

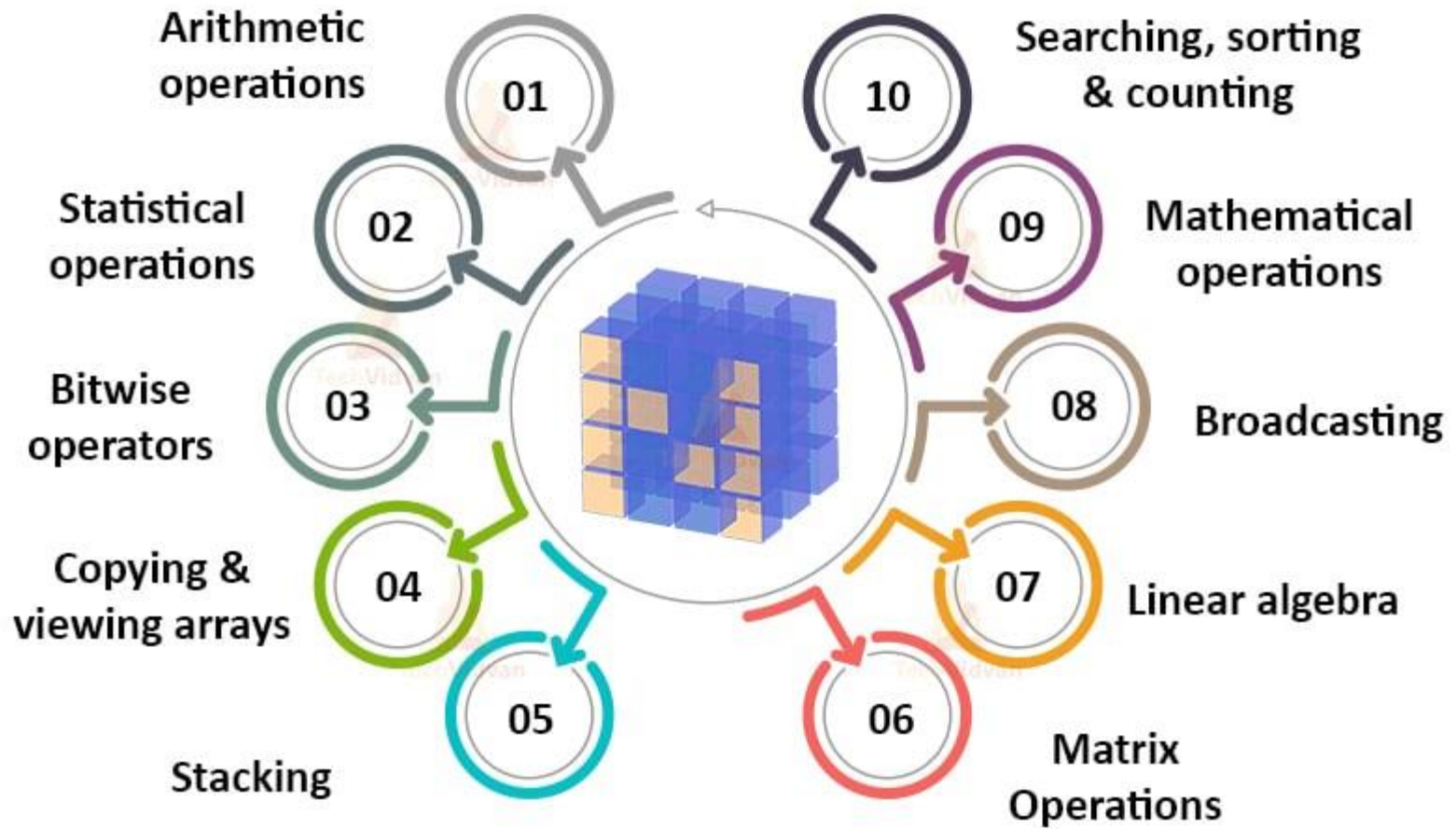
# მონაცემთა ანალიტიკა Python

---

ლექცია 3: NUMPY ბიბლიოთეკა - სკალარები, ვექტორები, მატრიცები. მატრიცების და ვექტორების ინდექსირება და ანათალი. ლოგიკური, არითმეტიკული და სტატისტიკური ოპერაციები მატრიცებზე. წრფივი ალგებრის საკითხები.

ლიკა სვანაძე  
lika.svanadze@btu.edu.ge

# Uses of NumPy



# Numpy

---

- Stands for Numerical Python
- The Numpy library provides specialized data structures, functions, and other tools for numerical computing and data analysis in Python
- Documentation: <https://numpy.org/doc/stable/user/quickstart.html>
- It is designed for efficiency on large arrays of data
- It provides
  - **ndarray** for creating multiple dimensional arrays
  - Internally stores data in a contiguous block of memory, independent of other built-in Python objects, **use much less memory** than built-in Python sequences.
  - Standard math functions for fast operations on entire arrays of data without having to write loops

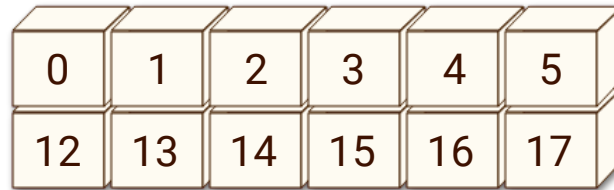
```
pip install numpy
```

# NumPy ndarray vs list

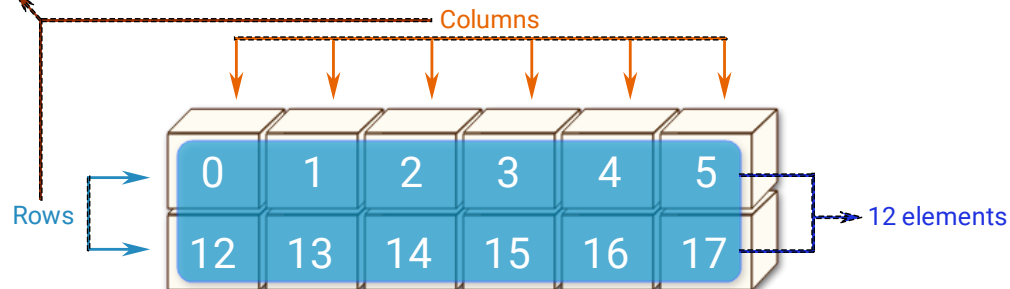
---

- One of the key features of NumPy is its N-dimensional array object, or ndarray, which is a fast, flexible container for large datasets in Python.
- NumPy-based algorithms are generally 10 to 100 times faster (or more) than their pure Python counterparts and use significantly less memory.

Original arrays:



2 dimensions



x.ndim

Number of dimensions:

2

x.size

Number of elements:

12

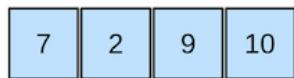
x.itemsize

Number of bytes for each element  
in the said array:

8

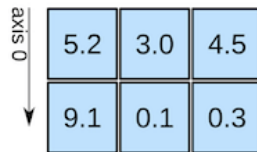
# NumPy ndarray

1D array



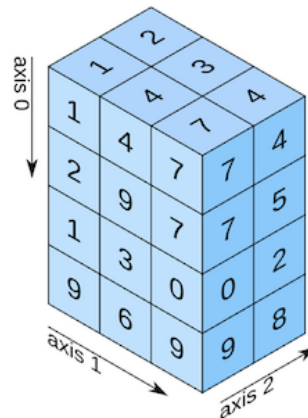
shape: (4,)

2D array



shape: (2, 3)

3D array

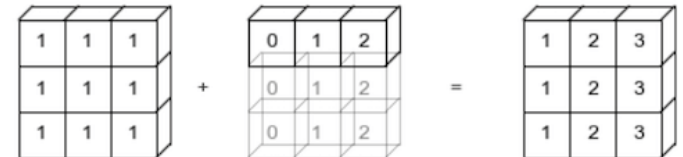


shape: (4, 3, 2)

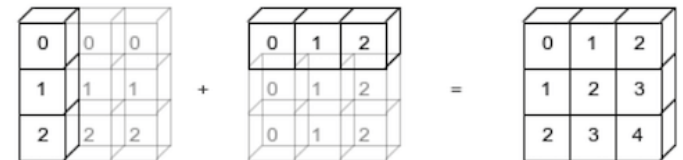
`np.arange(3)+5`



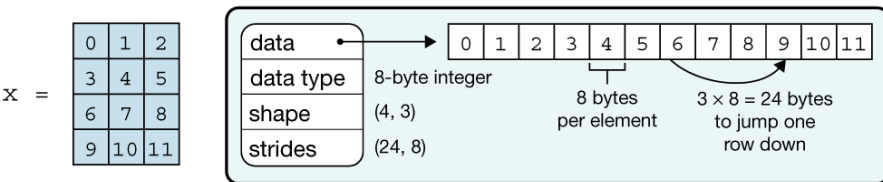
`np.ones((3,3))+np.arange(3)`



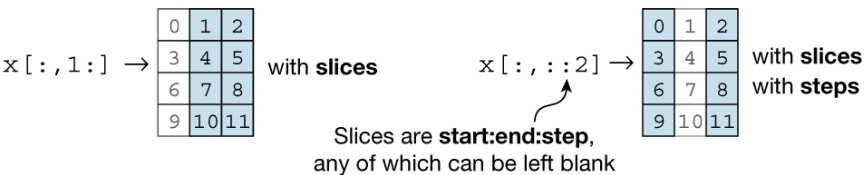
`np.arange(3).reshape((3,1))+np.arange(3)`



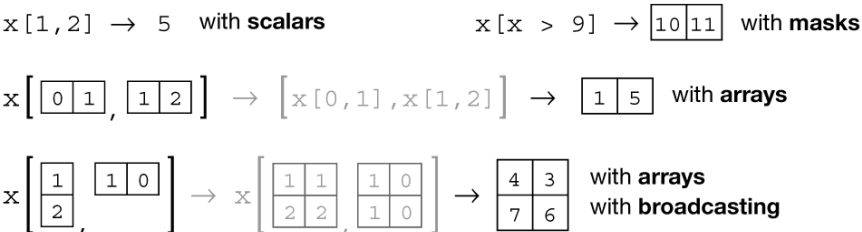
## a Data structure



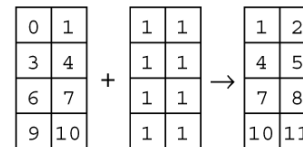
## b Indexing (view)



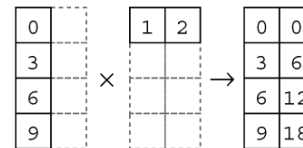
## c Indexing (copy)



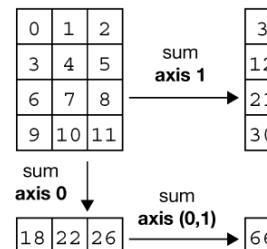
## d Vectorization



## e Broadcasting



## f Reduction



## g Example

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

```
In [2]: x = np.arange(12)
```

```
In [3]: x = x.reshape(4, 3)
```

```
In [4]: x
```

```
Out[4]:
```

```
array([[ 0,  1,  2],
       [ 3,  4,  5],
       [ 6,  7,  8],
       [ 9, 10, 11]])
```

```
In [5]: np.mean(x, axis=0)
```

```
Out[5]: array([4.5, 5.5, 6.5])
```

```
In [6]: x = x - np.mean(x, axis=0)
```

```
In [7]: x
```

```
Out[7]:
```

```
array([[ -4.5,  -4.5,  -4.5],
       [ -1.5,  -1.5,  -1.5],
       [  1.5,   1.5,   1.5],
       [  4.5,   4.5,   4.5]])
```

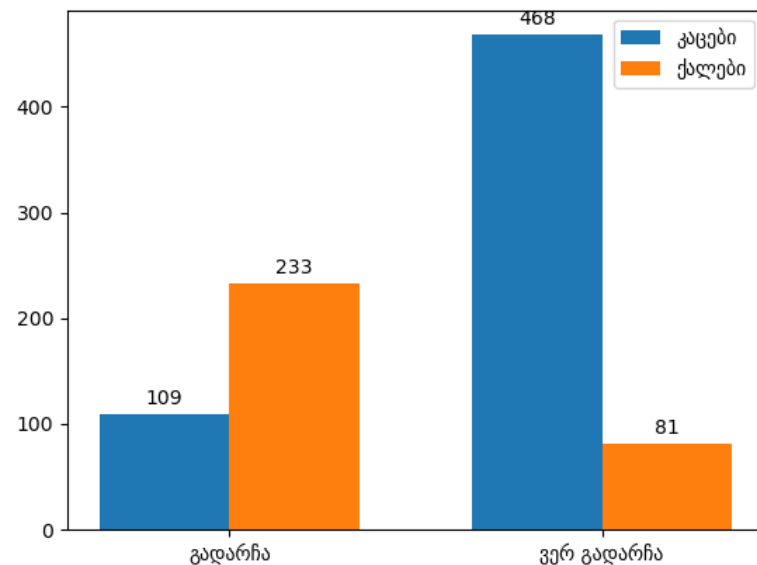
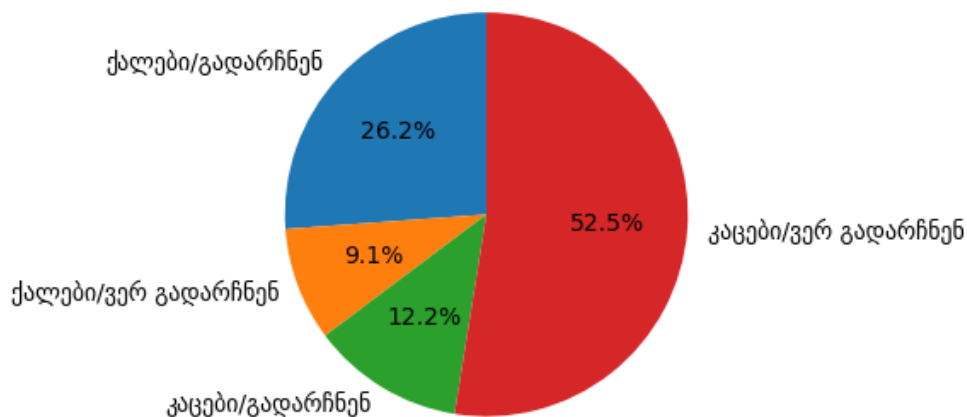
# მონაცემთა ანალიზი და ვიზუალიზაცია

- \* მას შემდეგ რაც ბაზიდან ინფორმაციას წავიკითხავთ, საჭიროა წამოღებული მონაცემების ანალიზი (მაგ. სტატისტიკური ანალიზი) და მათი ეფექტური ვიზუალიზაცია.
- \* მონაცემთა ანალიზი და ვიზუალიზაცია პროგრამირების ერთ-ერთი მნიშვნელოვანი კომპონენტია და პითონი ფართოდ გამოიყენება სხვადასხვა ტიპის მონაცემთა ანალიზისთვის.
- \* პითონში არსებობს სხვადასხვა მოდულები რომლებიც გამოიყენება მონაცემთა ანალიზისთვის, როგორიცაა numpy, pandas, ....., ხოლო ვიზუალიზაციისთვის იყენებენ matplotlib, seaborn.
- \* მონაცემები გრაფიკულად შესაძლებელია წარმოდგენილი იყოს ჰისტოგრამების, წრიული დიაგრამების (pie charts), plots (მათემატიკური ფუნქციები/გრაფიკები)-ის საშუალებით.
- \* Matplotlib-ის ოფიციალური საიტი: <https://matplotlib.org/>
- \* მოდულების დაყენება შესაძლებელია pip-ის მეშვეობით შემდეგნაირად:
  - \* pip install matplotlib
  - \* pip install numpy



# matplotlib - სავარჯიშო 1

- \* matplotlib ბიბლიოთეკის გამოყენებით შესაძლებელია შევქმნათ შემდეგი სახის გრაფიკული გამოსახულებები. იხილეთ პრაქტიკული სავარჯიშოების კოდი ლექცია 11-ის classwork.py ფაილში (ატვირთულია Google Classroom-ში). სურათზე მოცემულია titanic.sqlite ბაზის მონაცემების მიხედვით შესაბამისი დიაგრამები:



# matplotlib - სავარჯიშო 2

- \* სურათზე მოცემულია შობადობის სტატისტიკა საქართველოში წლების მიხედვით. გამოსახულია 3 წირი რომელიც ასახავს ჯამურ მონაცემებს და სქესის მიხედვით შობადობას. მონაცემები აღებულია საქართველოს სტატისტიკის ეროვნული სამსახურის ოფიციალური ვებ-გვერდიდან: <https://www.geostat.ge/>
- \* იხილეთ ამ სავარჯიშოს კოდი ლექცია 11-ის classwork.py ფაილში (ატვირთულია Google Classroom-ში):

