Машинное обучение

Лекция 11

Уменьшение размерности.

Михаил Гущин

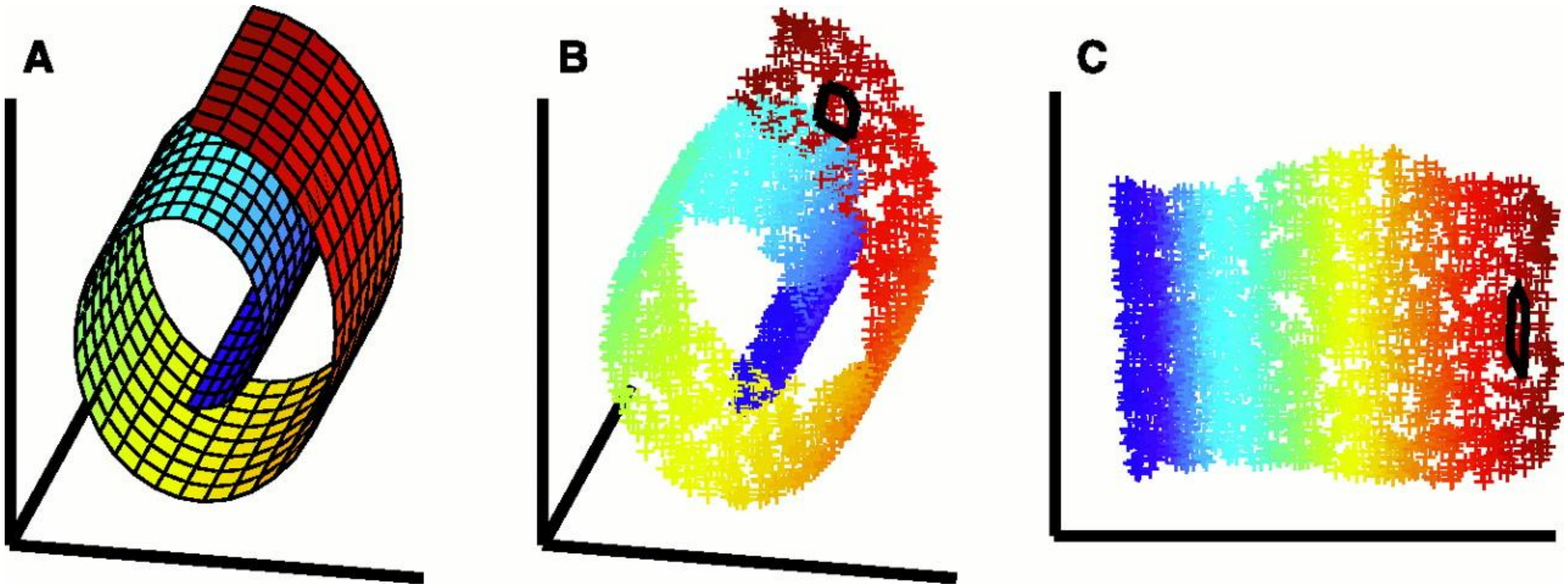
mhushchyn@hse.ru

НИУ ВШЭ, 2025

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

Уменьшение размерности (Dimensionality reduction)

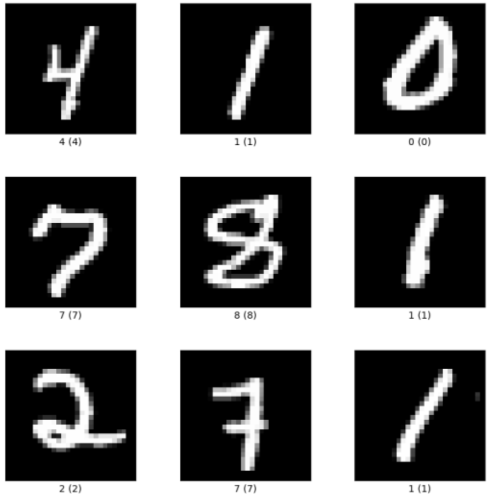
Постановка задачи

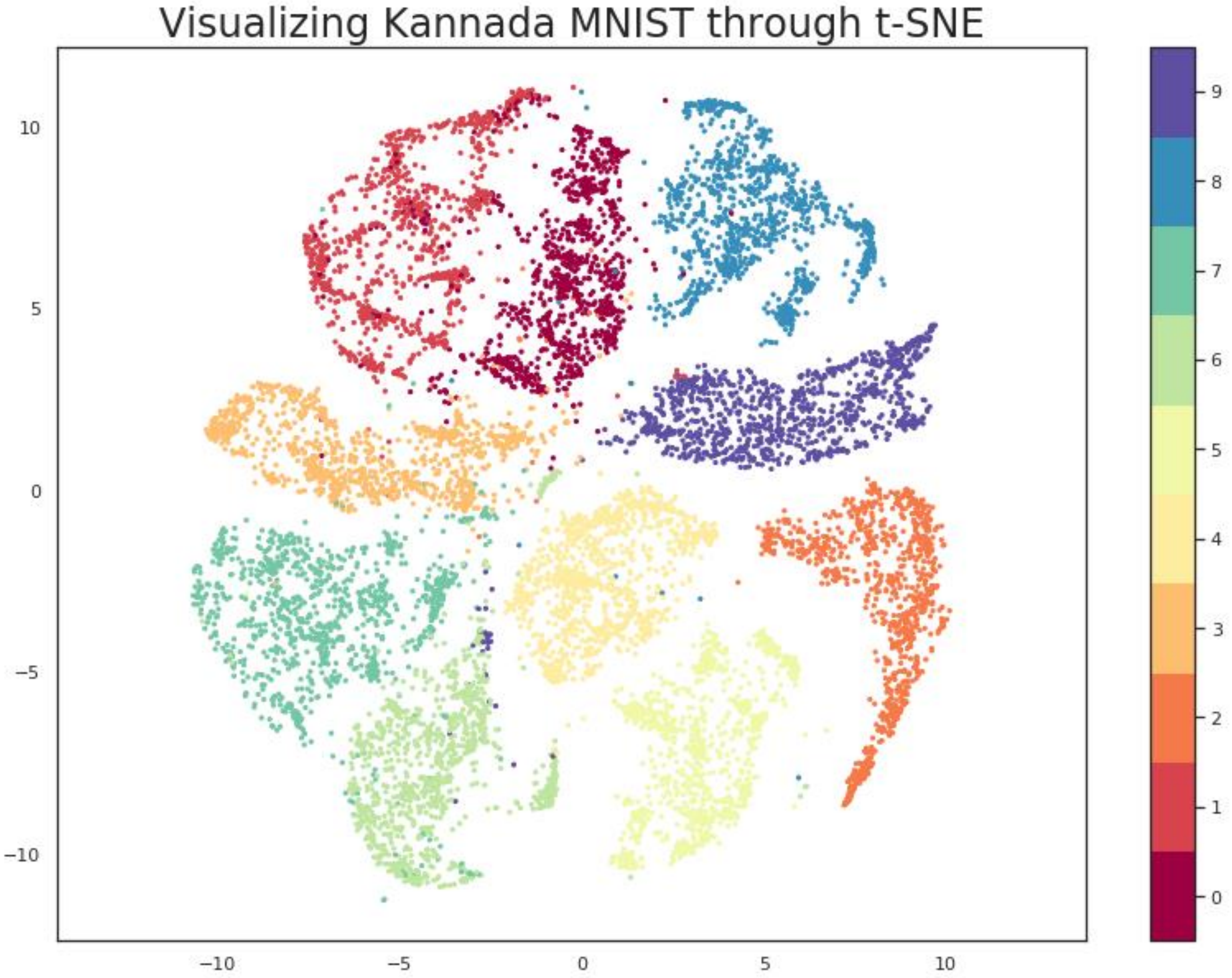


- Уменьшить размерность пространства входных признаков
- Минимизировать потери полезной информации
- Новые признаки – преобразование старых признаков

Пример

Уменьшение размерности:
28x28 –> 2

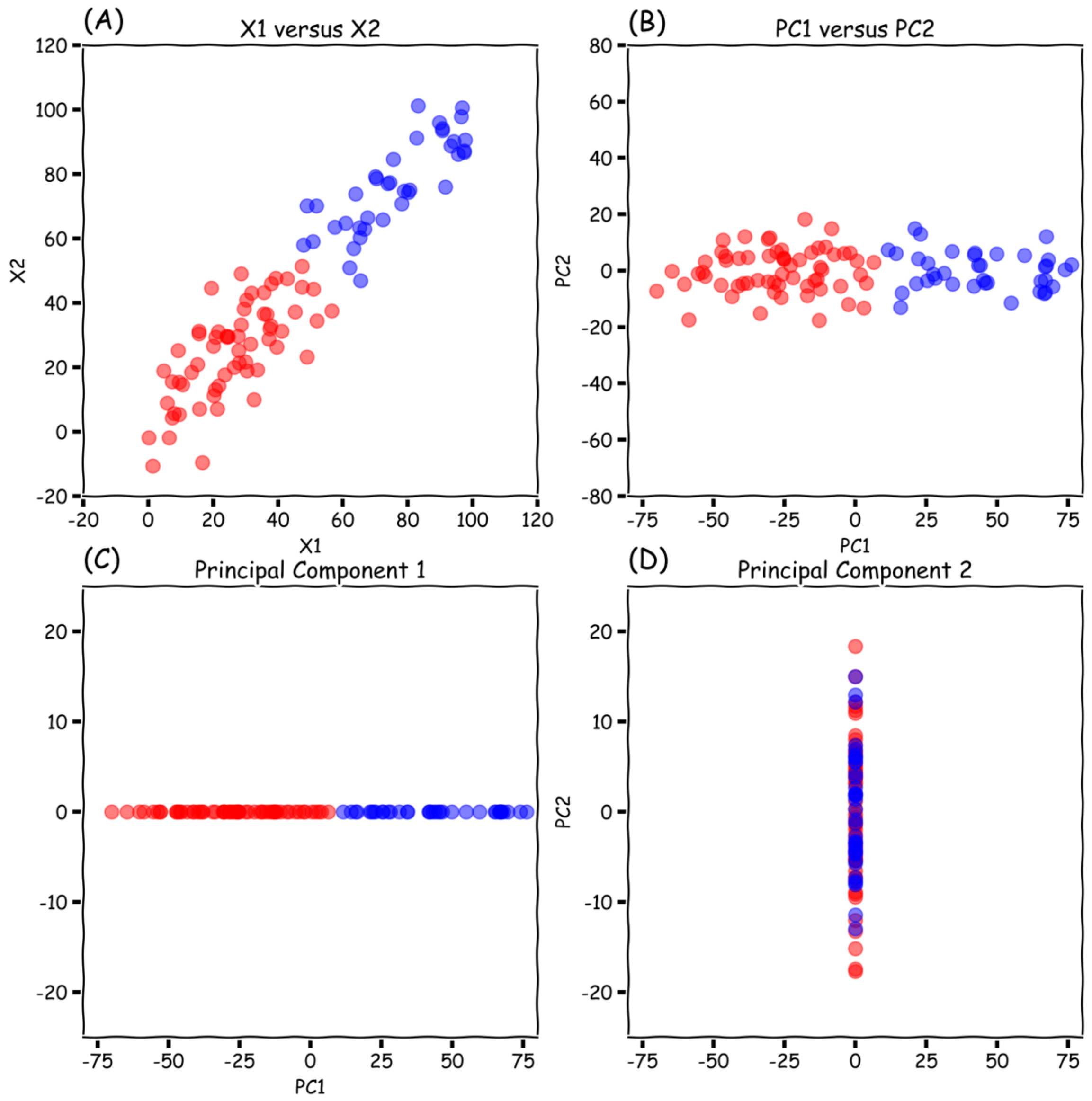




Метод главных компонент
(Principal Component Ananlisys) (PCA)

Интуиция

- Даны 2 класса в 2D пространстве (X1, X2)
- Данные расположены вдоль диагонали
- Повернем системы координат так, чтобы данные располагались вдоль одной из осей
- Получили новые признаки (PC1, PC2)
- Классы разделяются по PC1



Источник: https://www.robertoreif.com/blog/2018/1/9/pca

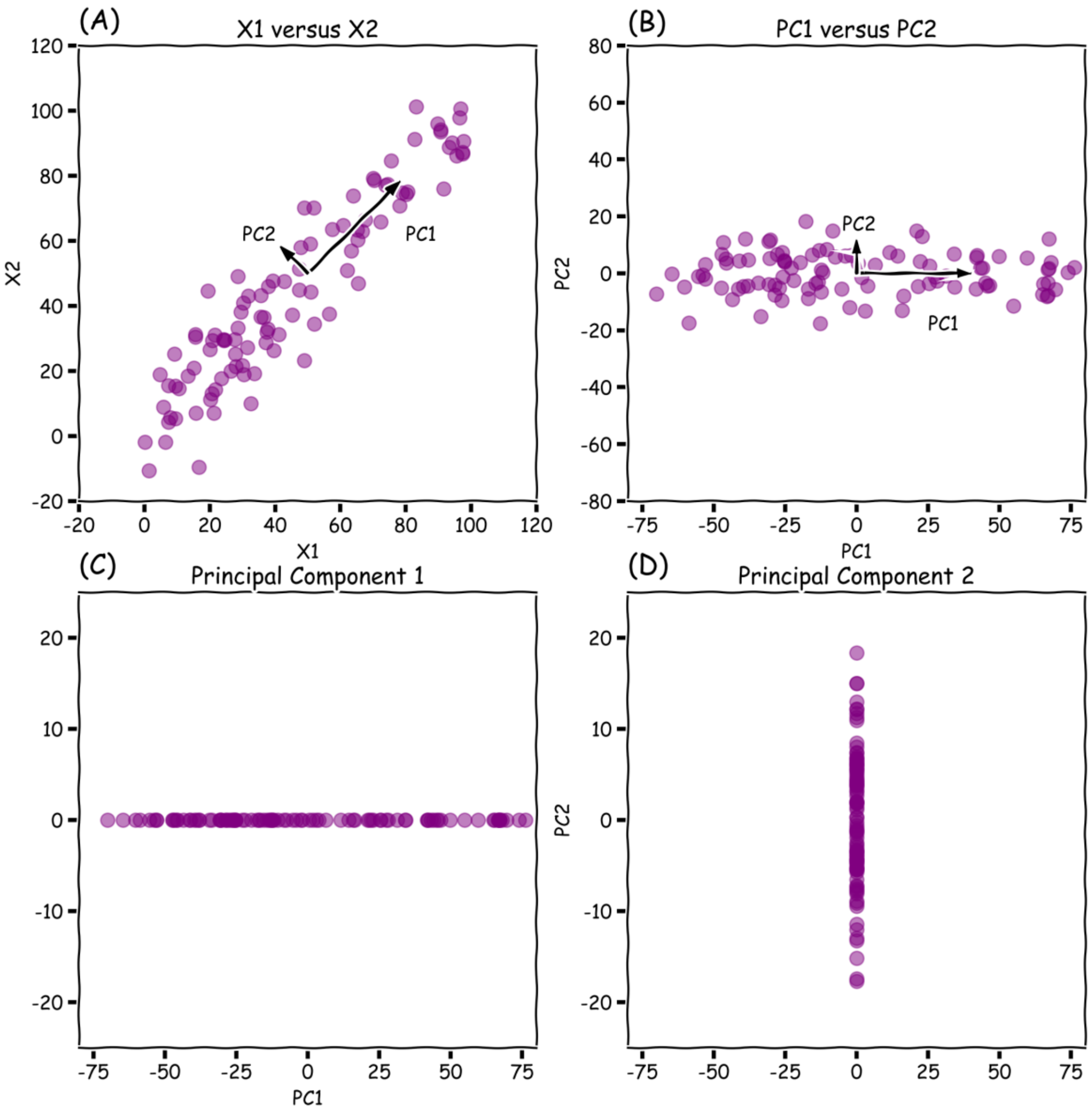
Задача

- Даны наблюдения с признаками (X1, X2, X3, …)
- Нужно найти главные компоненты (PC1, PC2, PC3, …)

- Мы будем обозначать главные компоненты как $a_1, a_2, a_3, \dots$
- При этом, будем требовать их ортогональность:

$$a_i^T a_i = 1$$

$$a_i^T a_j = 0, \qquad i \neq j$$



Поиск первой компоненты

- Пусть признаки в матрице наблюдений X нормированы (применили Standard Scaler)
- Пусть дан вектор a
- Тогда проекция вектора x_i на a задается как $z_i = a^T x_i$
- Посчитаем дисперсию z_i:

$$\sigma_a^2 = \frac{1}{n} \sum_{i=1}^{n} (z_i - \mu_z)^2$$

- Будем искать такой вектор a, чтобы

$$\sigma_a^2 \to \max_a$$

Поиск первой компоненты

- Распишем дисперсию z_i:

$$\sigma_a^2 = \frac{1}{n}\sum_{i=1}^{n}(z_i - \mu_z)^2 = \frac{1}{n}\sum_{i=1}^{n}(a^T x_i)^2 =$$

$$= \frac{1}{n}\sum_{i=1}^{n} a^T\left(x_i x_i^T\right)a = a^T\left(\frac{1}{n}\sum_{i=1}^{n} x_i x_i^T\right)a =$$

$$= a^T X^T X a = a^T C a \to \max_a$$

Поиск первой компоненты

- Тогда первую главную компоненту a_1 находим так:

$$\begin{cases} a_1 X^T X a_1 \rightarrow \max\limits_{a_1} \\ a_1^T a_1 = 1 \end{cases}$$

Поиск первой компоненты

- Тогда первую главную компоненту a_1 находим так:

$$\begin{cases} a_1 X^T X a_1 \to \max\limits_{a_1} \\ a_1^T a_1 = 1 \end{cases}$$

- Решаем максимизацией лагранжиана:

$$L = a_1 X^T X a_1 - \nu(a_1^T a_1 - 1) \to \max_{\nu, a_1}$$

$$\frac{\partial L}{\partial a_1} = 2X^T X a_1 - 2\nu a_1 = 0$$

$$X^T X a_1 = \nu a_1$$

Поиск первой компоненты

- Видим, что вектор первой главной компоненты – собственный вектор матрицы $X^T X$
- Но какой собственный вектор?
- Уже знаем, что

$$a_1 X^T X a_1 \to \max_{a_1}$$

  - И знаем, что

$$X^T X a_1 = \nu a_1$$

  - Тогда подставим в первое выражение:

$$\nu a_1^T a_1 = \nu \to \max$$

- Получили, что a_1 - собственный вектор с максимальным собственным числом.

Поиск второй компоненты

- Вторую главную компоненту a_2 находим так:

$$\begin{cases} a_2 X^T X a_2 \to \max\limits_{a_2} \\ a_2^T a_2 = 1 \\ a_2^T a_1 = 0 \end{cases}$$

Алгоритм PCA

- Нормируем матрицу наблюдений X
- Находим матрицу $C = X^T X$
- Находим первые k собственных векторов a_i и собственных значений λ_i:

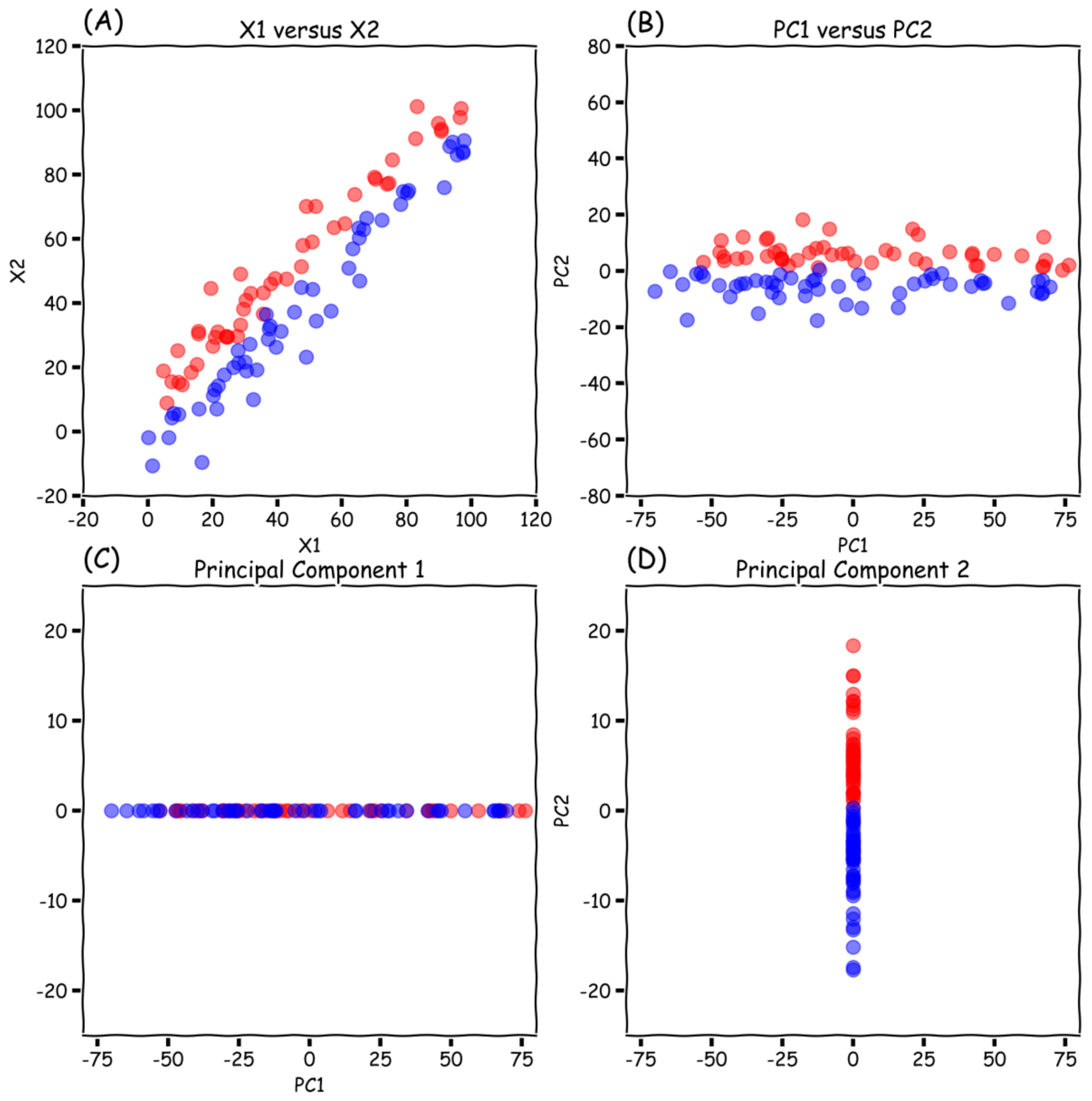
$$A = \begin{bmatrix} | & | & & | \\ a_1 & a_2 & \dots & a_k \\ | & | & & | \end{bmatrix}$$

- Делаем проекцию на собственные вектора:

$$Z = XA$$

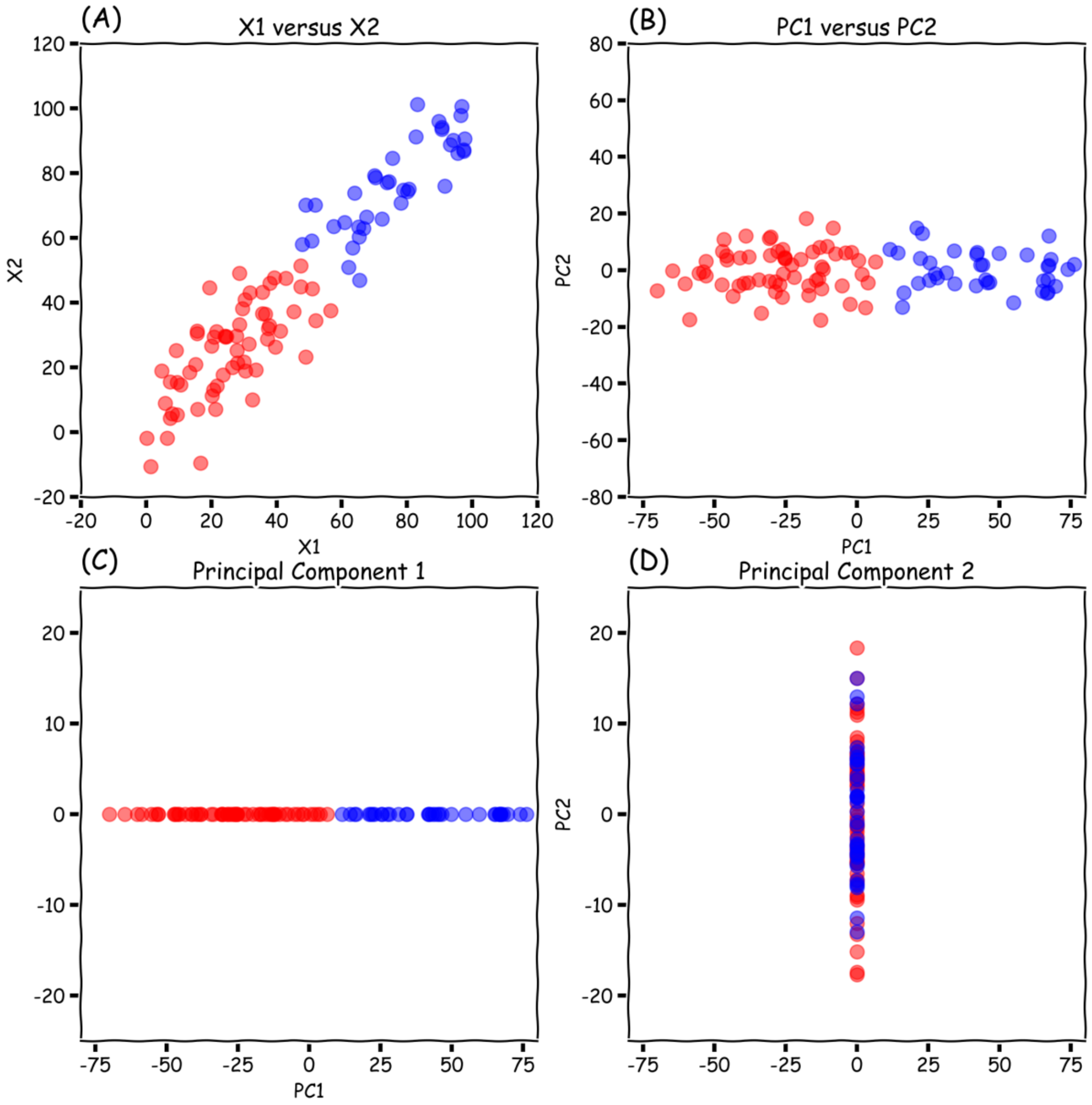
- Матрица Z – новая матрица наблюдений

Пример



Источник: https://www.robertoreif.com/blog/2018/1/9/pca

Пример



Источник: https://www.robertoreif.com/blog/2018/1/9/pca

Объясненная дисперсия (Explained variance)

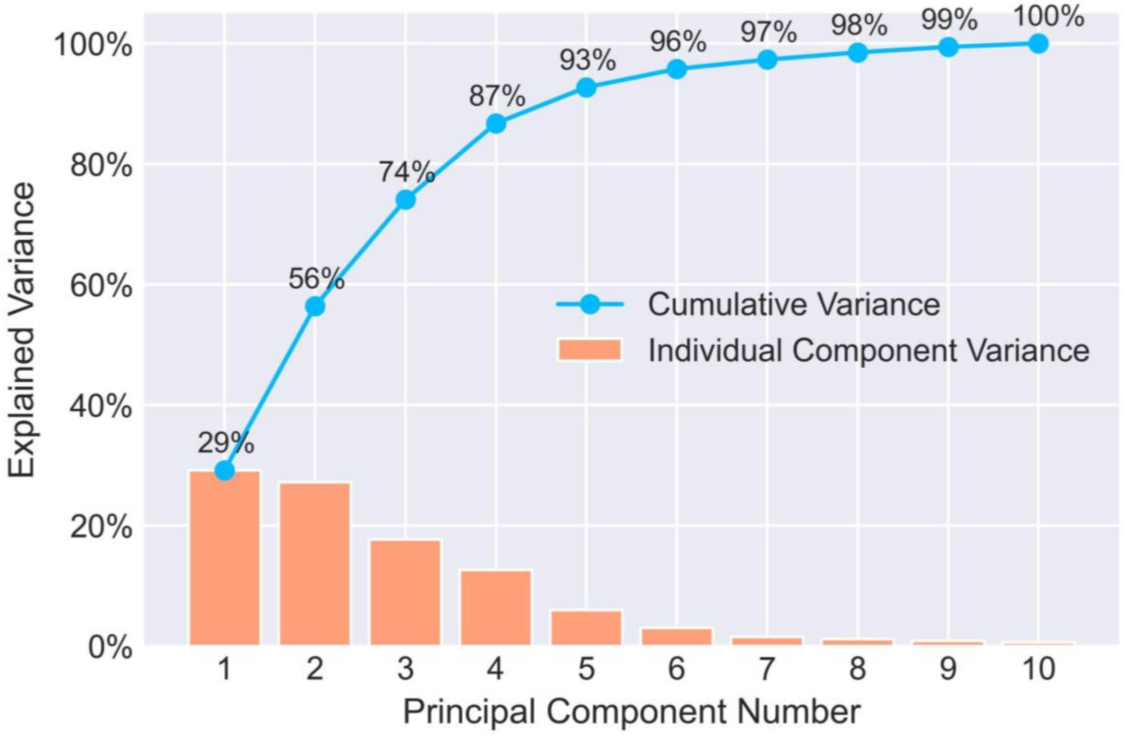
- Зачем нужны собственные значения λ_i?
- Они позволяют оценить дисперсию наблюдений вдоль собственных векторов
- Чем меньше дисперсия, тем больше наблюдения похожи друг на друга –> содержат меньше информации
- Собственные значения позволяют оценить потери информации, если мы выбросим какие-то собственные вектора

Объясненная дисперсия (Explained variance)

- Посчитаем объясненную дисперсию для собственного вектора a_i:

$$e_i = \frac{\lambda_i}{\sum_{d=1}^{D} \lambda_d}$$

  - где D – число собственных векторов матрицы C

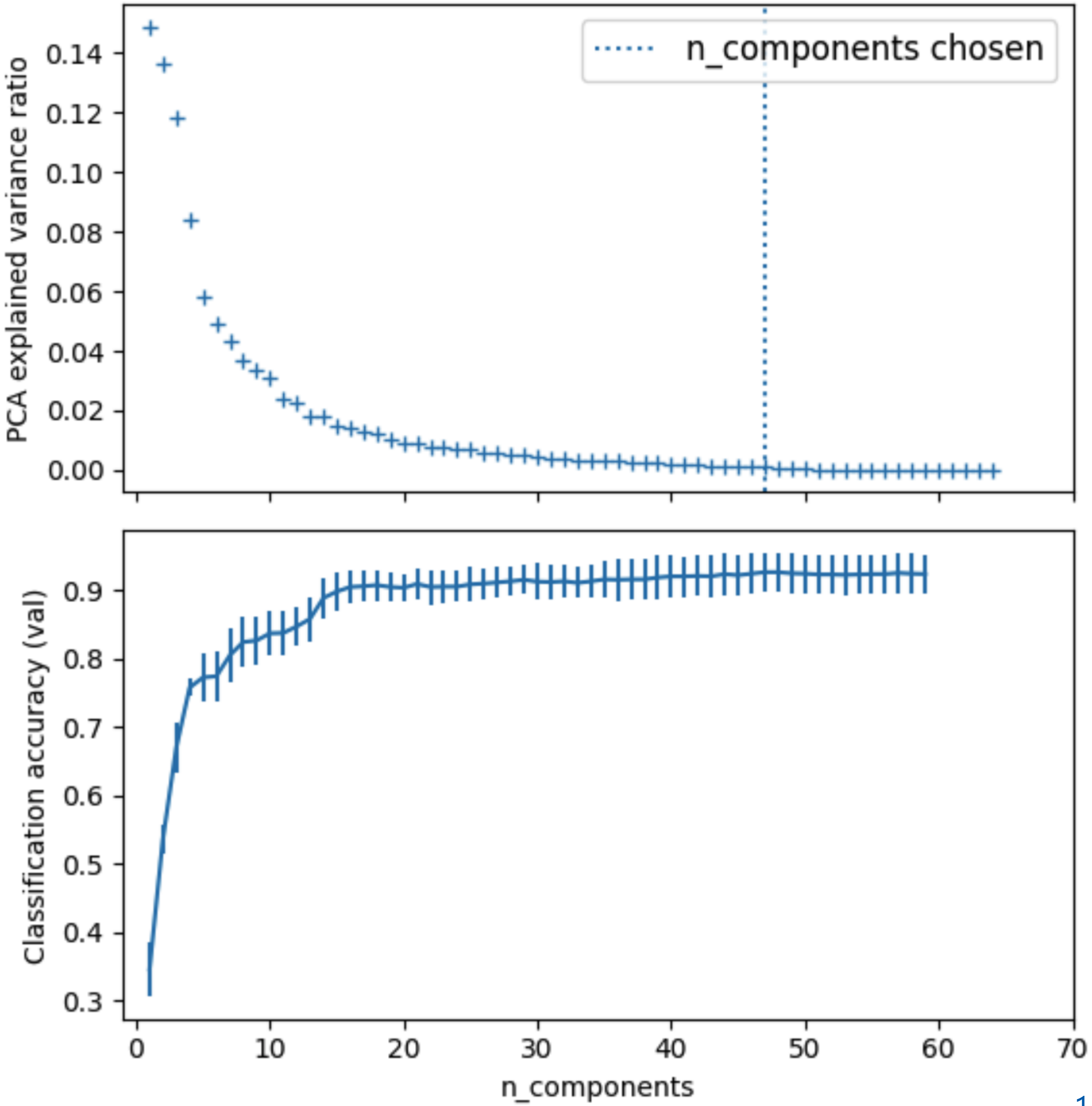


Пример

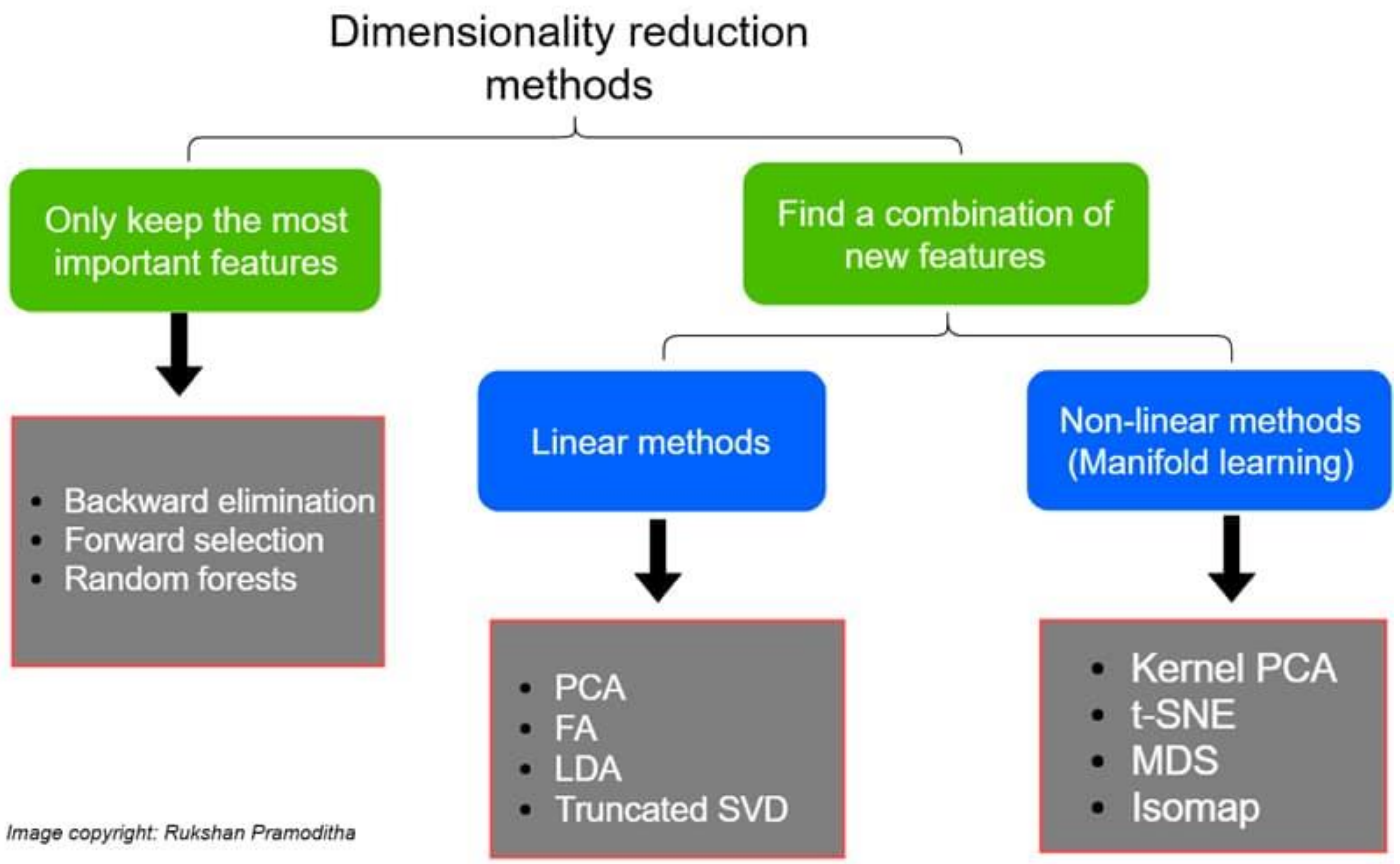
Зависимость объясненной дисперсии и качества классификации от числа главных компонент PCA

- После 20 компонент качество классификации почти перестает расти
- Оптимальное число компонент - 47

```
Best parameter (CV score=0.927):
{'logistic__C': 0.046415888336127774, 'pca__n_components': 47}
```



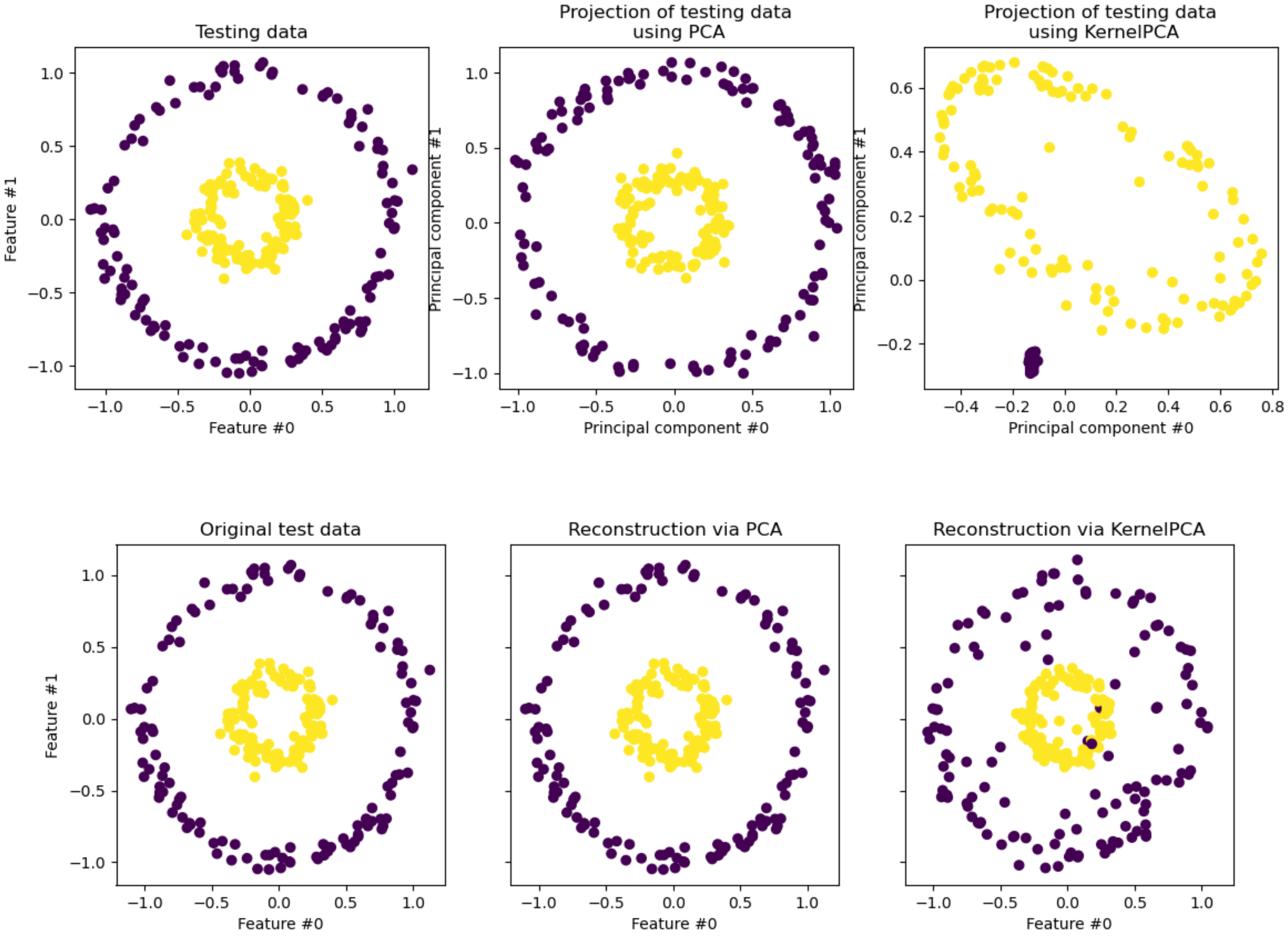
Другие методы уменьшения размерности


Dimensionality reduction methods
Only keep the most important features
Find a combination of new features
Backward elimination
Forward selection
Random forests
Linear methods
Non-linear methods (Manifold learning)
PCA
FA
LDA
Truncated SVD
Kernel PCA
t-SNE
MDS
Isomap
Image copyright: Rukshan Pramoditha

Методы уменьшения размерности

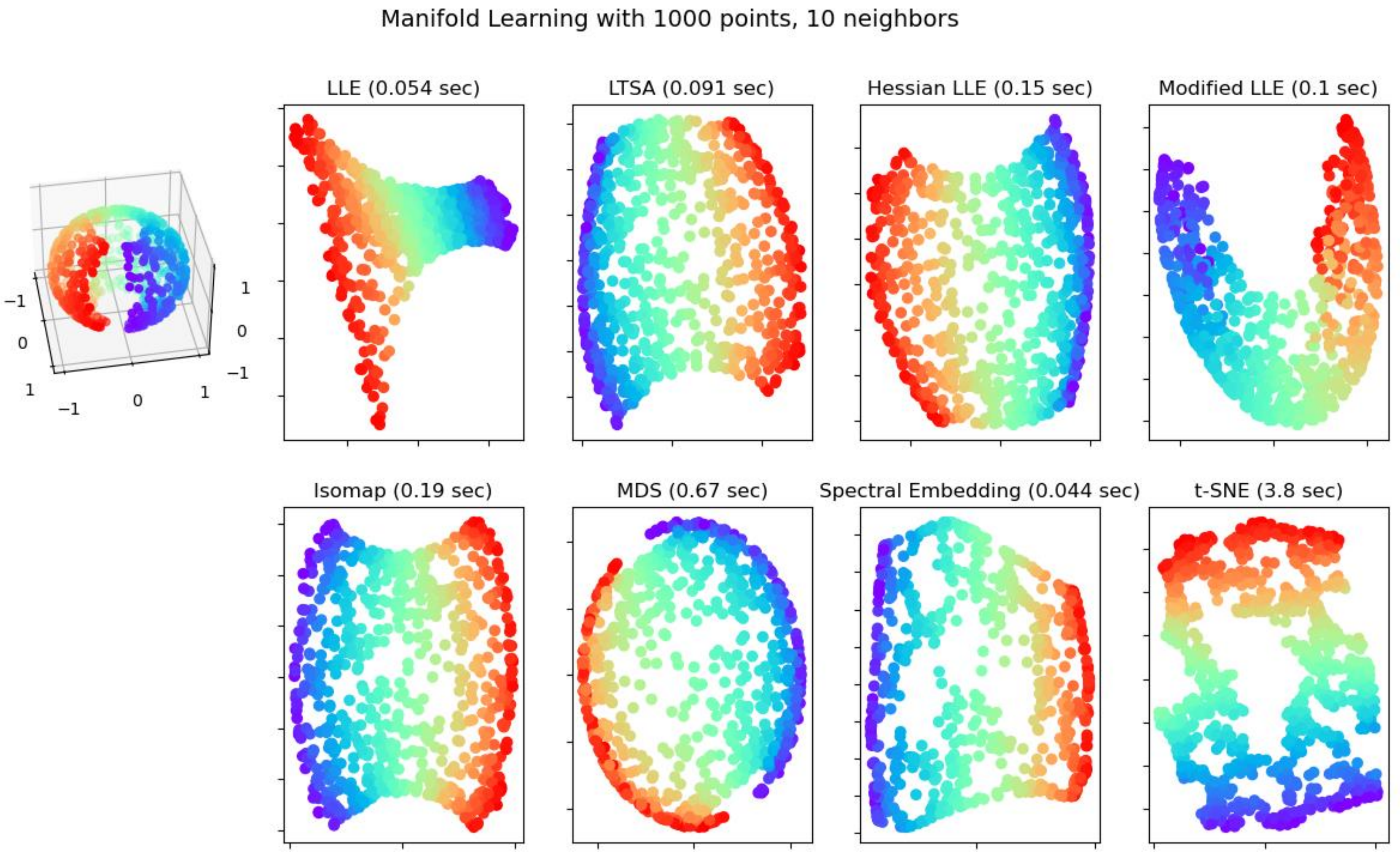
- PCA (https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html)
- Kernel PCA (https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.KernelPCA.html)
- t-SNE (https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html)
- Isomap (https://scikit-learn.org/stable/modules/generated/sklearn.manifold.Isomap.html)

Пример



Источник: https://scikit-learn.org/stable/auto_examples/decomposition/plot_kernel_pca.html#sphx-glr-auto-examples-decomposition-plot-kernel-pca-py

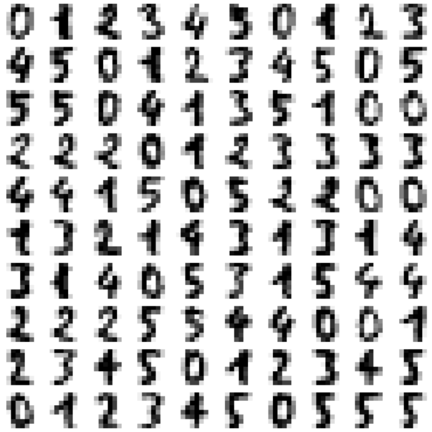
Пример



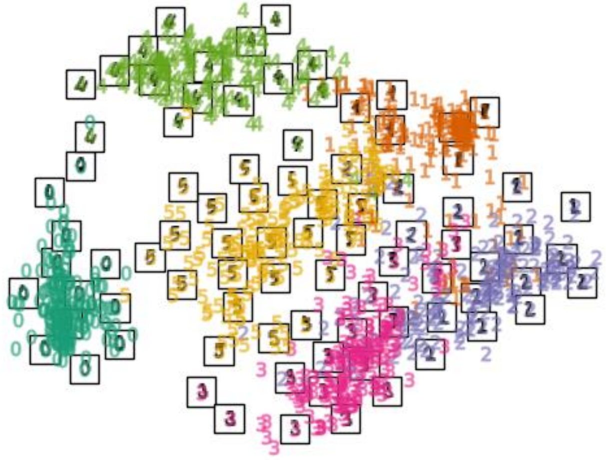
Источник: https://scikit-learn.org/stable/auto_examples/manifold/plot_manifold_sphere.html#sphx-glr-auto-examples-manifold-plot-manifold-sphere-py

Пример

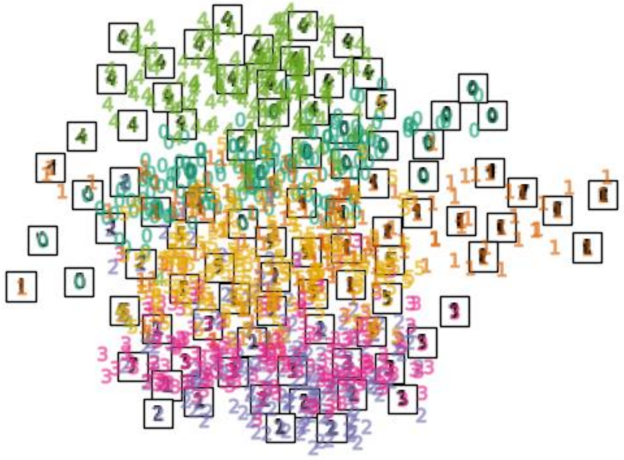
A selection from the 64-dimensional digits dataset



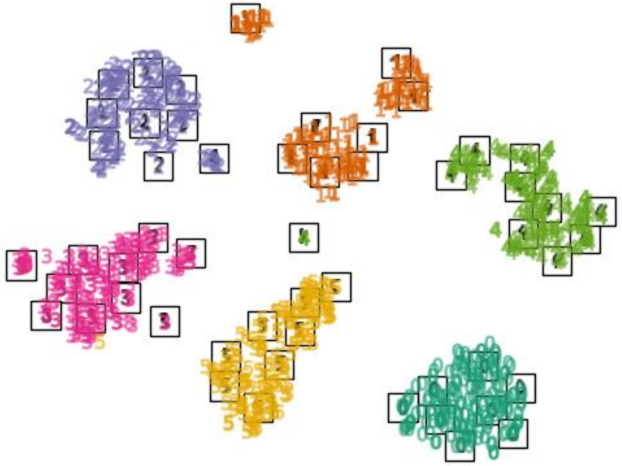
Isomap embedding (time 0.802s)



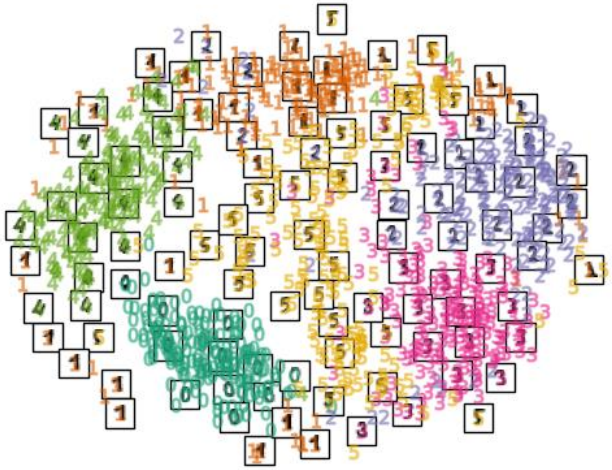
Truncated SVD embedding (time 0.003s)



t-SNE embeedding (time 2.681s)



MDS embedding (time 3.179s)



Источник: https://scikit-learn.org/stable/auto_examples/manifold/plot_lle_digits.html#sphx-glr-auto-examples-manifold-plot-lle-digits-py

Денойзинг с использованием PCA