



SVD. Eigenfaces. PCA. Линейные системы

Даня Меркулов

МФТИ. AI360

Сингулярное разложение матрицы

 $A_{m \times n}$

=

 $U_{m \times m}$ σ_1
 σ_2
 σ_3
⋮
 σ_r $\Sigma_{m \times n}$ $V^*_{n \times n}$

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

Сингулярное разложение матрицы

 $A_{m \times n}$

=

 $U_{m \times m}$ σ_1
 σ_2
 σ_3
...
 σ_r $\Sigma_{m \times n}$ $V_{n \times n}^*$

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

 $A_{m \times n}$

=

 $U_{m \times m}$ σ_1
 σ_2
 σ_3
...
 σ_r $\Sigma_{m \times n}$ $V^*_{n \times n}$

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

 $A_{m \times n}$

=

 $U_{m \times m}$ σ_1
 σ_2
 σ_3
...
 σ_r $\Sigma_{m \times n}$ $V^*_{n \times n}$

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 :
 $U\Sigma V^* = UD\Sigma(VD)^* = U\Sigma V^*$.

Теорема Экарта-Янга

Наилучшее приближение низкого ранга может быть вычислено с помощью SVD.

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

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

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:

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

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:

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

Пример 1

Найдите SVD следующей матрицы:

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

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

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

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

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

Пример 1

Найдите SVD следующей матрицы:

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

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

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

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

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

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

Пример 2

Решите упражнение Eigenfaces

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 проекций точек данных на наш вектор $w_{(1)}$, что приводит к следующей задаче оптимизации:

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

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:

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

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

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:

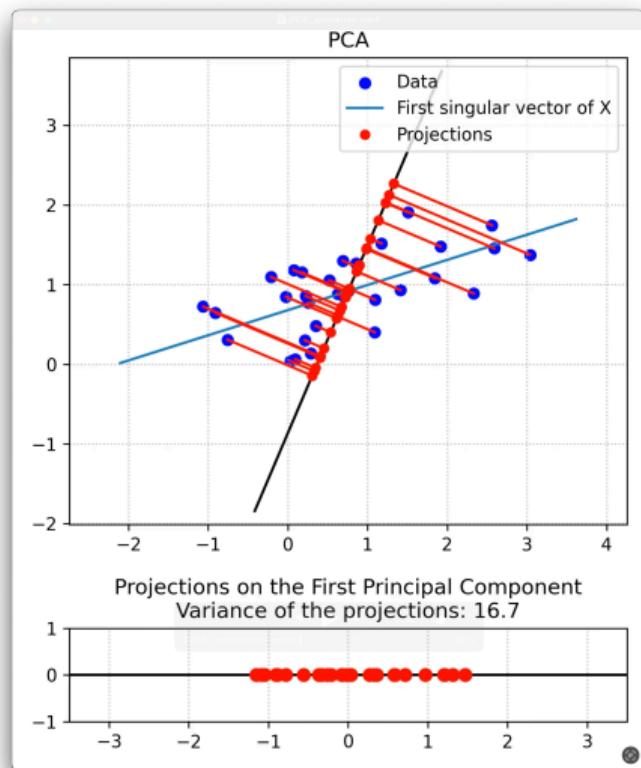
$$w_{(1)} = \arg \max_{\|w\|=1} \left\{ \sum_i (a_{(i)}^\top \cdot w)^2 \right\}$$

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

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

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

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 $w_{(1)}$, что приводит к следующей задаче оптимизации:

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

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

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

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

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

Вывод метода

Таким образом, мы можем заключить, что следующее отображение:

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

Вывод метода

Таким образом, мы можем заключить, что следующее отображение:

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

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

Вывод метода

Таким образом, мы можем заключить, что следующее отображение:

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

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:

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

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:

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

и транспонируем его:

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

Вывод метода

Таким образом, мы можем заключить, что следующее отображение: Теперь рассмотрим матрицу AA^\top :

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

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

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

Сначала мы запишем SVD нашей матрицы:

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

и транспонируем его:

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

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

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

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:

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

и транспонируем его:

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

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

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

В итоге:

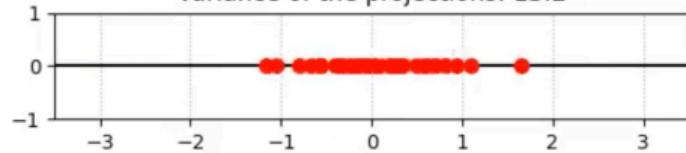
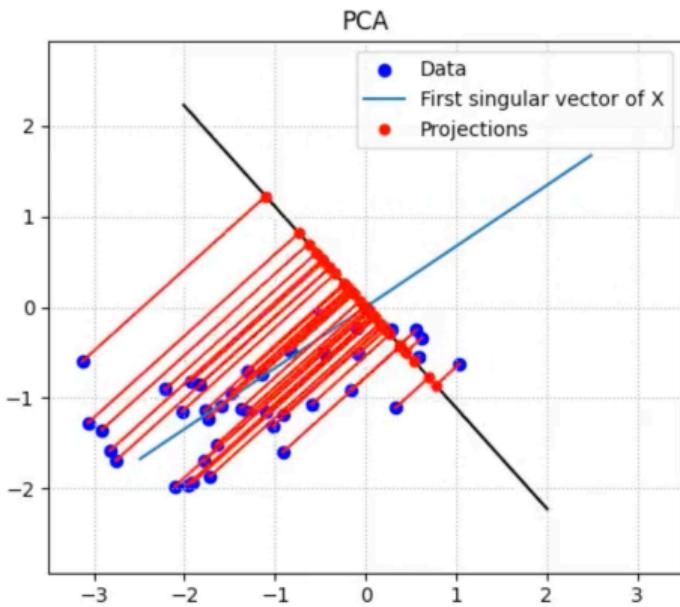
$$\begin{aligned} \Pi &= A \cdot W = \\ &= U \Sigma W^T W = U \Sigma \end{aligned}$$

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

$$\Pi_r = U_r \Sigma_r$$

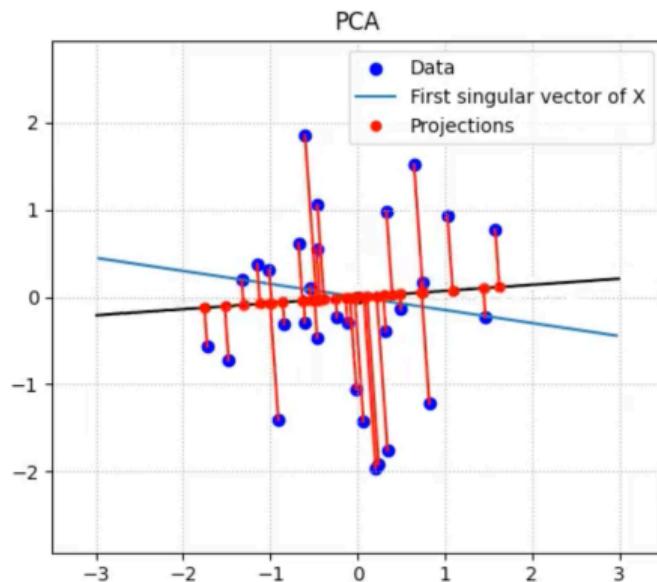
PCA. Упражнение 1

Что могло пойти не так с этим PCA?

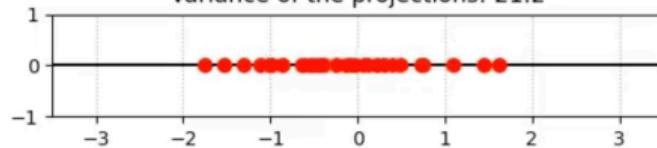


PCA. Упражнение 2

Что могло пойти не так с этим PCA?

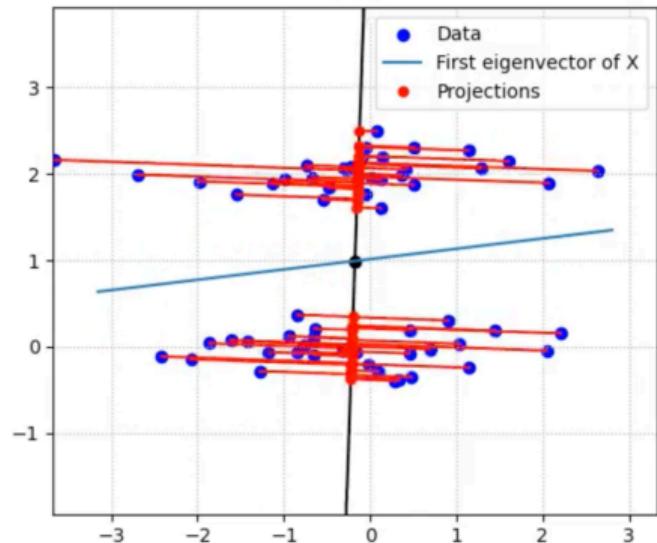


Projections on the First Principal Component
Variance of the projections: 21.2

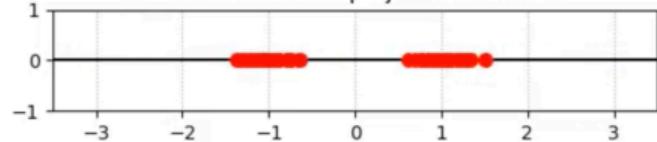


PCA. Упражнение 3

PCA

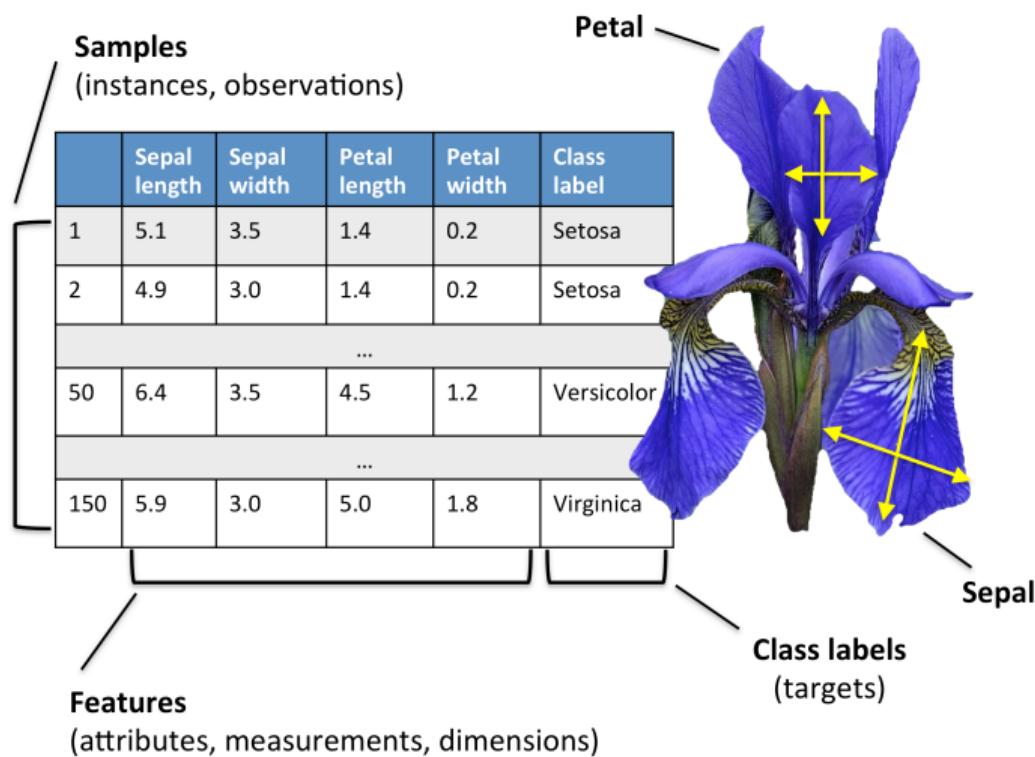


Projections on the First Principal Component
Variance of the projections: 65.7

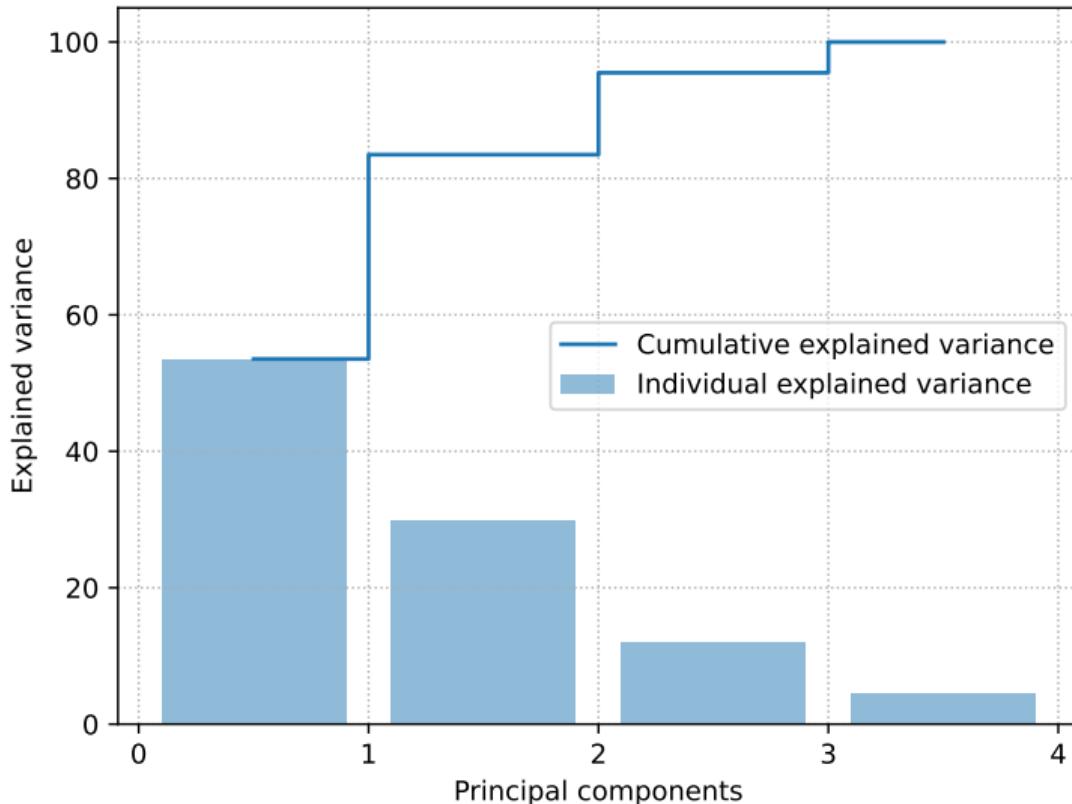


Что могло пойти не так с этим PCA?

Iris dataset variance



Iris dataset variance



Wine dataset variance



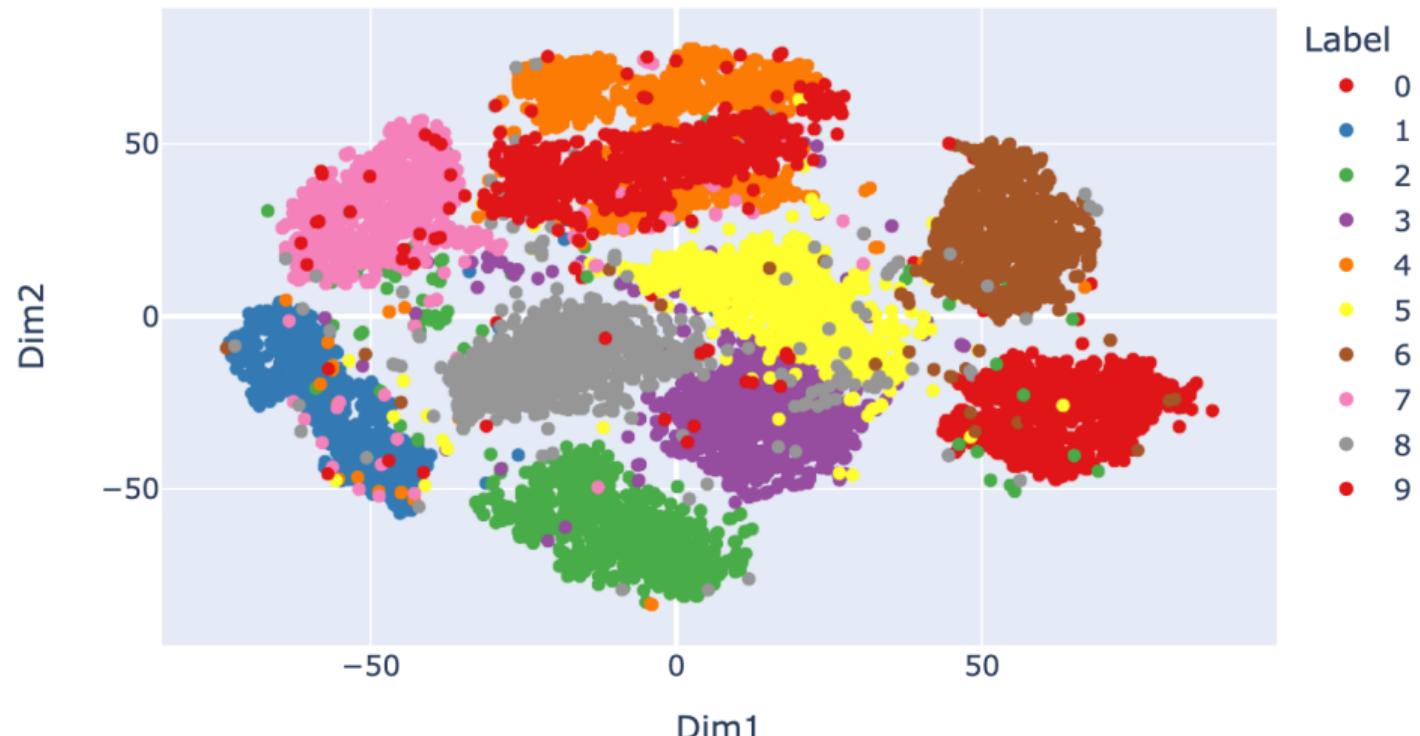
PCA on MNIST

2D PCA of MNIST



t-SNE on MNIST

2D t-SNE of MNIST



UMAP on MNIST

2D UMAP of MNIST

