

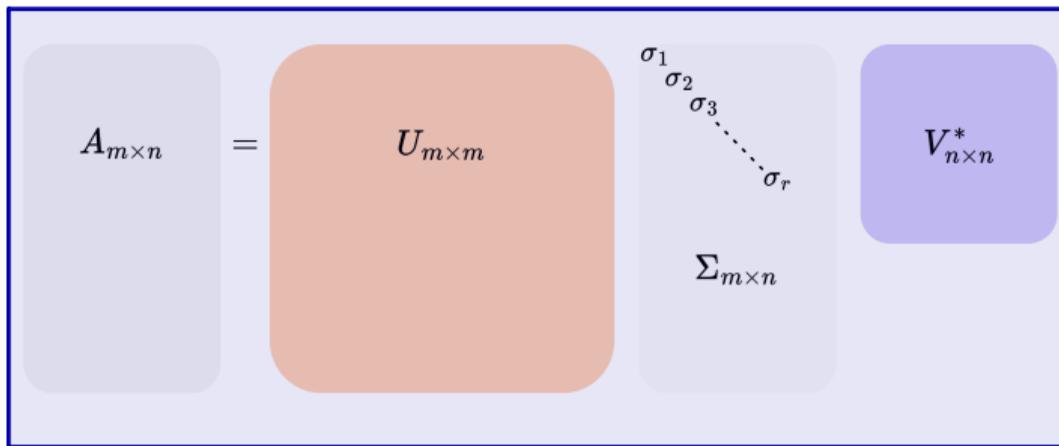
**SVD. Eigenfaces. PCA. Линейные системы**

**Даня Меркулов**

МФТИ. AI360

## Сингулярное разложение матрицы

## Сингулярное разложение матрицы



Для любой матрицы  $A \in \mathbb{R}^{m \times n}$  существует разложение:

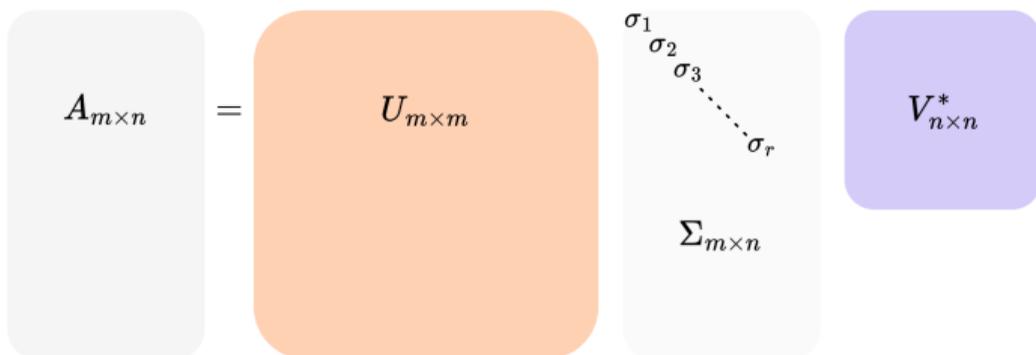
$$A = U \Sigma V^*,$$

где

- $U \in \mathbb{R}^{m \times m}$  - унитарная матрица левых сингулярных векторов

$U, \text{sigma}, v^T = \text{np.linalg.svd}(A)$

# Сингулярное разложение матрицы



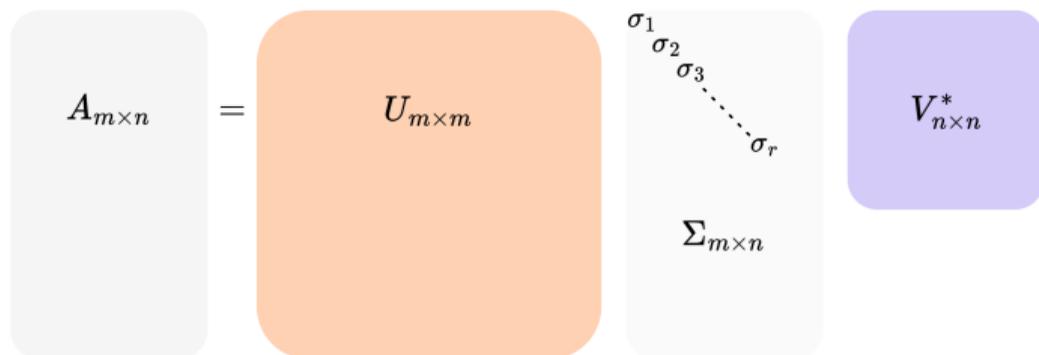
Для любой матрицы  $A \in \mathbb{R}^{m \times n}$  существует разложение:

$$A = U \Sigma V^*,$$

где

- $U \in \mathbb{R}^{m \times m}$  - унитарная матрица левых сингулярных векторов
  - $\Sigma \in \mathbb{R}^{m \times n}$  - диагональная матрица сингулярных чисел
- $$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{\min(m,n)} \geq 0$$

# Сингулярное разложение матрицы



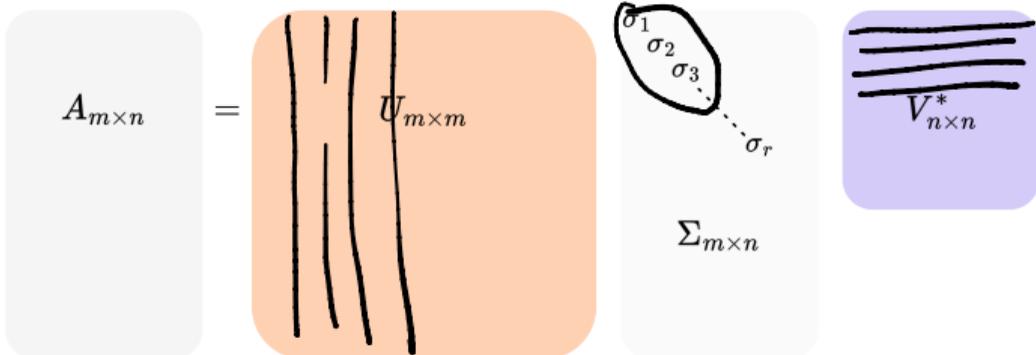
Для любой матрицы  $A \in \mathbb{R}^{m \times n}$  существует разложение:

$$A = U \Sigma V^*,$$

где

- $U \in \mathbb{R}^{m \times m}$  - унитарная матрица левых сингулярных векторов
- $\Sigma \in \mathbb{R}^{m \times n}$  - диагональная матрица сингулярных чисел  
 $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{\min(m,n)} \geq 0$
- $V \in \mathbb{R}^{n \times n}$  - унитарная матрица правых сингулярных векторов

# Сингулярное разложение матрицы



$$A = \sum_{i=1}^r \sigma_i \cdot u_i \cdot v_i^*$$

$$A_{y \times z} = \sum_{i=1}^{y \wedge z} \sigma_i \cdot u_i \cdot v_i^*$$

Для любой матрицы  $A \in \mathbb{R}^{m \times n}$  существует разложение:

$$A = U \Sigma V^*,$$

где

- $U \in \mathbb{R}^{m \times m}$  - унитарная матрица левых сингулярных векторов
- $\Sigma \in \mathbb{R}^{m \times n}$  - диагональная матрица сингулярных чисел  
 $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{\min(m,n)} \geq 0$
- $V \in \mathbb{R}^{n \times n}$  - унитарная матрица правых сингулярных векторов
- Сингулярные числа единственны. Если все сингулярные числа различны, то разложение единственно с точностью до унитарной диагональной матрицы  $D$ :  
 $U \Sigma V^* = U D \Sigma (V D)^* = U \Sigma V^*$ .

## Теорема Экарта-Янга

truncated SVD  
усечённое SVD

Наилучшее приближение низкого ранга может быть вычислено с помощью SVD.

💡 Пусть  $r < \text{rank}(A)$ ,  $A_r = U_r \Sigma_r V_r^*$ . Тогда

$$B = A_r$$

$$\min_{\text{rank}(B)=r} \|A - B\|_2 = \|A - A_r\|_2 = \sigma_{r+1}.$$

То же самое верно для  $\|\cdot\|_F$ , но  $\|A - A_r\|_F = \sqrt{\sigma_{r+1}^2 + \dots + \sigma_{\min(n,m)}^2}$ .

**Следствие:** вычисление наилучшего приближения ранга  $r$  эквивалентно установке  $\sigma_{r+1} = 0, \dots, \sigma_K = 0$ .

Ошибка

$$\min_{A_r} \|A - A_r\|_2 = \sigma_{r+1}, \quad \min_{A_r} \|A - A_r\|_F = \sqrt{\sigma_{r+1}^2 + \dots + \sigma_K^2}$$

вот почему важно смотреть на скорость убывания сингулярных значений.

## Пример 1

Найдите SVD следующей матрицы:

$m > n$

$$A = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

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

$m \times n$        $n \times n$        $n \times n$   
 $3 \times 1$        $1 \times 1$        $1 \times 1$

$$\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \cdot (1) \cdot (1)$$

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

$$U^T U = I$$

$$V^T V = I$$

$$U^T U = I$$

## Пример 1

Найдите SVD следующей матрицы:

$$A = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

$$\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} = U \Sigma V^T$$

The image shows the SVD decomposition of matrix A. The matrix A is written as a column vector  $\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$ . This is equated to the product of three matrices: U, Σ, and V<sup>T</sup>. Matrix U is a 3x1 column vector  $\begin{pmatrix} 1/\sqrt{14} \\ 2/\sqrt{14} \\ 3/\sqrt{14} \end{pmatrix}$ , with each element circled in red. Matrix Σ is a 3x1 column vector  $\begin{pmatrix} \sqrt{14} \\ 1 \\ 0 \end{pmatrix}$ , with the first element circled in black and the second element circled in red. Matrix V<sup>T</sup> is a 1x3 row vector  $\begin{pmatrix} 1 & 0 & 0 \end{pmatrix}$ , with the first element circled in red.

$$U^T U = I$$
$$\begin{pmatrix} 1/\sqrt{14} & 2/\sqrt{14} & 3/\sqrt{14} \end{pmatrix} \begin{pmatrix} 1/\sqrt{14} \\ 2/\sqrt{14} \\ 3/\sqrt{14} \end{pmatrix} = 1$$

## Пример 1

Найдите SVD следующей матрицы:

$$A = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

### Решение

1. Простейшая форма SVD выглядит так:

$$A = U\Sigma V^T = \begin{bmatrix} \frac{1}{\sqrt{14}} \\ \frac{2}{\sqrt{14}} \\ \frac{3}{\sqrt{14}} \end{bmatrix} [\sqrt{14}] [1]$$

## Пример 1

Найдите SVD следующей матрицы:

$$A = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

### Решение

1. Простейшая форма SVD выглядит так:

$$A = U\Sigma V^T = \begin{bmatrix} \frac{1}{\sqrt{14}} \\ \frac{2}{\sqrt{14}} \\ \frac{3}{\sqrt{14}} \end{bmatrix} [\sqrt{14}] [1]$$

2. Однако, если вы хотите использовать полную форму с квадратными сингулярными матрицами:

$$A = U\Sigma V^T = \begin{bmatrix} \frac{1}{\sqrt{14}} & \frac{1}{\sqrt{3}} & \frac{-5}{\sqrt{42}} \\ \frac{2}{\sqrt{14}} & \frac{1}{\sqrt{3}} & \frac{4}{\sqrt{42}} \\ \frac{3}{\sqrt{14}} & \frac{-1}{\sqrt{3}} & \frac{-1}{\sqrt{42}} \end{bmatrix} \begin{pmatrix} \sqrt{14} \\ 0 \\ 0 \end{pmatrix} [1]$$

*diag*

## Пример 1

Найдите SVD следующей матрицы:

$$A = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

### Решение

1. Простейшая форма SVD выглядит так:

$$A = U\Sigma V^T = \begin{bmatrix} \frac{1}{\sqrt{14}} \\ \frac{2}{\sqrt{14}} \\ \frac{3}{\sqrt{14}} \end{bmatrix} [\sqrt{14}] [1]$$

2. Однако, если вы хотите использовать полную форму с квадратными сингулярными матрицами:

$$A = U\Sigma V^T = \begin{bmatrix} \frac{1}{\sqrt{14}} & \frac{1}{\sqrt{3}} & \frac{-5}{\sqrt{42}} \\ \frac{2}{\sqrt{14}} & \frac{1}{\sqrt{3}} & \frac{4}{\sqrt{42}} \\ \frac{3}{\sqrt{14}} & \frac{-1}{\sqrt{3}} & \frac{-1}{\sqrt{42}} \end{bmatrix} \begin{bmatrix} \sqrt{14} \\ 0 \\ 0 \end{bmatrix} [1]$$

3. Вычислим  $A^T A$ :

$$A^T A = [1 \quad 2 \quad 3] \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = 1^2 + 2^2 + 3^2 = 14.$$

Сингулярные значения  $\sigma_i$  являются квадратными корнями из собственных значений  $A^T A$ . Поскольку  $A^T A$  является  $1 \times 1$  матрицей со значением 14, сингулярное значение равно  $\sigma = \sqrt{14}$ .

## Пример 1

Найдите SVD следующей матрицы:

$$A = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

### Решение

1. Простейшая форма SVD выглядит так:

$$A = U\Sigma V^T = \begin{bmatrix} \frac{1}{\sqrt{14}} \\ \frac{2}{\sqrt{14}} \\ \frac{3}{\sqrt{14}} \end{bmatrix} [\sqrt{14}] [1]$$

2. Однако, если вы хотите использовать полную форму с квадратными сингулярными матрицами:

$$A = U\Sigma V^T = \begin{bmatrix} \frac{1}{\sqrt{14}} & \frac{1}{\sqrt{3}} & \frac{-5}{\sqrt{42}} \\ \frac{2}{\sqrt{14}} & \frac{1}{\sqrt{3}} & \frac{4}{\sqrt{42}} \\ \frac{3}{\sqrt{14}} & \frac{-1}{\sqrt{3}} & \frac{-1}{\sqrt{42}} \end{bmatrix} \begin{bmatrix} \sqrt{14} \\ 0 \\ 0 \end{bmatrix} [1]$$

3. Вычислим  $A^T A$ :

$$A^T A = [1 \quad 2 \quad 3] \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = 1^2 + 2^2 + 3^2 = 14.$$

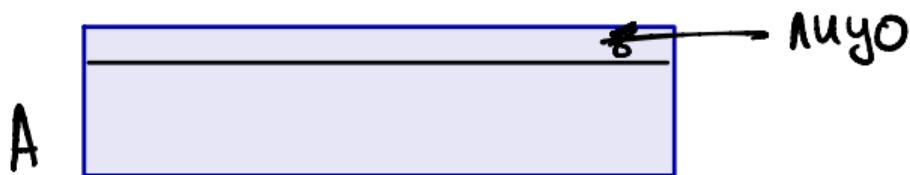
Сингулярные значения  $\sigma_i$  являются квадратными корнями из собственных значений  $A^T A$ . Поскольку  $A^T A$  является  $1 \times 1$  матрицей со значением 14, сингулярное значение равно  $\sigma = \sqrt{14}$ .

4. Поскольку  $V$  является  $n \times n$  ортогональной матрицей ( $1 \times 1$  в этом случае), она может быть  $V = [1]$  (или  $V = [-1]$ ). Мы выбираем  $V = [1]$ .

## Пример 2

Решите упражнение Eigenfaces

$$A_{1140 \times 1850} = U_{1140 \times 110} \sum_{1140 \times 1850} V^T_{1850 \times 1850}$$

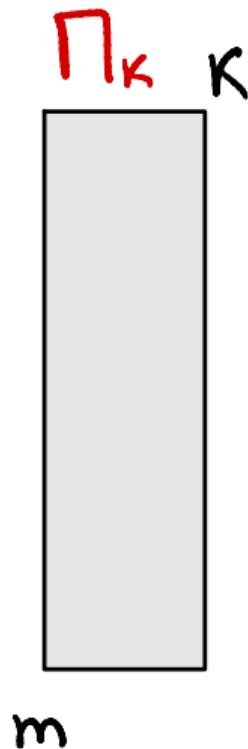
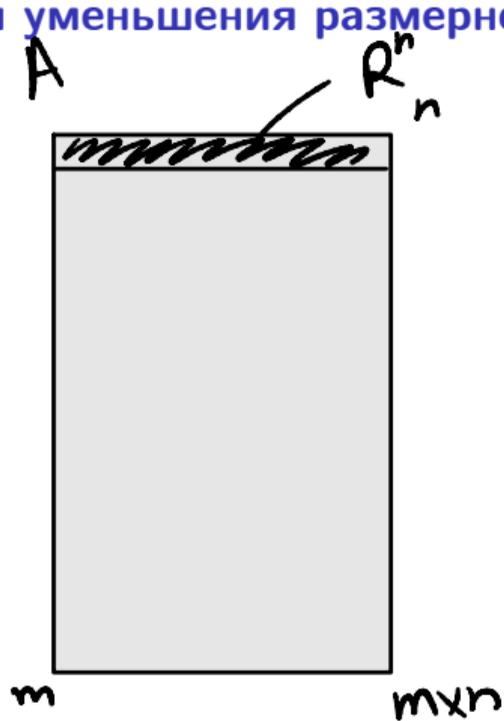


$$A_r \leftarrow \sum_{i=1}^r \sigma_i u_i v_i^T$$

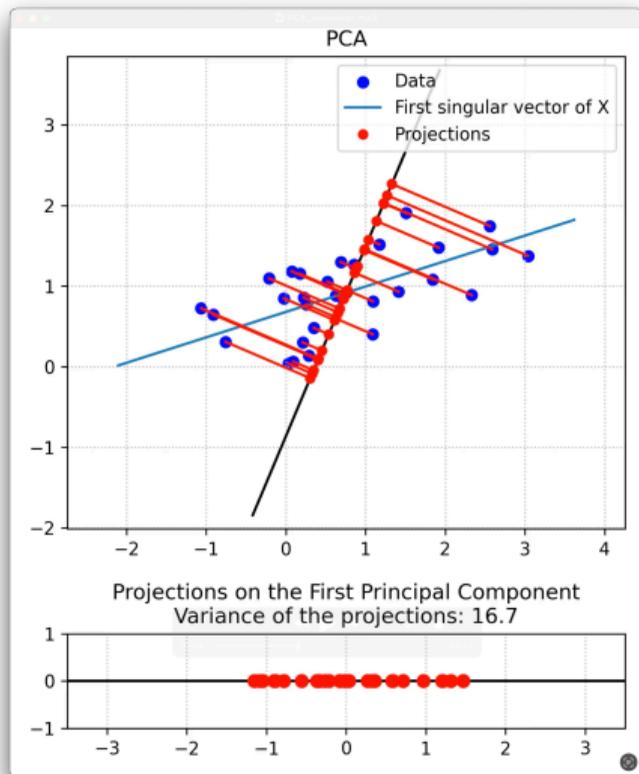


# Метод главных компонент

# Общая идея уменьшения размерности

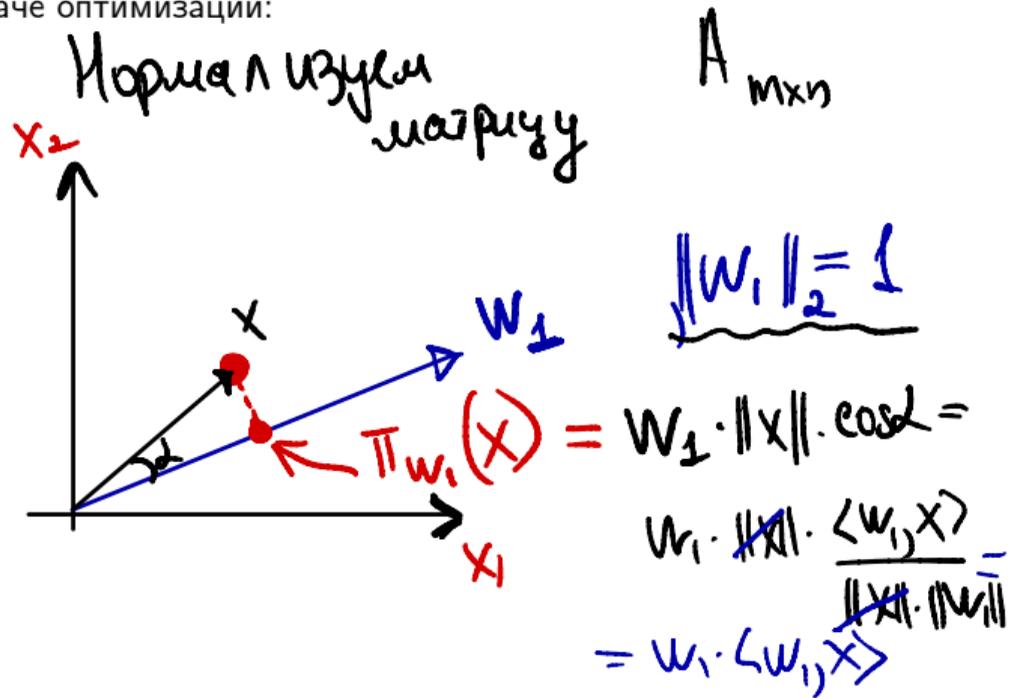


## Метод главных компонент как задача оптимизации

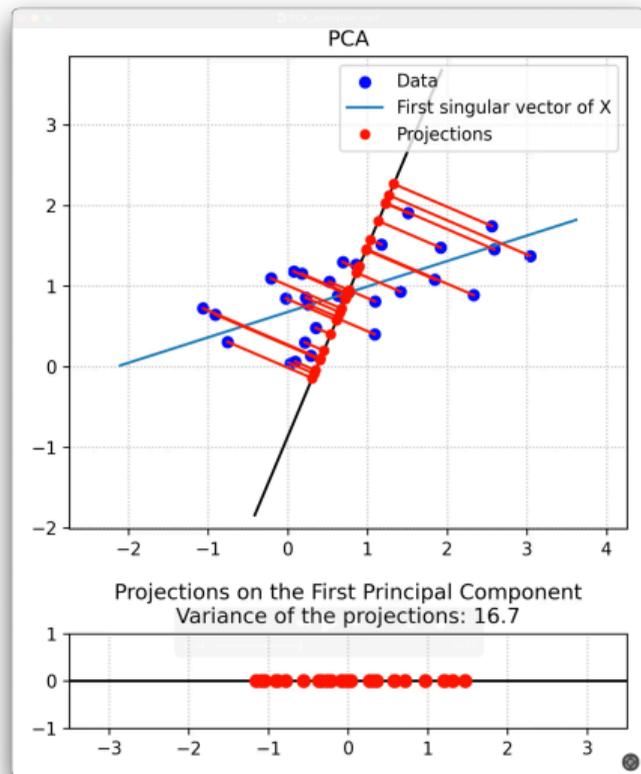


Первая компонента должна быть определена так, чтобы максимизировать дисперсию (вариабельность) проекции.

Предположим, что мы уже нормализовали данные, т.е.  $\sum_i a_i = 0$ , тогда дисперсия выборки станет суммой всех квадратов проекций точек данных на наш вектор  $w_{(1)}$ , что приводит к следующей задаче оптимизации:



## Метод главных компонент как задача оптимизации

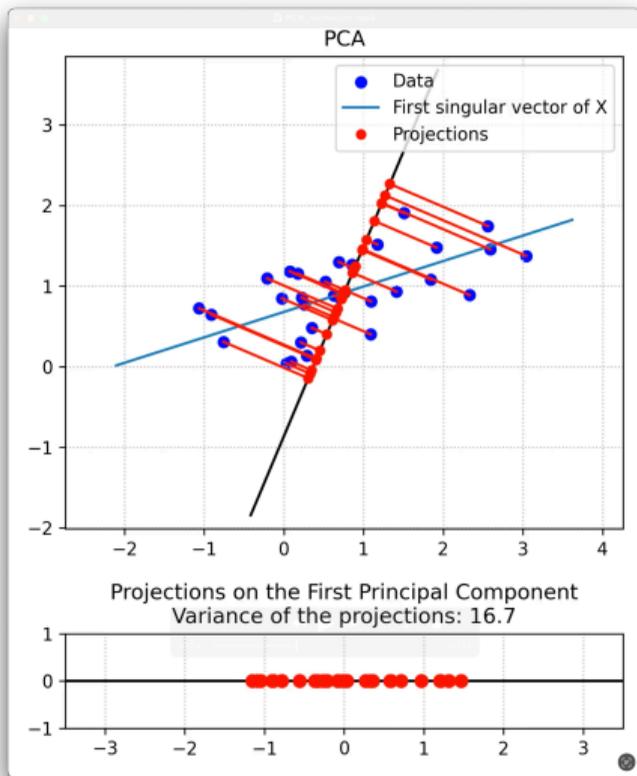


Первая компонента должна быть определена так, чтобы максимизировать дисперсию (вариабельность) проекции. Предположим, что мы уже нормализовали данные, т.е.  $\sum_i a_i = 0$ , тогда дисперсия выборки станет суммой всех квадратов проекции точек данных на наш вектор  $w_{(1)}$ , что приводит к следующей задаче оптимизации:

$$w_{(1)} = \arg \max_{\|w\|=1} \left\{ \sum_i (a_{(i)}^\top \cdot w)^2 \right\}$$

Hand-drawn diagram illustrating the variance formula. A horizontal axis has four red dots representing data points. A blue arrow points upwards from the axis, labeled with the origin '0'. The formula  $Q_{\text{дисперсия}} X = \mathbb{E} X^2 - (\mathbb{E} X)^2$  is written in red. A blue arrow points from the origin '0' to the term  $(\mathbb{E} X)^2$ . Below the axis, the formula  $\sum (a_i^\top w)^2$  is written in blue, with a blue arrow pointing from the origin '0' to this formula.

## Метод главных компонент как задача оптимизации

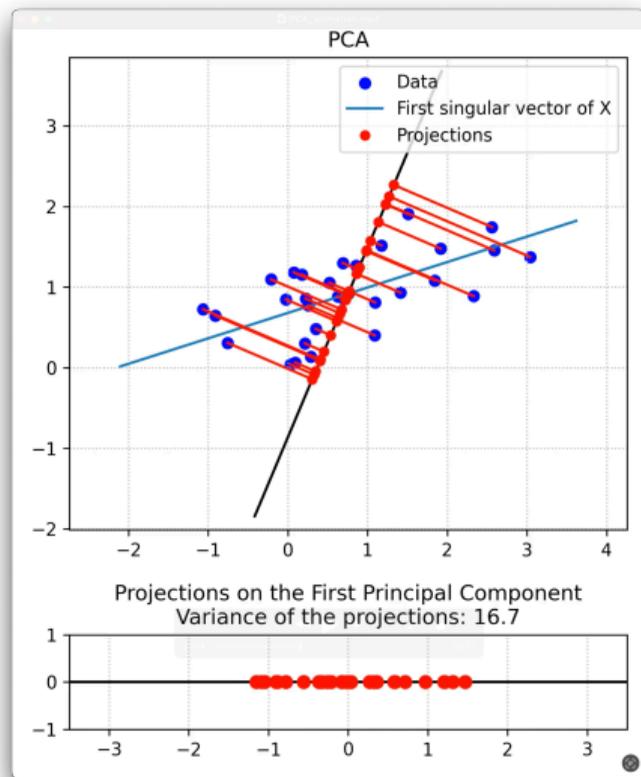


Первая компонента должна быть определена так, чтобы максимизировать дисперсию (вариабельность) проекции. Предположим, что мы уже нормализовали данные, т.е.  $\sum_i a_i = 0$ , тогда дисперсия выборки станет суммой всех квадратов проекций точек данных на наш вектор  $w_{(1)}$ , что приводит к следующей задаче оптимизации:

$$A w \quad \begin{matrix} m \times n & n \times 1 \end{matrix} \quad w_{(1)} = \arg \max_{\|w\|=1} \left\{ \sum_i (a_{(i)}^\top \cdot w)^2 \right\}$$
$$w_{(1)} = \arg \max_{\|w\|=1} \{ \|A w\|^2 \} = \arg \max_{\|w\|=1} \{ w^\top A^\top A w \}$$

Red arrows indicate the flow of the derivation from the first equation to the second, and from the second to the third.

## Метод главных компонент как задача оптимизации



Первая компонента должна быть определена так, чтобы максимизировать дисперсию (вариабельность) проекции. Предположим, что мы уже нормализовали данные, т.е.  $\sum_i a_i = 0$ , тогда дисперсия выборки станет суммой всех квадратов проекций точек данных на наш вектор  $\mathbf{w}_{(1)}$ , что приводит к следующей задаче оптимизации:

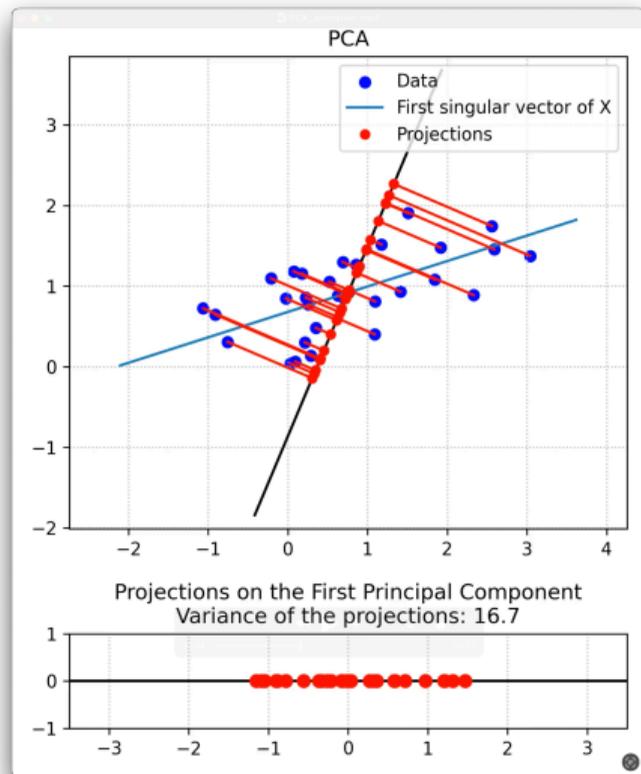
$$\mathbf{w}_{(1)} = \arg \max_{\|\mathbf{w}\|=1} \left\{ \sum_i (\mathbf{a}_{(i)}^\top \cdot \mathbf{w})^2 \right\}$$

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

так как мы ищем единичный вектор, мы можем переформулировать задачу:

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

## Метод главных компонент как задача оптимизации



Первая компонента должна быть определена так, чтобы максимизировать дисперсию (вариабельность) проекции. Предположим, что мы уже нормализовали данные, т.е.  $\sum_i a_i = 0$ , тогда дисперсия выборки станет суммой всех квадратов проекций точек данных на наш вектор  $\mathbf{w}_{(1)}$ , что приводит к следующей задаче оптимизации:

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

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

так как мы ищем единичный вектор, мы можем переформулировать задачу:

$$\max \frac{\mathbf{x}^\top \mathbf{B} \mathbf{x}}{\mathbf{x}^\top \mathbf{x}}$$

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

Известно, что для положительно полуопределенной матрицы  $\mathbf{A}^\top \mathbf{A}$  такой вектор это **собственный вектор**  $\mathbf{A}^\top \mathbf{A}$ , соответствующий наибольшему собственному значению.

## Вывод метода

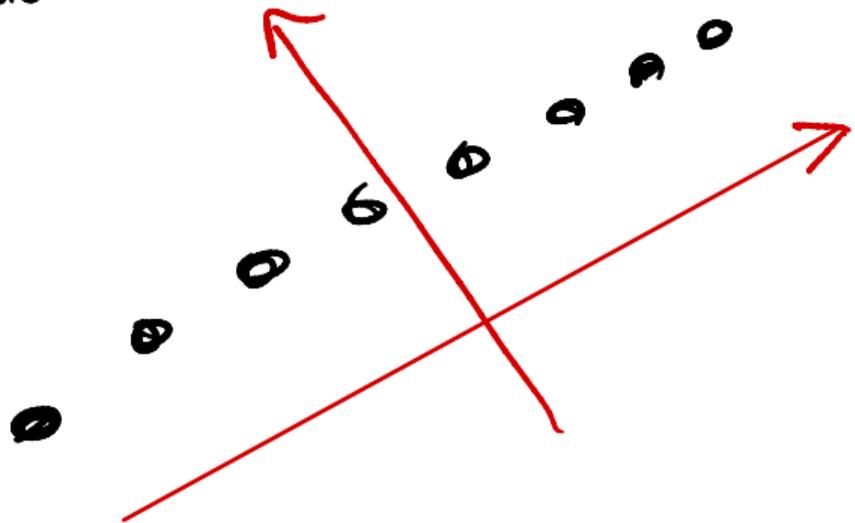
Таким образом, мы можем заключить, что следующее отображение:

$$\Pi_{n \times k} = A_{n \times d} \cdot W_{d \times k}$$

↑  
сжатые

← от U и W сжатие

$$A_{m \times n} = U_{m \times m} \Sigma_{m \times n} V^T_{n \times n}$$
$$A^T = V \Sigma U^T$$



## Вывод метода

Таким образом, мы можем заключить, что следующее отображение:

$$\Pi_{n \times k} = A_{n \times d} \cdot W_{d \times k}$$

описывает проекцию данных на  $k$  главных компонент, где  $W$  содержит первые (по величине собственных значений)  $k$  собственных векторов  $A^T A$ .

## Вывод метода

Таким образом, мы можем заключить, что следующее отображение:

$$\Pi_{n \times k} = A_{n \times d} \cdot W_{d \times k}$$

описывает проекцию данных на  $k$  главных компонент, где  $W$  содержит первые (по величине собственных значений)  $k$  собственных векторов  $A^T A$ .

Теперь мы кратко выведем, как SVD может привести нас к PCA.

Сначала мы запишем SVD нашей матрицы:

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

## Вывод метода

Таким образом, мы можем заключить, что следующее отображение:

$$\Pi_{n \times k} = A \cdot W_{\substack{n \times d \\ d \times k}}$$

описывает проекцию данных на  $k$  главных компонент, где  $W$  содержит первые (по величине собственных значений)  $k$  собственных векторов  $A^T A$ .

Теперь мы кратко выведем, как SVD может привести нас к PCA.

Сначала мы запишем SVD нашей матрицы:

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

и транспонируем его:

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

## Вывод метода

Таким образом, мы можем заключить, что следующее отображение:

$$\Pi_{n \times k} = A \cdot W_{n \times d \quad d \times k}$$

описывает проекцию данных на  $k$  главных компонент, где  $W$  содержит первые (по величине собственных значений)  $k$  собственных векторов  $A^T A$ .

Теперь мы кратко выведем, как SVD может привести нас к PCA.

Сначала мы запишем SVD нашей матрицы:

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

и транспонируем его:

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

сингулярные  
знач.  $A$

Теперь рассмотрим матрицу  $AA^T$ :

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

собств.  
энер. матрица  
 $A^T A$

которая соответствует разложению матрицы  $A^T A$ , где  $W$  - матрица собственных векторов  $A^T A$ , а  $\Sigma^2$  содержит собственные значения  $A^T A$ .

содержит  $n$  собственных  
векторов матрицы  
 $A^T A$

## Вывод метода

Таким образом, мы можем заключить, что следующее отображение:

$$\Pi = A \cdot W$$

$n \times k$       $n \times d$     $d \times k$

описывает проекцию данных на  $k$  главных компонент, где  $W$  содержит первые (по величине собственных значений)  $k$  собственных векторов  $A^T A$ .

Теперь мы кратко выведем, как SVD может привести нас к PCA.

Сначала мы запишем SVD нашей матрицы:

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

и транспонируем его:

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

Теперь рассмотрим матрицу  $AA^T$ :

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

которая соответствует разложению матрицы  $A^T A$ , где  $W$  - матрица собственных векторов  $A^T A$ , а  $\Sigma^2$  содержит собственные значения  $A^T A$ .

В итоге:

$$\begin{aligned} \Pi &= A \cdot W = \\ &= U \Sigma W^T W = U \Sigma \end{aligned}$$

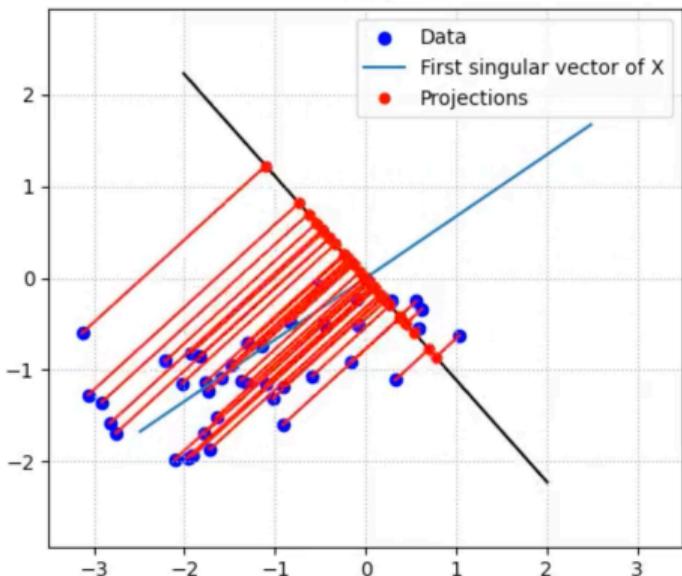
Последняя формула дает нам простой способ вычислить PCA через SVD с любым количеством главных компонент:

$$\Pi_r = U_r \Sigma_r$$

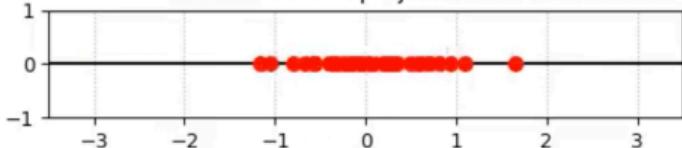
# PCA. Упражнение 1

Что могло пойти не так с этим PCA?

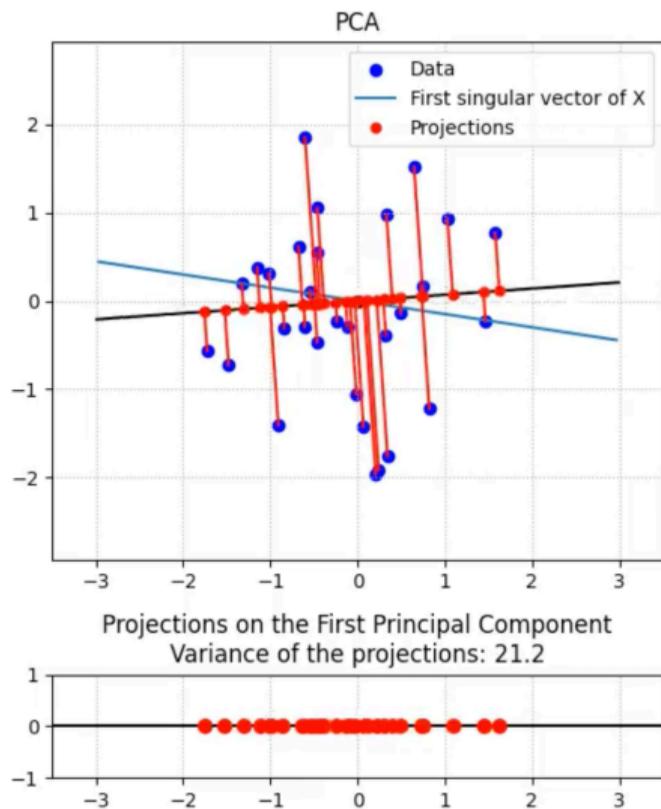
PCA



Projections on the First Principal Component  
Variance of the projections: 13.2



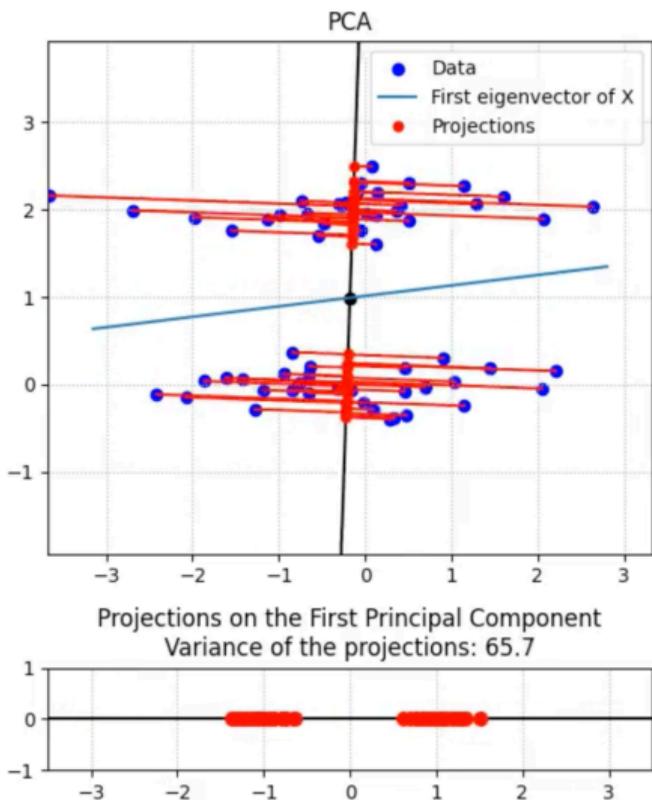
## PCA. Упражнение 2



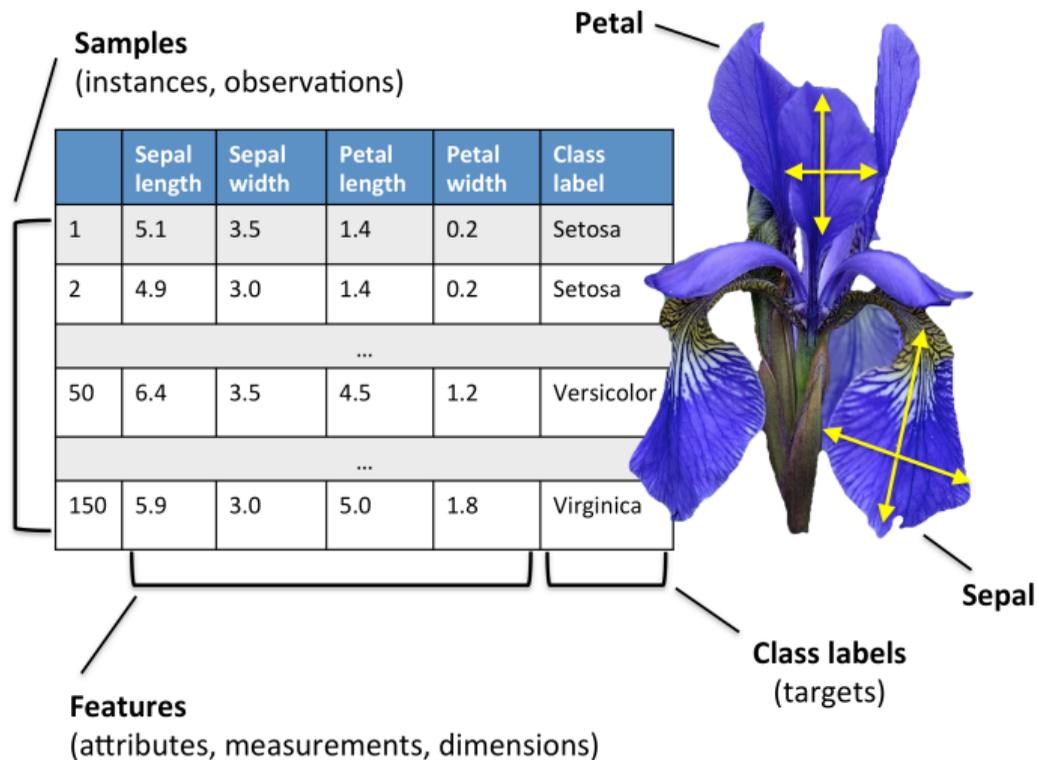
Что могло пойти не так с этим PCA?

# PCA. Упражнение 3

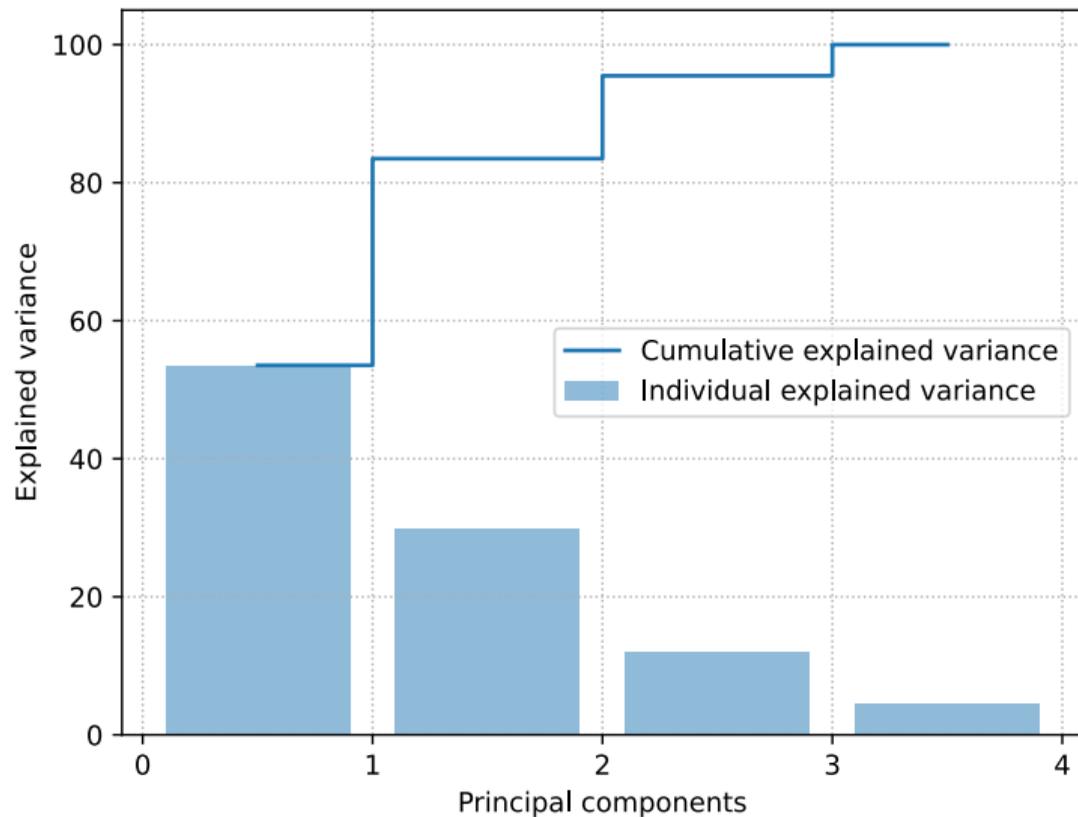
Что могло пойти не так с этим PCA?



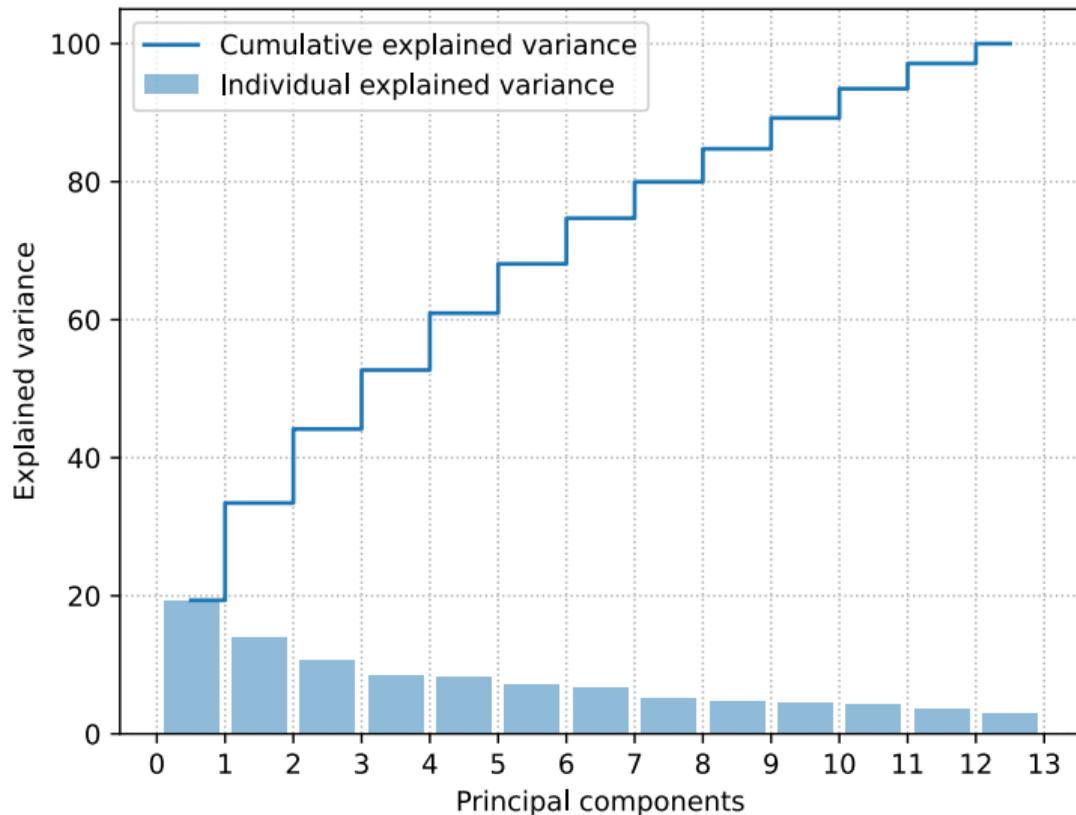
# Iris dataset variance



## Iris dataset variance

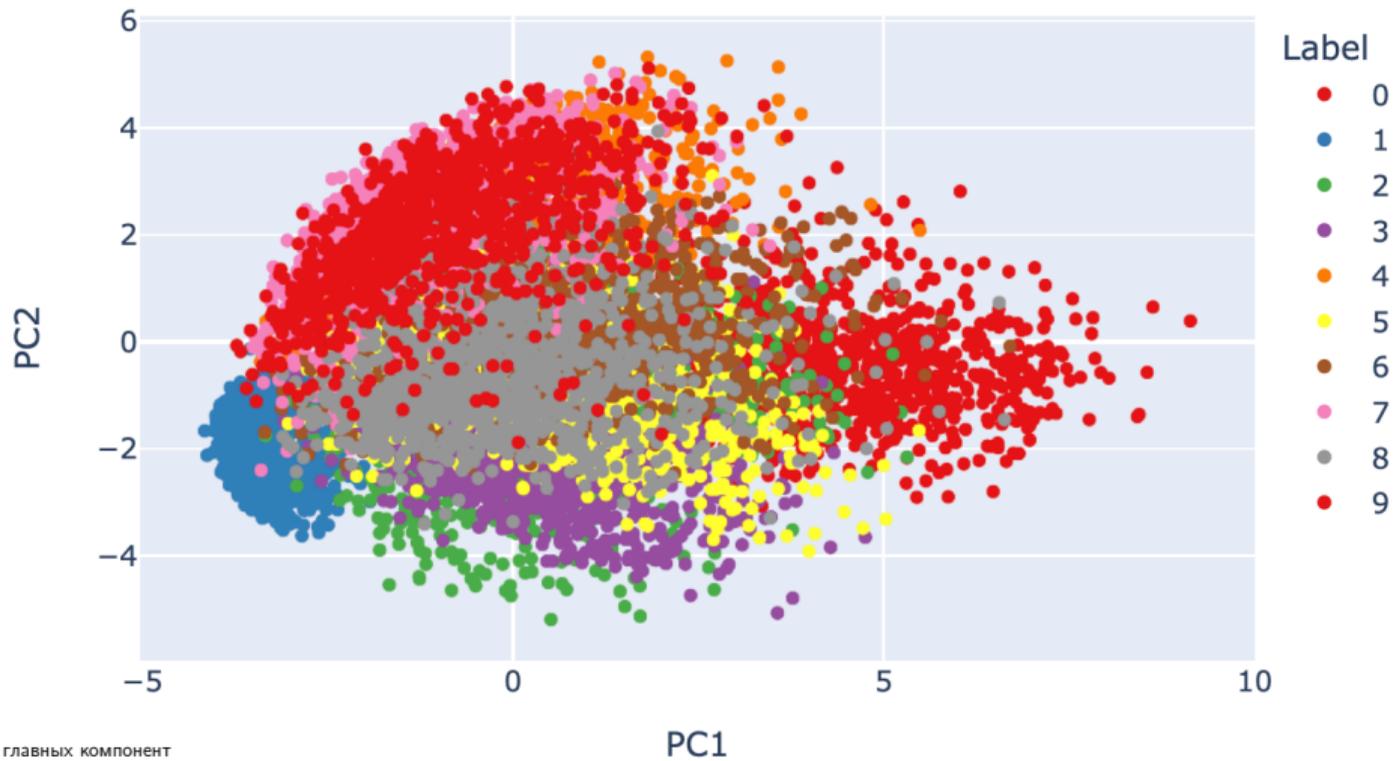


# Wine dataset variance



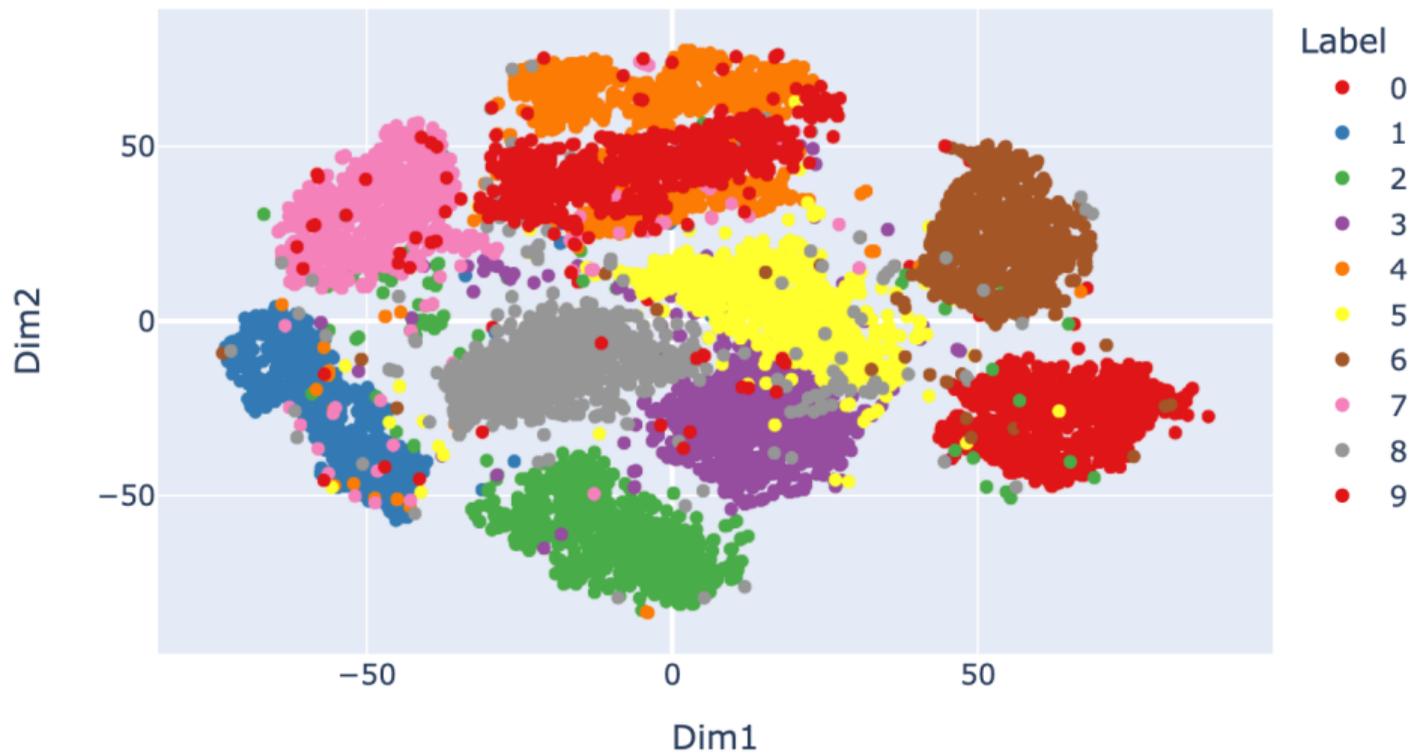
# PCA on MNIST

## 2D PCA of MNIST



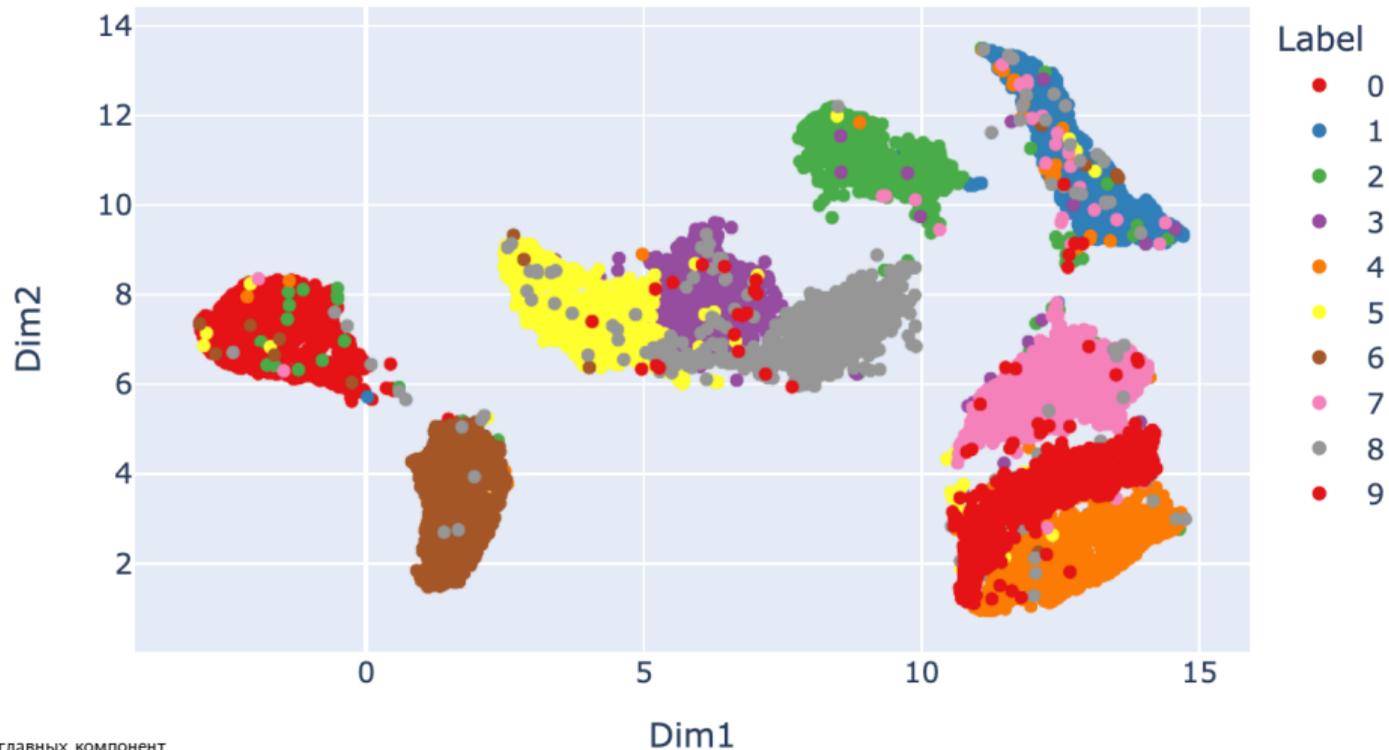
# t-SNE on MNIST

## 2D t-SNE of MNIST



# UMAP on MNIST

## 2D UMAP of MNIST



# Линейные системы

## Матричные разложения и линейные системы

В задаче наименьших квадратов (aka линейной регрессии) мы имеем измерения  $X \in \mathbb{R}^{m \times n}$  и  $y \in \mathbb{R}^m$  и ищем вектор  $\theta \in \mathbb{R}^n$  такой, что  $X\theta$  близок к  $y$ . Близость определяется как сумма квадратов разностей:

$$\sum_{i=1}^m (x_i^\top \theta - y_i)^2 \quad \|X\theta - y\|_2^2 \rightarrow \min_{\theta \in \mathbb{R}^n} \quad X\theta^* = y$$

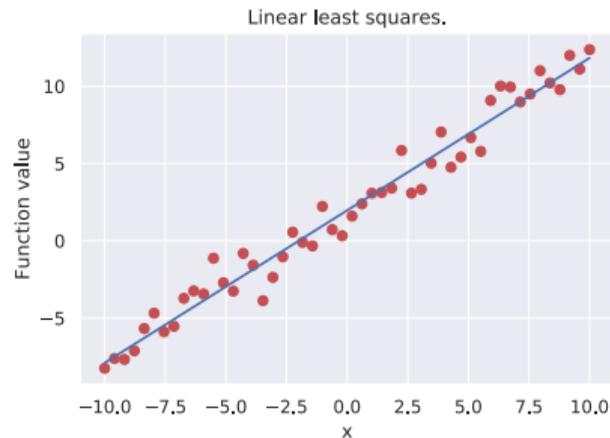
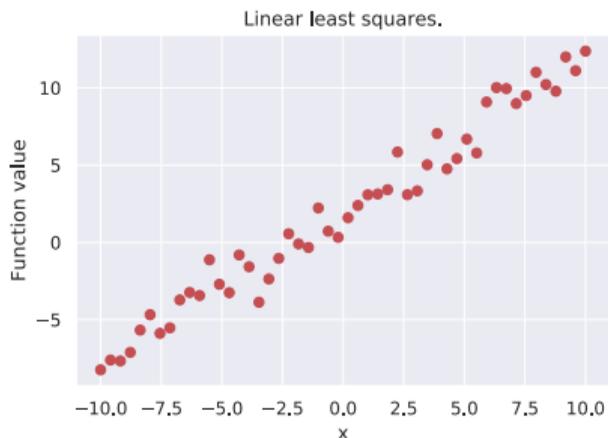


Рис. 7: Illustration of linear system aka least squares

# Матричные разложения и линейные системы

## Moore--Penrose inverse

Если матрица  $X$  относительно мала, мы можем записать и вычислить точное решение:

$$\theta^* = (X^T X)^{-1} X^T y = X^\dagger y,$$

# Матричные разложения и линейные системы

## Moore--Penrose inverse

Если матрица  $X$  относительно мала, мы можем записать и вычислить точное решение:

$$\theta^* = (X^T X)^{-1} X^T y = X^\dagger y,$$

где  $X^\dagger$  называется псевдо-обратной матрицей. Однако, этот подход возводит в квадрат число обусловленности задачи, что может быть проблемой для больших и плохо обусловленных задач.

# Матричные разложения и линейные системы

## Moore--Penrose inverse

Если матрица  $X$  относительно мала, мы можем записать и вычислить точное решение:

$$\theta^* = (X^T X)^{-1} X^T y = X^\dagger y,$$

где  $X^\dagger$  называется псевдо-обратной матрицей. Однако, этот подход возводит в квадрат число обусловленности задачи, что может быть проблемой для больших и плохо обусловленных задач.

## QR разложение

Для любой матрицы  $X \in \mathbb{R}^{m \times n}$  существует QR разложение:

$$X = Q \cdot R,$$

# Матричные разложения и линейные системы

## Moore--Penrose inverse

Если матрица  $X$  относительно мала, мы можем записать и вычислить точное решение:

$$\theta^* = (X^T X)^{-1} X^T y = X^\dagger y,$$

где  $X^\dagger$  называется псевдо-обратной матрицей. Однако, этот подход возводит в квадрат число обусловленности задачи, что может быть проблемой для больших и плохо обусловленных задач.

## QR разложение

Для любой матрицы  $X \in \mathbb{R}^{m \times n}$  существует QR разложение:

$$X = Q \cdot R,$$

где  $Q$  - ортогональная матрица (ее столбцы ортогональные единичные векторы) и  $R$  - верхняя треугольная матрица. Важно отметить, что поскольку  $Q^{-1} = Q^T$ , мы имеем:

$$QR\theta = y \quad \longrightarrow \quad R\theta = Q^T y$$

Теперь процесс нахождения  $\theta$  состоит из двух шагов:

1. Найдите QR разложение  $X$ .

# Матричные разложения и линейные системы

## Moore--Penrose inverse

Если матрица  $X$  относительно мала, мы можем записать и вычислить точное решение:

$$\theta^* = (X^T X)^{-1} X^T y = X^\dagger y,$$

где  $X^\dagger$  называется псевдо-обратной матрицей. Однако, этот подход возводит в квадрат число обусловленности задачи, что может быть проблемой для больших и плохо обусловленных задач.

## QR разложение

Для любой матрицы  $X \in \mathbb{R}^{m \times n}$  существует QR разложение:

$$X = Q \cdot R,$$

где  $Q$  - ортогональная матрица (ее столбцы ортогональные единичные векторы) и  $R$  - верхняя треугольная матрица. Важно отметить, что поскольку  $Q^{-1} = Q^T$ , мы имеем:

$$QR\theta = y \quad \longrightarrow \quad R\theta = Q^T y$$

Теперь процесс нахождения  $\theta$  состоит из двух шагов:

1. Найдите QR разложение  $X$ .
2. Решите треугольную систему  $R\theta = Q^T y$ , которая треугольная и, следовательно, легко решаемая.

# Матричные разложения и линейные системы

## Разложение Холецкого

Для любой положительно определенной матрицы  $A \in \mathbb{R}^{n \times n}$  существует разложение Холецкого:

$$X^T X = A = L^T \cdot L,$$

где  $L$  - нижняя треугольная матрица. Мы имеем:

$$L^T L \theta = y \quad \longrightarrow \quad L^T z_\theta = y$$

Теперь процесс нахождения  $\theta$  состоит из двух шагов:

1. Найдите разложение Холецкого  $X^T X$ .

Обратите внимание, что в этом случае ошибка пропорциональна квадрату числа обусловленности.

# Матричные разложения и линейные системы

## Разложение Холецкого

Для любой положительно определенной матрицы  $A \in \mathbb{R}^{n \times n}$  существует разложение Холецкого:

$$X^T X = A = L^T \cdot L,$$

где  $L$  - нижняя треугольная матрица. Мы имеем:

$$L^T L \theta = y \quad \longrightarrow \quad L^T z_\theta = y$$

Теперь процесс нахождения  $\theta$  состоит из двух шагов:

1. Найдите разложение Холецкого  $X^T X$ .
2. Найдите  $z_\theta = L \theta$  путем решения треугольной системы  $L^T z_\theta = y$

Обратите внимание, что в этом случае ошибка пропорциональна квадрату числа обусловленности.

# Матричные разложения и линейные системы

## Разложение Холецкого

Для любой положительно определенной матрицы  $A \in \mathbb{R}^{n \times n}$  существует разложение Холецкого:

$$X^T X = A = L^T \cdot L,$$

где  $L$  - нижняя треугольная матрица. Мы имеем:

$$L^T L \theta = y \quad \longrightarrow \quad L^T z_\theta = y$$

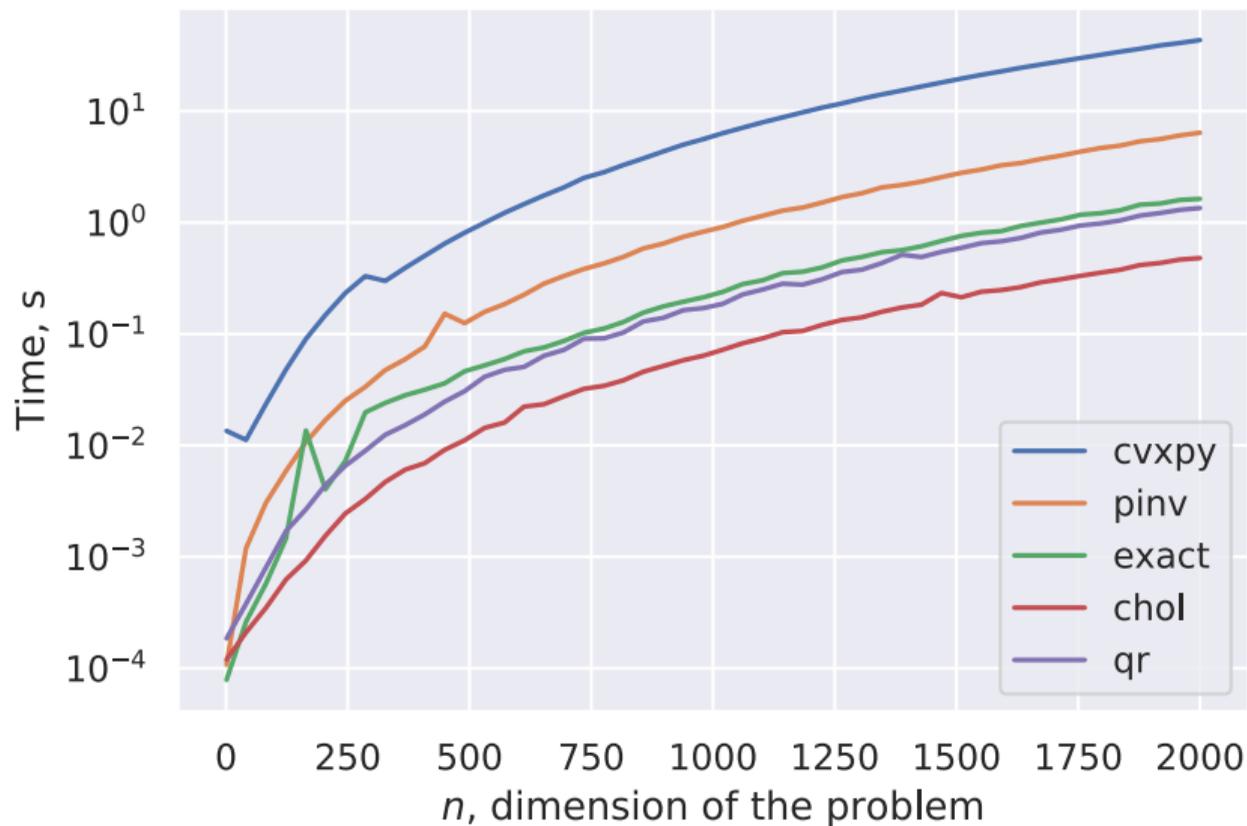
Теперь процесс нахождения  $\theta$  состоит из двух шагов:

1. Найдите разложение Холецкого  $X^T X$ .
2. Найдите  $z_\theta = L \theta$  путем решения треугольной системы  $L^T z_\theta = y$
3. Найдите  $\theta$  путем решения треугольной системы  $L \theta = z_\theta$

Обратите внимание, что в этом случае ошибка пропорциональна квадрату числа обусловленности.

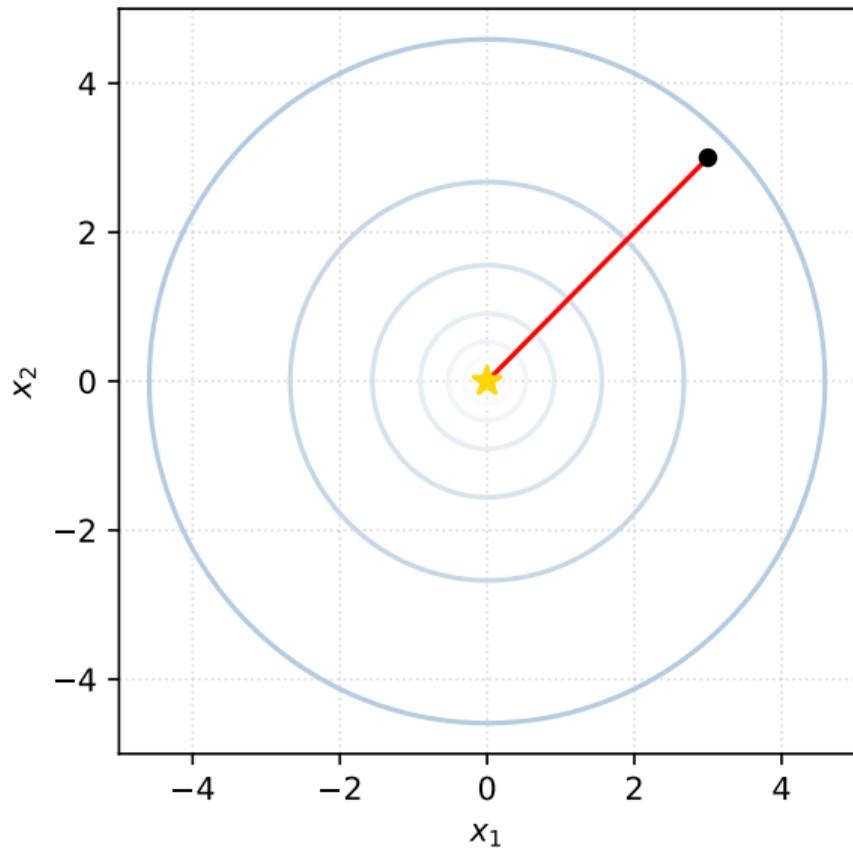
# Матричные разложения и линейные системы

Random square linear system

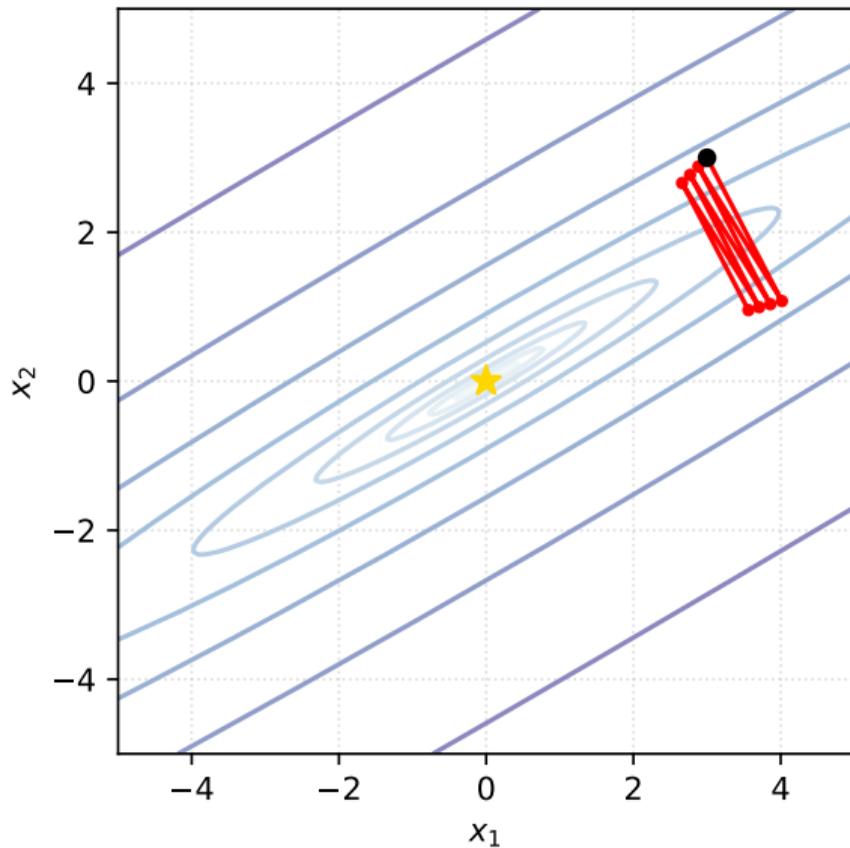


# Число обусловленности $\kappa$

$\kappa = 1.0$



$\kappa = 100.0$



## Матричные разложения и линейные системы

