Foundations of Data Science, Fall 2020

16. Principal Component Analysis

Prof. Dan Olteanu

DaST Data•(Systems+Theory)

University of Zurich[UZH]

December 8, 2020

https://lms.uzh.ch/url/RepositoryEntry/16830890400

https://uzh.zoom.us/j/96690150974?pwd=cnZmMTduWUtCeWoxYW85Z3RMYnpTZz09

Dimensionality Reduction

Real-world data can have redundant information

- Data may have correlated variables

  When one variable changes, the other changes as well

- Data synthesised by different teams may have redundancy

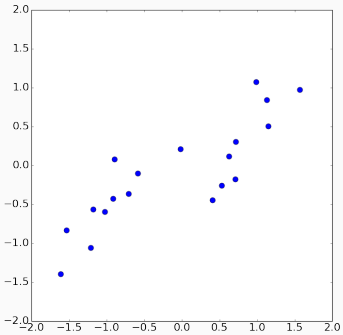
Dimensionality reduction

- Project the data into a lower dimensional subspace while still capturing the essence of the original data
- Relevant for visualisation, eg, projections into a 2D subspace
- Preprocessing step before applying other learning algorithms

Principal Component Analysis (PCA) used for dimensionality reduction

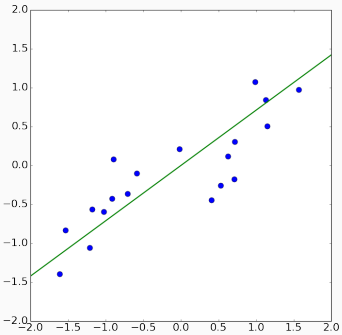
- Identifies a small number of directions which explain most variation in data

Principal Component Analysis



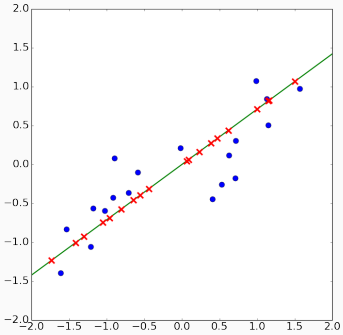
Which direction has the most variance?

Principal Component Analysis



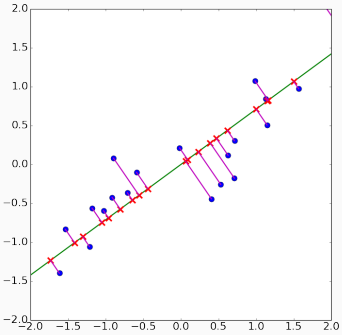
1st principal component = line passing through the multidimensional mean and minimises the sum of squares of the distances of the blue points from the line

Principal Component Analysis



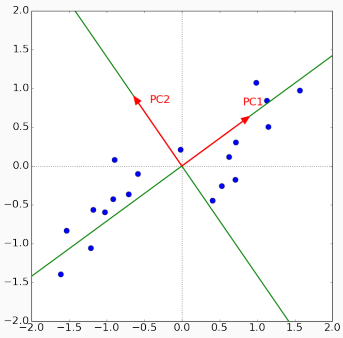
The red points are orthogonal projections of the blue points onto a vector representing the 1st principal component. The 2D points are projected onto 1D.

Principal Component Analysis



The magenta lines are the distances of the blue points from the 1st principal component. They are also called reconstruction errors.

Principal Component Analysis



2nd principal component = compute it as the 1st principal component, after correlation with 1st principal component has been subtracted from the points

2nd principal component is orthogonal to the 1st principal component

Two Equivalent Views of Principal Component Analysis

Maximum Variance

Finding k directions that maximise the variance in the data

Best Reconstruction

Finding k dimensional subspace with least reconstruction error

We will next discuss both views and show they are equivalent.

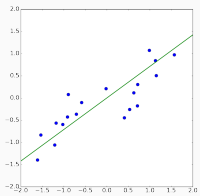
PCA Keeps Subspace with Maximum Variance

Aligns the points with the principal components

Moves as much of the variance as possible using an orthogonal transformation into the first few dimensions

The values in the remaining dimensions tend to be small and may be dropped with minimal loss of information

Is the optimal orthogonal transformation that keeps the subspace with the largest "variance"



PCA: Maximum Variance View

PCA is a *linear* dimensionality reduction technique

Finds the directions of maximum variance in the data $(\mathbf{x}_1, \ldots, \mathbf{x}_N)$

Find a set of orthonormal vectors $\mathbf{v}_1, \ldots, \mathbf{v}_k$

- The first principal component (PC) $\mathbf{v}_1$ is the direction of largest variance
- The second PC $\mathbf{v}_2$ is the direction of largest variance orthogonal to $\mathbf{v}_1$
- The i^{th} PC $\mathbf{v}_i$ is the direction of largest variance orthogonal to $\mathbf{v}_1, \ldots, \mathbf{v}_{i-1}$

PCA: Maximum Variance View

Given: i.i.d. data $\mathbf{x}_1, \ldots, \mathbf{x}_N$; data matrix $\mathbf{X} = [\mathbf{x}_1^\top, \ldots, \mathbf{x}_N^\top]^\top$

Goal: Find $\mathbf{v}_1 \in \mathbb{R}^D$, $\|\mathbf{v}_1\| = 1$, that maximizes $\|\mathbf{X}\mathbf{v}_1\|^2$

Solution: $\mathbf{v}_1 = \underset{\mathbf{v}_1 : \|\mathbf{v}_1\|=1}{\operatorname{argmax}} \left(\mathbf{v}_1^\top \mathbf{X}^\top \mathbf{X} \mathbf{v}_1 \right)$

- Let $\mathbf{z} = \mathbf{X}\mathbf{v}_1 = [\mathbf{x}_1^\top \mathbf{v}_1, \ldots, \mathbf{x}_N^\top \mathbf{v}_1]^\top$, so $z_i = \mathbf{x}_i^\top \mathbf{v}_1$.
- $\|\mathbf{X}\mathbf{v}_1\|^2$ is the variance of the projections of data onto $\mathbf{v}_1$ (mod const factor $\frac{1}{N}$)

$$\|\mathbf{X}\mathbf{v}_1\|^2 = \sum_{i=1}^N z_i^2 = \sum_{i=1}^N z_i^2 = \mathbf{z}^\top \mathbf{z} = \mathbf{v}_1^\top \mathbf{X}^\top \mathbf{X} \mathbf{v}_1$$

- This assumes that data is centred: $\sum_i \mathbf{x}_i = \mathbf{0}$
- Find $\mathbf{v}_2, \mathbf{v}_3, \ldots, \mathbf{v}_k$ that are all successively orthogonal to previous directions and maximise (as yet unexplained) variance

$$\widehat{\mathbf{X}}_j = \mathbf{X} - \sum_{s=1}^{j-1} \mathbf{X}\mathbf{v}_s\mathbf{v}_s^\top, \qquad 1 \le j \le k$$

Linear Algebra Background: Rayleigh quotient

$$R(\mathbf{M}, \mathbf{v}) = \frac{\mathbf{v}^\top \mathbf{M} \mathbf{v}}{\mathbf{v}^\top \mathbf{v}}$$

For positive semi-definite $\mathbf{M}$:

- The max value of $R(\mathbf{M}, \mathbf{v})$ is the largest eigenvalue of $\mathbf{M}$
- This is attained for $\mathbf{v}$ being the corresponding eigenvector

Applied to our problem:

- $\mathbf{v}_1 = \underset{\mathbf{v}_1 : \|\mathbf{v}_1\|=1}{\operatorname{argmax}} \left(\mathbf{v}_1^\top \mathbf{X}^\top \mathbf{X} \mathbf{v}_1 \right) = \operatorname{argmax} \left\{ \frac{\mathbf{v}_1^\top \mathbf{X}^\top \mathbf{X} \mathbf{v}_1}{\mathbf{v}_1^\top \mathbf{v}_1} \right\}$
- $\mathbf{X}^\top \mathbf{X}$ **is** positive semi-definite (cf. earlier lectures)
- Largest eigenvalue of $\mathbf{X}^\top \mathbf{X}$ = max value attained by $\mathbf{v}_1^\top \mathbf{X}^\top \mathbf{X} \mathbf{v}_1$ for $\|\mathbf{v}_1\| = 1$
- $\mathbf{v}_1$ is the corresponding eigenvector

PCA: Best Reconstruction View

Given: i.i.d. data $\mathbf{x}_1, \ldots, \mathbf{x}_N$; data matrix $\mathbf{X} = [\mathbf{x}_1^T, \ldots, \mathbf{x}_N^T]^T$

Goal: Find a k-dimensional linear projection that best represents the data

Solution in case of one PC:

- Let $\mathbf{v}_1$ be the direction of projection
- The point $\mathbf{x}_i$ is mapped to $\tilde{\mathbf{x}}_i = (\mathbf{v}_1 \cdot \mathbf{x}_i)\mathbf{v}_1$, where $\|\mathbf{v}_1\| = 1$
- Minimise reconstruction error $\sum_{i=1}^N \|\mathbf{x}_i - \tilde{\mathbf{x}}_i\|^2$

Solution in case of k PCs:

- Suppose $\mathbf{V}_k \in \mathbb{R}^{D\times k}$ is such that columns of $\mathbf{V}_k$ are orthogonal
- Project data $\mathbf{X}$ onto subspace defined by $\mathbf{V}$: $\mathbf{Z} = \mathbf{X}\mathbf{V}_k$
- Minimise reconstruction error $\sum_{i=1}^N \|\mathbf{x}_i - \mathbf{V}_k\mathbf{V}_k^T\mathbf{x}_i\|^2$

Equivalence between the Two Objectives: One PC Case

Let $\mathbf{v}_1$ be the direction of projection

The point $\mathbf{x}$ is mapped to $\tilde{\mathbf{x}} = (\mathbf{v}_1 \cdot \mathbf{x})\mathbf{v}_1$, where $\|\mathbf{v}_1\| = 1$

Maximum Variance

Find $\mathbf{v}_1$ that maximises $\sum_{i=1}^N (\mathbf{v}_1^T\mathbf{x}_i)^2 = \sum_{i=1}^N (\mathbf{v}_1 \cdot \mathbf{x}_i)^2$

Best Reconstruction

Find $\mathbf{v}_1$ that minimises:

$$\begin{aligned}\sum_{i=1}^N \|\mathbf{x}_i - \tilde{\mathbf{x}}_i\|^2 &= \sum_{i=1}^N \left(\|\mathbf{x}_i\|^2 - 2(\mathbf{x}_i \cdot \tilde{\mathbf{x}}_i) + \|\tilde{\mathbf{x}}_i\|^2\right) \\ &= \sum_{i=1}^N \left(\|\mathbf{x}_i\|^2 - 2(\mathbf{v}_1 \cdot \mathbf{x}_i)^2 + (\mathbf{v}_1 \cdot \mathbf{x}_i)^2\|\mathbf{v}_1\|^2\right) \\ &= \sum_{i=1}^N \|\mathbf{x}_i\|^2 - \sum_{i=1}^N (\mathbf{v}_1 \cdot \mathbf{x}_i)^2\end{aligned}$$

So the same $\mathbf{v}_1$ satisfies the two objectives

Finding Principal Components using Singular Value Decomposition

Recall the SVD $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ of $\mathbf{X} \in \mathbb{R}^{N\times D}$

- $\mathbf{\Sigma} \in \mathbb{R}^{N\times D}$ is diagonal with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_D \geq 0$
- $\mathbf{V} \in \mathbb{R}^{D\times D}$ is orthonormal matrix with right singular vectors
- $\mathbf{U} \in \mathbb{R}^{N\times N}$ is orthonormal matrix with left singular vectors

PCA can be associated with this SVD of $\mathbf{X}$:

The first k principal components are first k columns of $\mathbf{V}$

- PCA is the linear transformation $\mathbf{Z} = \mathbf{X}\mathbf{V}$
- Let $\mathbf{V}_k$ be the first k columns in $\mathbf{V}$
- Then, dimensionality reduction via PCA: $\mathbf{Z}_k = \mathbf{X}\mathbf{V}_k$

Finding Principal Components using Singular Value Decomposition

Since $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, we obtain

$$\mathbf{X}^T\mathbf{X} = (\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T)^T(\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T) = \mathbf{V}\mathbf{\Sigma}^T\underbrace{\mathbf{U}^T\mathbf{U}}_{\mathbf{I}_N}\mathbf{\Sigma}\mathbf{V}^T = \mathbf{V}\underbrace{\mathbf{\Sigma}^T\mathbf{\Sigma}}_{\mathbf{\Sigma}^2 \in \mathbb{R}^{D\times D}}\mathbf{V}^T$$

- the eigenvectors of $\mathbf{X}^T\mathbf{X}$ are the right singular vectors $\mathbf{v}_i$ of $\mathbf{X}$
- the eigenvalues of $\mathbf{X}^T\mathbf{X}$ are the squares of the singular values σ_i of $\mathbf{X}$
- Similarly, the eigenvectors $\mathbf{u}_j$ of $\mathbf{X}\mathbf{X}^T$ with eigenvalue σ^2 are the left singular vectors of $\mathbf{X}$ (this can be used in case $D > N$)

The PCA transformation $\mathbf{Z} = \mathbf{X}\mathbf{V} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\mathbf{V} = \mathbf{U}\mathbf{\Sigma}$. We can truncate $\mathbf{Z}$:

- For $\mathbf{Z}_k \in \mathbb{R}^{N\times k}$, we only need the first k largest singular values and their singular vectors: $\mathbf{Z}_k = \mathbf{U}_k\mathbf{\Sigma}_k = \mathbf{X}\mathbf{V}_k$
- Eckart-Young theorem: $\mathbf{Z}_k$ is nearest possible matrix of rank k to $\mathbf{X}$: The difference between the two has the smallest Frobenius norm

Algorithm for Finding PCs (when N > D)

Variant 1: Construct $\mathbf{X}^T\mathbf{X}$ in $O(D^2N)$ and its eigenvectors in $O(D^3)$

Variant 2: Iterative methods to get top k singular (right) vectors directly:

- Initiate $\mathbf{v}^0$ to be random unit norm vector
- Iterative Update:
  - $\mathbf{v}^{t+1} = \mathbf{X}^T\mathbf{X}\mathbf{v}^t$
  - $\mathbf{v}^{t+1} = \mathbf{v}^{t+1}/\|\mathbf{v}^{t+1}\|$

  until (approximate) convergence
- Update step takes $O(ND)$ time (compute $\mathbf{X}\mathbf{v}^t$ first, then $\mathbf{X}^T(\mathbf{X}\mathbf{v}^t)$)
- This gives the singular vector corresponding to the largest singular value
- Subsequent singular vectors obtained by choosing $\mathbf{v}^0$ orthogonal to previously identified singular vectors (this needs to be done at each iteration to avoid numerical errors creeping in)

Algorithm for Finding PCs (when D $\gg$ N)

Constructing the matrix $\mathbf{X}\mathbf{X}^T$ takes time $O(N^2D)$

Eigenvectors of $\mathbf{X}\mathbf{X}^T$ can be computed in time $O(N^3)$

The eigenvectors give the 'left' singular vectors, $\mathbf{u}_i$ of $\mathbf{X}$

To obtain $\mathbf{v}_i$, we use the fact that $\mathbf{v}_i = \sigma^{-1}\mathbf{X}^T\mathbf{u}_i$

Iterative method can be used directly as in the case when $N > D$

Revisiting PCA's Reconstruction Error

We have thin SVD: $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$

Let $\mathbf{V}_k$ be the matrix containing first k columns of $\mathbf{V}$

Projection onto k PCs: $\mathbf{Z} = \mathbf{X}\mathbf{V}_k = \mathbf{U}_k\mathbf{\Sigma}_k$, where $\mathbf{U}_k$ is the matrix of the first k columns of $\mathbf{U}$ and $\mathbf{\Sigma}_k$ is the $k \times k$ diagonal submatrix for $\mathbf{\Sigma}$ of the top k singular values

Reconstruction: $\tilde{\mathbf{X}} = \mathbf{Z}\mathbf{V}_k^T = \mathbf{U}_k\mathbf{\Sigma}_k\mathbf{V}_k^T$

Reconstruction error $= \sum_{i=1}^{N} \|\mathbf{x}_i - \mathbf{V}_k\mathbf{V}_k^T\mathbf{x}_i\|^2 = \sum_{j=k+1}^{D} \sigma_j^2$

This follows from the following calculations:

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \sum_{j=1}^{D} \sigma_j \mathbf{u}_j \mathbf{v}_j^T \qquad \tilde{\mathbf{X}} = \mathbf{U}_k\mathbf{\Sigma}_k\mathbf{V}_k^T = \sum_{j=1}^{k} \sigma_j \mathbf{u}_j \mathbf{v}_j^T$$

$$\|\mathbf{X} - \tilde{\mathbf{X}}\|_F = \sum_{j=k+1}^{D} \sigma_j^2$$

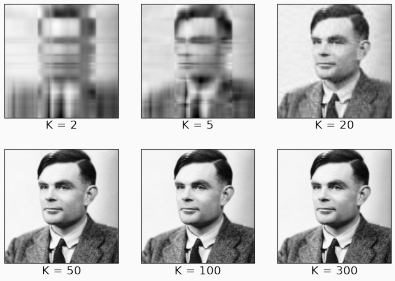
Reconstruction of an Image using PCA

Each image has m ($\approx 10k - 100k$) pixels and represented as a vector $\mathbf{x} \in \mathbb{R}^m$

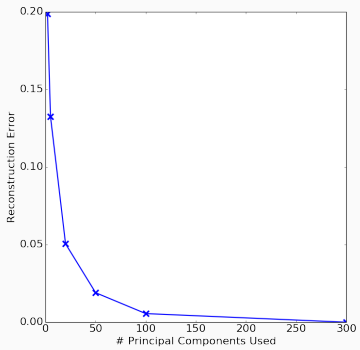
$x_1, \ldots, x_m$ represent the intensity of pixels in the image

Yet accurate representation obtained using $k \approx$ 10-100 PCs

Top k left singular vectors $\mathbf{u}_1, \ldots, \mathbf{u}_k$ called eigenfaces



How Many Principal Components to Pick?



Look for an 'elbow' in the curve of reconstruction error vs # PCs

How Many Principal Components to Pick?

If SVD is computed, fix relative error threshold $0 \leq t \leq 1$ and choose k such that

$$\frac{\|\mathbf{X} - \mathbf{X}_k\|_F^2}{\|\mathbf{X}\|_F^2} = \frac{\sigma_{k+1}^2 + \cdots + \sigma_r^2}{\sigma_1^2 + \cdots + \sigma_r^2} \leq t$$

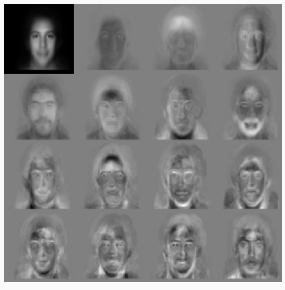
Recall $\|\cdot\|_F$ is the Frobenius norm.

Application: Eigenfaces

Eigenfaces = popular application of PCA for face detection and recognition

- Face detection: Identify faces in a given image
- Face Recognition: Classification (or search) problem to identify a certain person

Application: Eigenfaces



PCA on a dataset of face images. Each principal component can be thought of as being an 'element' of a face.

Source: http://vismod.media.mit.edu/vismod/demos/facerec/basic.html

Application: Eigenfaces

Detection: Each patch of the image can be checked to identify whether there is a face in it

Recognition: Map all faces in terms of their principal components. Then use some distance measure (nearest neighbour) on the projections to find faces that are most like the input image.

Why use PCA for face detection?

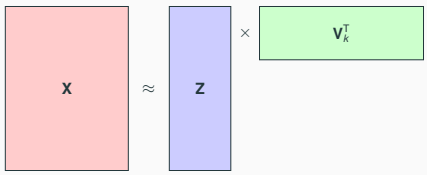
- Even though images can be large, we can use the $D \gg N$ approach to be efficient
- The final model (the PCs) can be quite compact, can fit on cameras, phones
- Works very well given the simplicity of the model

Application: Latent Semantic Analysis

$\mathbf{X}$ is an $N \times D$ matrix, D is the size of dictionary

For document i, $\mathbf{x}_i$ is a vector of keyword counts or frequencies

Reconstruction using k eigenvectors $\mathbf{X} \approx \mathbf{Z}\mathbf{V}_k^\top$, where $\mathbf{Z} = \mathbf{X}\mathbf{V}_k$



Document i is thus represented as a linear combination of the top k principal components with coefficients $\mathbf{z}_i$

$\langle \mathbf{z}_i, \mathbf{z}_j \rangle$ is likely a better notion of similarity than $\langle \mathbf{x}_i, \mathbf{x}_j \rangle$