Linear Systems and Applications

A Hands-On Python Workshop

Rhudaina Mohammad

Institute of Mathematics, UP Diliman

rzmohammad@up.edu.ph

23-25 May 2024 · CSRC Building
National Science Complex, UP Diliman

Workshop Schedule

Day 1	Python Basics and Programming Fundamentals.
Day 2	Introduction to Numerical Methods for Linear Systems and Applications.
Day 3	**Introduction to Dimensionality Reduction in Data Science.**

Dimensionality Reduction

High dimensionality is a common challenge in processing data from complex systems.

Dimensionality Reduction

Principal Component Analysis – an algorithm that decomposes high-dimensional data into its most statistically descriptive factors

Consider a dataset A of n samples – each with m features, represented as a matrix

1. Standardize data $A \in \mathbb{R}^{n \times m}$ along the features: Z
2. Calculate the covariance matrix for the features: $C = \frac{1}{n-1} Z^T Z$
3. Perform eigendecomposition on the covariance matrix: $C = X \Lambda X^{-1}$
4. Sort principal components (PCs) based on their eigenvalues from highest to lowest.
5. Calculate explained variance (eigenvalue : sum of eigenvalues) for each PC.
6. Reduce standardized data Z by the desired number p of PCs: $Z X_{p,\text{sorted}}$ where $X_{p,\text{sorted}} \in \mathbb{R}^{m \times p}$ contains the first p columns of X_{sorted}.

Reduced (Economy) SVD

When $n \geq m$, Σ has at most m nonzero elements on the diagonal, and X can be represented exactly using the reduced (or economy) SVD.

$$\underset{n\times m}{A} = \underbrace{\begin{bmatrix}\underset{n\times m}{\hat{U}} & \underset{n\times(n-m)}{\hat{U}^{\perp}}\end{bmatrix}}_{U\in\mathbb{R}^{n\times n}} \underbrace{\begin{bmatrix}\underset{m\times m}{\hat{\Sigma}} \\ 0\end{bmatrix}}_{\Sigma\in\mathbb{R}^{n\times m}} \underset{m\times m}{V^T} = \underset{n\times m}{\hat{U}}\;\underset{m\times m}{\hat{\Sigma}}\;\underset{m\times m}{V^T}$$

- The columns of U are called left singular vectors; while the columns of V are right singular vectors.
- The diagonal elements of $\hat{\Sigma}$ are called singular values (in decreasing order).
- The rank of A is equal to the number of nonzero singular values.

Dimensionality Reduction

- Since all nonzeros entries of Σ are in the diagonal, then we can write

$$A = U\Sigma V^T = \sum_{i=1}^{m} \sigma_i u_i v_i^T = \underbrace{\sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \cdots + \sigma_m u_m v_m^T}_{\text{sum of rank-1 matrices}}$$

  where singular values σ_i is the ith diagonal entry of Σ such that

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_m \geq 0$$

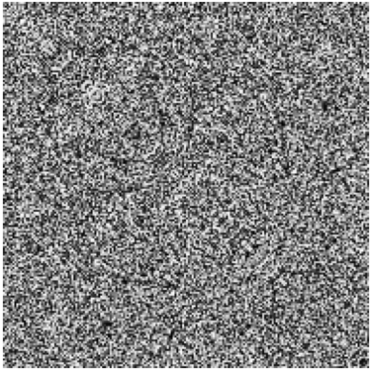
  Here, u_i and v_i are the ith columns of singular vectors U and V, respectively.

- For many systems, singular values decrease rapidly, and so, we can get a good approximation of A by truncating at some rank p:

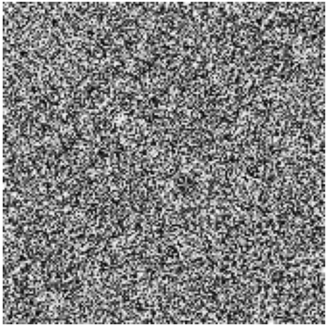
$$A \approx A_p = \sum_{i=1}^{p} \sigma_i u_i v_i^T = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \cdots + \sigma_p u_p v_p^T$$

Dimensionality Reduction

$A \in \mathbb{R}^{192\times168}$



$AA^T \in \mathbb{R}^{192\times192}$



$A^TA \in \mathbb{R}^{168\times168}$

- Both correlation matrices AA^T and A^TA are symmetric.
  - AA^T is formed by taking inner product of rows of A
  - A^TA is formed by taking inner product of columns of A

Dimensionality Reduction

- Note that

$$A = U\Sigma V^T = U \begin{bmatrix} \hat{\Sigma} \\ 0 \end{bmatrix} V^T$$

  Then,

$$\begin{aligned} AA^T &= U \begin{bmatrix} \hat{\Sigma} \\ 0 \end{bmatrix} \underbrace{V^T V}_{I} \begin{bmatrix} \hat{\Sigma} & 0 \end{bmatrix} U^T = U \begin{bmatrix} \hat{\Sigma} & 0 \\ 0 & 0 \end{bmatrix} U^T \\ A^T A &= V \begin{bmatrix} \hat{\Sigma} & 0 \end{bmatrix} \underbrace{U^T U}_{I} \begin{bmatrix} \hat{\Sigma} \\ 0 \end{bmatrix} V^T = V \hat{\Sigma}^2 V^T \end{aligned}$$

- To get the singular values, it is better to compute the eigendecomposition of $A^T A$, which is smaller and more manageable than AA^T.

Dimensionality Reduction

- Principal Component Analysis (PCA): eigendecomposition $X\Lambda X^{-1}$ of covariance matrix of standardized data A given by

$$C := \frac{1}{n-1}A^TA$$

  - If $A = U\Sigma V^T$, then we can write the covariance matrix

$$C = \frac{1}{n-1}\left(U\Sigma V^T\right)^T U\Sigma V^T = V\frac{1}{n-1}\underbrace{\Sigma^T}_{\Sigma}\underbrace{U^TU}_{I}\Sigma V^T = \frac{1}{n-1}V\Sigma^2V^T$$

  Hence, $W = V$ and $\Lambda = \frac{1}{n-1}\Sigma^2$, and so, the eigenvalues can be obtained from the singular values of A:

$$\lambda_k = \frac{\sigma_k^2}{n-1}$$

- SVD provides a numerically robust approach for computing the principal components.

Truncated SVD

$$\underset{n\times m}{A} = \underbrace{\left[\begin{array}{c|c|c} \underset{n\times p}{U_p} & * & * \end{array}\right]}_{U\in\mathbb{R}^{n\times n}} \underbrace{\left[\begin{array}{c|c} \underset{p\times p}{\Sigma_p} & 0 \\ \hline 0 & * \\ \hline \multicolumn{2}{c}{0} \end{array}\right]}_{\Sigma\in\mathbb{R}^{n\times m}} \underbrace{\left[\begin{array}{c} \underset{r\times m}{V_p^T} \\ \hline * \end{array}\right]}_{V\in\mathbb{R}^{m\times m}}$$

$$\approx \underset{n\times p}{U_p}\;\underset{p\times p}{\Sigma_p}\;\underset{p\times m}{V_p^T}$$

Image Compression via Reduced SVD

Consider a grayscale image with n pixels in the vertical direction and m pixels in the horizontal direction

- Let A be an $n \times m$ matrix representation of a 8-bit grayscale image
- pixel values denote light intensities from 0 (black) to 255 (white)

(UP Diliman, 2021)

$A \in \mathbb{R}^{4000 \times 3000}$

$p = 20$

about 50% image variance

$p = 200$

almost 80% image variance

Thank you for your attention!