Statistical Data Analysis

Jana de Wiljes

wiljes@uni-potsdam.de

www.dewiljes-lab.com

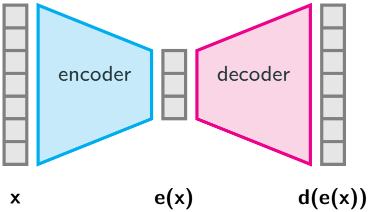
January 4, 2023

Dimension reduction

Dimension reduction

Goal: reducing the number of given features in a data set $x_i \in \mathcal{S}$ with $i \in \{1, \ldots, N\}$

- choose model class for the encoder $e \in \mathcal{E}$ and for the decoder $d \in \mathcal{D}$
- and appropriate loss functional $l(x, d(e(x)))$



Dimension reduction problem

For a given data $\mathcal{S}$ and fixed families of functions $\mathcal{E}$ and $\mathcal{D}$

$$(e^*, d^*) = \arg \min_{(e,d)\in\mathcal{E}\times\mathcal{D}} l(x, d(e(x))) \tag{1}$$

Matrix

Definition: Let K a field. A $m \times n$ **matrix** with entries in K ist a table

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix} \in K^{m \times n}$$

of elements $a_{ij} \in K$. m is the number of rows and n the number of columns of A. Let $A = (a_{ij}) \in K^{m \times n}$ and $B = (b_{jk}) \in K^{n \times r}$ be two matrices, so that the column number of A coincides with the number of rows of B. Then the product

$$C = A \cdot B = (c_{ik}) \in K^{m \times r}$$

is given via

$$c_{ik} = \sum_{j=1}^{n} a_{ij} b_{jk}.$$

Principal Component Analysis

Principal Component Analysis (PCA)

Goal: reducing the number of given features in a data set $x_i \in \mathcal{S}$ with $i \in \{1, \dots, N\}$ via a linear projection

- choose model class such that the combination of the encoder and decoder is $\{d(e(x)) = \sum_{j=1}^{k} \langle u_j, x \rangle u_j | u_1, \dots, u_k$ orthonomal basis of U $\}$ where U is k-dimensional subspace
- loss functional $l(x, d(e(x))) = \|x - d(e(x))\|^2$

Optimisation Problem: For a given data $\mathcal{S} = \{x_1, \dots, x_N\}$ where $x_i \in \mathbb{R}^d$ the associated optimisation problem is defined by

$$Q^* = \arg \min_{Q \in \mathbb{R}^{p \times k} \text{ with } Q^\top Q = I} \frac{1}{N} \sum_{i=1}^{N} \|x_i - \sum_{j=1}^{k} \langle u_j, x_i \rangle u_j\|^2$$

where $Q = \begin{pmatrix} | & & | \\ u_1 & \dots & u_k \\ | & & | \end{pmatrix} \in \mathbb{R}^{p \times k}$. Note that $Q^\top Q = I \in \mathbb{R}^{k \times k}$

Principal Component Analysis (PCA)

Given: data $\mathcal{S} = \{x_1, \ldots, x_N\}$ where $x_i \in \mathbb{R}^p$

Consider: the following optimisation problem

$$Q^* = \arg \min_{Q \in \mathbb{R}^{p \times k},\ Q^\top Q = I} \frac{1}{N} \sum_{i=1}^{N} \|x_i - \sum_{j=1}^{k} u_j \langle u_j, x_i \rangle\|^2$$

$$= \arg \min_{Q \in \mathbb{R}^{p \times k},\ Q^\top Q = I} \frac{1}{N} \sum_{i=1}^{N} \|x_i - QQ^T x_i\|^2$$

$$\underbrace{\begin{pmatrix} | & & | \\ u_1 & \ldots & u_k \\ | & & | \end{pmatrix}}_{Q \in \mathbb{R}^{p \times k}} \underbrace{\begin{pmatrix} -- & u_1 & -- \\ & \ldots & \\ -- & u_k & -- \end{pmatrix}}_{Q^\top \in \mathbb{R}^{k \times p}} \underbrace{\begin{pmatrix} x_i(1) \\ \\ x_i(p) \end{pmatrix}}_{x_i \in \mathbb{R}^{p \times 1}}$$

Note that

$$e(x_i) = Q^\top x_i \in \mathbb{R}^{k \times 1} \text{ (encoding)}$$
$$d(e(xi) = QQ^\top x_i = Qe(x_i) \in \mathbb{R}^{p \times 1} \text{ (decoding)}$$

Maximizing the data variance

Consider:

$$
\begin{aligned}
Q^* &= \arg \min_{Q \in \mathbb{R}^{p \times k},\ Q^\top Q = I} \frac{1}{N} \sum_{i=1}^{N} \|x_i - QQ^T x_i\|^2 \\
&= \arg \min_{Q \in \mathbb{R}^{p \times k},\ Q^\top Q = I} \frac{1}{N} \sum_{i=1}^{N} \|x_i\|^2 - 2\langle x_i, QQ^T x_i \rangle + \|QQ^T x_i\|^2 \\
&= \arg \min_{Q \in \mathbb{R}^{p \times k},\ Q^\top Q = I} \frac{1}{N} \sum_{i=1}^{N} \|x_i\|^2 - 2\langle x_i, QQ^T x_i \rangle + \langle x_i, QQ^T x_i \rangle \\
&= \arg \min_{Q \in \mathbb{R}^{p \times k},\ Q^\top Q = I} \frac{1}{N} \sum_{i=1}^{N} -\langle x_i, QQ^T x_i \rangle \\
&= \arg \max_{Q \in \mathbb{R}^{p \times k},\ Q^\top Q = I} \frac{1}{N} \sum_{i=1}^{N} \langle x_i, QQ^T x_i \rangle \\
&= \arg \max_{Q \in \mathbb{R}^{p \times k},\ Q^\top Q = I} \operatorname{trace}(Q^T \frac{1}{N} XX^\top Q) \\
&= \arg \max_{Q \in \mathbb{R}^{p \times k},\ Q^\top Q = I} \operatorname{trace}(D)
\end{aligned}
$$

Singular value decomposition

Singular value decomposition

Before: for quadratic matrices we had Eigenvalues and Eigenvectors that can be used to diagonalise a matrix

Now:

- similar concept for non quadratic matrices
- the corresponding scalars are called Singular values which opposed to the Eigenvalues are **always** real
- Although similarity exist singular value decomposition is not an generalization of Eigenvalues/Eigenvector approach
- the rank of a matrix can be determined in a numerical stable way

Group of orthogonal matrices

Definition:

- $\mathrm{GL}(n, \mathbb{R})$ general linear group of degree n is the set of $n \times n$ invertible matrices
- $\mathrm{O}(n) = \left\{ Q \in \mathrm{GL}(n, \mathbb{R}) \mid Q^{\mathsf{T}} Q = Q Q^{\mathsf{T}} = I \right\}.$

Singular value decomposition

Theorem: Let $A \in \mathbb{R}^{m\times n}$ be a matrix. Then $\sigma_1, \dots, \sigma_p \in \mathbb{R}$ with $\sigma_1 \geq \cdots \geq \sigma_p \geq 0$ as well as $U \in O(m)$ and $V \in O(n)$ exist, so that

$$U^t A V = \Sigma := \begin{pmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_p \\ \vdots & & 0 \\ \vdots & & \vdots \\ 0 & \cdots & 0 \end{pmatrix},$$

wobei $p = \min(m, n)$. The values σ_i are called **singular values** of A. A representation of the form $A = U\Sigma V^t$ is called **singular value decomposition** (**SVD**).

Example

- For a quadratic matrix:

$$A_1 = \begin{pmatrix} 4 & 12 \\ 12 & 11 \end{pmatrix} = \begin{pmatrix} 3/5 & 4/5 \\ 4/5 & -3/5 \end{pmatrix} \cdot \begin{pmatrix} 20 & 0 \\ 0 & 5 \end{pmatrix} \cdot \begin{pmatrix} 3/5 & 4/5 \\ -4/5 & 3/5 \end{pmatrix}$$

- SDV of orthogonal matrices:

$$A_2 = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

-

$$A_3 = \begin{pmatrix} 0.36 & 1.60 & 0.48 \\ 0.48 & -1.20 & 0.64 \end{pmatrix} = \begin{pmatrix} 0.8 & 0.6 \\ -0.6 & 0.8 \end{pmatrix} \cdot \begin{pmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 & 1 & 0 \\ 0.6 & 0 & 0.8 \\ -0.8 & 0 & 0.6 \end{pmatrix}$$

Remark

- Note that
$$\operatorname{rang}\begin{pmatrix}0 & 0\\ 0 & 0\end{pmatrix} = 0, \quad \operatorname{rang}\begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix} = 1.$$
- yet the two eigenvalues in both cases are 0 and 0. The singular values on the other hand are 0, 0 and 0, 1 respectively, i.e., in this case the eigenvalues do not tell you anything about the rank of the matrix but the number of singular values of the matrix correspond to its rank
- Consider for $\varepsilon > 0$:
$$A = \begin{pmatrix}0 & 1\\ \varepsilon & 0\end{pmatrix}.$$
Since $\chi_A(t) = t^2 - \varepsilon$ the corresponding eigenvalues are $\pm\sqrt{\varepsilon}$. The singular values are $\sigma_1 = 1, \sigma_2 = \varepsilon$ and for ε converging towards 0, the rank of matrix is converging towards 1

Proof

We construct a singular value decompostion of von A:
First we set $B := A^t A$. This is a real symmetric $n \times n$- matrix and has only real eigenvalues λ_i, which we will use the following indices for $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$. The corresponding eigenvectors are denoted $\{v_1, \ldots, v_n\}$ and form a basis of the $\mathbb{R}^n$.
Further we note that all λ_i are non-negativ, this is due to to two aspects: the eigenvectors are v_i orthonormal:

$$v_i^t \cdot B \cdot v_i = \lambda_i \cdot v_i^t \cdot v_i = \lambda_i$$

and due to the definition of B and since the scalar product is positiv definit:

$$v_i^t \cdot B \cdot v_i = v_i^t \cdot A^t \cdot A \cdot v_i = \langle Av_i, Av_i \rangle \geq 0.$$

Since $r := \operatorname{rang} A = \operatorname{rang} B$, we know that the first r eigenvalues $\lambda_1, \ldots, \lambda_r$ are strictly positiv.

Proof

We set for $i = 1, \dots, r$

$$u_i := \frac{1}{\sqrt{\lambda_i}} A v_i$$

and construct $m - r$ additional orthonormal vektors $u_{r+1}, \dots, u_m$, that are also orthonormal to the original $u_1, \dots, u_r$ so that all of them together form a basis of $\mathbb{R}^m$. We know construct the matrices U and V out of the column vectors u_i bzw. v_i:

$$U = (u_1 \ \dots \ u_m), \quad V = (v_1 \ \dots \ v_n).$$

The singularvalues of A are $\sigma_i := \sqrt{\lambda_i}$, for $i = 1, 2, \dots, r$ and $\sigma_i = 0$ for $i = r + 1, \dots, p$.

It remains to show that $A = U \Sigma V^t$ is indeed a singular value decomposition of A. Firstly note that V is orthogonal, since v_i form by construction an orthonormal basis.

Proof

The vectors u_i are an orthonormal basis as well, since forr $i, j = 1, \ldots, r$ gilt

$$u_i^t u_j = \frac{1}{\sqrt{\lambda_i \lambda_j}} v_i^t A^t A v_j = \frac{\lambda_j}{\sqrt{\lambda_i \lambda_j}} v_i^t v_j = \begin{cases} 1, & i = j, \\ 0, & i \neq j \end{cases}$$

and this orthonormal property is propergated onto $u_{r+1}, \ldots, u_m$.

It remains to show that $A = U\Sigma V^t$ holds:

$$U\Sigma V^t = \sum_{i=1}^{r} \sqrt{\lambda_i} u_i v_i^t = \sum_{i=1}^{r} A v_i v_i^t = \sum_{i=1}^{n} A v_i v_i^t = A \cdot \sum_{i=1}^{n} v_i v_i^t = A \cdot I = A.$$

This conclude the proof as we have constructed a singular value decomposition of A.
□

Singular value decomposition

Theorem: Let $A = U\Sigma V$ be the singular value decomposition of $A \in \mathbb{R}^{m\times n}$ with singular values $\sigma_1 \geq \cdots \geq \sigma_p$ für $p = \min(m, n)$. Let $u_1, \ldots, u_m$ and $v_1, \ldots, v_n$ denote the columns of U and V respectively. Then the following holds:

- $Av_i = \sigma_i u_i$ and $A^t u_i = \sigma_i v_i$ für $i = 1, 2, \ldots, p$.
- For $\sigma_1 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_p = 0$ follows that $\operatorname{rang} A = r$. Furthermore,

  $$\mathbf{Ker}(A) = \langle v_{r+1}, \ldots, v_n \rangle \text{ und } \mathbf{Im}(A) = \langle u_1, \ldots, u_r \rangle.$$

- the squares $\sigma_1^2, \ldots, \sigma_p^2$ of the singular values are the eigenvalues of $A^t A$ and of AA^t to the corresponding eigen vectors $v_1, \ldots, v_p$ and $u_1, \ldots, u_p$ respectively.

Golub-Reinsch algorithm

Input: A$\in \mathbb{R}^{n \times m}$,m $\leq n$, ϵ

- $\begin{bmatrix} B \\ 0 \end{bmatrix} = (U_1, \ldots, U_n)^\top A(V_1 \ldots V_n - 2)$ where U_i and V_j are householder transformations
- Set q=0
- **while** $(q < n)$
  1. set $B(i, i+1) = 0$ if for any $i = 1, \ldots, n-1$ $B(i, i+1) \leq \epsilon(|B(i, i)| + |B(i+1, i+1)|)$
  2. Determine the smallest p and the largest q so that B can be blocked as
  $$B = \begin{bmatrix} B_{1,1} & 0 & 0 \\ 0 & B_{2,2} & 0 \\ 0 & 0 & B_{3,3} \end{bmatrix} \tag{2}$$
  where $B_{3,3}$ is diagonal and $B_{2,2}$ has no zero superdiagonal entry.
  3. If $q = n$, set $\Sigma =$ the diagonal portion of B **STOP**.
  4. If for $i = p+1, \ldots, n-q-1$ $B_{i,i} = 0$, then
     - Apply Givens rotations so that $B_{i,i+1} = 0$ and $B_{2,2}$ is still upper bidiagonal.
  5. else Golub Kahan SVD step: This step is essentially applying the QR method to the symmetric tridiagonal matrix $T = BB$

Pseudoinverse

Remark: For symmetric matrices A the singular values are the absolut values of the eigenvalues of A. In case all eigenvalues are non-negativ, $A = S^t \Lambda S$ is the SVD.

Definition: Let $A \in \mathbb{R}^{m \times n}$. A matrix $A^+ \in \mathbb{R}^{n \times m}$ is called the **pseudoinverse** of A, if $\forall b \in \mathbb{R}^m$ the vector $x = A^+ b$ is the solution of the minimalisation problem

$$\text{Find } x \text{, so that } \|b - Ax\|_2 \text{ is minimal}$$

i.e., $\|b - AA^+ b\| = \min_{x \in \mathbb{R}^n} \|b - Ax\|$.

Motivation

Note: for a quadratic invertible matrix A the pseudoinverse is: $A^{+} = A^{-1}$

Application: in case the system $Ax = b$ does not have a solution, it is possible to obtain the best approximation $\tilde{x} = A^{+}b$ via the pseudoinverse A^{+} i.e., the one that minimizes the error $\|Ax - b\|$ (note that is the solution of the least squares problem.

Note that A^{+} can be consider as a lineare mapping. Then the following holds

- $$AA^{+} \colon \mathbb{R}^m \to \mathbf{Im}(A)$$
  is the orthogonal projection to image of A and
- $$A^{+}A \colon \mathbb{R}^n \to (\mathbf{Ker}A)^{\perp}$$
  is the orthogonal projection to the orthogonal complement von the kernel of A.

SVD and Pseudoinverse

Theorem: Let $A \in \mathbb{R}^{m \times n}$ and let $A = U\Sigma V^t$ be the corresponding singular value decomposition with singular value $\sigma_1 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_p = 0$. Then we define

$$\Sigma^+ = \begin{pmatrix} \frac{1}{\sigma_1} & & & 0 \\ & \ddots & & \\ & & \frac{1}{\sigma_r} & \\ & & & \\ 0 & & & 0 \end{pmatrix}$$

and the matrix $A^+ = V\Sigma^+U^t \in \mathbb{R}^{n \times m}$ is the pseudo inverse of A.

Pseudocode

Algorithm 1 PCA

Compute dot product matrix: $\mathbf{X}^\top\mathbf{X} = \sum_{i=1}^{N}(\mathbf{x}_i - \mu)^\top(x_i - \mu)$;
Eigenanalysis $\mathbf{X}^\top\mathbf{X} = \mathbf{V}\Lambda\mathbf{V}^\top$;
Compute $\mathbf{U} = \mathbf{X}\mathbf{V}\Lambda^{-\frac{1}{2}}$;
Keep specific number of first components: $\mathbf{U}_k = [u_1, \ldots, u_k]$;
Compute k features: $\mathbf{Y} = \mathbf{U}_k^\top\mathbf{X}$;

Note that:

$$\mathbf{X} = \mathbf{U}\Lambda^{\frac{1}{2}}\mathbf{V}^\top$$

Further

$$\begin{aligned}\mathbf{X}^\top\mathbf{X} &= (\mathbf{U}\Lambda^{\frac{1}{2}}\mathbf{V}^\top)^\top\mathbf{U}\Lambda^{\frac{1}{2}}\mathbf{V}^\top \\ &= \mathbf{V}\Lambda^{\frac{1}{2}}\mathbf{U}^\top\mathbf{U}\Lambda^{\frac{1}{2}}\mathbf{V}^\top \\ &= \mathbf{V}\Lambda\mathbf{V}^\top\end{aligned}$$

Autoencoders

- unsupervised artificial neural network (feed forward)
- Two steps:
  - **Encoder:** learns how to efficiently compress and encode data
  - **Decoder:** learns how to reconstruct the data back from the reduced encoded representation to a representation that is as close to the original input as possible.