Machine Learning

Aula 17 – Redução de dimensionalidade

2021 – Engenharia
Fábio Ayres <fabioja@insper.edu.br>

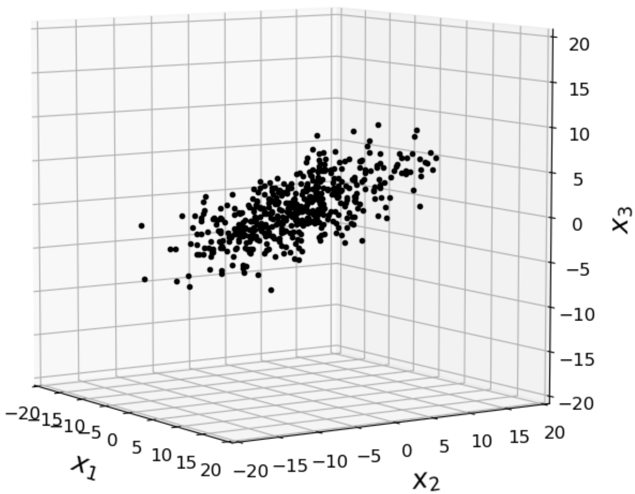
Redução de dimensionalidade

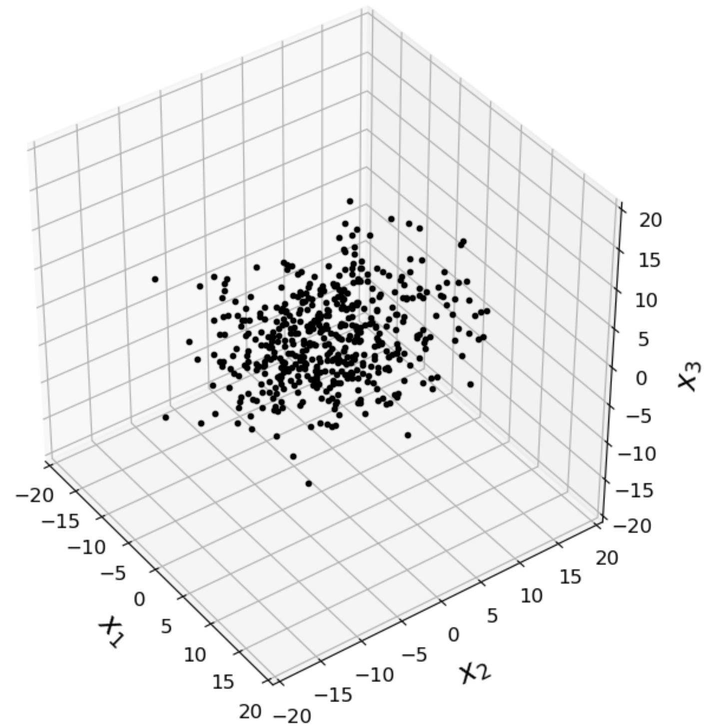
- Como visualizar um dataset de dimensão $n = 200$?

- Será que em um dataset de dimensão $n = 200$ realmente temos tudo isso de informação independente?

- Como "enxugar" um dataset?

Exemplo





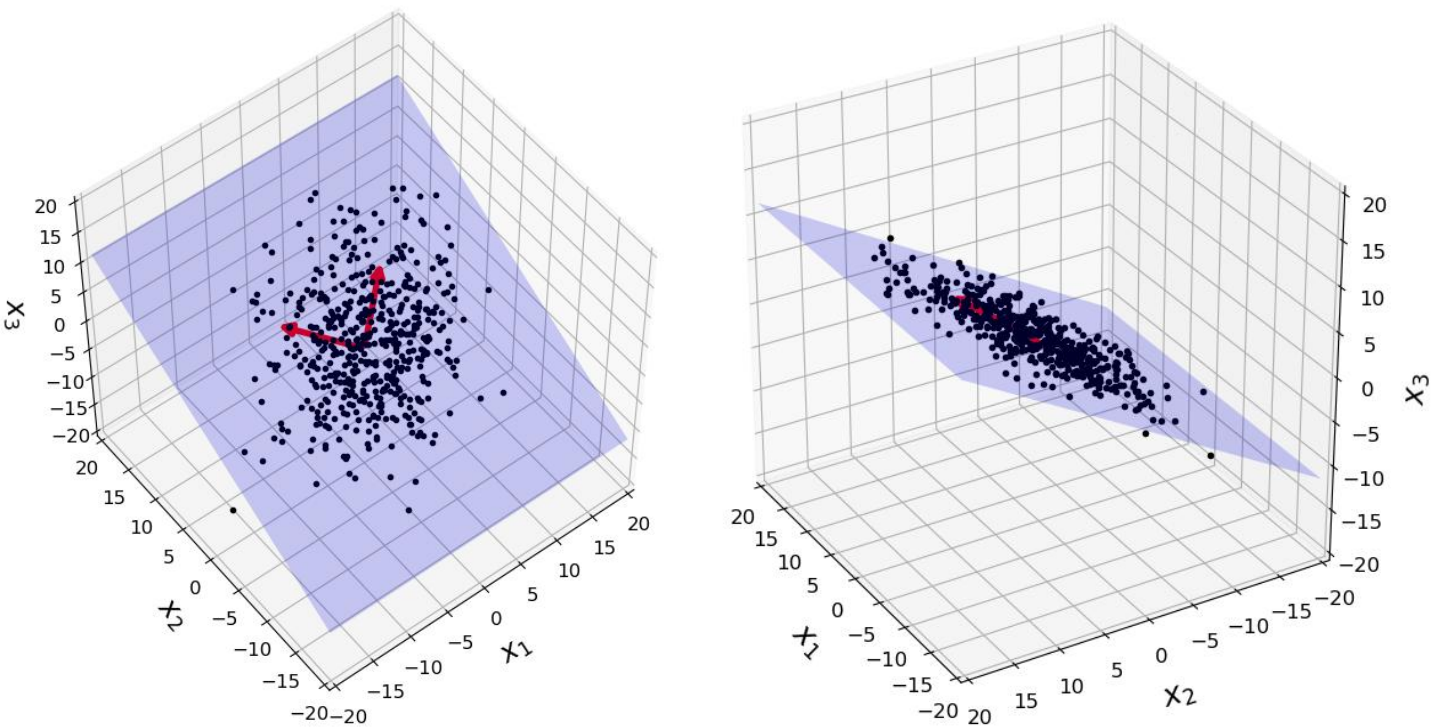
Ideia: projeção

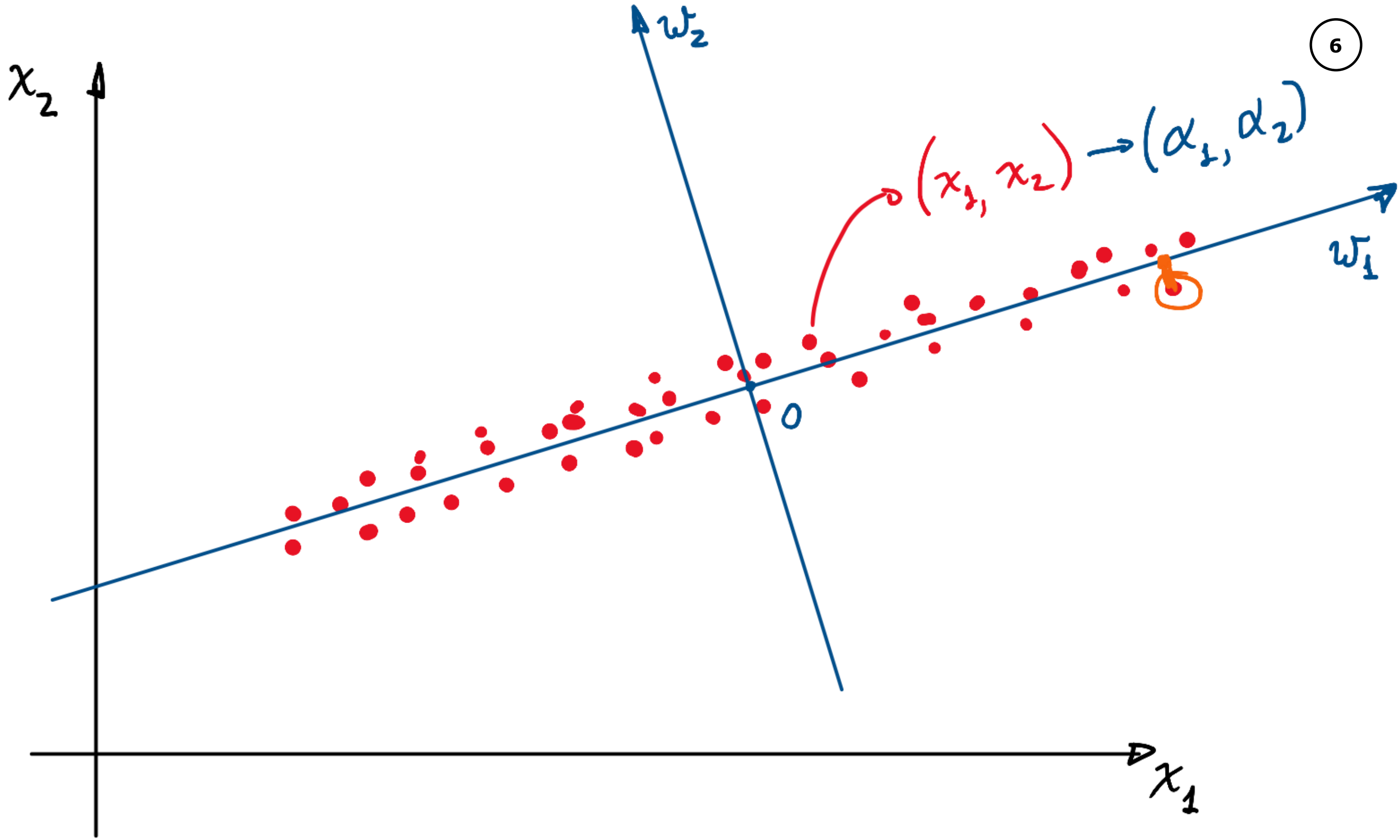
Podemos projetar esses dados em um hiperplano de menor dimensão!

Assim só precisamos guardar:

- A origem do plano
- Os vetores determinando o hiperplano
- As coordenadas de projeção por ponto: $d \ll n$

Exemplo





x_2
w_2
$(x_1, x_2) \rightarrow (\alpha_1, \alpha_2)$
w_1
0
x_1

Aproximações sucessivas

- $d = 0$: não guardo nenhuma coordenada por ponto!
    - Guardo apenas o centroide da nuvem!

- $d > 0$: começo a gerar componentes de projeção por ponto.

i) guardar 1 vez por dataset

$p, w_1, \ldots, w_d$

vetores no espaço original

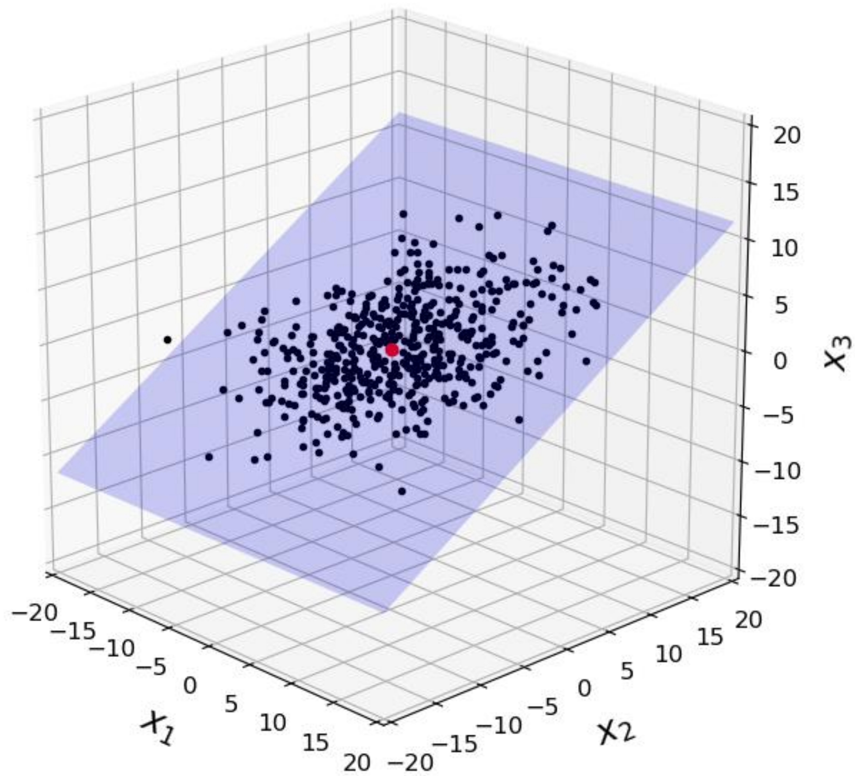
$$\hat{x}_i = \underbrace{p}_{\text{origem}} + \overbrace{\alpha_{i1} w_1 + \alpha_{i2} w_2 + \cdots + \alpha_{id} w_d}^{(\alpha_1^{(i)}, \alpha_2^{(i)}, \ldots, \alpha_d^{(i)})}$$

direções.

$\rightarrow d \ll n$ componentes

aproximação de $x_i = \underbrace{(x_1^{(i)}, x_2^{(i)}, \ldots, x_n^{(i)})}_{n \text{ componentes}}$

Zero-ésima aproximação



Zero-ésima aproximação

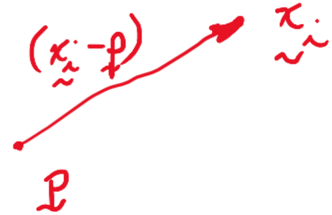
- Vamos chamar de $\hat{x}_i$ a aproximação do ponto x_i

- Como temos uma aproximação de um ponto só, $\hat{x}_i = p$ para algum ponto p. Que ponto é esse?

Aproximar cada ponto x_i por $\hat{x}_i = p$

↳ todo mundo aproximado pelo mesmo ponto (!!!)

$$MSE = \frac{1}{m}\sum_{i=1}^{m} \|\underset{\sim}{x}_i - \underset{\sim}{\hat{x}}_i\|^2 = \frac{1}{m}\sum_{i=1}^{m} \|\underset{\sim}{x}_i - \underset{\sim}{p}\|^2$$



Qual o $\underset{\sim}{p}$ que minimiza o MSE?

$$x_i - p = \begin{bmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{in} \end{bmatrix} - \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix}$$

x_i (ponto original)

p (aproximação)

11

$$\|x_i - p\| = \left[(x_{i1}-p_1)^2 + (x_{i2}-p_2)^2 + \dots + (x_{in}-p_n)^2\right]^{1/2}$$

$$\|x_i - p\|^2 = (x_{i1}-p_1)^2 + (x_{i2}-p_2)^2 + \dots + (x_{in}-p_n)^2$$

$$= (x_i - p)^T (x_i - p)$$

$$\underbrace{\left[(x_{i1}-p_1),\ (x_{i2}-p_2),\ \dots,\ (x_{in}-p_n)\right]}_{\substack{\text{matriz linha} \\ 1\times n}} \qquad \underbrace{\begin{bmatrix}(x_{i1}-p_1)\\(x_{i2}-p_2)\\\vdots\\(x_{in}-p_n)\end{bmatrix}}_{\substack{\text{matriz coluna} \\ n\times 1}}$$

$$\|x_i - p\|^2 = (x_i - p)^T (x_i - p) = x_i^T x_i \underbrace{- p^T x_i - x_i^T p}_{-2p^T x_i} + p^T p$$

$$MSE = \frac{1}{m}\sum_{i=1}^{m} \|x_i - p\|^2 = \frac{1}{m}\sum_{i=1}^{m} \left(x_i^T x_i - 2p^T x_i + p^T p \right)$$

$$= \frac{1}{m}\sum_{i=1}^{m} x_i^T x_i - 2p^T \cdot \frac{1}{m}\sum_{i=1}^{m} x_i + p^T p$$

$$\frac{\partial MSE}{\partial p} = \frac{\partial}{\partial p}\left(\cancelto{0}{\frac{1}{m}\sum_{i=1}^{m} x_i^T x_i} \underbrace{- 2p^T \cdot \frac{1}{m}\sum_{i=1}^{m} x_i}_{-2 \cdot \underbrace{\frac{1}{m}\sum_{i=1}^{m} x_i}_{\bar{x}}} + \underbrace{p^T p}_{2p} \right) = 0$$

$$\Rightarrow 2p - 2\bar{x} = 0 \Rightarrow \boxed{p = \bar{x}}$$

Zero-ésima aproximação

$$MSE = \frac{1}{m}\sum_{i=1}^{m}\|x_i - p\|^2 = \frac{1}{m}\sum_{i=1}^{m}(x_i - p)^T(x_i - p)$$

$$= \frac{1}{m}\sum_{i=1}^{m}\left(x_i^T x_i - 2p^T x_i + p^T p\right)$$

$$\frac{\partial}{\partial p}MSE = \frac{1}{m}\sum_{i=1}^{m} -2x_i + 2p = 0$$

$$\Rightarrow p = \frac{1}{m}\sum_{i=1}^{m} x_i$$

O ponto ótimo
é o centroide!

Próximas aproximações

- Agora vamos descobrir a próxima aproximação: $d = 1$

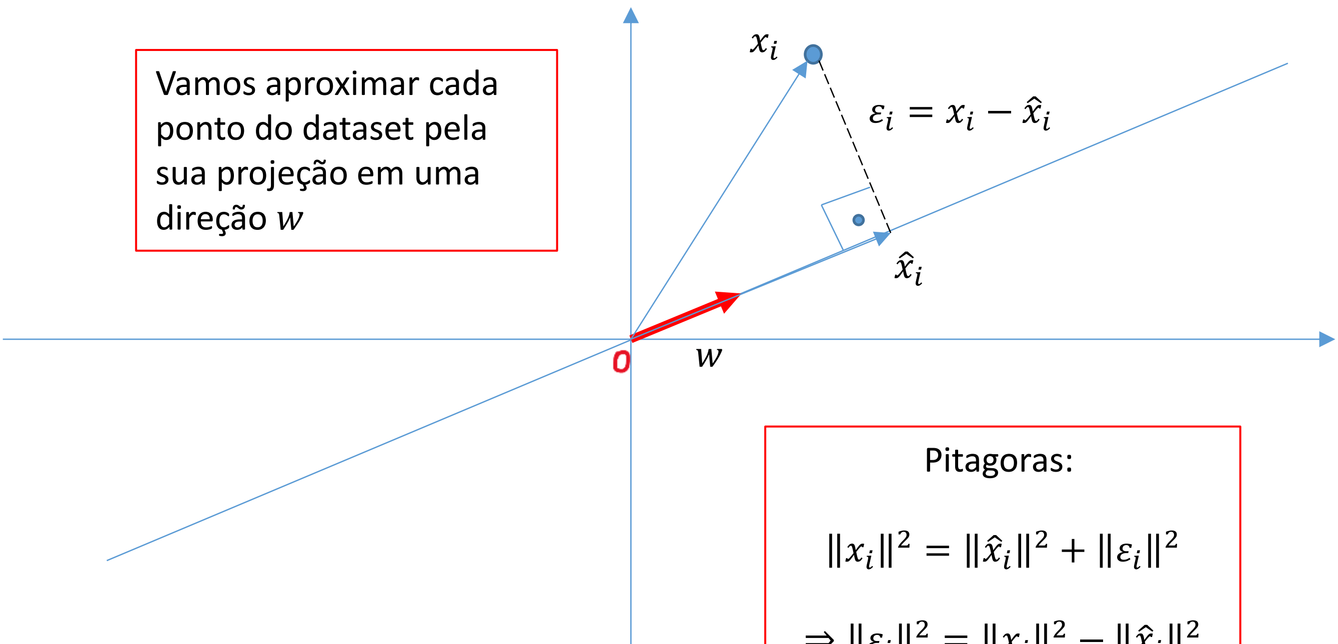
- Vamos remover a zero-ésima aproximação do dataset:

$$x_i \leftarrow (x_i - p)$$

- Nosso dataset (remanescente) tem uma nuvem de pontos cujo centróide é a origem

Próximas aproximações

Vamos aproximar cada ponto do dataset pela sua projeção em uma direção w



Pitagoras:

$$\|x_i\|^2 = \|\hat{x}_i\|^2 + \|\varepsilon_i\|^2$$

$$\Rightarrow \|\varepsilon_i\|^2 = \|x_i\|^2 - \|\hat{x}_i\|^2$$

Próximas aproximações

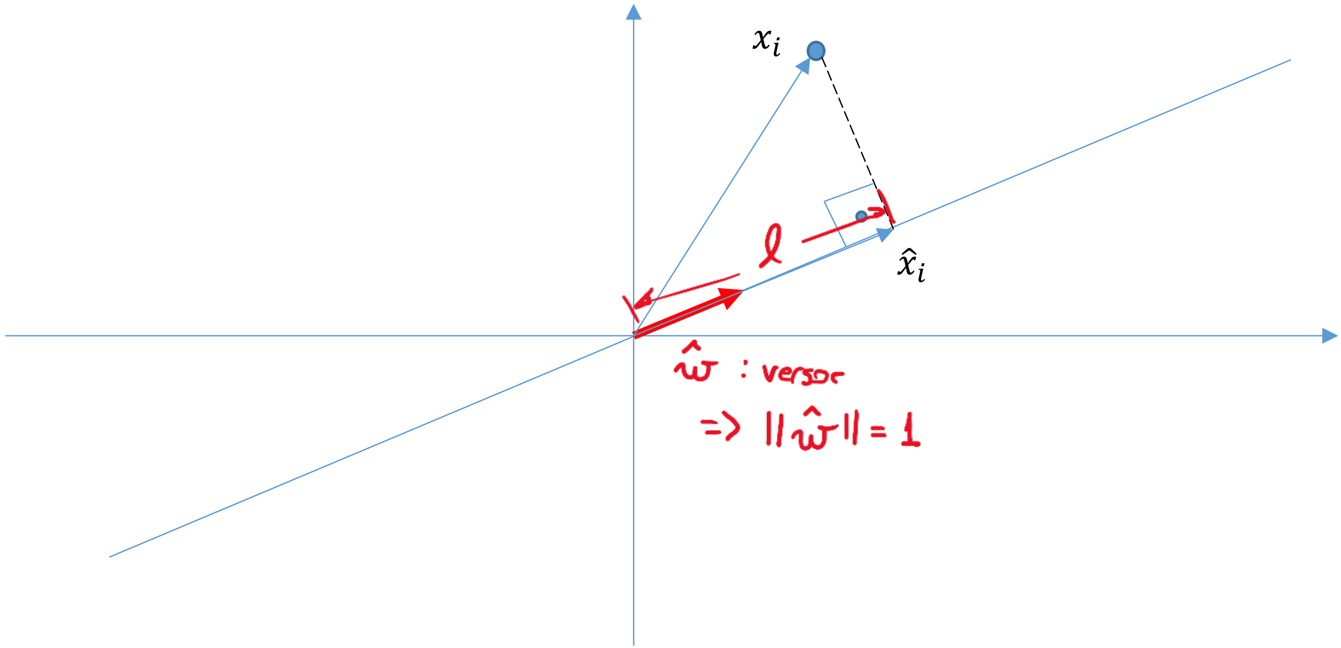
$$MSE = \frac{1}{m}\sum_{i=1}^{m}\|\varepsilon_i\|^2 = \frac{1}{m}\sum_{i=1}^{m}\left(\|x_i\|^2 - \|\hat{x}_i\|^2\right)$$

$$= \frac{1}{m}\sum_{i=1}^{m}\|x_i\|^2 - \frac{1}{m}\sum_{i=1}^{m}\|\hat{x}_i\|^2$$

Portanto,

$$w_1 = \arg\min_{w} MSE = \arg\max_{w} \underbrace{\frac{1}{m}\sum_{i=1}^{m}\|\hat{x}_i\|^2}_{M(w)}$$

$$\begin{cases} \ell = x_i \cdot \hat{w} \\ \hat{x}_i = \ell \, \hat{w} \end{cases} \Rightarrow \boxed{\hat{x}_i = (x_i \cdot \hat{w})\,\hat{w}}$$



$$\begin{cases} \hat{x}_i = (x_i \cdot \hat{w})\,\hat{w} \\ \hat{w} = \dfrac{1}{\|w\|}\, w \end{cases} \Rightarrow \hat{x}_i = \underbrace{\frac{(x_i \cdot w)}{\|w\|}\,\frac{w}{\|w\|}}_{\|w\| \cdot \|w\| = \|w\|^2}$$

$$\begin{cases} (\underset{\substack{\uparrow \\ \text{vetores}}}{x_i \cdot w}) = \underbrace{w^T x_i}_{\substack{\uparrow \\ \text{matrizes}}} \\ \|w\|^2 = w^T w \end{cases} \Rightarrow \boxed{\hat{x}_i = \frac{w^T x_i}{w^T w}\, w}$$

$$M(w) = \frac{1}{m}\sum_{i=1}^{m} \|\hat{x}_i\|^2 = \frac{1}{m}\sum_{i=1}^{m} \hat{x}_i^T x_i$$

$$\boxed{\hat{x}_i = \left(\frac{w^T x_i}{w^T w}\right) w} \Rightarrow M(w) = \frac{1}{m}\sum_{i=1}^{m} \left(\frac{w^T x_i}{w^T w}\right)^2 w^T w$$

numero

mat. coluna

$$= \frac{1}{m}\sum_{i=1}^{m} \frac{(w^T x_i)^2}{(w^T w)^{\cancel{2}}} \cdot \cancel{(w^T w)}$$

$$M(w) = \frac{1}{m}\sum_{i=1}^{m} \frac{(w^T x_i)^2}{w^T w}$$

$$M(w) = \frac{1}{m}\sum_{i=1}^{m}\frac{(w^T x_i)^2}{w^T w} = \frac{1}{m}\sum_{i=1}^{m}\frac{(w^T x_i)(x_i^T w)}{w^T w}$$

$$= \frac{1}{m}\,\frac{1}{w^T w}\sum_{i=1}^{m} w^T x_i x_i^T w$$

$$\overset{n\times 1}{[\ ]}\cdot\overset{1\times n}{[\ \ \ ]} = \overset{n\times n}{[\ \ \ \ \ ]}$$

$$\Rightarrow M(w) = \frac{1}{w^T w}\cdot w^T\underbrace{\left(\frac{1}{m}\sum_{i=1}^{m} x_i x_i^T\right)}w$$

$$C = \frac{1}{m}\sum_{i=1}^{m} x_i x_i^T$$

matriz de covariância

pto original menos média $\bar{x}$

$$\boxed{M(w) = \frac{w^T C w}{w^T w}}$$

maximizar $M(w) = \dfrac{w^T C w}{w^T w} = \underbrace{\dfrac{w^T}{\|w\|}}_{\hat{w}^T} \cdot C \cdot \underbrace{\dfrac{w}{\|w\|}}_{\hat{w}}$

$\Longleftrightarrow$ maximizar $\hat{w}^T C \hat{w}$ ← multiplicadores de Lagrange!

sujeito a $\hat{w}^T \hat{w} = 1$

$\Longleftrightarrow$ maximizar $L = \hat{w}^T C \hat{w} - \lambda(\hat{w}^T \hat{w} - 1)$

$\dfrac{\partial L}{\partial w} = 2C\hat{w} - 2\lambda\hat{w} = 0 \Rightarrow \boxed{C\hat{w} = \lambda\hat{w}}$: $\hat{w}$ é um autovetor de C, com autovalor λ

$$\begin{cases} L = \hat{w}^T C \hat{w} - \lambda(\hat{w}^T \hat{w} - 1) \\ C\hat{w} = \lambda \hat{w} \end{cases}$$

$$L = \hat{w}^T(\lambda \hat{w}) - \lambda \hat{w}^T \hat{w} + \lambda$$

$$= \lambda \hat{w}^T \hat{w} - \lambda \hat{w}^T \hat{w} + \lambda \implies L = \lambda$$

maximizar $L = \hat{w}^T C \hat{w} - \lambda(\hat{w}^T \hat{w} - 1)$

$\Updownarrow$

$\hat{w}$ é autovetor de C (matriz de covariancia dos dados) associado ao maior autovalor de C

Maximizando as projeções

$$\hat{x}_i = \text{projeção de } x_i \text{ na direção } w$$

$$\Rightarrow \hat{x}_i = \left(\frac{w^T x_i}{w^T w}\right) w$$

$$M(w) = \frac{1}{m}\sum_{i=1}^{m} \|\hat{x}_i\|^2 = \frac{1}{m}\sum_{i=1}^{m} \hat{x}_i^T \hat{x}_i$$

$$= \frac{1}{m}\sum_{i=1}^{m} \left(\frac{w^T x_i}{w^T w}\right)^2 w^T w = \frac{1}{m}\sum_{i=1}^{m} \frac{\left(w^T x_i\right)^2}{w^T w}$$

Maximizando as projeções

$$M(w) = \frac{1}{m}\sum_{i=1}^{m}\frac{\left(w^T x_i\right)^2}{w^T w} = \frac{1}{m}\sum_{i=1}^{m}\frac{w^T x_i x_i^T w}{w^T w}$$

$$\Rightarrow M(w) = \frac{w^T C w}{w^T w}$$

onde

$$C = \frac{1}{m}\sum_{i=1}^{m} x_i x_i^T$$

Matriz de covariância

linhas

colunas

$$X = \begin{bmatrix} \boxed{x_1^T} \\ x_2^T \\ \vdots \\ x_m^T \end{bmatrix} \qquad X^T = \begin{bmatrix} \boxed{x_1} & x_2 & \cdots & x_m \end{bmatrix}$$

$$\frac{1}{m} X^T X = \frac{1}{m} \begin{bmatrix} x_1 & x_2 & \cdots & x_m \end{bmatrix} \begin{bmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_m^T \end{bmatrix} = \frac{1}{m} \sum_{i=1}^{m} x_i x_i^T$$

Logo: $C = \frac{1}{m} X^T X$, que é a matriz de covariância dos dados!

Maximizando as projeções

Portanto, queremos achar w que maximiza

$$M(w) = \frac{w^T C w}{w^T w}$$

E agora?

Autovalores e autovetores

- Imagine o que acontece se w for um autovetor (dentre vários) da matriz C: $Cw = \lambda w$
- Portanto

$$M(w) = \frac{w^T C w}{w^T w} = \lambda \frac{w^T w}{w^T w} = \lambda$$

- Agora temos uma estratégia para maximizar $M(w)$:

$w =$ autovetor de C correspondente ao maior autovalor

Próximas aproximações?

Repita o processo:

- Remova a aproximação feita até o momento

$$X_k = X - \hat{X}_{k-1}$$

- Calcule a matriz de covariância atual

$$C_k = Cov[X_k] = \frac{1}{m} X_k^T X_k$$

- Calcule o autovetor do maior autovalor da matriz C_k

Principal Component Analysis

Ufa! Se você chegou até aqui, parabéns! Você acabou de derivar o método das componentes principais, um dos algoritmos mais importantes da estatística!

Principal Component Analysis

Calcule os parâmetros da PCA:

1. A média p das amostras

2. A matriz de covariância C

3. Os d autovetores w_k correspondentes aos maiores autovalores de C

Principal Component Analysis

Agora calcule as componentes principais de cada ponto de dados:

1. Remova a média p de cada amostra x_i

2. Projete os pontos resultantes nas direções principais

$$\hat{x}_i = p + \alpha_{i1} w_1 + \alpha_{i2} w_2 + \cdots + \alpha_{id} w_d$$

Principal Component Analysis

- Medida de “desempenho”: soma dos autovalores
  - Representa o quanto da variabilidade original do dataset foi capturada no dataset reduzido

- Aplicações
  - Análise das direções: podem indicar os aspectos mais importantes do dataset
  - Compressão: representar aproximadamente o dataset com menos dados
  - Visualização: projetar em 2D ou 3D para observar como os dados se espalham
  - Redução de computação: treinar modelos mais rapidamente sem perder muita qualidade

Principal Component Analysis

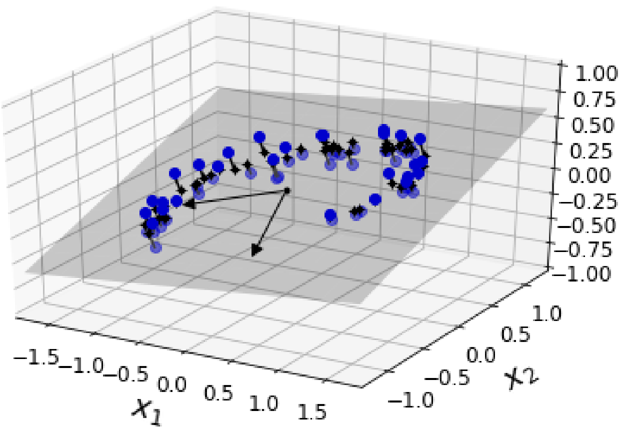
Implementações:

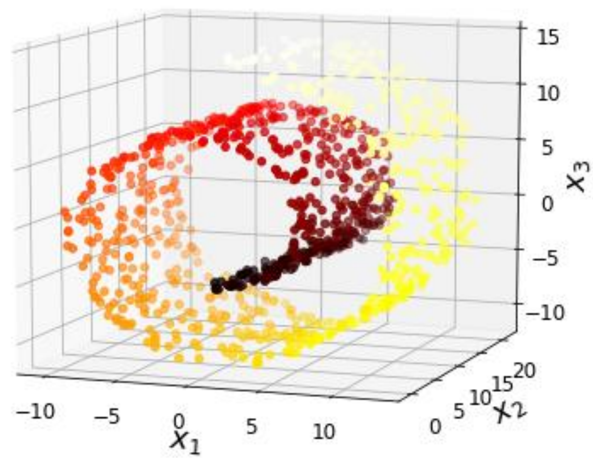
- Decomposição por Valores Singulares (SVD: Singular Value Decomposition)
- PCA incremental
- PCA aleatorizada

Estude mais sobre essas técnicas em seu livro texto

Limitações

- Só projeta os pontos em hiperplanos
- E se os pontos estão em outras superfícies?



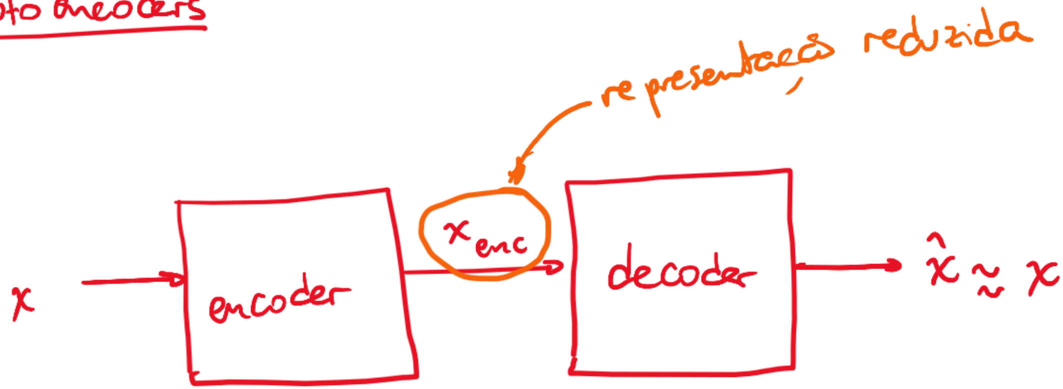


Outras técnicas

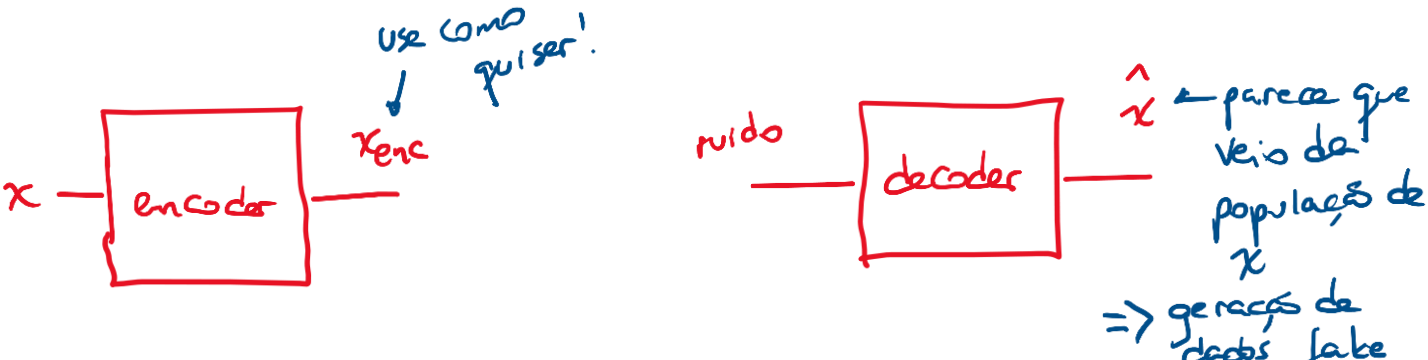
Existem técnicas mais sofisticadas para redução de dimensionalidade, que lidam com a não-linearidade:

- Kernel PCA
- Local Linear Embedding
- t-Distributed Stochastic Neighbor Embedding (t-SNE)

Auto encoders



"entra com x para prever x !!!"



Insper