

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
**Федеральное государственное автономное образовательное
учреждение высшего образования**
«СЕВЕРОКАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Кафедра инфокоммуникаций
Институт цифрового развития

ОТЧЁТ

по лабораторной работе №3.2

Дисциплина: «Программирование на Python»

Тема: «Основы работы с библиотекой NumPy»

Выполнила: студентка 2 курса,
группы ИВТ-б-о-21-1
Диченко Дина Алексеевна

Ставрополь 2023

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

Практическая часть:

1. Создала общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT и выбранный Вами язык программирования (выбор языка программирования будет доступен после установки флажка Add .gitignore).

Owner * Repository name *

DinaDichenko / Lab_3.2 ✓

Great repository names are short and memorable. Need inspiration? How about [improved-winner?](#)

Description (optional)

☒ **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☐ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: Python ▼

Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)

License: MIT License ▼

You are creating a public repository in your personal account.

Create repository

Рисунок 1. Создание репозитория

2. Выполнила клонирование созданного репозитория на рабочий компьютер.

3. Организовала свой репозиторий в соответствии с моделью ветвления git-flow.

```
C:\Users\student-09-331\Downloads\Lab_3.2>git flow init
Which branch should be used for bringing forth production releases?
- develop
- main
Branch name for production releases: [main] main
Which branch should be used for integration of the "next release"?
- develop
Branch name for "next release" development: [develop] develop
How to name your supporting branch prefixes?
Feature branches? [feature/] fea
Bugfix branches? [bugfix/] bug
Release branches? [release/] rel
Hotfix branches? [hotfix/] hot
Support branches? [support/] sup
Version tag prefix? [] pre
Hooks and filters directory? [C:/Users/student-09-331/Downloads/Lab_3.2/.git/hooks] hook
```

Рисунок 2. Организация репозитория в соответствии с git-flow

4. Дополнила файл .gitignore необходимыми правилами для выбранного языка программирования, интерактивной оболочки Jupyter notebook и интегрированной среды разработки.

```
# Byte-compiled / optimized / DLL files
__pycache__/
*.py[cod]
*$py.class

# C extensions
*.so

# Distribution / packaging
.Python
build/
develop-eggs/
dist/
downloads/
eggs/
.eggs/
lib/
lib64/
parts/
sdist/
var/
wheels/
pip-wheel-metadata/
share/python-wheels/
*.egg-info/
.installed.cfg
*.egg
```

Рисунок 3. Изменение .gitignore

5. Проработала примеры лабораторной работы.

6. Решить задания в ноутбуках, выданных преподавателем.

```
In [6]: # основная структура данных - массив
a = np.array([1, 2, 3, 4, 5])
b = np.array([0.1, 0.2, 0.3, 0.4, 0.5])

print("a =", a)
print("b =", b)

a = [1 2 3 4 5]
b = [0.1 0.2 0.3 0.4 0.5]

Создайте массив с 5 любыми числами:
```

```
In [9]: a = np.array([1, 2, 3, 4, 5])

print(a)

[1 2 3 4 5]

Арифметические операции, в отличие от операций над списками, применяются поэлементно:
```

```
In [8]: list1 = [1, 2, 3]
array1 = np.array([1, 2, 3])

print("list1:", list1)
print('\tlist1 * 3:', list1 * 3)
print('\tlist1 + [1]:', list1 + [1])

print('array1:', array1)
print('\tarray1 * 3:', array1 * 3)
print('\tarray1 + 1:', array1 + 1)

list1: [1, 2, 3]
list1 * 3: [1, 2, 3, 1, 2, 3, 1, 2, 3]
list1 + [1]: [1, 2, 3, 1]
array1: [1 2 3]
array1 * 3: [3 6 9]
array1 + 1: [2 3 4]

Создайте массив из 5 чисел. Возведите каждый элемент массива в степень 3
```

Рисунок 4. Решение ноутбуков преподавателя

возвращаемое значение -- массив, содержащий результаты вычисления для каждого элемента (True -- да или False -- нет).

```
In [14]: print("a =", a)
print("\ta > 1: ", a > 1)
print("\nb =", b)
print("\tb < 0.5: ", b < 0.5)

print("\nОдновременная проверка условий:")
print("\t(a > 1) & (b < 0.5): ", (a > 1) & (b < 0.5))
print("\tА вот это проверяет, что a > 1 ИЛИ b < 0.5: ", (a > 1) | (b < 0.5))

a = [10 20 30 40 50]
a > 1: [ True  True  True  True  True]

b = [ 1  5 10  8  5]
b < 0.5: [False False False False False]

Одновременная проверка условий:
(a > 1) & (b < 0.5): [False False False False False]
А вот это проверяет, что a > 1 ИЛИ b < 0.5: [ True  True  True  True  True]

Создайте 2 массива из 5 элементов. Проверьте условие "Элементы первого массива меньше 6, элементы второго массива делятся на 3"
```

```
In [18]: mas1 = np.array([1, 3, 5, 7, 8])
mas2 = np.array([1, 3, 6, 8, 10])

print("mas1 = ", mas1)
print("\tmas1 > 6: ", mas1 > 6)
print("mas1 = ", mas1)
print("\tmas2 % 3 == 0: ", mas2 % 3 == 0)

mas1 = [1 3 5 7 8]
mas1 > 6: [False False False  True  True]
mas1 = [1 3 5 7 8]
mas2 % 3 == 0: [False  True  True False False]

Теперь проверьте условие "Элементы первого массива делятся на 2 или элементы второго массива больше 2"
```

Рисунок 5. Решение ноутбуков преподавателя

```
In [7]: import numpy as np

m1 = np.array([2, 4, 6, 8, 10, 12])
m2 = np.array([ 7, 11, 15, 18, 23, 29])

print((m1+m2)**2)

[ 81 225 441 676 1089 1681]
```

2. Выведите все элементы из первого массива, индексы которых соответствуют индексам тех элементов второго массива, которые больше 12 и дают остаток 3 при делении на 5.

```
In [21]: print(m1[np.where((m2 > 12) & (m2 % 5 == 3))])

[ 8 10]
```

3. Проверьте условие "Элементы первого массива делятся на 4, элементы второго массива меньше 14". (Подсказка: в результате должен получиться массив с True и False)

```
In [23]: print((m1 % 4 == 0) & (m2 < 14))

[False  True  False  False  False  False]
```

Рисунок 6. Решение ноутбуков преподавателя

- Найдите интересный для вас датасет. Например, можно выбрать датасет тут: <http://data.un.org/Explorer.aspx> (выбираете датасет, жмете на view data, потом download, выбирайте csv формат)
- Рассчитайте подходящие описательные статистики для признаков объектов в выбранном датасете
- Проанализируйте и прокомментируйте содержательно получившиеся результаты
- Все комментарии оформляйте строго в ячейках формата markdown

```
In [32]: import csv

In [27]: with open('organizations_gdp_hist.csv', 'r', newline='') as csvfile:
data = csv.reader(csvfile, delimiter=',')
total = []
gdp_variation = []
for row in data:
    if row[4] == "Ingreso mediano alto":
        total.append(float(row[6]))
        gdp_variation.append(float(row[8]))

In [29]: arr_t = np.array(total)
arr_v = np.array(gdp_variation)

In [30]: print(f"Среднее значение ВВП в Ингресо медиано альт: {np.mean(arr_t)}")
print(f"Среднее значение изменения ВВП: {np.mean(arr_v)}")

Среднее значение ВВП в Ингресо медиано альт: 1531038886.3048372
Среднее значение изменения ВВП: 0.9125664043340592

In [33]: print(f"Минимальное значение ВВП в Ингресо медиано альт: {np.min(arr_t)}")
print(f"Минимальное значение изменения ВВП: {np.min(arr_v)}")

Минимальное значение ВВП в Ингресо медиано альт: 0.0
Минимальное значение изменения ВВП: -5.34027530229791

In [34]: print(f"Максимальное значение ВВП в Ингресо медиано альт: {np.max(arr_t)}")
print(f"Максимальное значение изменения ВВП: {np.max(arr_v)}")

Максимальное значение ВВП в Ингресо медиано альт: 9412034299.23122
Максимальное значение изменения ВВП: 10.7456116066371
```

Рисунок 7. Решение ноутбуков преподавателя

7. Создать ноутбук, в котором выполнить решение индивидуального задания. Ноутбук должен содержать условие индивидуального задания. При решении индивидуального задания не должны быть использованы условный оператор if, а также операторы циклов while for, а только средства библиотеки NumPy. Привести в ноутбуке обоснование принятых решений. Номер варианта индивидуального задания необходимо уточнить у преподавателя.

Индивидуальное задание:

Дана целочисленная квадратная матрица. Определить:

- сумму элементов в тех столбцах, которые не содержат отрицательных элементов;
- минимум среди сумм модулей элементов диагоналей, параллельных побочной диагонали матрицы.

```
Индивидуальное задание (вариант 5):  
Дана целочисленная квадратная матрица. Определить:  


- сумму элементов в тех столбцах, которые не содержат отрицательных элементов;
- минимум среди сумм модулей элементов диагоналей, параллельных побочной диагонали матрицы.

  
In [12]: import numpy as np  
  
k = int(input("Введите размерность матрицы: "))  
matrix = np.random.randint(-10, 20, size=(k, k))  
Введите размерность матрицы: 5  
  
In [13]: print("Матрица:\n", matrix)  
Матрица:  
[[ 9 -9  0 12  0]  
 [11 14 -6 -10 -7]  
 [ 8 11 -7 11  2]  
 [10  8 13 -2 12]  
 [ 9 13 15 12 12]]  
  
In [29]: print("Столбцы, не содержащие отрицательные элементы:\n", matrix[:, matrix.min(axis=0) > 0])  
Столбцы, не содержащие отрицательные элементы:  
[[ 9]  
 [11]  
 [ 8]  
 [10]  
 [ 9]]  
  
In [31]: sumP = np.sum(matrix[:, matrix.min(axis=0) > 0])  
print("Сумма элементов в столбцах, которые не содержат отрицательные элементы:", sumP)  
Сумма элементов в столбцах, которые не содержат отрицательные элементы: 47  
  
In [63]: mod = np.fliplr(np.absolute(matrix))  
print("Отзеркаленная матрица модулей элементов\n", mod)  
Отзеркаленная матрица модулей элементов  
[[ 9 11  8 10  9]  
 [ 9 13 15 12 12]  
 [10  8 13 -2 12]  
 [ 8 11 -7 11  2]  
 [11 14 -6 -10 -7]]
```

Рисунок 8. Решение индивидуального задания 1

8. 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.

Вычислить среднеквадратическую ошибку

Вычислить разницу между прогнозируемым значением оценок и истинным, с помощью возведенная в квадрат и усредненная по всему набору данных.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_{true} - y_{pred})^2$$

n - число рассматриваемых объектов, которое в нашем случае равно 50.

y - переменные, которые будут предсказаны. В нашем случае это оценка ученика

ytrue - истинное значение переменной, то есть так называемый правильный ответ.

ypred - предполагаемое значение переменной.

```
In [9]: import numpy as np
```

y_true и ypred являются массивами numpy с одинаковой длиной

mean() возвращает среднее значение набора данных

```
In [23]: def mse_loss(y_true, ypred):  
         return ((y_true - ypred) ** 2).mean()
```

```
y_true=np.random.randint(1,5,50)  
ypred=np.random.randint(1,5,50)
```

```
In [24]: print("Прогнозируемые значения: ", ypred)  
         print("Среднее прогнозируемых значений: ", ypred.mean())  
         print("Истинные значения: ", y_true)  
         print("Среднее рогнозируемых значений: ", ypred.mean())  
         print("Среднеквадратическая ошибка: ", mse_loss(y_true, ypred))
```

```
Прогнозируемые значения: [3 1 1 1 4 4 3 3 1 2 4 4 2 3 3 1 4 4 1 1 1 3 2 3 4 2 3 2 2 3 2 1 2 1 3  
 1 3 1 1 3 2 1 3 4 2 4 1 4]  
Среднее прогнозируемых значений: 2.42  
Истинные значения: [4 4 3 3 2 1 1 3 3 2 1 1 2 1 2 4 1 2 2 3 2 1 2 2 3 3 4 2 3 2 4 4 4 2 3 3 1  
 2 3 3 2 2 3 2 2 2 3 2 1 4]  
Среднее рогнозируемых значений: 2.42  
Среднеквадратическая ошибка: 2.08
```

Рисунок 9. Решение индивидуального задания 2

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?

Объект ndarray (что значит N-размерный массив). Этот объект является многомерным однородным массивом с заранее заданным количеством элементов. Однородный — потому что практически все объекты в нем одного размера или типа. На самом деле, тип данных определен другим объектом NumPy, который называется dtype (тип-данных).

Каждый ndarray ассоциирован только с одним типом dtype.

3. Как осуществляется доступ к частям многомерного массива?

Иногда возникает задача взять не все элементы строки, а только часть: рассмотрим пример, когда нам из второй строки нужно извлечь все элементы, начиная с третьего.

```
>>> m[1, 2:]  
  
matrix([[7, 8]])
```

Запись 2: означает, что начиная с третьего столбца включительно (т.к. нумерация начинается с 0, то третий элемент имеет индекс 2) взять все оставшиеся в ряду элементы.

Аналогично предыдущему примеру, можно извлечь только часть столбца матрицы.

```
\>>> m[0:2, 1:3]  
  
matrix([[2, 3],  
[6, 7]])
```

Numpy позволяет извлекать произвольный набор столбцов или строк матрицы, строки (столбцы) которые нужно извлечь передаются в виде списка.

4. Как осуществляется расчет статистик по данным?

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)

4. Как выполняется выборка данных из массивов ndarray?

Самым замечательным в использовании boolean массивов при работе с ndarray является то, что их можно применять для построения выборок.

Например:

```
>>> less_then_5 = nums < 5
```

```
>>> less_then_5
```

```
array([ True,  True,  True,  True, False, False, False, False, False, False])
```

Если мы переменную `less_then_5` передадим в качестве списка индексов для `nums`, то получим массив, в котором будут содержаться элементы из `nums` с индексами равными индексам `True` позиций массива `less_then_5`.

Вывод: в результате выполнения работы были исследованы базовые возможности библиотеки NumPy языка программирования Python.