Eigenvalues and eigenvectors. Singular values and singular vectors. CP tensor decomposition. PCA

Daniil Merkulov

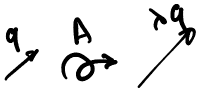
Applied Math for Data Science. Sberuniversity.

Eigenvalues and eigenvectors

Eigenvectors and eigenvalues

$A_{n \times n}$ ← квадр.

всегда сущ. набор из n СЗ (возможно, не уникальны)



A scalar value λ is an eigenvalue of the $n \times n$ matrix A if there is a nonzero vector q such that

$$\boxed{Aq = \lambda q.}$$

he vector q is called an eigenvector of A. The matrix A is nonsingular if none of its eigenvalues are zero. The eigenvalues of symmetric matrices are all real numbers, while nonsymmetric matrices may have imaginary eigenvalues. If the matrix is positive definite as well as symmetric, its eigenvalues are all positive real numbers.

$$\det(A - \lambda I) = 0$$ характер. ур-е

иногда СЗ – комплексные

– совпадать

Eigenvectors and eigenvalues

Theorem

$$A \succeq (\succ) 0 \Leftrightarrow \text{all eigenvalues of } A \text{ are } \geq (>) 0$$

Proof

1. $\rightarrow$ Suppose some eigenvalue λ is negative and let x denote its corresponding eigenvector. Then

$$Ax = \lambda x \rightarrow x^T Ax = \lambda x^T x < 0$$

which contradicts the condition of $A \succeq 0$.

Eigenvectors and eigenvalues

спектр – набор СЗ

Theorem

$$\boxed{A \succeq (\succ) 0} \Leftrightarrow \text{all eigenvalues of } A \text{ are } \geq (>) 0$$

Proof

1. $\rightarrow$ Suppose some eigenvalue λ is negative and let x denote its corresponding eigenvector. Then

$$Ax = \lambda x \rightarrow x^T A x = \lambda x^T x < 0$$

which contradicts the condition of $A \succeq 0$.

2. $\leftarrow$ For any symmetric matrix, we can pick a set of eigenvectors $v_1, \ldots, v_n$ that form an orthogonal basis of $\mathbb{R}^n$. Pick any $x \in \mathbb{R}^n$.

$$\begin{aligned} x^T A x &= (\alpha_1 v_1 + \ldots + \alpha_n v_n)^T A (\alpha_1 v_1 + \ldots + \alpha_n v_n) \\ &= \sum \alpha_i^2 v_i^T A v_i = \sum \alpha_i^2 \lambda_i v_i^T v_i \geq 0 \end{aligned}$$

here we have used the fact that $v_i^T v_j = 0$, for $i \neq j$.

Eigendecomposition (spectral decomposition)

$$Q^TQ = I$$

Suppose $A \in S_n$, i.e., A is a real symmetric $n \times n$ matrix. Then A can be factorized as

$$A = \begin{pmatrix} a_1 & & 0 \\ & \ddots & \\ 0 & & a_n \end{pmatrix} = I \cdot \begin{pmatrix} a_1 & & 0 \\ & \ddots & \\ 0 & & a_n \end{pmatrix} \cdot I^T$$

$$\boxed{A = Q\Lambda Q^T,}$$

на диагонали стоят её СЗ

//

пример

$$\begin{pmatrix} 1-\lambda & 0 & 0 & 0 \\ 0 & 2-\lambda & 0 & 0 \\ 0 & 0 & 3-\lambda & 0 \\ 0 & 0 & 0 & 4-\lambda \end{pmatrix}_{4\times 4} - \lambda \begin{pmatrix} 1 & & 0 \\ & 1 & \\ & & 1 \\ 0 & & & 1 \end{pmatrix}$$

диагональная

$a \in \mathbb{R}^n$

$\operatorname{diag}(a) \in \mathbb{R}^{n\times n}$

$\det[\operatorname{diag}(a)]$

$$\det = (1-\lambda)(2-\lambda)(3-\lambda)(4-\lambda) = 0$$

$$\left[\begin{array}{l} \lambda = 1 \\ \lambda = 2 \\ \lambda = 3 \\ \lambda = 4 \end{array}\right.$$

$A \in \mathbb{R}^{n\times n}$ $\operatorname{diag}(A) \in \mathbb{R}^n$

[1] A good cheat sheet with matrix decomposition is available at the NLA course website.

Eigendecomposition (spectral decomposition)

Suppose $A \in S_n$, i.e., A is a real symmetric $n \times n$ matrix. Then A can be factorized as

$$A = Q \Lambda Q^T,$$

where $Q \in \mathbb{R}^{n \times n}$ is orthogonal, i.e., satisfies $Q^T Q = I$, and $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n)$. The (real) numbers λ_i are the eigenvalues of A and are the roots of the characteristic polynomial $\det(A - \lambda I)$. The columns of Q form an orthonormal set of eigenvectors of A. The factorization is called the spectral decomposition or (symmetric) eigenvalue decomposition of A. [1]

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ & a_{22} & a_{23} & a_{24} \\ 0 & & a_{33} & a_{34} \\ & & & a_{44} \end{pmatrix}$$

[1] A good cheat sheet with matrix decomposition is available at the NLA course website.

Eigendecomposition (spectral decomposition)

Suppose $A \in S_n$, i.e., A is a real symmetric $n \times n$ matrix. Then A can be factorized as

$$A = Q\Lambda Q^T,$$

where $Q \in \mathbb{R}^{n\times n}$ is orthogonal, i.e., satisfies $Q^TQ = I$, and $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n)$. The (real) numbers λ_i are the eigenvalues of A and are the roots of the characteristic polynomial $\det(A - \lambda I)$. The columns of Q form an orthonormal set of eigenvectors of A. The factorization is called the spectral decomposition or (symmetric) eigenvalue decomposition of A. [1]

We usually order the eigenvalues as $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$. We use the notation $\lambda_i(A)$ to refer to the i-th largest eigenvalue of $A \in S$. We usually write the largest or maximum eigenvalue as $\lambda_1(A) = \lambda_{\text{max}}(A)$, and the least or minimum eigenvalue as $\lambda_n(A) = \lambda_{\text{min}}(A)$.

[1] A good cheat sheet with matrix decomposition is available at the NLA course website.

Eigenvalues

The largest and smallest eigenvalues satisfy

$$\lambda_{\text{min}}(A) = \inf_{x \neq 0} \frac{x^T A x}{x^T x}, \qquad \lambda_{\text{max}}(A) = \sup_{x \neq 0} \frac{x^T A x}{x^T x}$$

Eigenvalues

The largest and smallest eigenvalues satisfy

$$\lambda_{\text{min}}(A) = \inf_{x \neq 0} \frac{x^T A x}{x^T x}, \qquad \lambda_{\text{max}}(A) = \sup_{x \neq 0} \frac{x^T A x}{x^T x}$$

and consequently $\forall x \in \mathbb{R}^n$ (Rayleigh quotient):

$$\lambda_{\text{min}}(A) x^T x \leq x^T A x \leq \lambda_{\text{max}}(A) x^T x$$

Eigenvalues

$$\|A\|_2 = \sigma_{MAX}(A)$$

The largest and smallest eigenvalues satisfy

$$A^{-1} = \begin{pmatrix} \frac{1}{10} & 0 \\ 0 & 1 \end{pmatrix}$$

$$\lambda_{\min}(A) = \inf_{x \neq 0} \frac{x^T A x}{x^T x}, \qquad \lambda_{\max}(A) = \sup_{x \neq 0} \frac{x^T A x}{x^T x}$$

and consequently $\forall x \in \mathbb{R}^n$ (Rayleigh quotient):

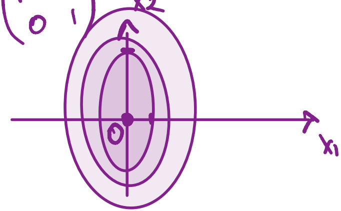
$$A = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$$

$$\lambda_{\min}(A) x^T x \leq x^T A x \leq \lambda_{\max}(A) x^T x$$

$$f(x) = \frac{1}{2} x^T A x$$

The **condition number** of a nonsingular matrix is defined as

$$A = \begin{pmatrix} 10 & 0 \\ 0 & 1 \end{pmatrix} \qquad K = 10$$



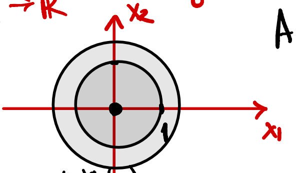
$$\boxed{\kappa(A) = \|A\| \|A^{-1}\|}$$

квадрат. ф-ция

$$f : \mathbb{R}^n \to \mathbb{R}$$

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$K = 1$$



$$f(x) = \frac{1}{2} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}^T \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}^T \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = x_1 \cdot x_1 + x_2 \cdot x_2 = x_1^2 + x_2$$

Eigenvalues

The largest and smallest eigenvalues satisfy

$$\lambda_{\text{min}}(A) = \inf_{x \neq 0} \frac{x^T A x}{x^T x}, \qquad \lambda_{\text{max}}(A) = \sup_{x \neq 0} \frac{x^T A x}{x^T x}$$

and consequently $\forall x \in \mathbb{R}^n$ (Rayleigh quotient):

$$\lambda_{\text{min}}(A) x^T x \leq x^T A x \leq \lambda_{\text{max}}(A) x^T x$$

The **condition number** of a nonsingular matrix is defined as

в случае $A \in S^n_{++}$

$\lambda(A) = \sigma(A)$

$$\kappa(A) = \|A\| \|A^{-1}\|$$

If we use spectral matrix norm, we can get:

$$\boxed{\kappa(A) = \frac{\sigma_{\text{max}}(A)}{\sigma_{\text{min}}(A)}}$$

If, moreover, $A \in \mathbb{S}^n_{++}$: $\kappa(A) = \dfrac{\lambda_{\text{max}}(A)}{\lambda_{\text{min}}(A)}$

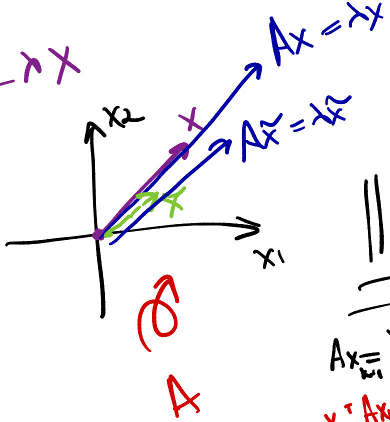
PageRank

A для графа перех. вер-тей

$$Ax^* = 1 \cdot x^*$$

x^* – собств. вектор A, соотв. СЗ $\lambda = 1$

$$Ax = \lambda x$$



SVD

$$\|x\| = 1$$

$$Ax_{k+1} = \lambda x_{k+1} \quad \Big| \; x_{k+1}^T \cdot$$

$$x_{k+1}^T A x_{k+1} = \lambda \cdot x_{k+1}^T \cdot x_{k+1}$$

$$\lambda = \frac{x_{k+1}^T A x_{k+1}}{x_{k+1}^T x_{k+1}} = \langle x_{k+1}, Ax_{k+1} \rangle$$

Power Method:

Вход: $A \in \mathbb{R}^{n \times n}$

Задача: найти СВ(A), соотв. макс СЗ по мод.

Алгоритм:

```
$x_0 = np.randn(n)$
for k in range(n_iters):
    $\tilde{x}_{k+1} = A \cdot x_k$
    $x_{k+1} = \frac{\tilde{x}_{k+1}}{\|\tilde{x}_{k+1}\|}$
```

при этом $\lambda_{k+1} = \langle x_{k+1}, Ax_{k+1} \rangle$

Singular value decomposition

Suppose $A \in \mathbb{R}^{m \times n}$ with rank $A = r$. Then A can be factored as

$$\boxed{A = U\Sigma V^T}$$

$V^T V = I$

унитарная

$U^T U$

unitary

унитарная

$\mathrm{diag}(\sigma_1 \ldots\ldots \sigma_n)$

Singular value decomposition

Suppose $A \in \mathbb{R}^{m \times n}$ with rank $A = r$. Then A can be factored as

$$A = U \Sigma V^T$$

where $U \in \mathbb{R}^{m \times r}$ satisfies $U^T U = I$, $V \in \mathbb{R}^{n \times r}$ satisfies $V^T V = I$, and Σ is a diagonal matrix with $\Sigma = \text{diag}(\sigma_1, ..., \sigma_r)$, such that

Singular value decomposition

Suppose $A \in \mathbb{R}^{m \times n}$ with rank $A = r$. Then A can be factored as

$$A = U \Sigma V^T$$

where $U \in \mathbb{R}^{m \times r}$ satisfies $U^T U = I$, $V \in \mathbb{R}^{n \times r}$ satisfies $V^T V = I$, and Σ is a diagonal matrix with $\Sigma = \text{diag}(\sigma_1, ..., \sigma_r)$, such that

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0.$$

Singular value decomposition

Suppose $A \in \mathbb{R}^{m \times n}$ with rank $A = r$. Then A can be factored as

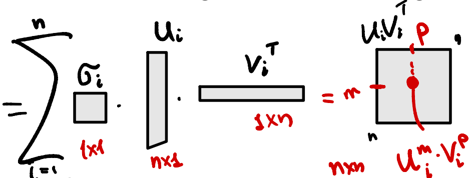
$$A = U \Sigma V^T$$

where $U \in \mathbb{R}^{m \times r}$ satisfies $U^T U = I$, $V \in \mathbb{R}^{n \times r}$ satisfies $V^T V = I$, and Σ is a diagonal matrix with $\Sigma = \text{diag}(\sigma_1, ..., \sigma_r)$, such that

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0.$$

This factorization is called the **singular value decomposition (SVD)** of A. The columns of U are called left singular vectors of A, the columns of V are right singular vectors, and the numbers σ_i are the singular values. The singular value decomposition can be written as

$$\boxed{A = \sum_{i=1}^{r} \sigma_i u_i v_i^T,}$$



where $u_i \in \mathbb{R}^m$ are the left singular vectors, and $v_i \in \mathbb{R}^n$ are the right singular vectors.

Singular value decomposition

Question

Suppose, matrix $A \in \mathbb{S}^n_{++}$. What can we say about the connection between its eigenvalues and singular values?

Singular value decomposition

Question

Suppose, matrix $A \in \mathbb{S}^n_{++}$. What can we say about the connection between its eigenvalues and singular values?

Question

How do the singular values of a matrix relate to its eigenvalues, especially for a symmetric matrix?

Skeleton decomposition

Simple, yet very interesting decomposition is Skeleton decomposition, which can be written in two forms:

$$A = UV^T \quad A = \hat{C}\hat{A}^{-1}\hat{R}$$

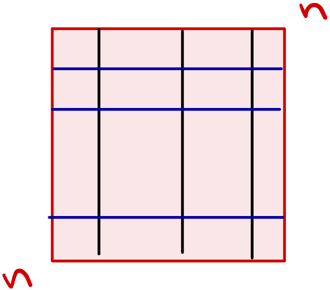
Skeleton decomposition

Simple, yet very interesting decomposition is Skeleton decomposition, which can be written in two forms:

$$A = UV^T \quad A = \hat{C}\hat{A}^{-1}\hat{R}$$

The latter expression refers to the fun fact: you can randomly choose r linearly independent columns of a matrix and any r linearly independent rows of a matrix and store only them with the ability to reconstruct the whole matrix exactly.

если ваша матрица ранга r



всю матрицу можно восстановить по r строк + r столбу

$rn + rn = 2rn$

вместо n^2

Skeleton decomposition

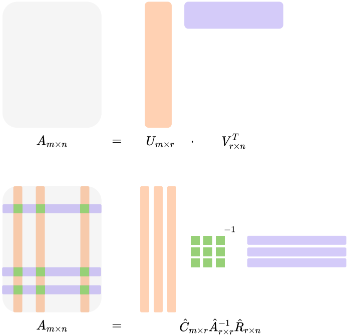
Simple, yet very interesting decomposition is Skeleton decomposition, which can be written in two forms:

$$A = UV^T \quad \boxed{A = \hat{C}\hat{A}^{-1}\hat{R}}$$

The latter expression refers to the fun fact: you can randomly choose r linearly independent columns of a matrix and any r linearly independent rows of a matrix and store only them with the ability to reconstruct the whole matrix exactly.

Use cases for Skeleton decomposition are:

- Model reduction, data compression, and speedup of computations in numerical analysis: given rank-r matrix with $r \ll n, m$ one needs to store $\mathcal{O}((n+m)r) \ll nm$ elements.



$A_{m\times n}$ = $U_{m\times r}$ · $V^T_{r\times n}$
-1
$A_{m\times n}$ = $\hat{C}_{m\times r}\hat{A}^{-1}_{r\times r}\hat{R}_{r\times n}$

Figure 1: Illustration of Skeleton decomposition

Skeleton decomposition

Simple, yet very interesting decomposition is Skeleton decomposition, which can be written in two forms:

$$A = UV^T \quad A = \hat{C}\hat{A}^{-1}\hat{R}$$

The latter expression refers to the fun fact: you can randomly choose r linearly independent columns of a matrix and any r linearly independent rows of a matrix and store only them with the ability to reconstruct the whole matrix exactly.

Use cases for Skeleton decomposition are:

- Model reduction, data compression, and speedup of computations in numerical analysis: given rank-r matrix with $r \ll n, m$ one needs to store $\mathcal{O}((n+m)r) \ll nm$ elements.
- Feature extraction in machine learning, where it is also known as matrix factorization

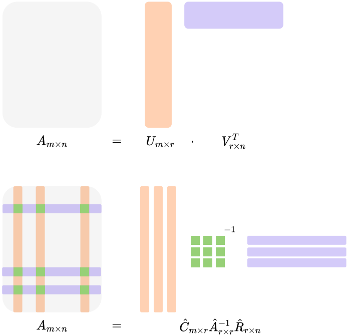


Figure 1: Illustration of Skeleton decomposition

Skeleton decomposition

$\tilde{B} = (\underset{\det B = 0}{B} + 1e\text{-}4 \cdot I) \quad \det \tilde{B} \neq 0$

Simple, yet very interesting decomposition is Skeleton decomposition, which can be written in two forms:

$$A = UV^T \quad A = \hat{C}\hat{A}^{-1}\hat{R}$$

The latter expression refers to the fun fact: you can randomly choose r linearly independent columns of a matrix and any r linearly independent rows of a matrix and store only them with the ability to reconstruct the whole matrix exactly.

Use cases for Skeleton decomposition are:

- Model reduction, data compression, and speedup of computations in numerical analysis: given rank-r matrix with $r \ll n, m$ one needs to store $\mathcal{O}((n+m)r) \ll nm$ elements.
- Feature extraction in machine learning, where it is also known as matrix factorization
- All applications where SVD applies, since Skeleton decomposition can be transformed into truncated SVD form.

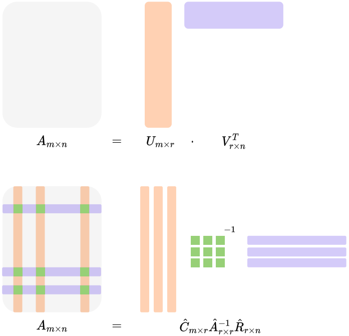
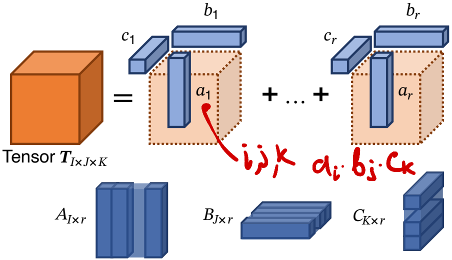


Figure 1: Illustration of Skeleton decomposition

Canonical tensor decomposition

One can consider the generalization of Skeleton decomposition to the higher order data structure, like tensors, which implies representing the tensor as a sum of r primitive tensors.

ранг канонического тензорного разложения



i,j,k $\quad a_i \cdot b_j \cdot c_k$

Figure 2: Illustration of Canonical Polyadic decomposition

Example

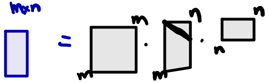
Note, that there are many tensor decompositions: Canonical, Tucker, Tensor Train (TT), Tensor Ring (TR), and others. In the tensor case, we do not have a straightforward definition of *rank* for all types of decompositions. For example, for TT decomposition rank is not a scalar, but a vector.

Determinant and trace

$$A_r = U_r \cdot \Sigma_r \cdot V_r^T$$

The determinant and trace can be expressed in terms of the eigenvalues

$$A_{m\times n} = U \cdot \Sigma \cdot V^T$$



$$\det A = \prod_{i=1}^{n} \lambda_i, \qquad \operatorname{tr} A = \sum_{i=1}^{n} \lambda_i$$

Теория: $W \quad X_{in}$

Практика: $X_{in}^T \cdot W^T$

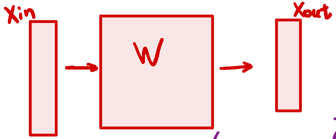
$$UV^T \cdot X_{in}$$

$$(X_{in}^T V \cdot U^T \cdot)$$

The determinant has several appealing (and revealing) properties. For instance,

- $\det A = 0$ if and only if A is singular;

Полносвязный слой:



Связь Skeleton и SVD: truncated

$$A_{m\times n} = \underbrace{U}_{m\times r} \underbrace{\Sigma}_{r\times r} \underbrace{V^T}_{r\times n} = \tilde{U} \cdot \tilde{V}^T$$

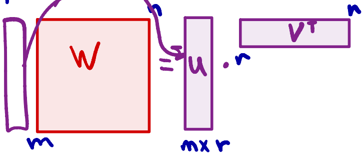
$\cdot\ \tilde{U}_{m\times r} = U\Sigma$

$\cdot\ \tilde{V}^T_{r\times n} = V^T$

$$\Sigma = \begin{pmatrix} \sigma_1 \ldots \sigma_r & 0 \\ 0 & 0\ 0\ 0\ 0 \end{pmatrix}$$

$$X_{out} = \sigma(W \cdot X_{in} + b)$$

$$W = UV^T$$



Determinant and trace

The determinant and trace can be expressed in terms of the eigenvalues

$$\boxed{\text{det}A = \prod_{i=1}^{n} \lambda_i,} \quad \boxed{\text{tr}A = \sum_{i=1}^{n} \lambda_i}$$

The determinant has several appealing (and revealing) properties. For instance,

- $\text{det}A = 0$ if and only if A is singular;
- $\text{det}AB = (\text{det}A)(\text{det}B)$;

Determinant and trace

The determinant and trace can be expressed in terms of the eigenvalues

$$\text{det}A = \prod_{i=1}^{n} \lambda_i, \qquad \text{tr}A = \sum_{i=1}^{n} \lambda_i$$

The determinant has several appealing (and revealing) properties. For instance,

- $\text{det}A = 0$ if and only if A is singular;
- $\text{det}AB = (\text{det}A)(\text{det}B)$;
- $\text{det}A^{-1} = \frac{1}{\text{det}\ A}$.

Determinant and trace

The determinant and trace can be expressed in terms of the eigenvalues

$$\text{det}A = \prod_{i=1}^{n} \lambda_i, \qquad \text{tr}A = \sum_{i=1}^{n} \lambda_i$$

The determinant has several appealing (and revealing) properties. For instance,

- $\text{det}A = 0$ if and only if A is singular;
- $\text{det}AB = (\text{det}A)(\text{det}B)$;
- $\text{det}A^{-1} = \frac{1}{\text{det} \ A}$.

Determinant and trace

The determinant and trace can be expressed in terms of the eigenvalues

$$\text{det}A = \prod_{i=1}^{n} \lambda_i, \qquad \text{tr}A = \sum_{i=1}^{n} \lambda_i$$

The determinant has several appealing (and revealing) properties. For instance,

- $\text{det}A = 0$ if and only if A is singular;
- $\text{det}AB = (\text{det}A)(\text{det}B)$;
- $\text{det}A^{-1} = \frac{1}{\text{det} \ A}$.

Don't forget about the cyclic property of a trace for arbitrary matrices A, B, C, D (assuming, that all dimensions are consistent):

$$\text{tr}(ABCD) = \text{tr}(DABC) = \text{tr}(CDAB) = \text{tr}(BCDA)$$

Determinant and trace

The determinant and trace can be expressed in terms of the eigenvalues

$$\text{det} A = \prod_{i=1}^{n} \lambda_i, \qquad \text{tr} A = \sum_{i=1}^{n} \lambda_i$$

The determinant has several appealing (and revealing) properties. For instance,

- $\text{det} A = 0$ if and only if A is singular;
- $\text{det} AB = (\text{det} A)(\text{det} B)$;
- $\text{det} A^{-1} = \frac{1}{\text{det } A}$.

Don't forget about the cyclic property of a trace for arbitrary matrices A, B, C, D (assuming, that all dimensions are consistent):

$$\boxed{\text{tr}(ABCD) = \text{tr}(DABC) = \text{tr}(CDAB) = \text{tr}(BCDA)}$$

Question

How does the determinant of a matrix relate to its invertibility?

First-order Taylor approximation

The first-order Taylor approximation, also known as the linear approximation, is centered around some point x_0. If $f : \mathbb{R}^n \to \mathbb{R}$ is a differentiable function, then its first-order Taylor approximation is given by:

$$f^I_{x_0}(x) = f(x_0) + \nabla f(x_0)^T (x - x_0)$$

Where:

- $f(x_0)$ is the value of the function at the point x_0.

First-order Taylor approximation

The first-order Taylor approximation, also known as the linear approximation, is centered around some point x_0. If $f : \mathbb{R}^n \to \mathbb{R}$ is a differentiable function, then its first-order Taylor approximation is given by:

$$f_{x_0}^I(x) = f(x_0) + \nabla f(x_0)^T (x - x_0)$$

Where:

- $f(x_0)$ is the value of the function at the point x_0.
- $\nabla f(x_0)$ is the gradient of the function at the point x_0.

First-order Taylor approximation

The first-order Taylor approximation, also known as the linear approximation, is centered around some point x_0. If $f : \mathbb{R}^n \to \mathbb{R}$ is a differentiable function, then its first-order Taylor approximation is given by:

$$f^I_{x_0}(x) = f(x_0) + \nabla f(x_0)^T (x - x_0)$$

Where:

- $f(x_0)$ is the value of the function at the point x_0.
- $\nabla f(x_0)$ is the gradient of the function at the point x_0.

First-order Taylor approximation

The first-order Taylor approximation, also known as the linear approximation, is centered around some point x_0. If $f : \mathbb{R}^n \to \mathbb{R}$ is a differentiable function, then its first-order Taylor approximation is given by:

$$\boxed{f^I_{x_0}(x) = f(x_0) + \nabla f(x_0)^T (x - x_0)}$$

b f k x

Where:

- $f(x_0)$ is the value of the function at the point x_0.
- $\nabla f(x_0)$ is the gradient of the function at the point x_0.

It is very usual to replace the $f(x)$ with $f^I_{x_0}(x)$ near the point x_0 for simple analysis of some approaches.

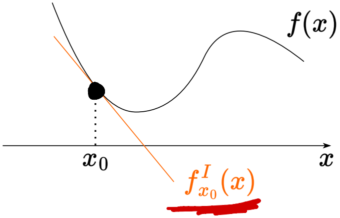


Figure 3: First order Taylor approximation near the point x_0

Second-order Taylor approximation

The second-order Taylor approximation, also known as the quadratic approximation, includes the curvature of the function. For a twice-differentiable function $f : \mathbb{R}^n \to \mathbb{R}$, its second-order Taylor approximation centered at some point x_0 is:

$$f_{x_0}^{II}(x) = f(x_0) + \nabla f(x_0)^T (x - x_0) + \frac{1}{2}(x - x_0)^T \nabla^2 f(x_0)(x - x_0)$$

Where $\nabla^2 f(x_0)$ is the Hessian matrix of f at the point x_0.

Second-order Taylor approximation

The second-order Taylor approximation, also known as the quadratic approximation, includes the curvature of the function. For a twice-differentiable function $f : \mathbb{R}^n \to \mathbb{R}$, its second-order Taylor approximation centered at some point x_0 is:

$$\boxed{f_{x_0}^{II}(x) = f(x_0) + \nabla f(x_0)^T (x - x_0) + \frac{1}{2}(x - x_0)^T \nabla^2 f(x_0)(x - x_0)}$$

Where $\nabla^2 f(x_0)$ is the Hessian matrix of f at the point x_0.
When using the linear approximation of the function is not sufficient one can consider replacing the $f(x)$ with $f_{x_0}^{II}(x)$ near the point x_0. In general, Taylor approximations give us a way to locally approximate functions. The first-order approximation is a plane tangent to the function at the point x_0, while the second-order approximation includes the curvature and is represented by a parabola. These approximations are especially useful in optimization and numerical methods because they provide a tractable way to work with complex functions.

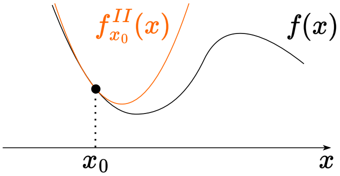


Figure 4: Second order Taylor approximation near the point x_0

Exercises

$$\rho_i = \frac{\partial \log R(x)}{\partial \log x} = \frac{\partial R(x)}{R(x)} \frac{x_i}{\partial x_i} \qquad \sum \rho_i = 1, \rho_i > 0$$

$$\rho_i = \frac{x_i \cdot (\Sigma x)_i}{x^T \Sigma x}$$

- Linear Least Squares

$$\frac{1}{2} x^T \Sigma x - \sum_{i=1}^{n} \rho_i^{des} \log x_i$$

$$f(x) \to \min_{x \in \mathbb{R}^n} \qquad \nabla f = 0$$

$$x^T \Sigma x \, \rho_i = x_i \cdot (\Sigma x)_i$$

Exercises

$$\underbrace{\frac{1}{2}x^T\Sigma x - \sum_{i=1}^{n} \rho_i^{des} \log x_i}_{f} \longrightarrow \min_{x \in \mathbb{R}^n}$$

$$\nabla f = 0$$

$$\frac{\partial}{\partial x_k}\left(\frac{1}{2}x^T\Sigma x - \sum_{i=1}^{n} \rho_i^{des} \log x_i\right) =$$

- Linear Least Squares
- Stupid, but important idea on matrix multiplication

$$x^T\Sigma x = \langle x, \Sigma x \rangle =$$

$$= \sum_{k=1}^{n} \underbrace{x_k \cdot (\Sigma x)_k}_{\rho^k_{des}} = \sum_{k=1}^{n} \rho_k^{des} = 1$$

$$= \frac{1}{\cancel{2}} \cdot \cancel{2}(\Sigma x)_k - \rho_k^{des} \cdot \frac{1}{x_k} = 0$$

$$\boxed{\rho_k^{des} = \frac{(\Sigma x)_k \cdot x_k}{x^T\Sigma x}}$$

Exercises

- Linear Least Squares
- 🐍Stupid, but important idea on matrix multiplication
- 🐍Problems

Exercises

- Linear Least Squares
- Stupid, but important idea on matrix multiplication
- Problems
- How to calculate minimum and maximum eigenvalue of the hessian matrix of linear least squares problem? What about binary logistic regression?

Applications > Principal component analysis

Principal component analysis

1 Intuition

Imagine, that you have a dataset of points. Your goal is to choose orthogonal axes, that describe your data the most informative way. To be precise, we choose first axis in such a way, that maximize the variance (expressiveness) of the projected data. All the following axes have to be orthogonal to the previously chosen ones, while satisfy largest possible variance of the projections.

Let’s take a look at the simple 2d data. We have a set of blue points on the plane. We can easily see that the projections on the first axis (red dots) have maximum variance at the final position of the animation. The second (and the last) axis should be orthogonal to the previous one.

? source

This idea could be used in a variety of ways. For example, it might happen, that projection of complex data on the principal plane (only 2 components) bring you enough intuition for clustering. The picture below plots projection of the labeled dataset onto the first to principal components (PCs), we can clearly see, that only two vectors (these PCs) would be enough to differ Finnish people from Italian in particular dataset (celiac disease (Dubois et al. 2010)) source

НОРМАЛИЗАЦИЯ

2 Problem

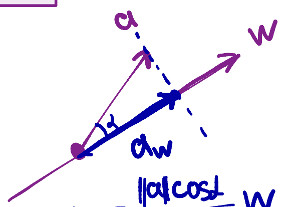
The first component should be defined in order to maximize variance. Suppose, we’ve already normalized the data, i.e. $\boxed{\sum_i a_i = 0}$, then sample variance will become the sum of all squared projections of data points to our vector $\mathbf{w}_{(1)}$, which implies the following optimization problem:

$$\boxed{\mathbf{w}_{(1)} = \underset{\|\mathbf{w}\|=1}{\arg\max} \left\{ \sum_i \left(\mathbf{a}_{(i)}^\top \cdot \mathbf{w} \right)^2 \right\}}$$

$a_i \in \mathbb{R}^n$

$w_1 \in \mathbb{R}^n$

$$\underset{w}{\arg\max}\ \mathrm{Var}\ (\text{проекции})$$



$$\cos\alpha = \frac{\langle a, w\rangle}{\|a\| \cdot \|w\|}$$

$$a_w = \frac{\|a\| \cos\alpha}{\|w\|} w = $$

$$= \frac{\langle a, w\rangle \cdot \cancel{\|a\|}}{\cancel{\|a\|} \cdot \|w\| \cdot \|w\|} \cdot w =$$

$\|w\| = 1$

$$= \langle a, w\rangle \cdot w$$

or

$\sum_{i=1}^{n}(a_i^T w)^2$

$$\mathbf{w}_{(1)} = \arg\max_{\|\mathbf{w}\|=1} \{\|\mathbf{A}\mathbf{w}\|^2\} = \arg\max_{\|\mathbf{w}\|=1} \{\mathbf{w}^\top \mathbf{A}^\top \mathbf{A}\mathbf{w}\}$$

A m

since we are looking for the unit vector, we can reformulate the problem:

$\|Aw\|^2 = \langle Aw, Aw \rangle = (Aw)^T \cdot Aw = w^T A^T A w$

$$\boxed{\mathbf{w}_{(1)} = \arg\max \left\{ \frac{\mathbf{w}^\top \mathbf{A}^\top \mathbf{A}\mathbf{w}}{\mathbf{w}^\top \mathbf{w}} \right\}}$$

$\leftarrow$ задача поиска СВ матрицы $A^T A$

It is known, that for positive semidefinite matrix $A^\top A$ such vector is nothing else, but eigenvector of $A^\top A$, which corresponds to the largest eigenvalue. The following components will give you the same results (eigenvectors).

$\frac{w^T A^T A \cdot w}{w^T w} = \lambda$

So, we can conclude, that the following mapping:

$$\underset{n\times k}{\Pi} = \underset{n\times d}{A} \cdot \underset{d\times k}{W}$$

РАЗМЕРНОСТЬ ДАННЫХ

РАЗМЕРНОСТЬ ПРОЕКЦИИ

кол-во данных

$w^T A^T A w = \lambda \cdot w^T w$

$w^T \cdot$

$w \neq 0$

describes the projection of data onto the k principal components, where W contains first (by the size of eigenvalues) k eigenvectors of $A^\top A$.

$$\boxed{A^T A w = \lambda \cdot w}$$

Now we'll briefly derive how SVD decomposition could lead us to the PCA.

$\Pi =$

Firstly, we write down SVD decomposition of our matrix:

$$\underline{A = U\Sigma W^\top}$$

and to its transpose:

$$\begin{aligned}
\underline{A^\top} &= (U\Sigma W^\top)^\top \\
&= (W^\top)^\top \Sigma^\top U^\top \\
&= W\Sigma^\top U^\top \\
&= \underline{W\Sigma U^\top}
\end{aligned}$$

Then, consider matrix $AA^\top$:

$$
\begin{aligned}
A^\top A &= (W\Sigma U^\top)(U\Sigma V^\top) \\
&= W\Sigma I\Sigma W^\top \\
&= W\Sigma\Sigma W^\top \\
&= W\Sigma^2 W^\top
\end{aligned}
$$

Which corresponds to the eigendecomposition of matrix $A^\top A$, where W stands for the matrix of eigenvectors of $A^\top A$, while Σ^2 contains eigenvalues of $A^\top A$.

At the end:

$$
\begin{aligned}
\Pi &= A \cdot W = \\
&= U\Sigma \overset{=I}{W^\top W} = \underline{U\Sigma}
\end{aligned}
$$

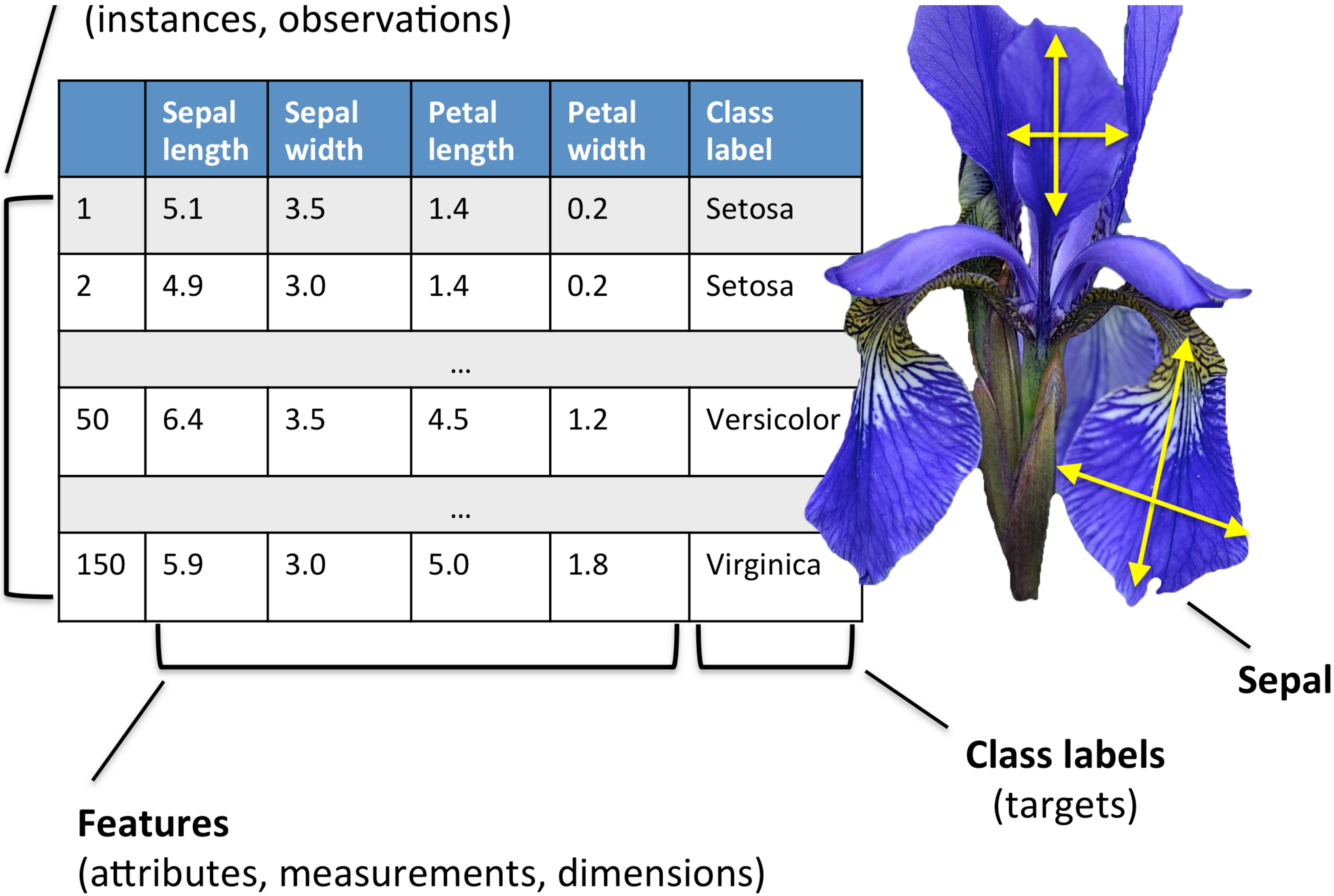
The latter formula provide us with easy way to compute PCA via SVD with any number of principal components:

$$\boxed{\Pi_r = U_r\Sigma_r}$$

3 Examples

3.1 🌼 Iris dataset

Consider the classical Iris dataset



(instances, observations)

	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
…					
50	6.4	3.5	4.5	1.2	Versicolor
…					
150	5.9	3.0	5.0	1.8	Virginica

Sepal

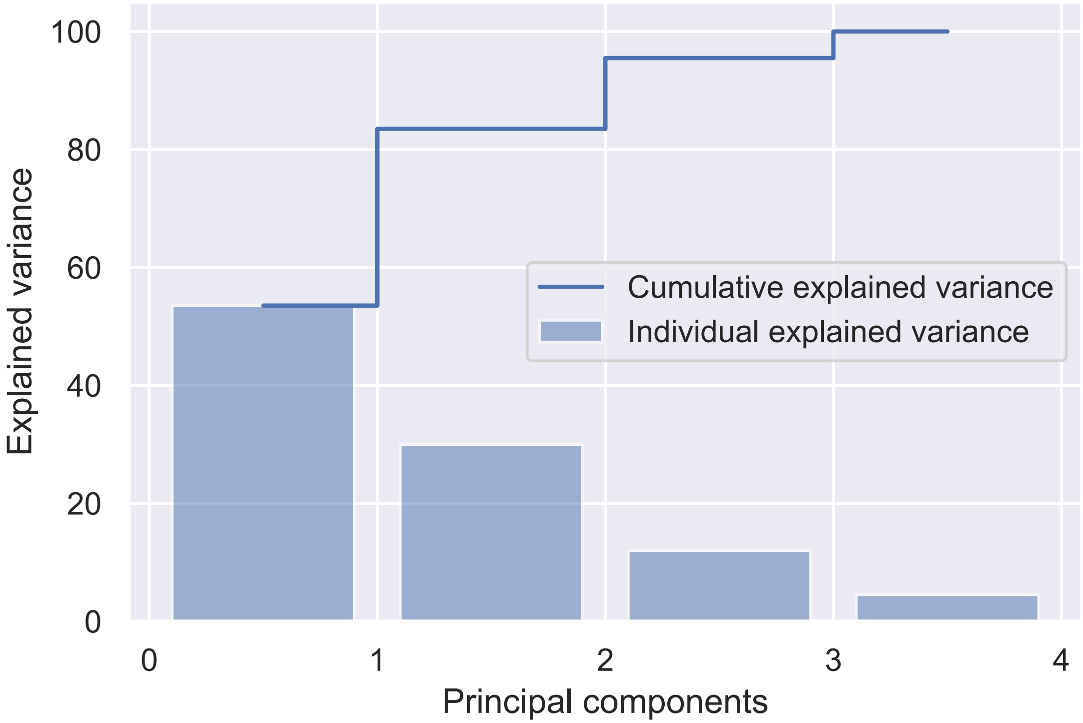
Class labels
(targets)

Features
(attributes, measurements, dimensions)

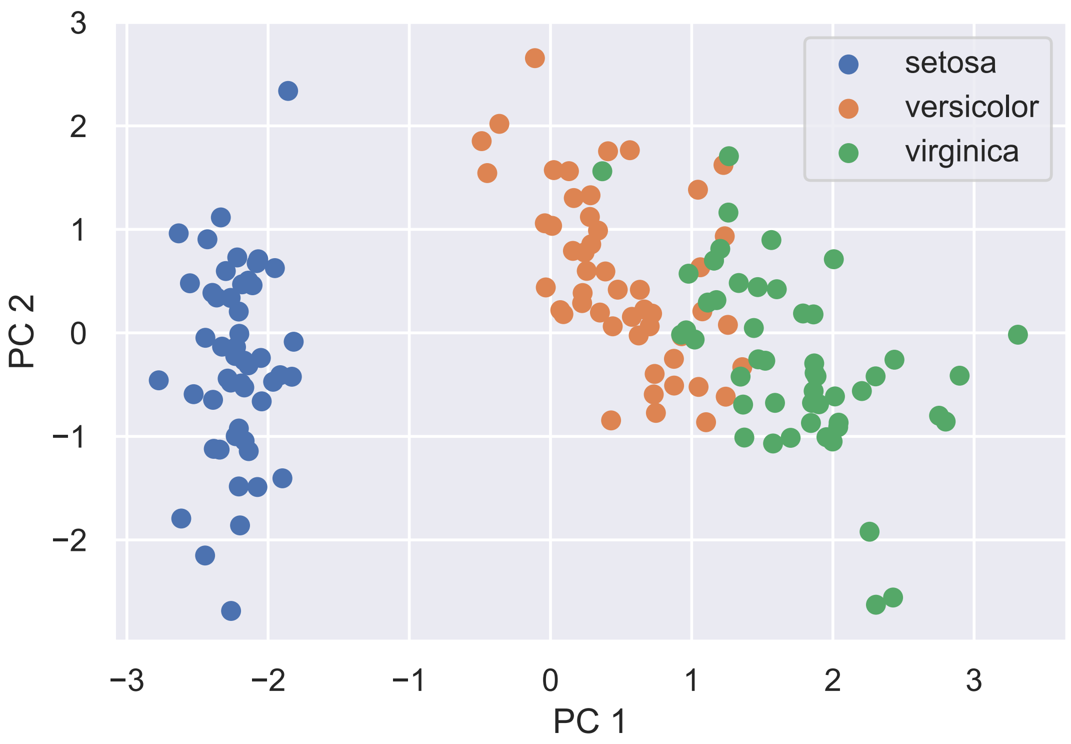
Illustration

source

We have the dataset matrix $A \in \mathbb{R}^{150 \times 4}$



Illustration



Illustration

Любая $A = U\Sigma V^T$

eigen faces

$$A_{n\times d} = U\Sigma W^T$$

$W^k_{k\times d}$

k собств. векторов

$$2 \to \Pi_{n\times k} = A_{n\times d} \cdot W^T_{d\times k}$$

4 Code

Open In Colab{: .btn }

5 Related materials

- Wikipedia
- Blog post
- Blog post