EE445 Mod3-Lec1: Spectral Properties of Matrices

References:

- [CE-OptMod]: Chapter 3.3, 4, 5

Outline

1. Review Eigenvalues & Eigenvectors
2. Symmetric Matrices
3. Introduction to Singular values and SVD

Why are Spectral Properties Important in ML+OPT?

- Computational efficiency
- Analysis
- Dimensionality reduction
- Numerical stability

How will we see it used?

1. Kernel methods
2. Principle component analysis (unsupervised ML)
3. Principle component regression
4. (time permitting) spectral clustering

Reminder: Eigenvalues & Eigenvectors

Some basics:

- **Def. (Characteristic Polynomial)**:
- **Def. (Left/Right Eigenvector-value pair)**:
- **Orthogonality**:

Reminder: Eigenvalues & Eigenvectors

Why important?

- Many ML algorithms involve transforming the matrix A into simpler, or *canonical forms*, from which it is easy to compute its eigenvalues and eigenvectors.
- These transformations are called **similarity transformations**

Similarity transforms

- **Def. [Similarity Transform]**:

- **Proposition.** Similar matrices A and B has the same eigenvalues.

- Some special matrices are similar to diagonal matrices—i.e., for some matrices A, there is a similarity transform S such that $\Lambda = S^{-1}AS$ is diagonal, and Λ contains the eigenvalues of A.
- These matrices are called **diagonalizable**.

Part 2. Special Matrices [Symmetric and Positive (semi) definite]

Symmetric Matrices

Symmetric Matrix:

- Symmetric matrices are one of the most important matrices in linear algebra and ML
- **Mod2-L4**: we often use kernel matrices $K = [K(x^{(i)}, x^{(j)})]$ and these are symmetric—i.e., $K = K^\top$—since $K(x^{(i)}, x^{(j)}) = K(x^{(j)}, x^{(i)})$
- **Mod2-L2**: Gram matrices $A^\top A$ and $AA^\top$ are symmetric;
  - in fact we can study all kinds of properties of a matrix A such as the range and null spaces using these gram matrices (cf. Finite Rank Operator Lemma)

Symmetric Matrices: Examples

The graph Laplacian is a symmetric matrix

Sample covariance matrix

Hessian of a function:

Quadratic Functions

- Symmetric matrices play an important role not just in ML but also OPT
- We have seen how to formulate least squares regression as a optimization problem with a quadratic objective:

$$\|Ax - b\|_2^2 = (Ax - b)^\top (Ax - b)$$

- A quadratic function $f : \mathbb{R}^n \to \mathbb{R}$ is a second-order multivariate polynomial in x, that is a function containing a linear combination of all possible monomials of degree at most two—i.e.,

Quadratic Functions

- using properties of symmetric matrices, we can express any quadratic function as a quadratic form.

Symmetric Matrices: Eigendecomposition (Spectral Theorem)

- Every symmetric matrix A can be **diagonalized** as $A = V\Lambda V^\top$ with V formed by the orthonormal eigenvectors of A and $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n)$ a diagonal matrix of the eigenvalues of A

$$A = \underbrace{\begin{bmatrix} | & \cdots & | \\ v_1 & \cdots & v_n \\ | & \cdots & | \end{bmatrix}}_{V} \underbrace{\begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \lambda_{n-1} & 0 \\ 0 & \cdots & 0 & \lambda_n \end{bmatrix}}_{\Lambda} \underbrace{\begin{bmatrix} - & v_1^\top & - \\ \vdots & \vdots & \vdots \\ - & v_n^\top & - \end{bmatrix}}_{V^\top}$$

  or equivalently, $A = \lambda_1 v_1 v_1^\top + \cdots + \lambda_n v_n v_n^\top$ (i.e., weighted sum of dyads)
- Additionally, $VV^\top = V^\top V = I$
- This factorization property and the fact that S has n orthogonal eigenvectors are two **important properties** for a symmetric matrix.

Example Problem: Eigenvalues are Real

Problem: Consider a symmetric matrix A. Show that the eigenvalues of A are real.
Solution.

Example Problem: Orthogonality of Eigenvectors

Problem: Consider a symmetric matrix A. Show that eigenvectors corresponding to distinct eigenvalues are orthogonal.

Solution.

Matrix powers with eigendecomposition

- Recall from **Mod1** we saw many applications with matrix powers such as computing the number of paths of length k in a graph
- For symmetric matrices, computing matrix powers is easy

Positive Definite Matrices

- Another important class of matrices are positive definite matrices
- The matrix A is **positive definite** if $\langle Ax, x \rangle > 0$; sometimes we write $A \succ 0$
- And, A is **positive semidefinite (PSD)** if $\langle Ax, x \rangle \geq 0$; sometimes we write $A \succeq 0$
- Positive definite matrices need not be symmetric, but often we are interested in positive definite symmetric matrices
- Eigenvalues: let $\lambda_1(A) \geq \cdots \geq \lambda_n(A)$ be the order set of eigenvalues of $A = A^\top$

$$A \succeq 0 \iff \lambda_i(A) \geq 0, \ \forall i \in \{1, \dots, n\}$$

$$A \succ 0 \iff \lambda_i(A) > 0, \ \forall i \in \{1, \dots, n\}$$

Example Problem

Problem: Show that $A \succeq 0 \iff \lambda_i(A) \geq 0, \ \forall i \in \{1, \ldots, n\}$
Solution.

Examples of PSD Matrices from ML and OPT

- **Mod2-L4**: we often use kernel matrices $K = [K(x^{(i)}, x^{(j)})]$ and these are symmetric and in general PSD
- **Mod2-L2**: Gram matrices $A^\top A$ and $AA^\top$ are PSD

Problem: Show that Gram and Kernel matrices are PSD.
solution.

Example Problem

Problem. Show that a matrix A is PSD if and only if $A = B^\top B$ for some real matrix B.
Solution.

Part 3. SVD

Overview

- We just talked about special classes of matrices that have a nice decomposition in terms of their eigenvalues—namely, symmetric PSD matrices.
- Now, we will talk about a matrix decomposition that every matrix has—i.e., `SVD`
- And, it is fundamentally related to a key ML analysis tool: `PCA`

Matrix Decomposition

- Matrix decomposition, also known as matrix factorization, involves describing a given matrix using its constituent elements.
- Recall that you saw QR decomposition in **Module 1** and then its use in **Module 2** (e.g., solving least squares, in particular sparse problems)
- Perhaps the most known and widely used matrix decomposition method is the **Singular-Value Decomposition**, or `SVD`.
- All matrices have an `SVD`, which makes it more stable than other methods, such as the eigen-decomposition.
- We will see the `SVD` is useful for computing the pseudoinverse efficiently and for dimensionality reduction

Singular Value Decomposition

What is SVD?

- One can generalize eigenvalues/vectors to non-square matrices, in which case they are called singular vectors and singular values.
- The SVD is a unique matrix decomposition that exists for every matrix $A \in \mathbb{R}^{m\times n}$:

$$A = U\Sigma V^\top$$

where $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ are *unitary* matrices, and $\Sigma \in \mathbb{R}^{m\times n}$ is a matrix with non-negative entries on the diagonal and zeros on the off diagonal.

- Unitary: $UU^\top = I$ and $VV^\top = I$

SVD as a Dyadic Exanpsion

An equivalent way to express the SVD $A = U\Sigma V^\top$ is as a dyadic expansion:

- That is, the SVD expresses A as a nonnegative linear combination of $\min\{m, n\}$ rank-1 matrices
- the singular values provide the multipliers
- the outer products of the left and right singular vectors provide the rank-1 matrices.

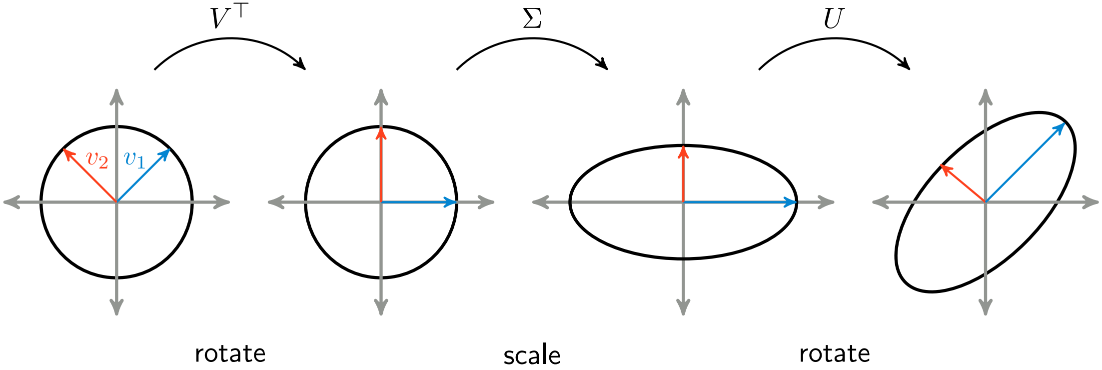
SVD

- The diagonal entries of Σ are called the **singular values** of A
- The column vectors of V are called the **right singular vectors** of A
- The column vectors of U are called the **left singular vectors** of A.
- The number of nonzero singular values is equal to the rank of the matrix A.

$$\underbrace{A}_{m \times n} = \underbrace{U}_{m \times m}\,\underbrace{\Sigma}_{m \times n}\,\underbrace{V^\top}_{n \times n}$$

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \ddots & 0 \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \sigma_n \\ 0 & \cdots & 0 & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 \end{bmatrix}$$

Geometric View of SVD



Unpacking the SVD

- Let $A \in \mathbb{R}^{m\times n}$
- **Fact 1**. Both $A^\top A \in \mathbb{R}^{n\times n}$ and $AA^\top \in \mathbb{R}^{m\times m}$ are symmetric square matrices:
- **Fact 2**. Both $A^\top A$ and $AA^\top$ share the same non-zero eigenvalues:

Unpacking the SVD

- According to the othogonally diagonalizable property of symmetric matrices, the matrices $A^\top A$ and $AA^\top$ can be decomposed as following:

- **How to obtain the SVD?**: Compute by diagonalizing the PSD symmetric matrices $A^\top A$ and $AA^\top$

Using the SVD to Compute Pseudo Inverses

- It turns out that using the `SVD` we have a very easy way to compute the pseudo-inverse of A—i.e., $A^\dagger = (A^\top A)^{-1} A^\top$ which we saw in **Mod1 & Mod2**

Matrix Norms and Connections to Singular values

- Matrix norms and singular values have special relationships.
- **Forbenius Norm**:

$$\|A\|_F = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2 \right)^{1/2} = (\mathsf{Tr}(A^\top A))^{1/2}$$

- **Matrix p-norm**: matrix p-Norm is defined as the largest scalar that you can get for a unit vector

$$\|A\|_p = \sup_{x \neq 0} \frac{\|Ax\|_p}{\|x\|_p} = \max_{\|x\|_p = 1} \|Ax\|_p$$

- Aside: supremum $\sup(\cdot)$ is the " least upper bound" of its argument

Matrix Norms: Spectral Norm

- **Spectral Norm (Matrix 2-norm)**: Largest singular value of the matrix $\sigma_1(A)$

- **Fact**: show that $\|A\|_F = \sqrt{\sum_{i=1}^{\min\{m,n\}} \sigma_i^2(A)}$ using the fact that $\|A\|_F = \sqrt{\mathsf{Tr}(A^\top A)}$

Reduced SVD & Low Rank Approximation

- Rank of Λ is $r \implies$ there are r non-zero eigenvalues of the matrices $A^\top A$ and $AA^\top$
- Reduced `SVD`:

Low Rank Structure

$$\underbrace{A}_{m \times n} = \underbrace{Y}_{m \times r} \; \underbrace{Z^\top}_{r \times n}$$

Low Rank Structure

$$A = uv^\top =$$

$$A = uv^\top + wz^\top =$$

Low Rank Approximation

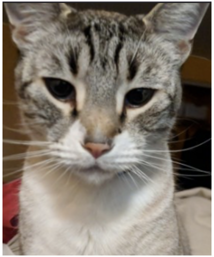
- Low Rank Approximation: take only top k-singular values and corresponding dyads in the dyadic expansion

- Low Rank Approximation is an important tool for many applications including
  - Linear system identification: approximating matrix is Hankel structured. (You saw this in `M2-N2.ipynb`)
  - ML: feature space dimensionality reduction
  - Recommender systems: matrix completion
  - Distance matrix completion where there is a positive definiteness constraint.
  - Natural language processing where the approximation is non-negative.
  - Image or video compression

Example: Compression

- **Compression.** A low-rank approximation provides a (lossy) compressed version of the data matrix.
  - The original matrix A is described by mn numbers, while describing Y and $Z^\top$ requires only $k(m+n)$ numbers.
  - When k is small relative to m and n, replacing the product of m and n by their sum is a big win.
  - With images, a modest value of k (say 100 or 150) is usually enough to achieve approximations that look a lot like the original image.

bender in color



gray scale

rank 2 bender

rank 20 bender

rank 100 bender

Optimality of Low Rank Approximation

- The low rank approximation obtained via SVD is optimal in the following sense.
- Recall the Forbenius norm:

- i.e., ℓ_2-norm (i.e., usual Euclidean norm) applied to the matrix as if it were a vector
- **Theorem [Eckat-Young-Mirsky]**.

How to choose k?

- When producing a low-rank matrix approximation, we have been taking as a parameter the target rank k.
- **Ideal Setting**: the singular values of A give strong guidance
  - if the top few singular values are big and the rest are small, then the obvious solution is to take k equal to the number of "big values".
- **Less Ideal Setting**: take k as small as possible subject to obtaining a useful approximation, where what "useful" means depends on the application.
  - e.g., a common rule of thumb is to choose k such that the sum of the top k singular values is at least c times as big as the sum of the other singular values, where c is a domain-dependent constant (like 10, say).

Next Up

- Next lecture we will talk about PCA, and show that PCA reduces to SVD and is fundamentally connected to low rank approximations.