Lecture 21

Do Well!

20 min

PCA

Statistical learning
- CV
- Regularization

Predictors
Training
Architectures
Unsupervised
Stat Learning ←
Learning & people

Lecture Notes VII – Principal Component Analysis

Marina Meilă
mmp@uwaterloo.ca

With Thanks to Pascal Poupart & Gautam Kamath
Cheriton School of Computer Science
University of Waterloo

March 19, 2026

3. Representing X, x^i

$X = U\sqrt{\tilde{\Sigma}}\,V^T$ $\quad n \times D \quad$ SVD(X)

$u^i_j = u_{ij}\tilde{\sigma}_j$

$\Downarrow$

$x^i = v_1 \cdot u^i_1 + v_2 \cdot u^i_2 + \cdots + v_D \cdot u^i_D$ $\rightarrow$ scalar coefficients

in basis V

$U = [u_{ij}]_{\substack{i=1:n \\ j=1:D}}$

$x^i \longleftrightarrow (u^i_1 \cdots u^i_D) = \tilde{x}^i$

$X = \begin{bmatrix} (x^1)^T \\ \cdots \\ (x^n)^T \end{bmatrix}$ $\quad n \times D$

$\tilde{\sigma}_1 \geq \tilde{\sigma}_2 \geq \cdots \tilde{\sigma}_d \geq \cdots \tilde{\sigma}_D$

$v_1 \quad v_2 \quad v_d \quad v_d$

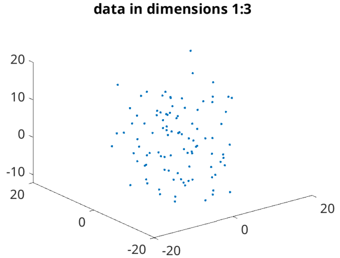
Principal e-values vectors

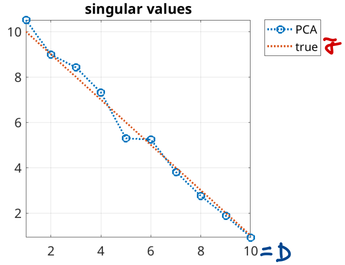
1 $\mathrm{Var}(X) = \frac{1}{n} X^T X = = \frac{1}{n} V \tilde{\Sigma} V^T$ $\quad D \times D$

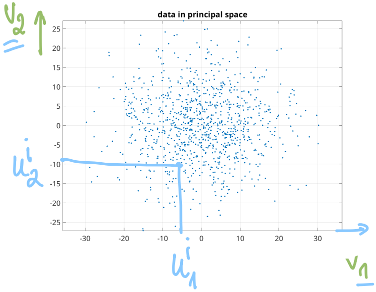
2 $G = XX^T = U\tilde{\Sigma}U^T$ $\quad n \times n$

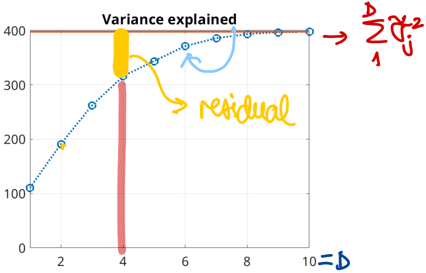
Example – Gaussian data

$D=10$









$\sum_{j=1}^{4} \tilde{\sigma}_j^2$

PCA is "optimal" basis

$$x^i = \underbrace{v_1 \cdot u_1^i + v_2 \cdot u_2^i}_{d} + \underbrace{\cdots + v_D \cdot u_D^i}_{\text{residual}}$$

$$\tilde{x}^i = \sum_{j=1}^{d} u_j^i v_j$$

projected on B

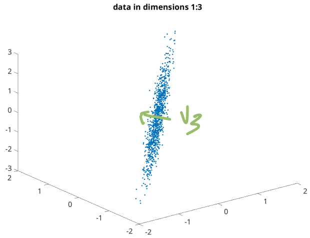
$$MSE = \mathcal{L}_{LS}(B) = \frac{1}{n} \sum_{i=1}^{n} \left(\|x^i\|^2 - \|\tilde{x}^i\|^2 \right)$$

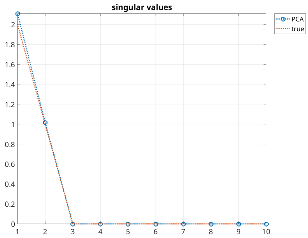
some subspace

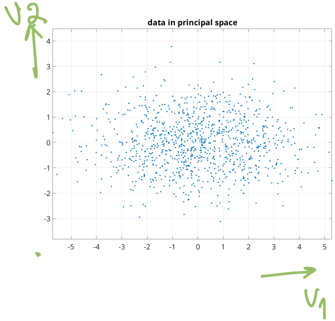
$$\mathcal{L}_{LS}(V) = \boxed{\frac{1}{n} \sum_{j=d+1}^{D} \sigma_j^2} = \min_B \mathcal{L}_{LS}(B)$$

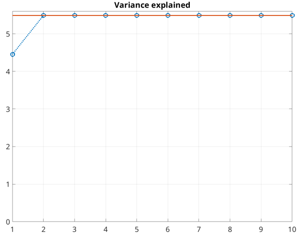
PCA is optimal

Example – Gaussian data 2D



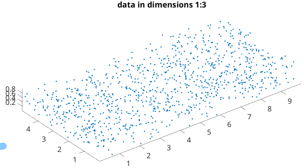


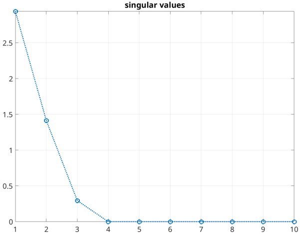




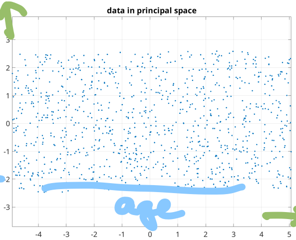
Example – Brick

d=3 sufficient

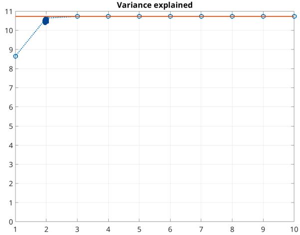




plane in math

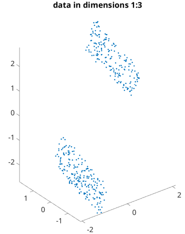


age



Example – clusters

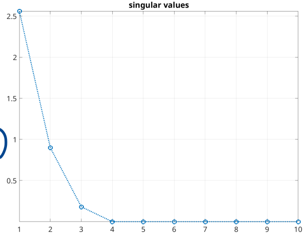
d=3

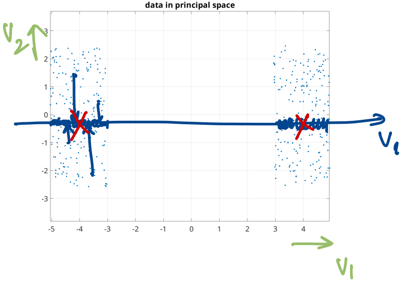


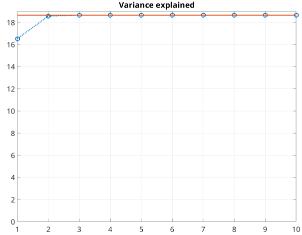
K=2 clusters

project on

PCA(K-1)







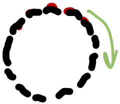
PCA Summary

PCAnalysis

- Reduces data dimensiont from D to d
- Linear operation (projection)
- “Optimal” linear method to reduce dimension
- Can discover if data is low-dimensional
- For clustering – recommended pre-processing: PCA in $K-1$ dimensions
- Limitation: fails to discover non-linear low dimensional structure

→ manifold learning/ non-lin dim red/ embedding

- Understand, explore, interpret data



time of day

Model Selection

- select K for clustering
- select K for K-NN
- compare n.n. architectures
- — " — different predictors {DT, nn, K-NN, lin regr, ...}

"Alg": Given $\mathcal{D}$

1. Train $f_1, f_2, \dots f_M$ predictors on $\mathcal{D}$
2. Select $f^* = \text{"best"}\{f_{1:M}\}$

Model sel

BIC only for Max Likelihood, cheap!! score

CV any models, any Loss, more computation!! score

CV cross-Validation

Assume $\mathcal{D}'$ available, $|\mathcal{D}'| = n'$ from same distribution

1. ... $f_{1:M}$ trained on $\mathcal{D}$ ← training data
2. calculate $Loss(f_m; \mathcal{D}')$ = validation loss

   validation data

3. $f^* = \underset{m=1:M}{\text{argmin}}\, Loss(f_m; \mathcal{D}')$

$n' = ?$ → 1 value

$n' \leq 1000$

n → p parameters

Plenty of data

n ~~$\gg$~~ 1000

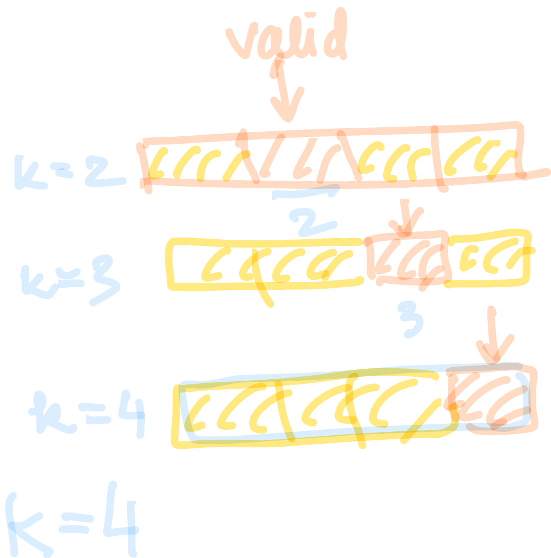
K-fold CV

1. Partition $\mathcal{D} = \mathcal{D}_1 \cup \dots \cup \mathcal{D}_K$ disjoint, $|\mathcal{D}_1| \approx \frac{n}{K}$
2. for $k = 1 : K$
   - train $f^{(k)}_{1:M}$ on $\mathcal{D}_{-k}$
   - compute $L(f^{(k)}_m ; \mathcal{D}_k) = L^{(k)}_m$

$$\underbrace{\mathcal{D}_{-k}}_{\text{train}} = \mathcal{D} \setminus \underbrace{\mathcal{D}_k}_{\text{validate}}$$

$$\bar{L}_m = \frac{1}{K} \sum_k L^{(k)}_m$$

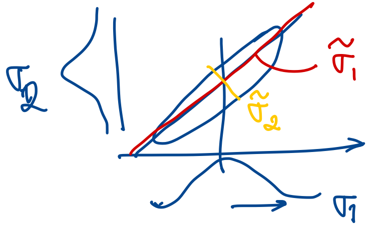
$$m^* = \underset{m}{\arg\min}\ \bar{L}_m$$



$D = 2$

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ & \sigma_2^2 \end{bmatrix}$$

$Var(x_1)$



$\sigma_1^2 + \sigma_2^2 = \tilde{\sigma}_1^2 + \tilde{\sigma}_2^2$

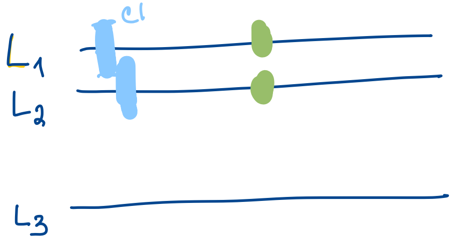
(Questions after class)

$L(f(X))$ r.v. $\rightarrow$ $\underset{\text{mean}}{\text{var}} = \sigma^2$

$L = E_{px}[L(f(x), y)]$

$\hat{L} = \frac{1}{n'} \Sigma L(f(x), y)$

$Var\,\hat{L} = \frac{\sigma^2}{n'}$, $std = \frac{\sigma}{\sqrt{n'}}$



n' small

n' large