized eigenvalue problem; record runtime and memory usage as s varies and across sketch types.
4. **Plot key findings**:
   - Accuracy (or subspace distance) vs. sketch dimension s.
   - Runtime vs. sketch dimension s.
   - Accuracy vs. runtime tradeoff curves.
   - Comparing sketch types in accuracy, subspace error, and runtime.
5. **Analyze tradeoffs** between accuracy and speed to identify the minimum sketch dimension achieving near-optimal accuracy and the sketching operators that offer the best speed–accuracy performance in practice.

References

[1] Elnaz Barshan, Ali Ghodsi, Zohreh Azimifar, and Mansoor Zolghadri Jahromi. Supervised principal component analysis: Visualization, classification and regression on subspaces and submanifolds. *Pattern Recognition*, 44(7):1357–1371, 2011.

[2] Yuji Nakatsukasa and Joel A. Tropp. Fast and accurate randomized algorithms for linear systems and eigenvalue problems. *SIAM Journal on Matrix Analysis and Applications*, 45(2):1183–1214, 2024.