Machine Learning: Principal Component Analysis (PCA)

by Elena Battini Sönmez
İstanbul Bilgi University

Dimensionality Reduction:

- High dimension is challenging and redundant
- Idea1: Reduce the dimensionality by feature combination

    Example: $x=[x_1,x_2,x_3,x_4]'$, $f(x)=y$, $y=[x_1+x_2,x_3+x_4]$

Ideally, the new vector y should retain all discriminant information of x

- The best f(x) is most likely a non-linear function, for simplicity, we assume it is a linear mapping, which can be written as a matrix:

$$W \cdot x = y, W \in \Re^{k \times d}, x \in \Re^{d \times 1}, y \in \Re^{k \times 1}, k < d$$

Dimensionality Reduction:

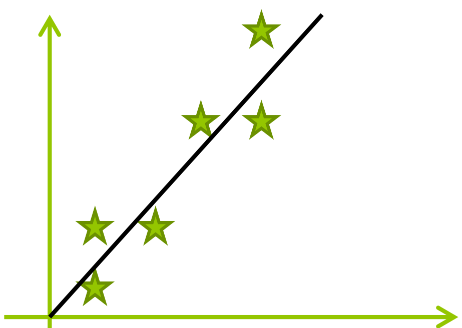
- Principal Component Analysis (PCA)
- Fisher Linear Discriminant

Principal Component Analysis:

Main idea: to seek for the most accurate data representation in a lower dimensional space

Example in 2D: data set={(2,1)(2,3)(4,3)(5,6)(7,6)(7,9)}, card(dataset)=6

bad projection line-large error

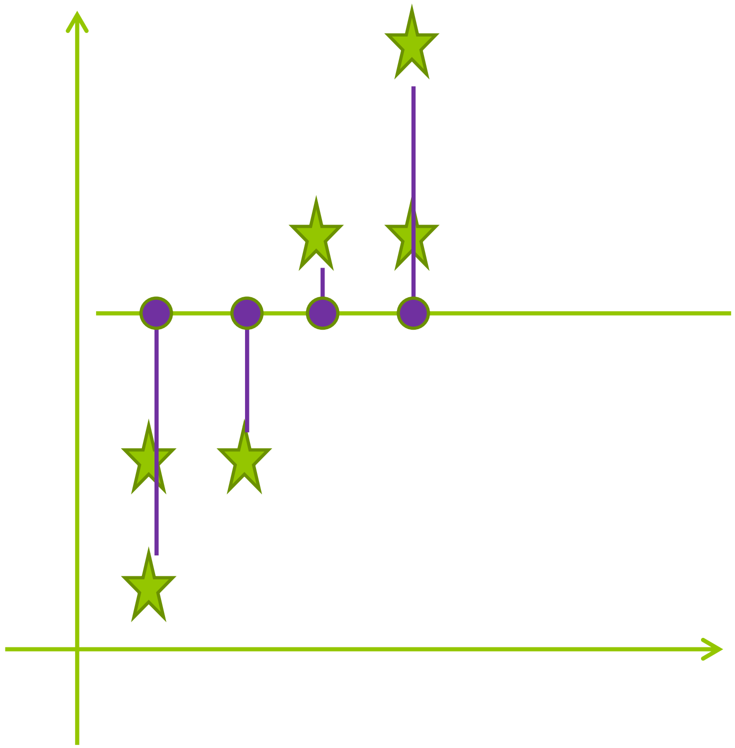


good projection line

- Notice that the best projection line is the one having maximum variance

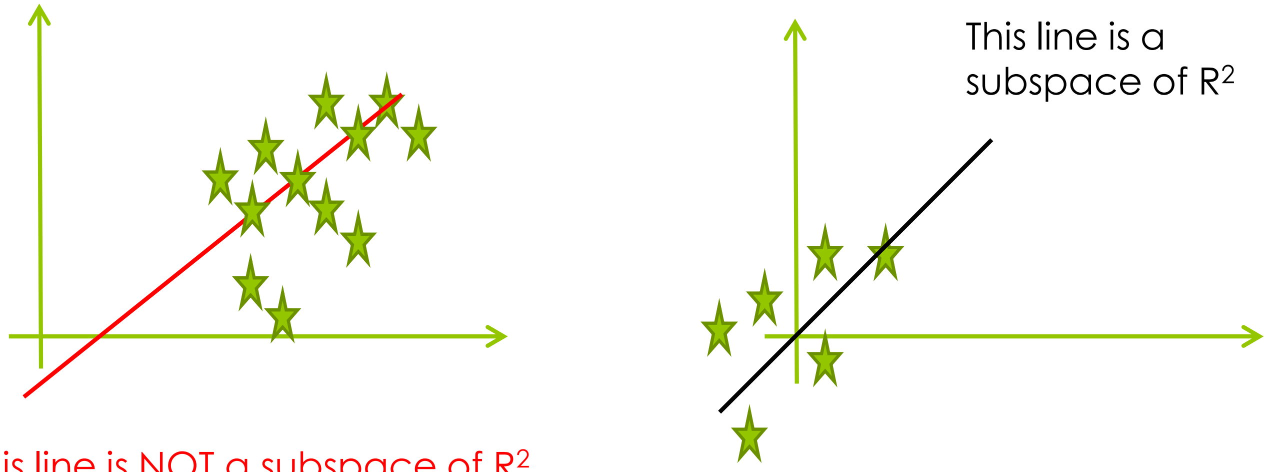
Projections and Errors:

- To project a point into a line we draw the perpendicular line from that point into the line
- sample's error: distance between original point, ★, and the projected one, ●
- The total error is the sum of samples' error



PCA calculation: Important point

- Remeber that a subspace must contain the zero vector



This line is NOT a subspace of R^2

PCA Calculation:

- Before PCA subtract the sample mean from the data:

$$x - \frac{1}{n}\sum_{i=1}^{n} x_i = x - \mu_i$$

- We want to find the most accurate representation of data in some subspace W which has dimension k<d
- Let $\{e_1, e_2, \ldots, e_k\}$ be an orthonormal basis for W, vector $x_1 \in W,\ x_1 = \sum_{i=1}^{k} \alpha_{1,i} e_i$
- The error in this representation: $error_1 = \left\| x_1 - \sum_{i=1}^{k} \alpha_{1,i} \cdot e_i \right\|^2$

Obs: error_1 is the length of the violet line (2 slides before)

PCA Calculation:

- The total error is the sum over all errors, having n data points x_j:

$$J(e_1, e_2, \cdots, e_k, \alpha_{11}, \alpha_{12}, \cdots, \alpha_{nk}) = \sum_{j=1}^{n} \left\| x_j - \sum_{i=1}^{k} \alpha_{ji} e_i \right\|^2$$

- Goal: how to minimize J(.)?

- Remember:

$$(a-b)^2 = a^2 - 2ab + b^2$$

PCA Calculation:

- Let us simplify J(.) first:

$$J(e_1, e_2, \cdots, e_k, \alpha_{11}, \alpha_{12}, \cdots, \alpha_{nk}) = \sum_{j=1}^{n} \left\| x_j - \sum_{i=1}^{k} \alpha_{ji} e_i \right\|^2 =$$

$$= \sum_{i=1}^{n} \left\| x_i \right\|^2 - 2\sum_{j=1}^{n} x_j^t \left(\sum_{i=1}^{k} \alpha_{ji} e_i \right) + \sum_{j=1}^{n} \sum_{i=1}^{k} \alpha_{ji}^2 \left\| e_i \right\|^2 =$$

$$= \sum_{i=1}^{n} \left\| x_i \right\|^2 - 2\sum_{j=1}^{n} \sum_{i=1}^{k} \alpha_{ji} x_j^t e_i + \sum_{j=1}^{n} \sum_{i=1}^{k} \alpha_{ji}^2$$

- Remember:

d(ax)=a and dx²=2x

PCA Calculation:

$$J(e_1,e_2,\cdots,e_k,\alpha_{11},\alpha_{12},\cdots,\alpha_{nk}) = \sum_{i=1}^{n}\|x_i\|^2 - 2\sum_{j=1}^{n}\sum_{i=1}^{k}\alpha_{ji}x_j^t e_i + \sum_{j=1}^{n}\sum_{i=1}^{k}\alpha_{ji}^2$$

- Take the partial derivatives with respect to : α_{ml}

$$\frac{\partial}{\partial \alpha_{ml}} J(e_1,e_2,\cdots,e_k,\alpha_{11},\alpha_{12},\cdots,\alpha_{nk}) = -2x_j^t e_l + 2\alpha_{ml}$$

- Thus the optimal value for $\alpha_{ml} = x_m^t e_l$
- Plug the optimal value into J(.):

$$J(e_1,e_2,\cdots,e_k) = \sum_{i=1}^{n}\|x_i\|^2 - 2\sum_{j=1}^{n}\sum_{i=1}^{k}(x_j^t e_i)x_j^t e_i + \sum_{j=1}^{n}\sum_{i=1}^{k}(x_j^t e_i)^2 =$$

$$\sum_{i=1}^{n}\|x_i\|^2 - \sum_{j=1}^{n}\sum_{i=1}^{k}(x_j^t e_i)^2$$

PCA Calculation:

$$J(e_1, e_2, \cdots, e_k) = \sum_{i=1}^{n} \|x_i\|^2 - \sum_{j=1}^{n} \sum_{i=1}^{k} (x_j^t e_i)^2$$

- Rewrite J(.) using: $(a^{\dagger}b)^2 = (a^{\dagger}b)^{\dagger}(a^{\dagger}b) = (b^{\dagger}a)(a^{\dagger}b) = b^{\dagger}(aa^{\dagger})b$

$$J(e_1, e_2, \cdots, e_k) = \sum_{i=1}^{n} \|x_i\|^2 - \sum_{i=1}^{k} e_i^t \left(\sum_{j=1}^{n} \left(x_j x_j^t \right) \right) e_i$$

Where $S = \sum_{j=1}^{n} x_j x_j^t$ is the scatter matrix

Notice that the scatter matrix is equal to (n-1) time the covarianze matrix!!!

PCA Calculation:

$$J(e_1, e_2, \cdots, e_k) = \sum_{i=1}^{n} \|x_i\|^2 - \sum_{i=1}^{k} e_i^t \left(\sum_{j=1}^{n} \left(x_j x_j^t \right) \right) e_i = \sum_{i=1}^{n} \|x_i\|^2 - \sum_{i=1}^{k} e_i^t S e_i$$

- Minimize J(.) is equivalent to maximize: $\sum_{i=1}^{k} e_i^t S e_i$
- We want also to enforce the constraints: $e_i^t e_i = 1$
- Using the Lagrange multipliers method, we can write:

$$u(e_1, e_2, \cdots, e_k) = \sum_{i=1}^{k} e_i^t S e_i - \sum_{j=1}^{k} \lambda_j (e_j^t e_j - 1)$$

PCA Calculation:

- It can be shown that:

$$\frac{d}{dx}(x^t Ax) = 2Ax \text{ and } \frac{d}{dx}(x^t x) = 2x$$

$$u(e_1, e_2, \cdots, e_k) = \sum_{i=1}^{k} e_i^t S e_i - \sum_{j=1}^{k} \lambda_j (e_j^t e_j - 1)$$

$$\frac{\partial}{\partial e_m} u(e_1, e_2, \cdots, e_k) = 2Se_m - 2\lambda_m e_m = 0 \quad Se_m = \lambda_m e_m$$

Therefore, e_m is the eigenvector of the scatter matrix S!!!

Replacing: "Se_i" with "$\lambda_i e_i$" into eq. J(.) { previous slide}

Constant term

$$J(e_1, e_2, \cdots, e_k) = \sum_{i=1}^{n} \|x_i\|^2 - \sum_{i=1}^{k} e_i^t S e_i = \sum_{i=1}^{n} \|x_i\|^2 - \sum_{i=1}^{k} \lambda_i \|e_i\|^2 = \sum_{i=1}^{n} \|x_i\|^2 - \sum_{i=1}^{k} \lambda_i$$

Therefore to minimize J take for the basis of W the k biggest engenvectors of S

PCA and data approximation:

- Let $\{e_1, e_2, \ldots, e_d\}$ be all d eigenvectors of the scatter matrix S, sorted from biggest to little
- Obs: we are in d (and not k) dimension!!!
- Without any approximation:

$$x_i = \sum_{j=1}^{d} \alpha_j e_j = \underbrace{\alpha_1 e_1 + \alpha_2 e_2 + \cdots + \alpha_{1k} e_k}_{\text{PCA approximation of } x_i} + \underbrace{\alpha_{k+1} e_{k+1} + \cdots + \alpha_d e_d}_{\text{error of approximation}}$$

- Therefore, PCA uses the k biggest eigenvectors of the scatter matrix of the data in $\Re^d$ to project the data into new dimension k, k<d.

PCA pseudo code:

- Input: D={x1,x2,...,xn} data set of "n" d-dimensional samples
- Center the data: $Cx = x_i - \frac{1}{n}\sum_{i=1}^{n} x_i$
- Compute the scatter matrix: $S = \sum_{i=1}^{n} Cx \cdot Cx', \dim(S) = d \times d$
- Select the k biggest eigenvectors of S: E=[e_1, ..., e_k]
- Down-sample all data: y=$E^{\dagger}$Cx

➡ Obs: dim(E)=d×k, dim(x)=d, dim(y)=k, k<d

Drawbacks of PCA:

- PCA is designed for accurate data representation and not for data classifcation
- It preserves as much variance in data as possible
- It works only if-when the direction of max variance preserves class distinctions ... however the direction of max variance can be useless for classification