# Mathematical Background for the SVD–LDA Project

Numerical Methods and Algorithms

## 1 Linear algebra preliminaries

We work over the real numbers. Vectors are columns, and matrices are written with capital letters. For a vector $x \in \mathbb{R}^n$, we use the Euclidean norm

$$\|x\|_2 = \sqrt{x^\mathsf{T} x}.$$

For a matrix $A \in \mathbb{R}^{m\times n}$, the Frobenius norm is

$$\|A\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij}^2} = \sqrt{\operatorname{trace}(A^\mathsf{T} A)}.$$

A square matrix $A \in \mathbb{R}^{n\times n}$ is *symmetric* if $A = A^\mathsf{T}$. A real symmetric matrix has the following important properties:

- All eigenvalues are real.
- There exists an orthonormal basis of eigenvectors; i.e. $A = Q\Lambda Q^\mathsf{T}$ with $Q$ orthogonal and $\Lambda$ diagonal.

An orthogonal matrix $Q \in \mathbb{R}^{n\times n}$ satisfies $Q^\mathsf{T} Q = I_n$. Its columns are an orthonormal basis of $\mathbb{R}^n$.

## 2 Singular value decomposition (SVD)

### 2.1 Definition and basic properties

**Theorem 1** (Singular value decomposition)**.** *Let $A \in \mathbb{R}^{m\times n}$ have rank $r$. Then there exist orthogonal matrices $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ and a diagonal matrix*

$$\Sigma = \begin{bmatrix} \sigma_1 & & & & & \\ & \sigma_2 & & & & \\ & & \ddots & & & \\ & & & \sigma_r & & \\ & & & & 0 & \\ & & & & & \ddots \end{bmatrix} \in \mathbb{R}^{m\times n}$$

*with singular values*

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$$

*such that*

$$A = U\Sigma V^\mathsf{T}.$$

The columns of $U$ are called *left singular vectors* of $A$; the columns of $V$ are *right singular vectors.*

**Proposition 1** (Relation to eigenvalues)**.** *Let $A = U\Sigma V^\mathsf{T}$ be an SVD. Then*

$$A^\mathsf{T} A = V\Sigma^\mathsf{T}\Sigma V^\mathsf{T}, \qquad AA^\mathsf{T} = U\Sigma\Sigma^\mathsf{T} U^\mathsf{T}.$$

*In particular,*

- *The eigenvalues of $A^\mathsf{T} A$ and $AA^\mathsf{T}$ are $\sigma_1^2, \dots, \sigma_r^2$ (and possibly some zeros).*
- *The right singular vectors of $A$ are eigenvectors of $A^\mathsf{T} A$.*
- *The left singular vectors of $A$ are eigenvectors of $AA^\mathsf{T}$.*

Thus one can think of the singular values as square roots of the eigenvalues of $A^\mathsf{T} A$.

## 2.2 Truncated SVD and best rank-$k$ approximation

Let $A = U\Sigma V^\mathsf{T}$ be an SVD with singular values $\sigma_1 \geq \cdots \geq \sigma_r > 0$. We can write $A$ as

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^\mathsf{T},$$

where $u_i$ and $v_i$ are the $i$-th columns of $U$ and $V$, respectively.

For an integer $k$ with $1 \leq k \leq r$, the *truncated SVD of rank $k$* is

$$A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^\mathsf{T}.$$

**Theorem 2** (Best rank-$k$ approximation (Eckart–Young))**.** *Let $A \in \mathbb{R}^{m \times n}$ have SVD $A = U\Sigma V^\mathsf{T}$. For $1 \leq k \leq r$, the truncated SVD $A_k$ solves*

$$\|A - A_k\|_F = \min_{\operatorname{rank}(B) \leq k} \|A - B\|_F.$$

*Moreover,*

$$\|A - A_k\|_F^2 = \sum_{i=k+1}^{r} \sigma_i^2.$$

Thus among *all* matrices of rank at most $k$, $A_k$ is the closest to $A$ in Frobenius norm.

## 2.3 Energy and effective rank

Define the "energy" (squared Frobenius norm) of $A$ by

$$\|A\|_F^2 = \sum_{i=1}^{r} \sigma_i^2.$$

The energy captured by the first $k$ singular values is

$$E(k) = \frac{\sum_{i=1}^{k} \sigma_i^2}{\sum_{j=1}^{r} \sigma_j^2}.$$

By the theorem above, $\|A - A_k\|_F^2 = \sum_{i=k+1}^{r} \sigma_i^2$, so $E(k)$ measures how much of the total variance (or information) is preserved by $A_k$.

**Definition 1** (Effective rank)**.** Fix a threshold $\alpha \in (0, 1)$, for example $\alpha = 0.9$ or $\alpha = 0.95$. The effective rank of $A$ at level $\alpha$ is

$$r_\alpha(A) = \min\{k : E(k) \geq \alpha\}.$$

If $r_{0.9}(A)$ is small, then most of the information in $A$ can be captured in a low-dimensional subspace.

# 3 The power method for eigenvalues

Suppose $A \in \mathbb{R}^{n\times n}$ is real symmetric, with eigenvalues

$$\lambda_1, \lambda_2, \dots, \lambda_n,$$

ordered so that $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$. Let $q_1, \dots, q_n$ be an orthonormal basis of eigenvectors, so $Aq_i = \lambda_i q_i$.

## 3.1 Algorithm

The *power method* approximates the dominant eigenvalue $\lambda_1$ and its eigenvector.

Given a starting vector $x^{(0)} \neq 0$ and integers $k = 0, 1, 2, \dots$, define

$$y^{(k+1)} = Ax^{(k)}, \qquad x^{(k+1)} = \frac{y^{(k+1)}}{\|y^{(k+1)}\|_2}.$$

At iteration $k$ we can form the Rayleigh quotient

$$\lambda^{(k)} = (x^{(k)})^{\mathsf{T}} A x^{(k)}.$$

We stop when $|\lambda^{(k)} - \lambda^{(k-1)}|/|\lambda^{(k)}|$ is below a tolerance.

## 3.2 Convergence idea

Write $x^{(0)}$ in the eigenbasis:

$$x^{(0)} = c_1 q_1 + c_2 q_2 + \cdots + c_n q_n,$$

with $c_1 \neq 0$ (this holds for almost all starting vectors).

Then

$$A^k x^{(0)} = c_1 \lambda_1^k q_1 + c_2 \lambda_2^k q_2 + \cdots + c_n \lambda_n^k q_n = \lambda_1^k \left( c_1 q_1 + c_2 \left(\frac{\lambda_2}{\lambda_1}\right)^k q_2 + \cdots \right).$$

Since $|\lambda_2/\lambda_1| < 1$, the terms with $i \geq 2$ decay geometrically. After normalization, $x^{(k)}$ converges to $\pm q_1$, and $\lambda^{(k)}$ converges to $\lambda_1$.

The rate of convergence is governed by the ratio $|\lambda_2/\lambda_1|$; if the dominant eigenvalue is well separated, the method converges faster.

# 4 PCA and its connection to SVD

Principal Component Analysis (PCA) is a method for reducing the dimension of data while preserving as much variance as possible.

## 4.1 Data matrix and covariance

Suppose we have $N$ data vectors $x_1, \dots, x_N \in \mathbb{R}^d$. Stack them into a data matrix

$$X = \begin{bmatrix} x_1^{\mathsf{T}} \\ x_2^{\mathsf{T}} \\ \vdots \\ x_N^{\mathsf{T}} \end{bmatrix} \in \mathbb{R}^{N\times d}.$$

Let $\mu \in \mathbb{R}^d$ be the sample mean

$$\mu = \frac{1}{N} \sum_{i=1}^{N} x_i.$$

We define centered data $\tilde{x}_i = x_i - \mu$ and the centered data matrix

$$\tilde{X} = \begin{bmatrix} \tilde{x}_1^\top \\ \vdots \\ \tilde{x}_N^\top \end{bmatrix}.$$

The sample covariance matrix is

$$C = \frac{1}{N-1} \tilde{X}^\top \tilde{X} \in \mathbb{R}^{d \times d}.$$

### 4.2 PCA as an eigenvalue problem

The principal components are defined as eigenvectors of $C$. If

$$C v_j = \lambda_j v_j,$$

with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \geq 0$, then $v_1$ is the direction along which the projected data has maximum variance; $v_2$ is the direction of next-largest variance, and so on.

Projecting onto the first $k$ principal components gives a $k$-dimensional representation

$$z_i = \begin{bmatrix} v_1^\top \tilde{x}_i \\ \vdots \\ v_k^\top \tilde{x}_i \end{bmatrix} \in \mathbb{R}^k.$$

### 4.3 Connection to SVD

Consider the SVD of the centered data matrix:

$$\tilde{X} = U \Sigma V^\top, \qquad \tilde{X} \in \mathbb{R}^{N \times d}.$$

Then

$$\tilde{X}^\top \tilde{X} = V \Sigma^\top \Sigma V^\top.$$

Thus

- The columns of $V$ are eigenvectors of $C$.
- The eigenvalues of $C$ are proportional to the squared singular values:

$$C = \frac{1}{N-1} \tilde{X}^\top \tilde{X} = V \left( \frac{\Sigma^\top \Sigma}{N-1} \right) V^\top.$$

In particular, PCA can be computed via the SVD of the centered data matrix.

## 5 Two-class Linear Discriminant Analysis (LDA)

Whereas PCA is unsupervised (it does not use labels), Linear Discriminant Analysis (LDA) is a *supervised* method that aims to find directions that separate classes.

We focus on the two-class case with labels 0 and 1.

## 5.1 Scatter matrices

Let $X_0$ be the set of feature vectors from class 0 and $X_1$ from class 1. Let

$$\mu_0 = \frac{1}{n_0} \sum_{x \in X_0} x, \qquad \mu_1 = \frac{1}{n_1} \sum_{x \in X_1} x$$

be class means and $\mu$ the overall mean. The *within-class scatter matrix* is

$$S_W = \sum_{x \in X_0} (x - \mu_0)(x - \mu_0)^\top + \sum_{x \in X_1} (x - \mu_1)(x - \mu_1)^\top.$$

The *between-class scatter matrix* can be defined as

$$S_B = (\mu_1 - \mu_0)(\mu_1 - \mu_0)^\top,$$

up to a constant scaling. For two classes, $S_B$ has rank 1.

## 5.2 Optimization formulation

LDA seeks a projection vector $w \in \mathbb{R}^d$ that maximizes the Rayleigh quotient

$$J(w) = \frac{w^\top S_B w}{w^\top S_W w}.$$

Intuitively, we want the projected class means to be far apart, while the within-class spread in the projected space is small.

**Proposition 2.** *Assume $S_W$ is invertible. Then the maximizer of $J(w)$ (up to scaling) is*

$$w \propto S_W^{-1}(\mu_1 - \mu_0).$$

*Idea of the proof.* Write

$$J(w) = \frac{\left(w^\top(\mu_1 - \mu_0)\right)^2}{w^\top S_W w}.$$

Let $v = S_W^{1/2} w$, where $S_W^{1/2}$ is a matrix such that $S_W^{1/2}(S_W^{1/2})^\top = S_W$. Then

$$J(w) = \frac{\left(v^\top S_W^{-1/2}(\mu_1 - \mu_0)\right)^2}{v^\top v}.$$

For fixed $S_W^{-1/2}(\mu_1 - \mu_0)$, this quotient is maximized when $v$ is parallel to $S_W^{-1/2}(\mu_1 - \mu_0)$, by the Cauchy–Schwarz inequality. Therefore,

$$w \propto S_W^{-1}(\mu_1 - \mu_0).$$

□

Thus in the two-class case, we do not need to solve a general eigenvalue problem; it suffices to solve a linear system $S_W w = \mu_1 - \mu_0$, possibly with a small regularization term to handle singular $S_W$.

### 5.3 Classification rule in 1D

Once we have $w$, we project a feature vector $x$ to a scalar

$$z = w^\top x.$$

Let

$$m_0 = \frac{1}{n_0} \sum_{x \in X_0} w^\top x, \qquad m_1 = \frac{1}{n_1} \sum_{x \in X_1} w^\top x$$

be the projected class means. A natural decision threshold is the midpoint

$$\tau = \frac{m_0 + m_1}{2}.$$

Then the LDA classifier is

$$\hat{y}(x) = \begin{cases} 1, & \text{if } w^\top x \geq \tau, \\ 0, & \text{otherwise.} \end{cases}$$

## 6 Numerical conditioning and the matrix $A^\top A$

### 6.1 Condition number

The (2-norm) condition number of an invertible matrix $A$ is

$$\kappa_2(A) = \|A\|_2 \|A^{-1}\|_2.$$

For a symmetric positive definite matrix, $\|A\|_2$ is its largest eigenvalue and $\|A^{-1}\|_2$ is $1/\lambda_{\min}(A)$, so

$$\kappa_2(A) = \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)}.$$

### 6.2 Why $A^\top A$ can be problematic

Let $A \in \mathbb{R}^{m \times n}$ have singular values $\sigma_1 \geq \cdots \geq \sigma_r > 0$. Then

$$\kappa_2(A) = \frac{\sigma_1}{\sigma_r}, \qquad \kappa_2(A^\top A) = \frac{\sigma_1^2}{\sigma_r^2} = \kappa_2(A)^2.$$

Thus forming $A^\top A$ *squares the condition number*, potentially magnifying numerical errors.

For example, when computing an SVD via the eigenvalue decomposition of $A^\top A$, this squaring of the condition number can reduce accuracy compared to algorithms that operate directly on $A$ (such as bidiagonalization followed by symmetric QR).

### 6.3 Regularization in LDA

In the LDA setting, $S_W$ may be singular or ill-conditioned, especially in high-dimensional problems with relatively few samples. A common remedy is to add a small multiple of the identity:

$$S_W^{(\text{reg})} = S_W + \lambda I_d,$$

with $\lambda > 0$ small (e.g. $\lambda = 10^{-6}$). We then solve

$$S_W^{(\text{reg})} w = \mu_1 - \mu_0.$$

This improves conditioning and stabilizes the computation of $w$.

## 7 Summary

The project combines several ideas:

- The SVD factorization $A = U\Sigma V^\top$ and its interpretation via singular values and singular vectors.
- Truncated SVD as the best rank-$k$ approximation in Frobenius norm, motivating image compression.
- The relationship between the singular values of $A$ and the eigenvalues of $A^\top A$.
- The power method as a basic iterative algorithm for computing dominant eigenpairs of symmetric matrices.
- PCA as an eigenvalue/SVD-based method for unsupervised dimensionality reduction.
- Two-class LDA as a supervised method that uses scatter matrices and an $S_W^{-1}(\mu_1 - \mu_0)$ direction for classification.
- Numerical conditioning issues arising from forming $A^\top A$, and the use of regularization in LDA.

These tools provide the mathematical foundation for the implementation tasks: computing SVD-based features from images, designing rank-$k$ approximations, and training an LDA classifier in a low-dimensional feature space.