

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

**«Лабораторная работа 2. Основы работы
с библиотекой NumPy»**

**ОТЧЕТ
по лабораторной работе №2
дисциплины
«Основы распознавания образов»**

Выполнил:
Луценко Дмитрий Андреевич
2 курс, группа ПИЖ-б-о-21-1,
09.03.04 «Программная инженерия»,
направленность (профиль) «Разработка
и сопровождение программного
обеспечения», очная форма обучения

(подпись)

Проверил:

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2022 г.

Лабораторная работа 2. Основы работы с библиотекой NumPy

Цель работы: исследовать базовые возможности библиотеки NumPy языка программирования Python.

Ход работы:

18. Дана целочисленная квадратная матрица. Определить:

- сумму элементов в тех строках, которые не содержат отрицательных элементов;
- минимум среди сумм элементов диагоналей, параллельных главной диагонали матрицы.

```
m = np.matrix('1 -2 3 54 -4; 43 1 8 21 64; -983 27 0 44 32 ; 11 22 33 44 55; 7 2 98 -45 23')
print(m)

[[ 1 -2  3 54 -4]
 [43  1  8 21 64]
 [-983 27  0 44 32]
 [11 22 33 44 55]
 [ 7  2 98 -45 23]]
```

1. Для нахождения суммы элементов в строках, не содержащих отриц. элементов воспользуемся функцией where

```
In [33]: np.where((m >= 0).all(axis=1), m.sum(axis=1), "String contains negative element")
Out[33]: array(['String contains negative element',
               ['137'],
               ['String contains negative element'],
               ['165'],
               ['String contains negative element']], dtype='<U32')
```

2. Сначала определим размер матрицы и создадим список с суммой элементов каждой диагонали, а затем найдём наименьшее число в этом списке

```
In [40]: size = m.shape
n = size[0]

min_elems = [(np.diagonal(m, offset = i, axis1=0, axis2=1)).sum() for i in range(-n, n)]
print(min_elems)
print(f'Minimal sum of elements: ', min(min_elems))

[0, 7, 13, -863, 58, 69, 105, 56, 118, -4]
Minimal sum of elements: -863
```

Рисунок 1 – Индивидуальное задание

Создать ноутбук, в котором выполнить решение вычислительной задачи (например, задачи из области физики, экономики, математики, статистики и т. д.), условие которой предварительно необходимо согласовать с преподавателем.

1.Г. За период с 1960 по 1986 г. прирост производительности труда в США составил 35% прироста производительности труда в Японии. Темп роста этого показателя составил в США 0,5% в год, а уровень Японии составлял в 1960 г. 22% уровня США. Если указанная тенденция сохранится, когда Япония догонит США по производительности труда?

Рисунок 2 – Экономическая задача

```
In [1]: import numpy as np
```

Эта прогнозная задача опирается на принцип экстраполяции, то есть распространения нынешних тенденций в будущее.

Расчеты строятся следующим образом. Обозначим через x темп роста производительности труда в Японии, тогда: $0,35 * x$; — это темп роста производительности в США, равный 0,5%.

Отсюда $x = 0,5/0,35$

```
In [12]: x = 0.5 / 0.35  
np.around(x, decimals = 2)
```

```
Out[12]: 1.43
```

Пусть t — искомое число лет начиная с 1960 г. Тогда уравнение для определения t таково:

$$(1,0143)^t * 22 = 100 * (1,005)^t$$

В результате, логарифмируя это уравнение, получаем

$$t * \log(1,00925) = \log(100/22)$$

```
In [23]: t = np.log(100/22) / np.log(1.00925)  
print(f'Это произойдёт через', np.around(t, decimals = 1), 'года')
```

Это произойдёт через 164.4 года

Рисунок 2 – Решение экономической задачи

Вывод: в ходе лабораторной работы были исследованы базовые возможности библиотеки NumPy языка программирования Python.

Ответы на контрольные вопросы:

1. Каково назначение библиотеки NumPy? NumPy — это библиотека для языка программирования Python, которая предоставляет в распоряжение разработчика инструменты для эффективной работы с многомерными массивами и высокопроизводительные вычислительные алгоритмы.

2. Что такое массивы ndarray? Narray — это объект библиотеки NumPy, который означает n -мерный массив.

3. Как осуществляется доступ к частям многомерного массива?

По элементам, например $m[1, 0]$;

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

по строкам, например $m[1, :]$;

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

по столбцам, например `m[:, 2]`;

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

по частям строки матрицы, например `m[1, 2:]`;

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

по частям столбца матрицы, например `m[0:2, 1]`;

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

по непрерывной части матрицы `m[0:2, 1:3]`;

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

либо по произвольным строкам/столбцам, например `>>cols = [0, 1, 3]`
`>>m[:, cols]`

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

4. Как осуществляется расчет статистик по данным? Для определения размерности массива используется `shape`, для вычисления статистики по строкам в функцию нужно передать параметр `axis=1`, а по столбцам `axis=0`.

Функции для вычисления статистик:

Имя метода	Описание
argmax	Индексы элементов с максимальным значением (по осям)
argmin	Индексы элементов с минимальным значением (по осям)
max	Максимальные значения элементов (по осям)
min	Минимальные значения элементов (по осям)
mean	Средние значения элементов (по осям)
prod	Произведение всех элементов (по осям)
std	Стандартное отклонение (по осям)
sum	Сумма всех элементов (по осям)
var	Дисперсия (по осям)

5. Как выполняется выборка данных из массивов `ndarray`?