Mathematics for Machine Learning

Lecture 10
(27.06.2024)

Dimensionality Reduction

Mahammad Namazov

Table of contents

Covariance Matrix

Introduction

Computation

Significance

Introduction

- In Probability Theory, we have seen covariance represents how vary two variables together;
- In case there are 2 variables, covariance matrix will be 2x2 matrix, which diagonal elements are variance of corresponding variable;

$$\Sigma = \begin{matrix} Var(X) & Cov(X,Y) \\ Cov(Y,X) & Var(Y) \end{matrix}$$

- Now assume, you have a dataset which includes n data, where each data has k features;
- Covariance Matrix based on this dataset represents the co-variation among features across all data in our dataset;
  - **Question: What will be the dimensions of Covariance Matrix of this dataset?**

Computation

- Now assume $\boldsymbol{X}$ is dataset, where each row of the matrix is a single datapoint, j^{th} column will be j^{th} feature:

$$\boldsymbol{X} = \begin{bmatrix} \cdots & \boldsymbol{x_1^T} & \cdots \\ \vdots & \vdots & \vdots \\ \cdots & \boldsymbol{x_n^T} & \cdots \end{bmatrix}$$

- Steps:
  - Compute mean for j^{th} feature across all data points:
  - Create a mean vector, where j^{th} element is mean of j^{th} feature:
  - Center each data point using their representations and mean vector:
  - Covariance Matrix will be:

- $\mu_j = \frac{1}{N}\sum_i^n \boldsymbol{X}_{i,j}$
- $\mathrm{M} = [\mu_1 \quad \mu_2 \quad \cdots \quad \mu_k]$
- $\overline{\boldsymbol{X}}_i = \boldsymbol{X}_i - M$
- $Cov(X) = \frac{1}{N-1}\boldsymbol{X}^T\boldsymbol{X}$

Note: There is another method, where columns are data. Computation steps are same, but contents are different because of dimensional compatibility (i.e., np.cov(.))

Significance

- From the computation we see that, matrix represents how each feature change through each data point in a given dataset;
- Also notice that computation centers each datapoint around zero then compute the covariance;
- Computation is same as we did in Probability Theory, when co-occurrence probability is 1;
- It is a **symmetric matrix**, which diagonal elements are variances of corresponding features;

Eigendecomposition

Eigenvalues

R-eigenvectors

L-eigenvectors

Eigen values

- From linear transformation we know that to transform a vector $\boldsymbol{v} \in V \subseteq \mathbb{R}^n$ into $\boldsymbol{w} \in W \subseteq \mathbb{R}^n$, we need a matrix which is square matrix $A_{n\times \mathrm{n}}$;
- However, there are such vectors in V so that transformation only scales them (i.e., multiplication by this matrix A will be same with multiplication by some scalar λ);

$$A\boldsymbol{v} = \lambda\boldsymbol{v}$$

- These vectors are called eigenvectors, and these scalars are called eigenvalues;

Right Eigenvectors

- Right Eigenvectors are vectors which are transformation invariant by given linear transformation matrix. However, this matrix can scale the given matrix by some specific constant (which are right eigenvalues);

- Computation can be given by the following steps:
  - Starting from given equation, which indicates transformation can scale only:
    $$A\boldsymbol{v} = \lambda\boldsymbol{v}$$
  - Representing specific transformation that transforms the given vector into origin point:
    $$(A - \lambda I)\boldsymbol{v} = \mathbf{0}$$
  - For a non-zero vector the equation above possible if and only if (WHY?):
    $$\det(A - \lambda I) = \mathbf{0} \Rightarrow \{\lambda_1, \dots, \lambda_n\}$$
  - Once we found eigenvalues, we can find eigenvectors by using them:
    $$(A - \lambda I)\boldsymbol{v} = \mathbf{0}$$

Left Eigenvectors

- Left eigenvectors define hyperplanes that are transformation invariant with respect to the given transformation matrix:

$$\boldsymbol{w}^T A = \lambda \boldsymbol{w}^T$$

- Computation steps are same as in right eigenvector computation:
  - Starting from given equation, which indicates transformation can scale only:

$$\boldsymbol{w}^T A = \lambda \boldsymbol{w}^T$$

  - Representing specific transformation that transforms the given vector into origin point:

$$\boldsymbol{w}^{\mathbf{T}}(A - \lambda I) = \mathbf{0}$$

  - For a non-zero vector the equation above possible if and only if (WHY?):

$$\det(A - \lambda I) = \mathbf{0} \Rightarrow \{\lambda_1, \dots, \lambda_n\}$$

  - Once we found eigenvalues, we can find eigenvectors by using them:

$$\boldsymbol{w}^T(A - \lambda I) = \mathbf{0}$$

Properties

- If you can decompose a matrix, it means it is diagonalizable matrix:
  - i.e., Only diagonalizable matrices can be decomposed by eigenvalues;
- If at least one eigenvalue is zero, then matrix is singular (determinant?);
- When you have a symmetric matrix, eigen-decomposition can be written as:

$$A = V\Lambda V^{-1} = V\Lambda V^{T}$$

- On the other hand, in such scenario, V is orthogonal matrix;
- Let's check the inverse of A: A^{-1}
  - Inverse application inverses the application;
  - Lost information from diagonal matrices;

$$A = \begin{bmatrix} a_{1,1} & 0 & 0 \\ 0 & a_{2,2} & 0 \\ 0 & 0 & a_{3,3} \end{bmatrix}$$

Singular Value Decomposition

Idea

Right SV

Left SV

Applications

Idea

- Square matrices are special matrices: $m = n$
- Thus, eigen-decomposition cannot be seen as generic decomposition;
- What about Singular Value Decomposition?
- We can decompose any matrix in a way that, it can be shown as composition of some matrices that each indicates significant properties;
- Mathematical Representation for a matrix A:

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

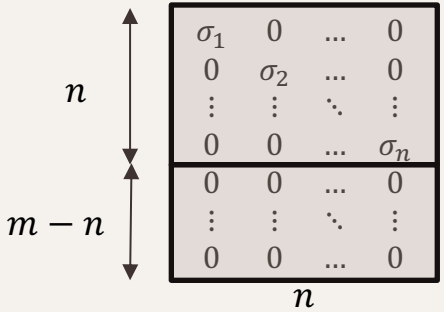
Singular Matrix

- Any matrix A has singular value decomposition, and it is unique:

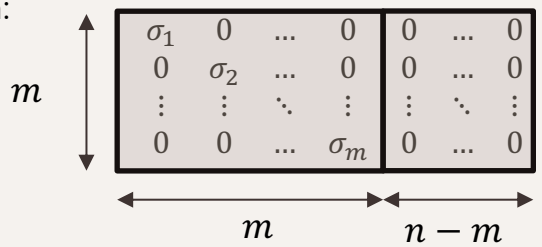
$$A = U\Sigma V^T$$

- Σ is singular value matrix, which is a diagonal matrix of shape ($m \times n$):
- Form of Σ will always be related to matrix shape;

- When m>n:



- When m<n:



Left Singular Vectors

- Any matrix A has singular value decomposition, and it is unique:

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

- U is a unitary matrix of shape $(m \times m)$:
- How to represent these features linearly independent from each other?
- You have m features per data, then you will need m vectors for that:

$$U = \begin{bmatrix} \vdots & \vdots & \dots & \vdots \\ \boldsymbol{u_1} & \boldsymbol{u_2} & \dots & \boldsymbol{u_m} \\ \vdots & \vdots & \dots & \vdots \end{bmatrix}$$

- Remember what is unitary matrix is?

$$U^T U = UU^T = I \Rightarrow U^T = U^{-1}$$

Right Singular Vectors

- Any matrix A has singular value decomposition, and it is unique:

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

- V is a unitary matrix of shape $(n \times n)$:
- How to represent these samples linearly independent?
- You have n samples, then you will need n vectors for that:

$$V = \begin{bmatrix} \vdots & \vdots & \dots & \vdots \\ \boldsymbol{v_1} & \boldsymbol{v_2} & \dots & \boldsymbol{v_n} \\ \vdots & \vdots & \dots & \vdots \end{bmatrix}$$

- However, we will use the transpose of this matrix:

$$V^T = \begin{bmatrix} \cdots & \boldsymbol{v_1^T} & \dots \\ \dots & \boldsymbol{v_2^T} & \dots \\ \vdots & \vdots & \vdots \\ \dots & \boldsymbol{v_n^T} & \dots \end{bmatrix}$$

Interpretation

- Any matrix A has singular value decomposition, and it is unique:
$$A = U\Sigma V^T$$
- What is A^TA?
  - Is it not covariance between samples (scaled by number of samples)?
- What if we do it with the equation above?
$$A^TA = V\tilde{\Sigma}^T\tilde{U}^T\tilde{U}\tilde{\Sigma}V^T$$
- Then we can use properties of unitary matrices:
$$A^TA = V\tilde{\Sigma}^T\tilde{\Sigma}V^T$$
- If we analyze $\tilde{\Sigma}^T\tilde{\Sigma}$ and use S^2 for simplicity:
$$A^TA = VS^2V^T$$
- Remember symmetric matrices and diagonal matrices?
  - If yes, then you will see that it is eigen-decomposition of A^TA, where V is right eigen-vectors of A^TA.

Interpretation

- Any matrix A has singular value decomposition, and it is unique:

$$A = U\Sigma V^T = \tilde{U}\tilde{\Sigma}V^T$$

- What is AA^T?
  - Is not it correlation between features?
- What if we do it with the equation above?

$$AA^T = \tilde{U}\tilde{\Sigma}V^T V\tilde{\Sigma}^T\tilde{U}^T$$

- Then we can use properties of unitary matrices:

$$AA^T = \tilde{U}\tilde{\Sigma}\tilde{\Sigma}^T\tilde{U}^T$$

- If we analyze $\tilde{\Sigma}\tilde{\Sigma}^T$ and use S^2 for simplicity:

$$AA^T = US^2U^T$$

- Same is applied here:
  - Then U is eigen-vectors of AA^T

Interpretation

- Any matrix A has singular value decomposition, and it is unique:
$$A = U\Sigma V^T$$
- Singular values that makes Σ, are ordered in decreasing way:
$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n$$
- Singular Value Decomposition: Linear Combination of 1 rank matrices:
$$A = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \cdots + \sigma_n u_n v_n^T$$
  - What we did was, eliminating $m - n$ columns from U, $m - n$ rows from Σ, where $m > n$;
$$A = \hat{U}\hat{\Sigma}V^T$$
- What if we want such rank that is less than n?
  - For instance, $\rho = r < n$
  - Keep elements till r, remove the rest;
$$A \cong \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \cdots + \sigma_r u_r v_r^T$$
- We simply reduced the dimensionality (Young-Eckard Theorem):
$$A = \bar{U}\bar{\Sigma}\bar{V}^T$$

Principal Component Analysis

Idea

Computation

Significance

Idea

- Same idea as SVD, but we extract the most significant data from all representations;
- Suppose you have such scenario:
  - You have m samples, where each has n properties;
  - SVD showed us that, these features can be decomposed in an ordered way;
  - Order is determined according to the information;
- What if we can decompose a matrix with its principal components with this idea?
- Then we choose r components that carries most of the information!

Computation

- Let's define the matrix A, which combines all of the information:
- We have m samples, where each sample has n features (i.e., $\boldsymbol{a_i} \in \mathbb{R}^n$);
  - We generate mean matrix $\bar{A}$;
  - Compute mean vector over all samples:

$$\overline{\boldsymbol{a}} = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{a_i}$$

  - Make such matrix, that each row is $\overline{\boldsymbol{a}}$:

$$\bar{A} = [\mathbf{1}]_{m\times 1}\overline{\boldsymbol{a}}$$

  - Find normalized matrix with respect to the mean:

$$B = A - \bar{A}$$

  - Generate Covariance Matrix with respect to the features:

$$C = B^T B$$

  - Use eigen-decomposition of Covariance Matrix C:

$$C = V\Lambda V^{-1}$$

$$A = \begin{bmatrix} \dots & \boldsymbol{a_1} & \dots \\ \dots & \boldsymbol{a_2} & \dots \\ \vdots & \vdots & \vdots \\ \dots & \boldsymbol{a_m} & \dots \end{bmatrix}$$

Voila:

$$T = BV$$

Details

- Do you recall the notion of SVD?
- What we do is using SVD on B:

$$B = U\Sigma V^T$$

- Then multiplying two sides by V:

$$BV = U\Sigma$$

- Amount of variance is captured by principal components is determined by eigenvalues in Λ:

$$\lambda_i = \sigma_i^2$$

- How can I choose r principal components that can be used to capture %95 of all variances?

$$\frac{\sum_{j=1}^{r} \lambda_i}{\sum_{j=1}^{n} \lambda_i}$$

Conclusion

Summary

Takeaways

References

Summary

- When number of variables are greater than 2, we cannot interpret distribution with a table;
- Correlation of such variable is always 1;
- Matrix decomposition is significant tool to extract valuable information from matrices;
- Eigen-decomposition – Singular Value Decomposition
- PCA is an algorithm that we can reduce the dimensionality;
- The resulting matrix will have same dimensions, but information that captured will be different;

Takeaways

- Applying a transformation matrix to a variable, transforms covariance respectively;
- Covariance matrix is useful not only for probability distributions but also for data;
- Using Matrix Decomposition, we can diagonalize any matrix;
- For eigen-decomposition, such diagonal matrix includes eigenvalues;
- For SVD, such diagonal matrix includes singular values;
- Singular values are ordered with respect to their significance;

References

- Further information to read:
  - Deisenroth, M. P., Faisal, A. A., & Ong, C. S. (2020). *Mathematics for machine learning*. Cambridge University Press.
    - Chapter 10
    - Chapter 4, Sections 2, 4, 5

The End

Thanks for your attention!

Mahammad Namazov