Dimensionality reduction methods

Objective: Reduce number of dims. without loose vital information/representation in the data.

Why?

- Compression
- Visualization
- "Curse of dimensionality"
  ↳ $N \sim 5^d$

$\underline{X} \longrightarrow \underline{Z}$ $X \in \mathbb{R}^n,\ Z \in \mathbb{R}^d$

n-dim   d-dim $d < n$

Two categories: Feature selection

- Selects certain features without transforming them

Feature extraction

- Creates new features based on the original by some function.

Multidimensionality scaling (MDS)

In the category of feature extraction.

Given N samples, we consider all distances d_{ij} to be given using some metric (like L_2 or L_1)

$\rightarrow \sqrt{(x_1^i - x_1^j)^2 + \dots + (x_n^i - x_n^j)^2}$

$\rightarrow |x_1^i - x_2^j| + \dots + |x_n^i - x_n^j|$

$i, j = 1, 2, \dots, N$

↳ MDS places these in a d-dim space so that these distances are preserved as best as possible!

Another similarity measure : Inner product.

$$\underline{x}^i \cdot \underline{x}^j = x_1^i \cdot x_1^i + x_2^i x_2^j + \dots + x_n^i \cdot x_n^j$$

Gram matrix $\underline{\underline{B}}$: Contains all possible inner products between the N samples

$$\underline{\underline{X}} = \begin{bmatrix} \underline{x}^1 & \longrightarrow \\ \underline{x}^2 & \\ \vdots & \\ \underline{x}^N & \longrightarrow \end{bmatrix} \in \mathbb{R}^{N \times n}$$

$\underline{\underline{B}}_x = \underline{\underline{X}} \cdot \underline{\underline{X}}^T \in \mathbb{R}^{N \times N}$, where each element ij is an inner product between $\underline{x}^i$ and $\underline{x}^j$

MDS : $\underline{\underline{Y}} \cdot \underline{\underline{Y}}^T \approx \underline{\underline{X}} \cdot \underline{\underline{X}}^T$, where

$$\underline{\underline{Y}} = \begin{bmatrix} \underline{y}^1 & \longrightarrow \\ \vdots & \\ \underline{y}^N & \longrightarrow \end{bmatrix} \in \mathbb{R}^{N \times d}$$

To achieve these reduced dimensional $\underline{y}$ samples it can be shown that it amounts to

a) Find the eigenvalues $\{\lambda_i\}_{i=1}^{N}$ of the Gram matrix $\underline{\underline{B}}_x$, and corresponding eigenvectors $\{\underline{e}_i\}$ $\underline{e}_i \in \mathbb{R}^N$

b) Choose a subset of the d largest eigenvals.

c) Construct a diagonal matrix $\underline{\underline{\Lambda}}_d^{1/2}$ with the d largest eigenvals on the diagonal (sorted) NB!

d) Construct a matrix $\underline{\underline{E}}_d$, where the columns are the eigenvectors corresponding to the d largest eigenvals.

e) $\underline{y}^i = \left(\underbrace{\underline{\underline{E}}_d}_{N \times d} \cdot \underbrace{\underline{\underline{\Lambda}}_d^{1/2}}_{d \times d}\right)^i$, $i = 1, \ldots, N$ (the rows)

Benefits of Feature extraction :

- "Fast" way of reducing complexity $O(N^3)$
- Preserves the important information

Downsides :

- What do the new dims represent?

Feature selection can fix this "explainability" problem.

Feature selection

- Keep relevant, remove irrelevant
  - ↳ Redundant, some features correlates with others.
- Find best subset
  - ↳ Least number of features contributing the most to the accuracy.

Forward selection: Add features one by one and check error towards validation set.
↳ Stop when error is lowest.

PS: FS needs a supervised dataset $\{\underline{x}^i, y^i\}_{i=1}^{N}$

this is not neccessary in feature extraction!

Drawbacks:
- Can be costly computationally
  Should be impl. with crossvalidation
- Do not neccessarily find all redundant features / best subsets.
- Needs labels
- Selected features depends heavily on model / pred. task.

Benefits
- Explainability
- Easily understood algorithm.

Dim. reduksjon metoder

Mål : Redusere antall dim. uten å miste vital informasjon / representasjon i data.

Why? : ↳ Compresjon ↳ Visualisering ↳ Computational complexity.

$$\underline{X} \rightarrow \underline{Y} \qquad \underline{X} \in \mathbb{R}^n, \quad \underline{\underline{Y}} \in \mathbb{R}^d$$

n-dim, n features → d-dim, d features

$d < n$

1a) Feature selection : Selekterer noen features uten å transformere de

Feature extraction : Lager nye features basert på de opprinnelige vha. en funksjon.

MDS : I kategorien extraction

Basert på å bevare lik "avstand" mellom vektorene i sample matrisene

$\underline{\underline{X}}$ sml. med $\underline{\underline{Y}}$

Avstandsmål : Indre-produkt $\underline{X}_i \cdot \underline{X}_j^T$

Gram matrise : $B_x = \underline{\underline{X}} \cdot \underline{\underline{X}}^T \in \mathbb{R}^{N \times N}$

↳ Inneholder alle "avstandene" mellom alle samples

MDS: $\underline{\underline{Y}} \cdot \underline{\underline{Y}}^T \approx \underline{\underline{X}} \cdot \underline{\underline{X}}^T$, der

$$\underline{\underline{Y}} = \begin{bmatrix} \underline{y}^1 \longrightarrow \\ \vdots \\ \underline{y}^N \longrightarrow \end{bmatrix} \in \mathbb{R}^{N \times d}$$

Hvordan? Algoritme:

a) Finn egenverdiene $\{\lambda_i\}_{i=1}^{N}$ til $\underline{\underline{B}}_x$ og egenvektorene $\{\underline{e}_i\}$, $\underline{e}_i \in \mathbb{R}^N$

b) Velg et subset av de d største eg.verdier og eg.vektorer $\{\lambda_i\}_{i=1}^{d}$ $\{\underline{e}_i\}_{i=1}^{d}$

c) Konstruer en diag. matrise $\underline{\underline{\Lambda}}_d^{1/2}$ med de d største λ_i på diag (sortert)

d) Konstruer matrisa $\underline{\underline{E}}_d$ hvor kollonnene er eg.vektorene tilsvarende eg.verdiene.

e) $\underline{y}_i = (\underbrace{\underline{\underline{E}}_d}_{N \times d} \cdot \underbrace{\underline{\underline{\Lambda}}_d^{1/2}}_{d \times d})^i$, $i = 1, \dots, N$ (radene)

Bruksområder: Der avstander mellom features er en viktig faktor å bevare

1b) Hvordan bruke gitte avstander mellom samples til å oppnå $B = X^T X$ (3)

Formel slide 23

$$D = \begin{bmatrix} 0 & d_{12} & d_{13} & \cdots & d_{17} \\ & 0 & & & \vdots \\ & & 0 & & \vdots \\ & & & \ddots & d_{67} \\ & & & & 0 \end{bmatrix}$$

(antar 7 samples)

Indreprod. avstandsmål $d_{ij} = \|x_i - x_j\|^2 \leftarrow$ Euklidsk avstandsmål

$\hookrightarrow \underline{\underline{B}} = \underline{\underline{X}}\,\underline{\underline{X}}^T = -\frac{1}{2} C D C^T$ $\quad N \times N$

$C = I_N - \frac{1}{N} J_N$, der $I_N = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & 0 \cdots & 0 \\ 0 & 0 & 1 \cdots & 0 \\ \vdots & & \ddots & \\ & & & 1 \end{pmatrix}$

$J_N = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & & & \\ 1 & \cdots & \cdots & 1 \end{pmatrix}$

Derfra videre til å redusere til 2 dim.

$\underline{y}_1, \underline{y}_2, \ldots, \underline{y}_N \in \mathbb{R}^2$