

Introduction to Machine Learning

Principal Component Analysis

Varun Chandola

Computer Science & Engineering
State University of New York at Buffalo
Buffalo, NY, USA
chandola@buffalo.edu



University at Buffalo
Department of Computer Science
and Engineering
School of Engineering and Applied Sciences



Principal Components Analysis

- Introduction to PCA

- Principle of Maximal Variance

- Defining Principal Components

- Dimensionality Reduction Using PCA

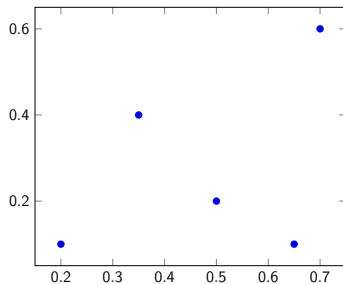
- PCA Algorithm

- Recovering Original Data

- Eigen Faces

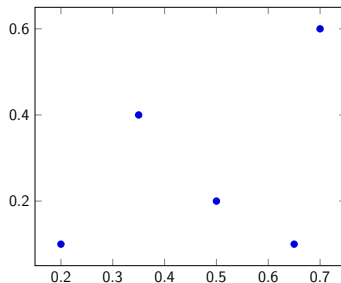
Introduction to PCA

- Consider the following data points



Introduction to PCA

- ▶ Consider the following data points



- ▶ *Embed* these points in 1 dimension
- ▶ What is the best way?
 - ▶ **Along the direction of the maximum variance**
 - ▶ Why?

Why Maximal Variance?

- ▶ Least loss of information
- ▶ Best capture the “spread”

Why Maximal Variance?

- ▶ Least loss of information
- ▶ Best capture the “spread”
- ▶ What is the direction of maximal variance?
- ▶ Given any direction ($\hat{\mathbf{u}}$), the projection of \mathbf{x} on $\hat{\mathbf{u}}$ is given by:

$$\mathbf{x}_i^\top \hat{\mathbf{u}}$$

- ▶ Direction of maximal variance can be obtained by maximizing

$$\begin{aligned} \frac{1}{N} \sum_{i=1}^N (\mathbf{x}_i^\top \hat{\mathbf{u}})^2 &= \frac{1}{N} \sum_{i=1}^N \hat{\mathbf{u}}^\top \mathbf{x}_i \mathbf{x}_i^\top \hat{\mathbf{u}} \\ &= \hat{\mathbf{u}}^\top \left(\frac{1}{N} \sum_{i=1}^N \mathbf{x}_i \mathbf{x}_i^\top \right) \hat{\mathbf{u}} \end{aligned}$$

Finding Direction of Maximal Variance

- Find:

$$\max_{\hat{\mathbf{u}}: \hat{\mathbf{u}}^\top \hat{\mathbf{u}} = 1} \hat{\mathbf{u}}^\top \mathbf{S} \hat{\mathbf{u}}$$

where:

$$\mathbf{S} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i \mathbf{x}_i^\top$$

- \mathbf{S} is the sample (empirical) covariance matrix of the mean-centered data

Defining Principal Components

- ▶ First PC: Eigen-vector of the (sample) covariance matrix with largest eigen-value
- ▶ Second PC?

Defining Principal Components

- ▶ First PC: Eigen-vector of the (sample) covariance matrix with largest eigen-value
- ▶ Second PC?
- ▶ Eigen-vector with next largest value
- ▶ Variance of each PC is given by λ_i
- ▶ Variance captured by first L PC ($1 \leq L \leq D$)

$$\frac{\sum_{i=1}^L \lambda_i}{\sum_{i=1}^D \lambda_i} \times 100$$

- ▶ What are eigen vectors and values?

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

\mathbf{v} is eigen vector and λ is eigen-value for the **square matrix \mathbf{A}**

- ▶ Geometric interpretation?

Dimensionality Reduction Using PCA

- ▶ Consider first L eigen values and eigen vectors
- ▶ Let \mathbf{W} denote the $D \times L$ matrix with first L eigen vectors in the columns (sorted by λ 's)
- ▶ PC score matrix

$$\mathbf{Z} = \mathbf{XW}$$

- ▶ Each input vector ($D \times 1$) is replaced by a shorter $L \times 1$ vector

PCA Algorithm

1. Center \mathbf{X}

$$\mathbf{X} = \mathbf{X} - \hat{\boldsymbol{\mu}}$$

2. Compute sample covariance matrix:

$$\mathbf{S} = \frac{1}{N-1} \mathbf{X}^T \mathbf{X}$$

3. Find eigen vectors and eigen values for \mathbf{S}
4. \mathbf{W} consists of first L eigen vectors as columns
 - ▶ Ordered by decreasing eigen-values
 - ▶ \mathbf{W} is $D \times L$
5. Let $\mathbf{Z} = \mathbf{XW}$
6. Each row in \mathbf{Z} (or \mathbf{z}_i^T) is the lower dimensional embedding of \mathbf{x}_i

Recovering Original Data

- ▶ Using \mathbf{W} and \mathbf{z}_i

$$\hat{\mathbf{x}}_i = \mathbf{W}\mathbf{z}_i$$

- ▶ **Average Reconstruction Error**

$$J(\mathbf{W}, \mathbf{Z}) = \frac{1}{N} \sum_{i=1}^N \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2$$

Theorem (Classical PCA Theorem)

Among all possible orthonormal sets of L basis vectors, PCA gives the solution which has the minimum reconstruction error.

- ▶ Optimal “embedding” in L dimensional space is given by $\mathbf{z}_i = \mathbf{W}^T \mathbf{x}_i$

Using PCA for Face Recognition

EigenFaces [1]

- ▶ **Input:** A set of images (of faces)
- ▶ **Task:** Identify if a new image is a face or not.



M. Turk and A. Pentland.

Face recognition using eigenfaces.

In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR '91., IEEE Computer Society Conference on*, pages 586–591, Jun 1991.