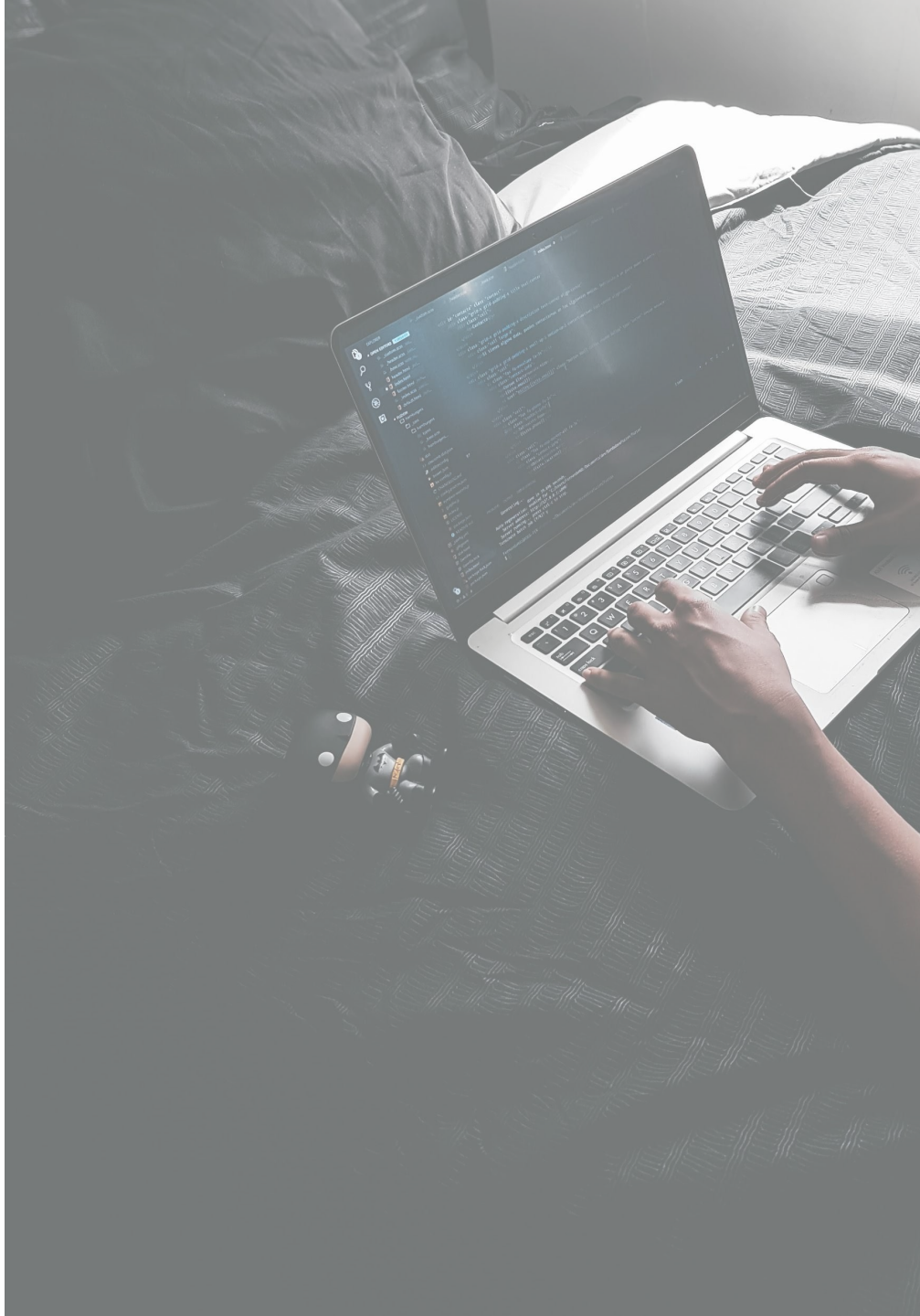
GeekBrains

Линейная алгебра с библиотекой Numpy



СТРУКТУРА КУРСА

Повторение основ (вводный урок)
1 урок

Библиотека Pandas
(загрузка данных, манипуляции с данными, агрегирование)
2 урока

Библиотеки Matplotlib и Seaborn
(визуализация данных, разведочный анализ, построение графиков и диаграмм)
1 урок

Библиотека Scipy
(стат. анализ и проверка статистических гипотез, оптимизация)
1 урок

Библиотека Numpy
(линейная алгебра, матрицы и векторы, мера близости)
1 урок

Библиотеки Scikit-learn
(предобработка данных, построение предиктивных ML-моделей)
1 урок

Feature Engineering
(повышение качества ML-моделей)
1 урок

ИТОГОВЫЙ ПРОЕКТ

GeekBrains

В ЭТОМ УРОКЕ

- Работа с матрицами
- Векторы и метрики расстояний
- Рекомендательные системы

БИБЛИОТЕКА NUMPY

- поддержка многомерных массивов, в т.ч. разреженных массивов (sparse arrays);
- поддержка математических функций, предназначенных для работы с многомерными массивами;
- совместимость со многими библиотеками;
- простота использования;
- открытый исходный код.

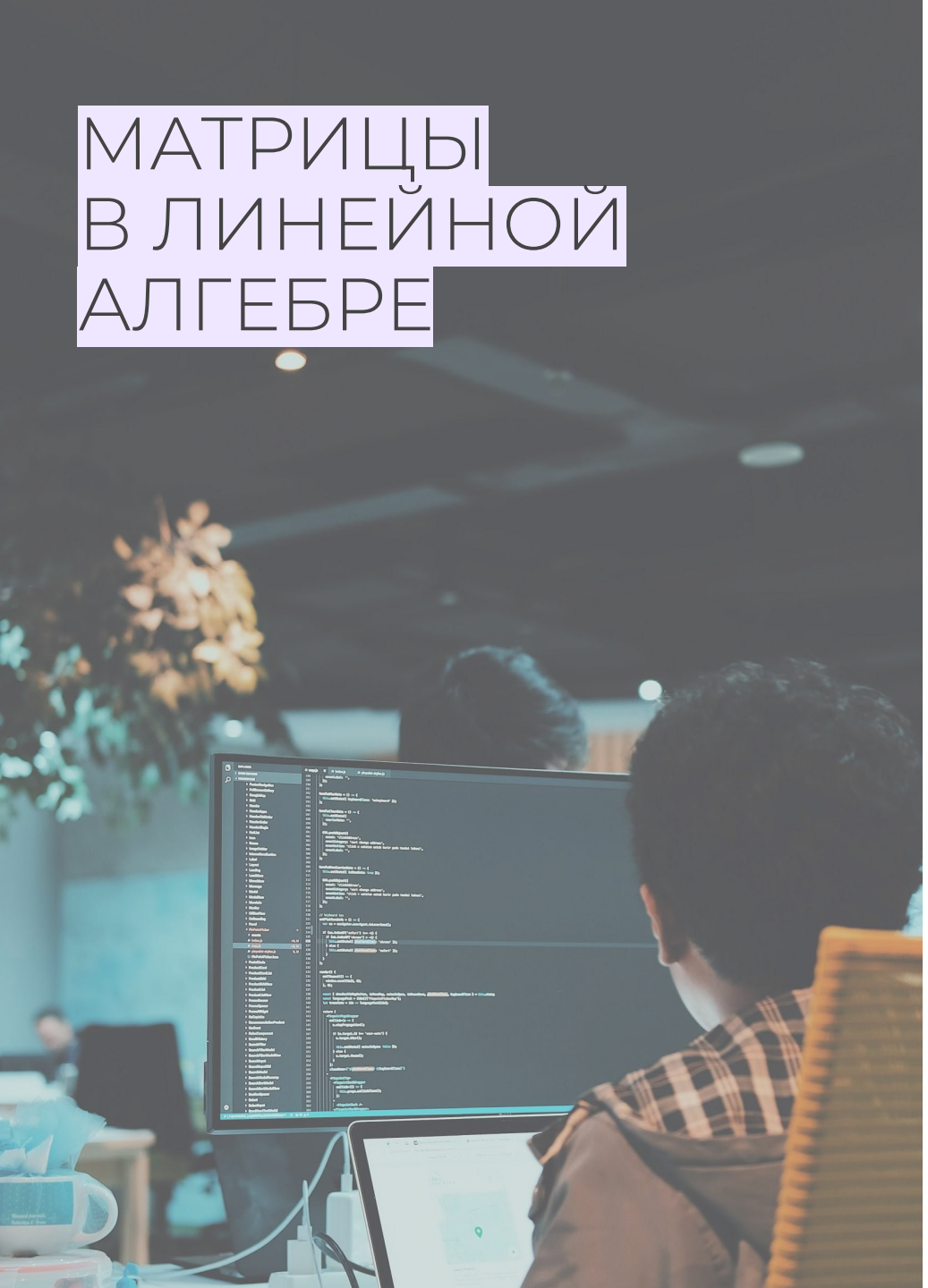
РОЛЬ NUMPY В ЭКОСИСТЕМЕ PYTHON

МАТРИЦЫ

GeekBrains

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.700	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5
1	7.8	0.880	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5
2	7.8	0.760	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8	5
3	11.2	0.280	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8	6
4	7.4	0.700	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5
5	7.4	0.660	0.00	1.8	0.075	13.0	40.0	0.9978	3.51	0.56	9.4	5
6	7.9	0.600	0.06	1.6	0.069	15.0	59.0	0.9964	3.30	0.46	9.4	5
7	7.3	0.650	0.00	1.2	0.065	15.0	21.0	0.9946	3.39	0.47	10.0	7
8	7.8	0.580	0.02	2.0	0.073	9.0	18.0	0.9968	3.36	0.57	9.5	7
9	7.5	0.500	0.36	6.1	0.071	17.0	102.0	0.9978	3.35	0.80	10.5	5
10	6.7	0.580	0.08	1.8	0.097	15.0	65.0	0.9959	3.28	0.54	9.2	5
11	7.5	0.500	0.36	6.1	0.071	17.0	102.0	0.9978	3.35	0.80	10.5	5
12	5.6	0.615	0.00	1.6	0.089	16.0	59.0	0.9943	3.58	0.52	9.9	5
13	7.8	0.610	0.29	1.6	0.114	9.0	29.0	0.9974	3.26	1.56	9.1	5
14	8.9	0.620	0.18	3.8	0.176	52.0	145.0	0.9986	3.16	0.88	9.2	5

МАТРИЦЫ В ЛИНЕЙНОЙ АЛГЕБРЕ



GeekBrains

Матрица:

$$A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \dots & \dots & \dots \\ a_{m1} & \dots & a_{mn} \end{pmatrix}$$

Вектор-строка:

$$B = \begin{pmatrix} b_1 & \dots & b_n \end{pmatrix}$$

Вектор-столбец:

$$C = \begin{pmatrix} c_1 \\ \dots \\ c_m \end{pmatrix}$$

МАТРИЦЫ В NUMPY

```
[1]  # импорт библиотеки
     import numpy as np
```

```
[2]  # создание матрицы
     a = np.array([[1, 2, 3],
                   [4, 5, 6],
                   [7, 8, 9]])
     a
```

```
array([[1, 2, 3],
       [4, 5, 6],
       [7, 8, 9]])
```

```
[3]  # создание вектора-столбца(строки)
     b = np.array([1,2,3,4])
     b
```

```
array([1, 2, 3, 4])
```

```
[4]  # импорт библиотеки
     import pandas as pd
```

```
     # получение датафрейма из numpy-массива
     df = pd.DataFrame(np.array([[1, 2, 3],
                                 [4, 5, 6],
                                 [7, 8, 9]]))
     df
```

	0	1	2
0	1	2	3
1	4	5	6
2	7	8	9

```
[6]  # получение numpy-массива из датафрейма
     df.values
```

```
array([[1, 2, 3],
       [4, 5, 6],
       [7, 8, 9]])
```

МЕТОДЫ NUMPY

- np.sum(a) - сумма элементов массива
- np.mean(a) - среднее значение элементов массива
- np.arange(0, 10) - массив от 0 до 9 (аналог range)
- np.random.shuffle(a) - случайная перестановка элементов массива
- np.concatenate((b, c)) - объединение нескольких массивов

ДЕЙСТВИЯ НАД МАТРИЦАМИ

Пример сложения матриц.

Даны две матрицы: $A = \begin{pmatrix} 2 & 4 \\ 6 & 7 \end{pmatrix}$; $B = \begin{pmatrix} 8 & 5 \\ 7 & 3 \end{pmatrix}$

$$C = A + B = \begin{pmatrix} 2+8 & 4+5 \\ 6+7 & 7+3 \end{pmatrix} = \begin{pmatrix} 10 & 9 \\ 13 & 10 \end{pmatrix}$$

```
import numpy as np
A = np.array([[2,4],
              [6,7]])
B = np.array([[8,5],
              [7,3]])
C = A + B
C
```

```
array([[10,  9],
       [13, 10]])
```

Пример умножения матриц.

$$C = A \times B = \begin{pmatrix} 2*8+4*7 & 2*5+4*3 \\ 6*8+7*7 & 6*5+7*3 \end{pmatrix} = \begin{pmatrix} 44 & 22 \\ 97 & 51 \end{pmatrix}$$

```
C = A.dot(B)
# C = np.dot(A,B)
C
```

```
array([[44, 22],
       [97, 51]])
```

ДЕЙСТВИЯ НАД МАТРИЦАМИ

Транспонирование:

$$A = \begin{pmatrix} 12 & -1 \\ -5 & 0 \end{pmatrix} \quad A^T = \begin{pmatrix} 12 & -5 \\ -1 & 0 \end{pmatrix}$$

```
import numpy as np
A = np.array([[12,-1],
              [-5,0]])
A.T
```

```
array([[12, -5],
       [-1,  0]])
```

Решение СЛАУ:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1 \\ a_{12}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2 \\ \ldots \\ a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n = b_m \end{cases}$$

$$AX = B$$

$$A = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \ldots & \ldots & \ldots & \ldots \\ a_{m1} & a_{m2} & \ldots & a_{mn} \end{pmatrix} \quad X = \begin{pmatrix} x_1 \\ x_2 \\ \ldots \\ x_n \end{pmatrix} \quad B = \begin{pmatrix} b_1 \\ b_2 \\ \ldots \\ b_m \end{pmatrix}$$

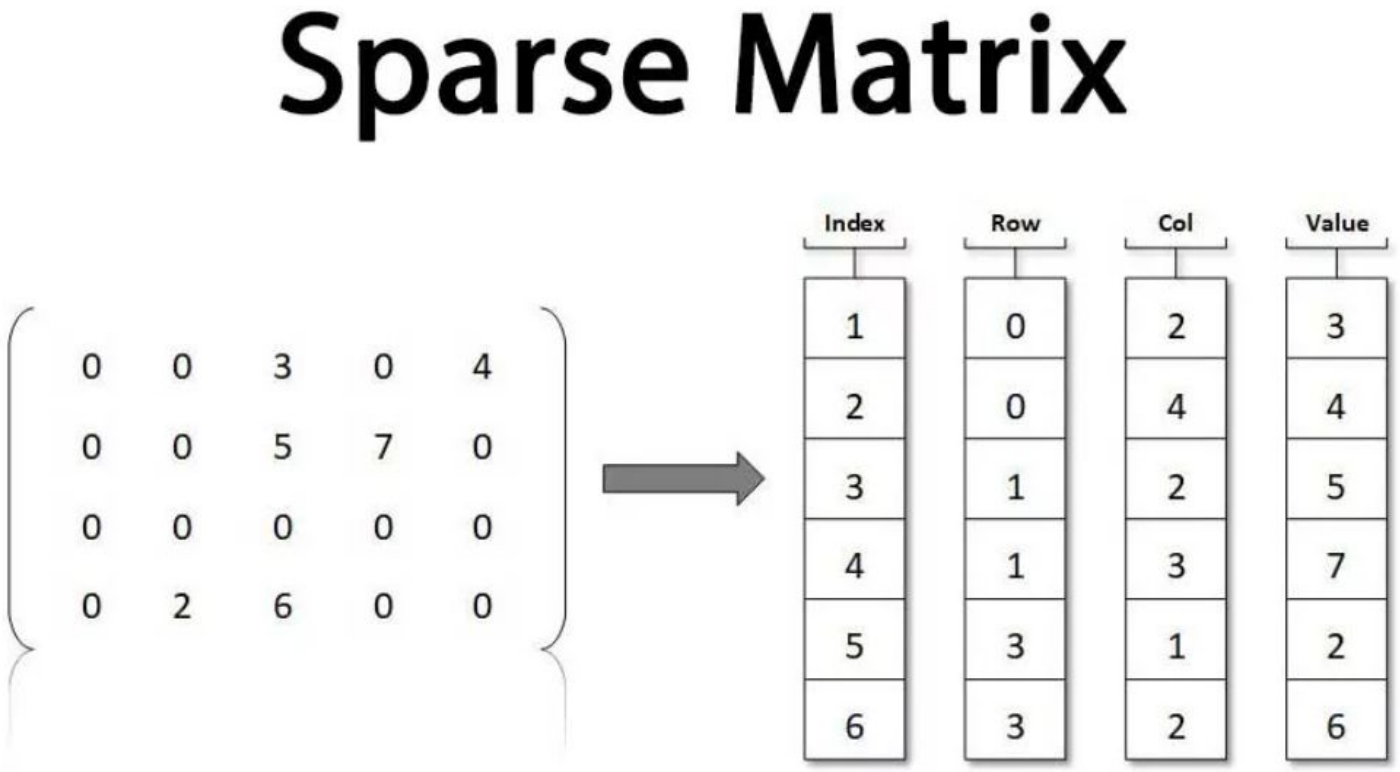
```
# СЛАУ
# 2x + 5y = 1
# 1x - 10y = 3

A = np.array([[2,5],
              [1,-10]])
B = np.array([1,3])

# решем систему уравнений
np.linalg.solve(A,B)
```

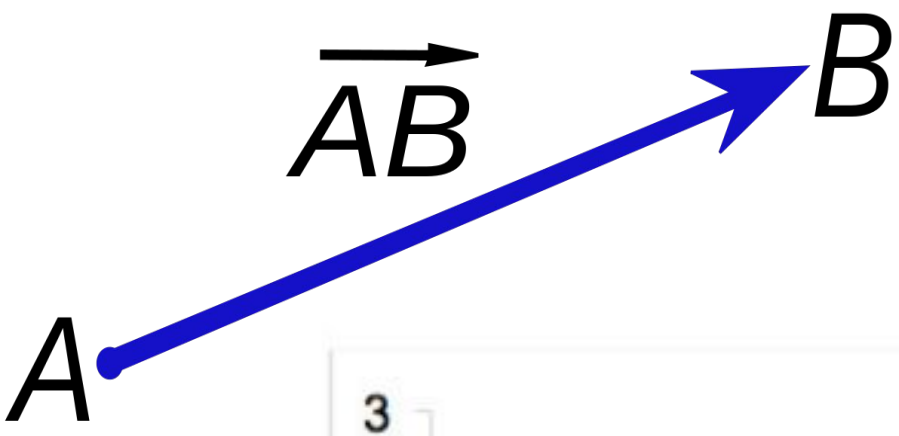
```
array([ 1. , -0.2])
```

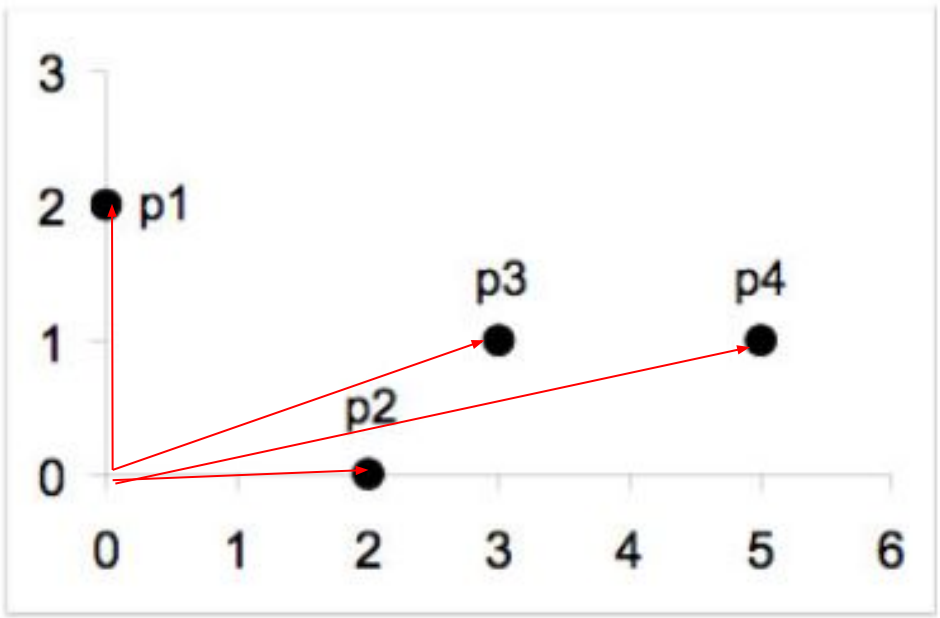
РАЗРЕЖЕННЫЕ МАТРИЦЫ




```
[ ]  from scipy.sparse import csr_matrix
     csr_array = csr_matrix(array)
```

ВЕКТОРЫ

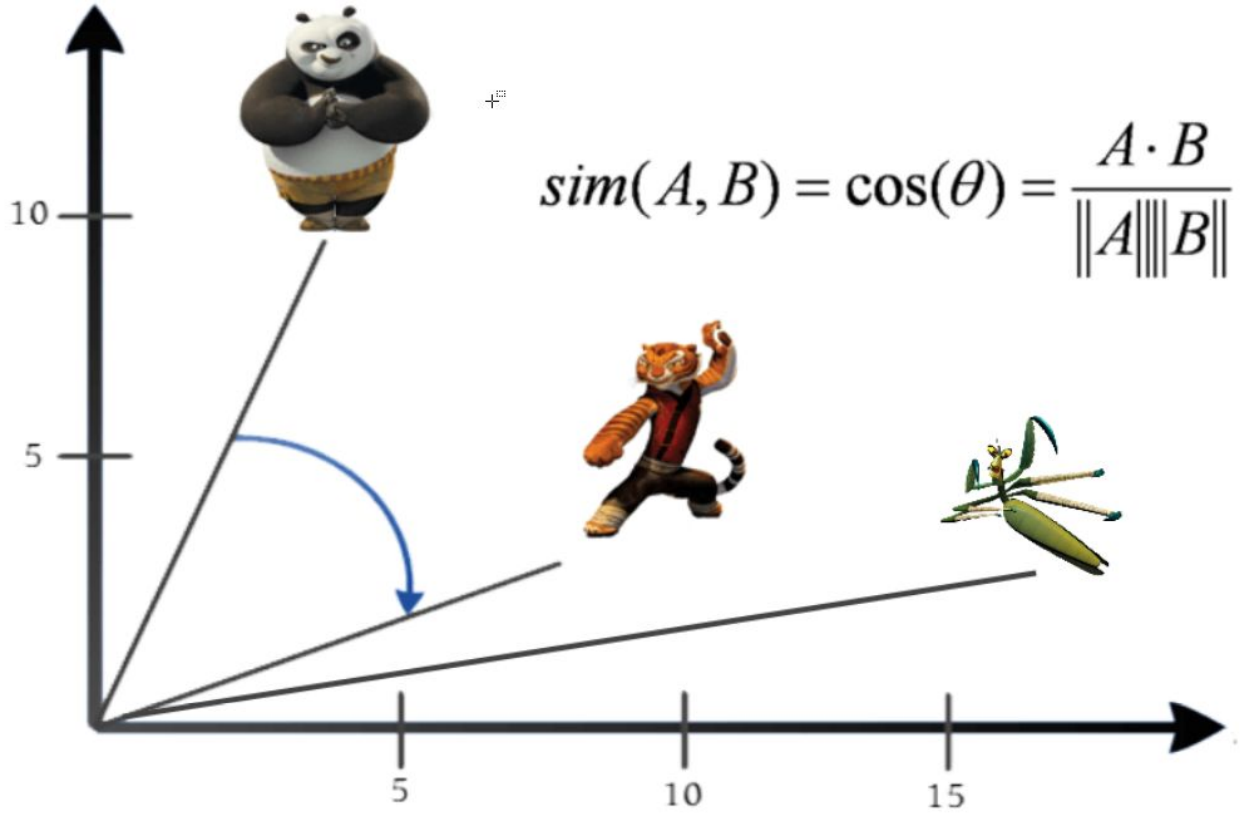




point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

ВЕКТОРЫ

Сила



$$sim(A,B) = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|}$$

```
[ ]  # определение величины угла между векторами
     def cosine(A, B):
       len_a = np.linalg.norm(a) # определение длин векторов
       len_b = np.linalg.norm(b)
       return np.dot(a, b) / (len_a * len_b)
```

```
[ ]  # величина угла в радианах
     np.arccos(cosine(A, B))
     # величина угла в градусах
     np.arccos(cosine(A, B))*360 / 2 / np.pi
```

Ловкость

Источник: medium.com

РЕКОМЕНДАТЕЛЬНЫЕ СИСТЕМЫ

Подобрали для вас

38 990 ₽

Распошивальная машина Janome Cover...

10 190 ₽

Беспроводные наушники Apple AirPo...

19 790 ₽

Швейная машина Janome Clio 325

44 990 ₽ ~~49 990 ₽~~

Часы SUUNTO 9 Baro

74 620 ₽ ~~97 970 ₽~~

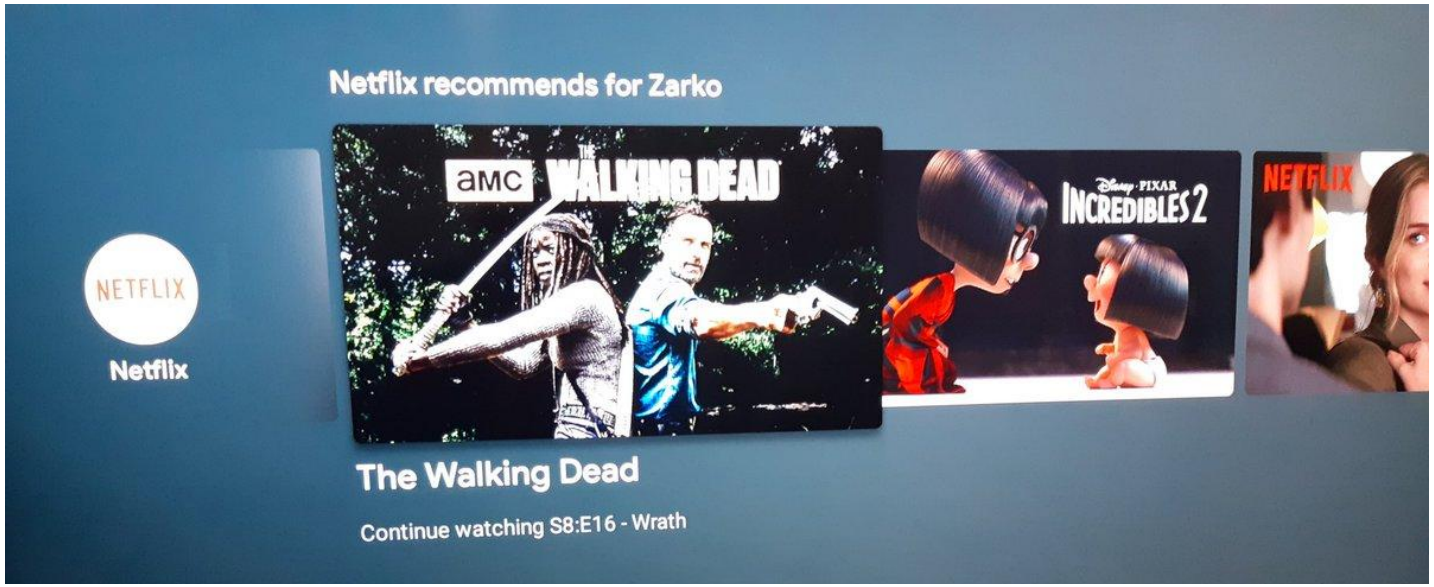
Вышивальная машина Janome Memory Craft...

380 ₽

Смесь Nutr 1 Premium

РЕКОМЕНДАТЕЛЬНЫЕ СИСТЕМЫ

NETFLIX



Netflix Prize

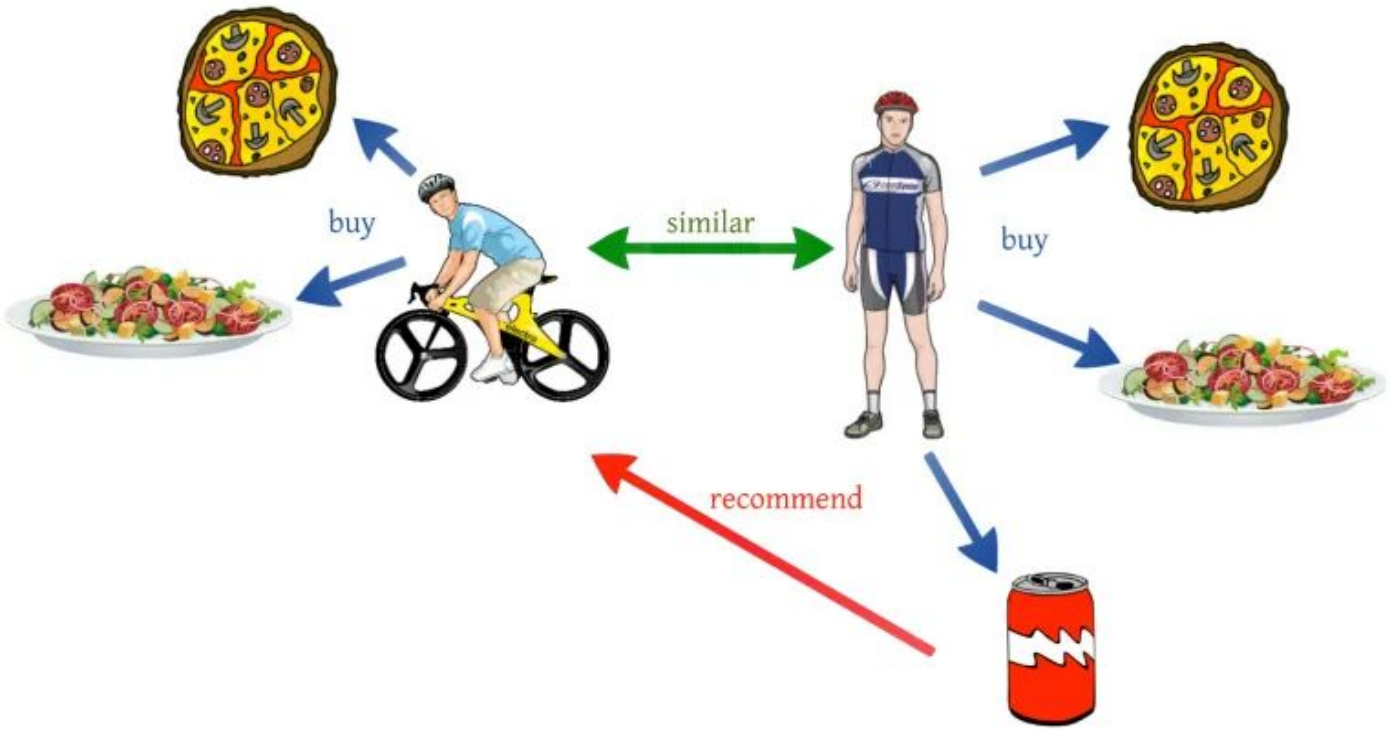
Rank	Team Name	Best Score	% Improvement	Last Submit Time
1	The Ensemble	0.8553	10.10	2009-07-26 18:38:22
2	BellKor's Pragmatic Chaos	0.8554	10.09	2009-07-26 18:18:28

Grand Prize - RMSE <= 0.8563

РЕКОМЕНДАТЕЛЬНЫЕ СИСТЕМЫ TIKTOK

COLLABORATIVE FILTERING

	item1	item2	item3	item4	item5	item6	item7	item8
user1						1	1	
user2	1	1	1					
user3		1		1	1		1	
user4	1	1		1		1	1	1

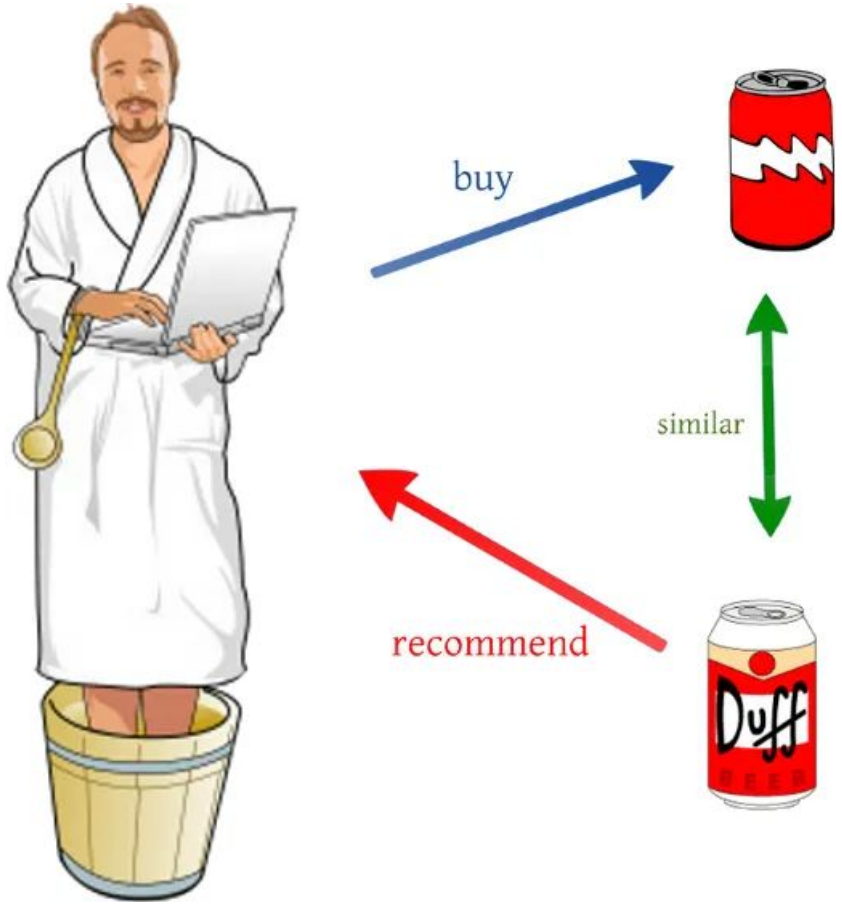


> people with similar taste to you like the thing you like.

Источник: dataaspirant.com

CONTENT-BASE FILTERING

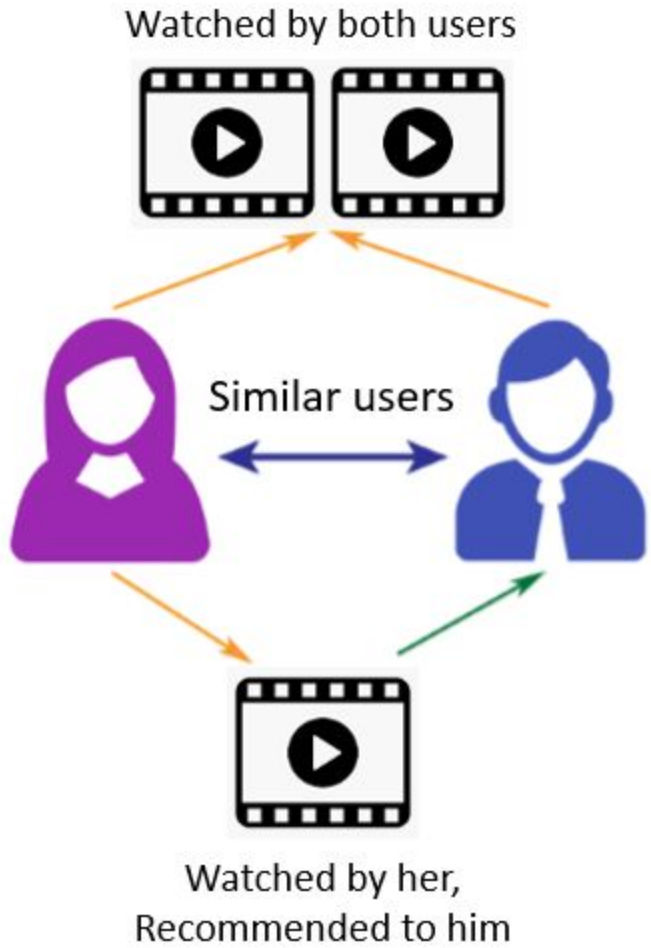
	user1	user2	user3	user4
item1		1		1
item2		1	1	1
item3		1		
item4			1	1
item5			1	
item6	1			1
item7	1		1	1
item8				1



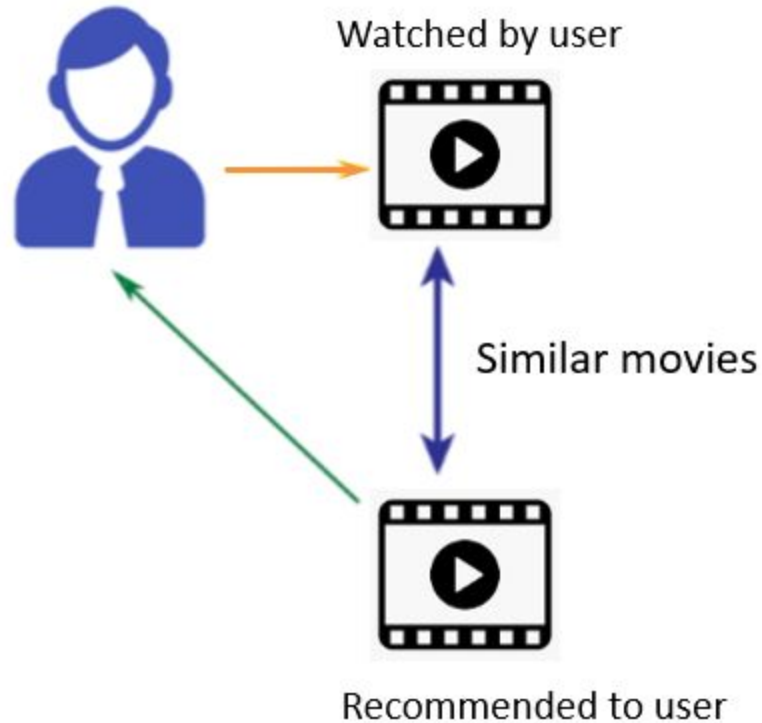
> people who liked this also liked these as well

Источник: dataaspirant.com

Collaborative Filtering



Content-Based Filtering



Google Colab

РЕЗЮМЕ УРОКА

- Вспомнили про матрицы и узнали как представлять данные в виде матриц в Numpy
- Научились измерять меру близости между векторами
- Разработали базовый алгоритм рекомендательной системы

ДОМАШНЕЕ ЗАДАНИЕ

Решить кейс:

Руководство решает внедрить фичу “С этим товаром покупают” в Интернет-магазине. Вам предлагается протестировать фичу на одном из товаров. Для тестирования фичи вам исходя из истории покупок в интернет-магазине нужно определить ТОП-10 наиболее близких товаров к исходному.

1. Используйте датасет с практики текущего урока.
2. Создайте матрицу item-customer (по строкам - товары, по столбцам - покупатели)
3. Проведите оценку мер близости товаров, получив матрицу item_item_sim_matrix со значениями косинусов между векторами товаров.
4. Отберите ТОП-10 похожих товаров по StockCode.
5. Выведите список ТОП-10 похожих товаров с названиями (Description) на экран.

Исходный товар - StockCode: 23166 Description: MEDIUM CERAMIC TOP STORAGE JAR

Формат - ссылка на ноутбук Colab.

Дополнительные материалы

1. Как работают рекомендательные системы
2. Рекомендательные системы: как помочь пользователю найти то, что ему нужно?
3. Введение в numpy видео
4. Рекомендательная система (content based) на данных Netflix
5. Рекомендательная система (collaborative filtering) на данных H&M

ВАШИ ВОПРОСЫ