10.2 Linear Dimensionality Reduction

Machine Learning 1: Foundations

Marius Kloft *(TUK)*

Kaiserslautern, 23–30 June 2020

Linear Dimensionality Reduction: Problem setting

- Given $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$
- find a k-dimensional linear subspace
- such that the data projected onto that space
- is as close to the original data as possible

Formal problem setting

- Given $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$
  - without loss of generality $\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i = 0$
    (we assume that the data has been centered in a pre-processing step)
- find a k-dimensional subspace
  - can write any such subspace as $\mathcal{U}_W := \mathrm{span}(\mathbf{w}_1, \ldots, \mathbf{w}_k)$ where $W := (\mathbf{w}_1, \ldots, \mathbf{w}_k) \in \mathbb{R}^{d \times k}$ is a orthonormal basis ($\mathbf{w}_i \perp \mathbf{w}_j$ for all $i \neq j$ and $\|\mathbf{w}_j\| = 1$)
- such that the data projected onto that space

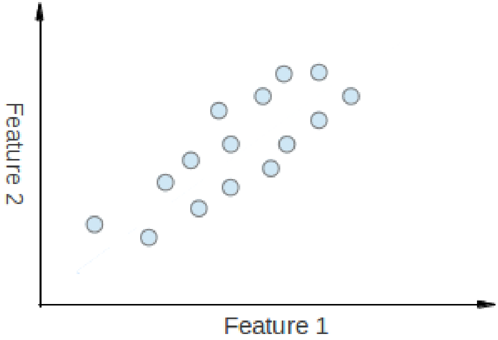
$$\Pi_{\mathcal{U}_W}(\mathbf{x}_i) := \arg\min_{\mathbf{x} \in \mathcal{U}_W} \|\mathbf{x} - \mathbf{x}_i\|^2, \quad i = 1, \ldots, n$$

- is as close to the original data as possible

How to measure “closeness”?

Example

- In the simplest case, we aim to find a $k = 1$-dimensional subspace
$$\mathcal{U}_W := \text{span}(\mathbf{w}_1) = \{c\mathbf{w}_1 : c \in \mathbb{R}\}$$
  - This is just a line!



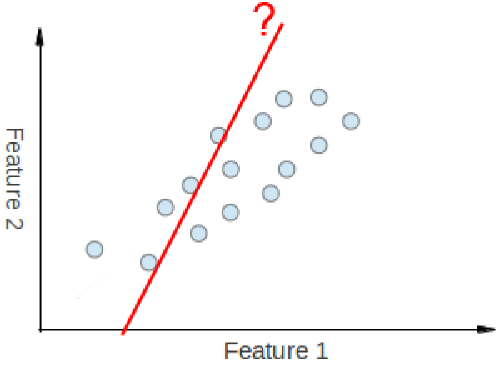
Which line?

Example

- In the simplest case, we aim to find a $k = 1$-dimensional subspace

$$\mathcal{U}_W := \text{span}(\mathbf{w}_1) = \{c\mathbf{w}_1 : c \in \mathbb{R}\}$$

  - This is just a line!



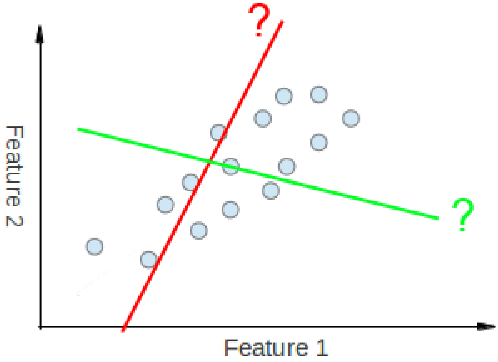
Which line?

Example

- In the simplest case, we aim to find a $k = 1$-dimensional subspace

$$\mathcal{U}_W := \text{span}(\mathbf{w}_1) = \{c\mathbf{w}_1 : c \in \mathbb{R}\}$$

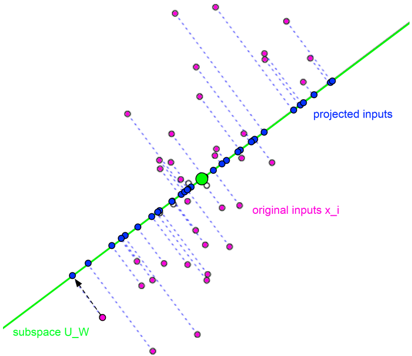
  - This is just a line!



Which line?

PCA Principle

Pick the subspace $\mathcal{U}_W$ with minimal average squared error $\frac{1}{n}\sum_{i=1}^{n} \|\mathbf{x}_i - \Pi_{\mathcal{U}_W}(\mathbf{x}_i)\|^2$.



Next, we compute $\Pi_{\mathcal{U}_W}(\mathbf{x}_i)$ explicitly.

Computing $\Pi_{\mathcal{U}_W}(\mathbf{x}_i)$ (1/2)

In the simple case $k = 1$, we have $\mathcal{U}_W := \mathrm{span}(\mathbf{w}_1)$ with $\|\mathbf{w}_1\| = 1$, we have

$$\Pi_{\mathcal{U}_W}(\mathbf{x}_i) \overset{\text{def.}}{=} \arg\min_{\mathbf{x}\in\mathcal{U}_W} \|\mathbf{x} - \mathbf{x}_i\|^2 = \arg\min_{\mathbf{x}\in\mathbb{R}^d:\, \exists\lambda\in\mathbb{R} \text{ with } \mathbf{x}=\lambda\mathbf{w}_1} \|\mathbf{x} - \mathbf{x}_i\|^2 .$$

Setting the derivative of $f(\lambda) := \|\lambda\mathbf{w}_1 - \mathbf{x}_i\|^2$ to zero reveals the optimum $\lambda^* = \mathbf{w}_1^\top\mathbf{x}_i$. Thus:

$$\Pi_{\mathcal{U}_W}(\mathbf{x}_i) = \lambda^*\mathbf{w}_1 = \mathbf{w}_1\lambda^* = \mathbf{w}_1\mathbf{w}_1^\top\mathbf{x}_i .$$

Computing $\Pi_{\mathcal{U}_W}(\mathbf{x}_i)$ (2/2)

In the general case $k \geq 1$, we have $\mathcal{U}_W := \text{span}(\mathbf{w}_1 \ldots, \mathbf{w}_k)$ with $\mathbf{w}_i \perp \mathbf{w}_j$ for all $i \neq j$ and $\|\mathbf{w}_1\| = \cdots = \|\mathbf{w}_k\| = 1$. Thus:

$$\begin{aligned}
\Pi_{\mathcal{U}_W}(\mathbf{x}_i) \quad &\stackrel{\text{def.}}{=} \quad \underset{\mathbf{x} \in \mathcal{U}_W}{\arg\min} \|\mathbf{x} - \mathbf{x}_i\|^2 \\
&= \quad \underset{\substack{\mathbf{x} \in \mathbb{R}^d \,:\, \exists \boldsymbol{\lambda} \in \mathbb{R}^k \\ \text{with } \mathbf{x} = \sum_{j=1}^k \lambda_j \mathbf{w}_j}}{\arg\min} \|\mathbf{x} - \mathbf{x}_i\|^2
\end{aligned}$$

Setting the derivative of $f(\boldsymbol{\lambda}) := \left\| \sum_{j=1}^k \lambda_j \mathbf{w}_j - \mathbf{x}_i \right\|^2$ to zero reveals the optimum $\lambda_j^* = \mathbf{w}_j^\top \mathbf{x}_i$ for all $j = 1, \ldots, k$. Thus:

$$\Pi_{\mathcal{U}_W}(\mathbf{x}_i) \;=\; \sum_{j=1}^k \lambda_j^* \mathbf{w}_j \;=\; \sum_{j=1}^k \mathbf{w}_j \lambda_j^* \;=\; \sum_{j=1}^k \mathbf{w}_j \mathbf{w}_j^\top \mathbf{x}_i \;=\; W W^\top \mathbf{x}_i.$$

Consequences

From $\Pi_{\mathcal{U}_W}(\mathbf{x}_i) = \sum_{j=1}^{k} \mathbf{w}_j \mathbf{w}_j^\top \mathbf{x}_i = WW^\top \mathbf{x}_i$ it follows:

- The projected data is

$$\begin{aligned}\hat{X} &:= (\Pi_{\mathcal{U}_W}(\mathbf{x}_1), \ldots, \Pi_{\mathcal{U}_W}(\mathbf{x}_n)) \\ &= \left(WW^\top \mathbf{x}_1, \ldots, WW^\top \mathbf{x}_n\right) \\ &= WW^\top X.\end{aligned}$$

- With respect to the basis $\{\mathbf{w}_1, \ldots, \mathbf{w}_k\}$ the coordinates of the projection of a point $\mathbf{x}_i$ are:

$$\tilde{\mathbf{x}}_i := \begin{pmatrix} \mathbf{w}_1^\top \mathbf{x}_i \\ \vdots \\ \mathbf{w}_k^\top \mathbf{x}_i \end{pmatrix} = W^\top \mathbf{x}_i.$$

  Thus the projected data in the coordinate system with basis $\{\mathbf{w}_1, \ldots, \mathbf{w}_k\}$ is:

$$\widetilde{X} := (\tilde{\mathbf{x}}_1, \ldots, \widetilde{\mathbf{x}}_n) = \left(W^\top \mathbf{x}_1, \ldots, W^\top \mathbf{x}_n\right) = W^\top X.$$

Principal component analysis

Thus our PCA principle, $\min \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{x}_i - \Pi_{\mathcal{U}_W}(\mathbf{x}_i)\|^2$, becomes:

Principal Component Analysis (PCA)

Let $k \in \{1, \ldots, d\}$ be the reduced dimensionality, and let the data matrix X be centered. Then **principal component analysis (PCA)** is given by:

$$W_* := \arg\min_{W \in \mathbb{R}^{d\times k}} \quad \sum_{i=1}^{n} \left\| x_i - WW^\top \mathbf{x}_i \right\|^2$$
$$\text{s.t.} \quad \mathbf{w}_i \perp \mathbf{w}_j \text{ for all } i \neq j \text{ and } \|\mathbf{w}_1\| = \cdots = \|\mathbf{w}_k\| = 1$$

The dimensionality-reduced data is:

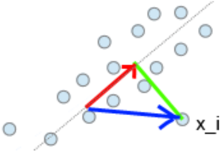
- in original coord. system: $\hat{X} := W_* {W_*}^\top X \in \mathbb{R}^{d\times n}$
- in k-dim. coordinate system*: $\widetilde{X} := {W_*}^\top X \in \mathbb{R}^{k\times n}$

How to solve the PCA optimization problem?

* Basis: $W = (\mathbf{w}_1, \ldots, \mathbf{w}_k)$.

Analysis

- Note that the PCA objective minimizes $\|\mathbf{x}_i - \Pi_{\mathcal{U}_W}(\mathbf{x}_i)\|^2$
- By the Pythagorean theorem,



$$\|\Pi_{\mathcal{U}_W}(\mathbf{x}_i)\|^2 + \|\mathbf{x}_i - \Pi_{\mathcal{U}_W}(\mathbf{x}_i)\|^2 = \|\mathbf{x}_i\|^2$$

- Thus:

$$\begin{aligned} &\arg\min_{W \in \mathbb{R}^{d \times k}} \textstyle\sum_{i=1}^n \|\mathbf{x}_i - \Pi_{\mathcal{U}_W}(\mathbf{x}_i)\|^2 \\ = \; &\arg\max_{W \in \mathbb{R}^{d \times k}} \textstyle\sum_{i=1}^n \|\Pi_{\mathcal{U}_W}(\mathbf{x}_i)\|^2 \end{aligned}$$

- Furthermore:

$$\sum_{i=1}^n \|\Pi_{\mathcal{U}_W}(\mathbf{x}_i)\|^2 = \sum_{i=1}^n \mathbf{x}_i^\top \underbrace{W^\top W}_{=I} \underbrace{WW^\top}_{=\sum_{j=1}^k \mathbf{w}_j \mathbf{w}_j^\top} \mathbf{x}_i$$

$$= \sum_{j=1}^k \mathbf{w}_j^\top \sum_{i=1}^n \mathbf{x}_i \mathbf{x}_i^\top \mathbf{w}_j = \sum_{j=1}^k \mathbf{w}_j^\top XX^\top \mathbf{w}_j.$$

Result of Derivation

Theorem

PCA can equivalently be written as

$$W_* := \arg\max_{W \in \mathbb{R}^{d\times k}} \quad \sum_{j=1}^{k} \mathbf{w}_j^\top XX^\top \mathbf{w}_j$$

$$\text{s.t.} \quad \mathbf{w}_i \perp \mathbf{w}_j \text{ for all } i \neq j \text{ and } \|\mathbf{w}_1\| = \cdots = \|\mathbf{w}_k\| = 1$$

In the special case $k = 1$: $\quad \mathbf{w}^* = \arg\max_{\mathbf{w}\in\mathbb{R}^d : \|\mathbf{w}\|=1} \mathbf{w}^\top XX^\top \mathbf{w}$.

The matrix $S_n := XX^\top$ is called “scatter matrix”

- Relation to sample covariance matrix $\hat{\Sigma}_n := \frac{1}{n}XX^\top$ (see ML2): $\quad S_n = n\widehat{\Sigma_n}$

One can show:

Theorem

The optimal PCA solution $W_* = (\mathbf{w}_1^*, \ldots, \mathbf{w}_k^*)$ is given by the k largest eigenvectors of the (centered) scatter matrix S_n.

Proof

Consider $k = 1$. The proof for $k > 1$ is analogue.

By the Lagrangian duality theorem (L), we have

$$\max_{\mathbf{w} \in \mathbb{R}^d : \|\mathbf{w}\|^2 = 1} \quad \mathbf{w}^\top X X^\top \mathbf{w} \tag{1}$$

$$\overset{(L)}{=} \quad \min_{\lambda \in \mathbb{R}} \quad \max_{\mathbf{w} \in \mathbb{R}^d} \quad \underbrace{\mathbf{w}^\top X X^\top \mathbf{w} - \lambda \left(\|\mathbf{w}\| \right)^2}_{=: \mathcal{L}(\mathbf{w})}$$

In the optimal point, we have:

$$0 \;=\; \nabla_{\mathbf{w}} \mathcal{L}(\mathbf{w}) \;=\; X X^\top \mathbf{w} - \lambda \mathbf{w},$$

which is equivalent to:

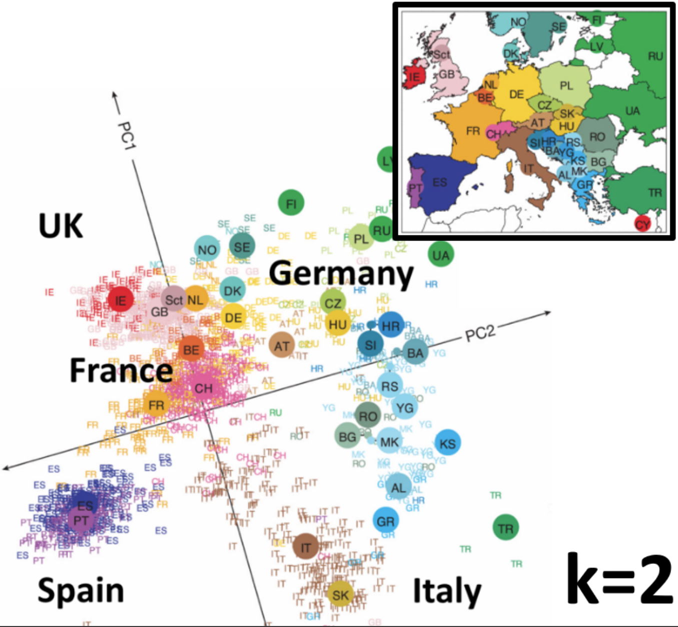
$$X X^\top \mathbf{w} \;=\; \lambda \mathbf{w}$$

This means the optimal $\mathbf{w}$ is an eigenvector of $XX^\top$. The objective in (1) is maximized for the largest eigenvalue λ. $\square$

PCA Algorithm

function PCA(parameter k, inputs $X = (\mathbf{x}_1, \ldots, \mathbf{x}_n) \in \mathbb{R}^{d \times n}$)
    compute sample mean $\hat{\boldsymbol{\mu}} := \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i$
    center each input: $\mathbf{x}_i \leftarrow \mathbf{x}_i - \hat{\boldsymbol{\mu}}$ and update X
    compute scatter matrix $S_n := XX^\top$
    compute k largest eigenvalues of S_n with eigenvectors $W = (\mathbf{w}_1, \ldots, \mathbf{w}_k)$
        (e.g., in MATLAB: [foo,W] = eig(S_n))
    return dim.-reduced data: $\widetilde{X} = W^\top X \in \mathbb{R}^{k \times n}$ and $\hat{X} = WW^\top X \in \mathbb{R}^{d \times n}$
end function

Example: Genomes of Europeans

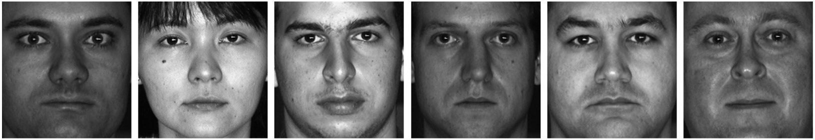


Example: Eigenfaces

A popular method is to apply PCA on portrait images

- the resulting eigenvectors are called **Eigenfaces**

Example images from the CMU PIE dataset:

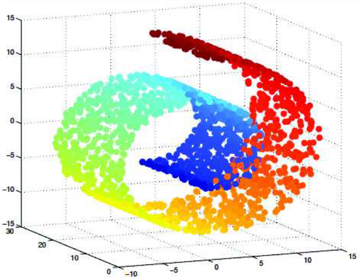


Mean face:

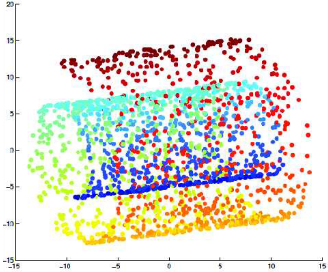
Top two eigenfaces:

What problems might you run into in practice?

Example: Swiss Role



swiss-role data (3-D)

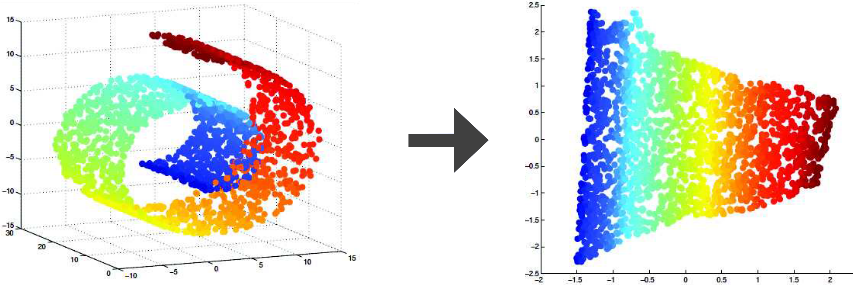


same data after PCA (2-D)

Why does PCA fail here?

PCA is a **linear** method and fails for **non-linear** data.

Better Solution:



Thus: need for **non-linear** methods for dimensionality reduction

The above plot has been produced by such a method:

- **kernel PCA**