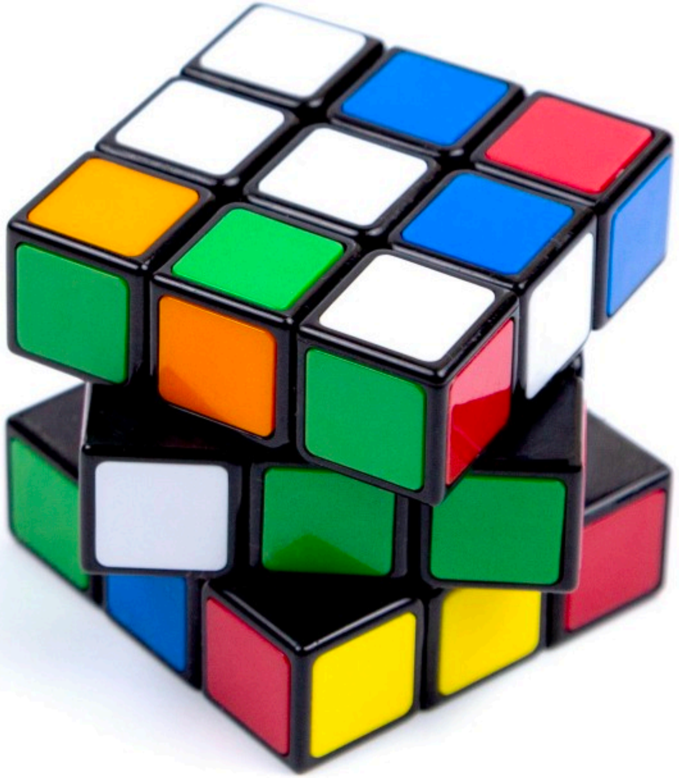
NumPy



Create GITHUB account

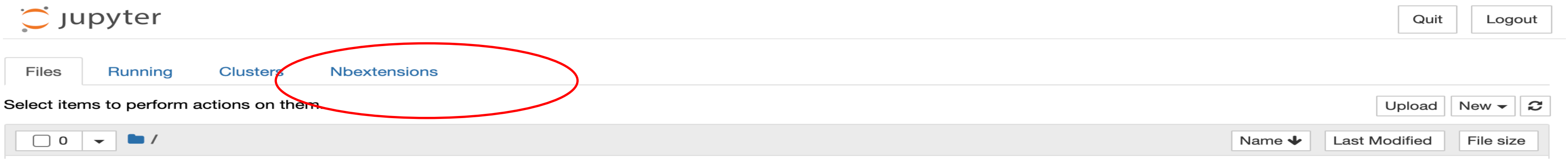
https://github.com/

Link gist with Jupyter notebook

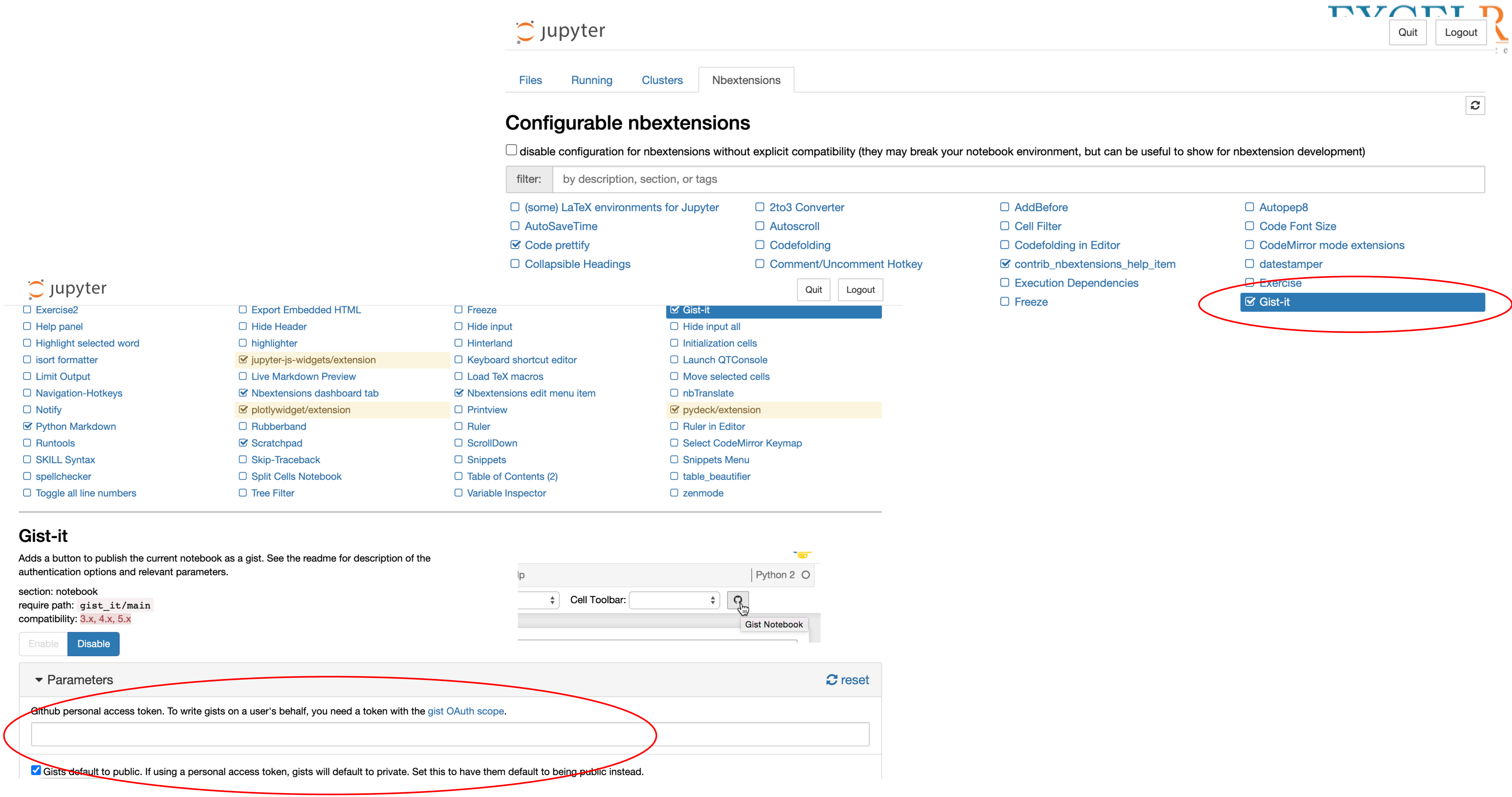
#Install jupyter extensions

Run below code in the notebook

```
!pip install jupyter_contrib_nbextensions
!jupyter contrib nbextension install --user
```



Select items to perform actions on them.



Configurable nbextensions

☐ disable configuration for nbextensions without explicit compatibility (they may break your notebook environment, but can be useful to show for nbextension development)

filter: by description, section, or tags

- ☐ (some) LaTeX environments for Jupyter
- ☐ AutoSaveTime
- ☑ Code prettify
- ☐ Collapsible Headings
- ☐ 2to3 Converter
- ☐ Autoscroll
- ☐ Codefolding
- ☐ Comment/Uncomment Hotkey
- ☐ AddBefore
- ☐ Cell Filter
- ☐ Codefolding in Editor
- ☑ contrib_nbextensions_help_item
- ☐ Execution Dependencies
- ☐ Freeze
- ☐ Autopep8
- ☐ Code Font Size
- ☐ CodeMirror mode extensions
- ☐ datestamper
- ☐ Exercise
- ☑ Gist-it

- ☐ Exercise2
- ☐ Help panel
- ☐ Highlight selected word
- ☐ isort formatter
- ☐ Limit Output
- ☐ Navigation-Hotkeys
- ☐ Notify
- ☑ Python Markdown
- ☐ Runtools
- ☐ SKILL Syntax
- ☐ spellchecker
- ☐ Toggle all line numbers
- ☐ Export Embedded HTML
- ☐ Hide Header
- ☐ highlighter
- ☑ jupyter-js-widgets/extension
- ☐ Live Markdown Preview
- ☑ Nbextensions dashboard tab
- ☑ plotlywidget/extension
- ☐ Rubberband
- ☑ Scratchpad
- ☐ Skip-Traceback
- ☐ Split Cells Notebook
- ☐ Tree Filter
- ☐ Freeze
- ☐ Hide input
- ☐ Hinterland
- ☐ Keyboard shortcut editor
- ☐ Load TeX macros
- ☑ Nbextensions edit menu item
- ☐ Printview
- ☐ Ruler
- ☐ ScrollDown
- ☐ Snippets
- ☐ Table of Contents (2)
- ☐ Variable Inspector
- ☑ Gist-it
- ☐ Hide input all
- ☐ Initialization cells
- ☐ Launch QTConsole
- ☐ Move selected cells
- ☐ nbTranslate
- ☑ pydeck/extension
- ☐ Ruler in Editor
- ☐ Select CodeMirror Keymap
- ☐ Snippets Menu
- ☐ table_beautifier
- ☐ zenmode

Gist-it

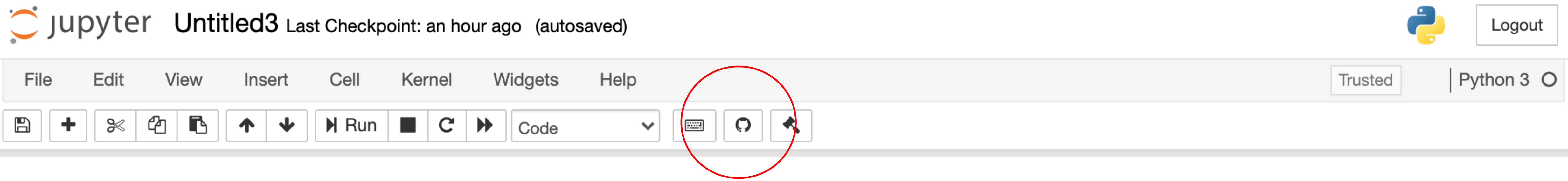
Adds a button to publish the current notebook as a gist. See the readme for description of the authentication options and relevant parameters.

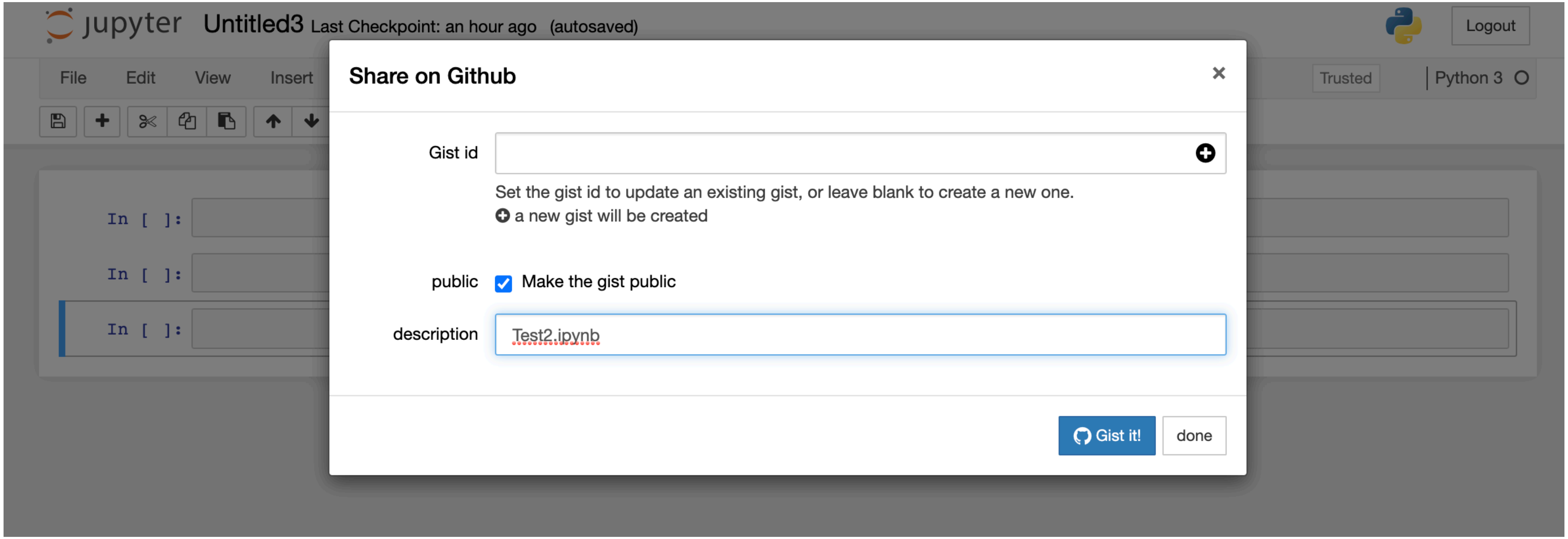
section: notebook
require path: `gist_it/main`
compatibility: 3.x, 4.x, 5.x

▾ Parameters ⟳ reset

Github personal access token. To write gists on a user's behalf, you need a token with the gist OAuth scope.

☑ Gists default to public. If using a personal access token, gists will default to private. Set this to have them default to being public instead.





Share on Github

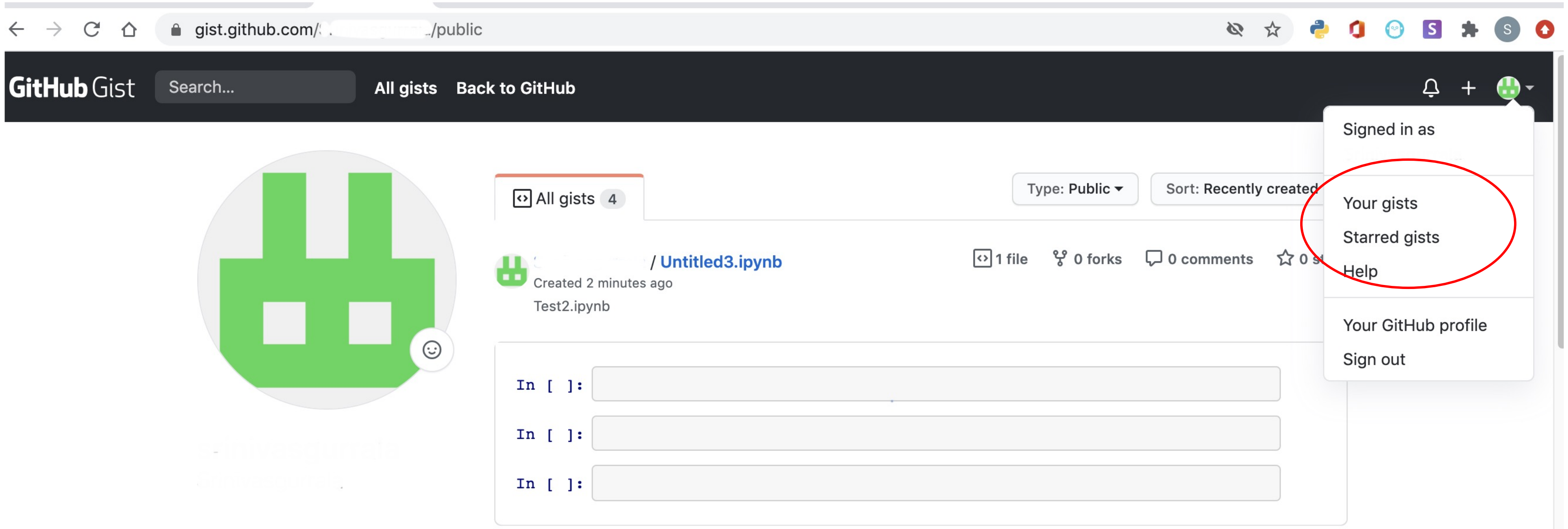
Gist id:

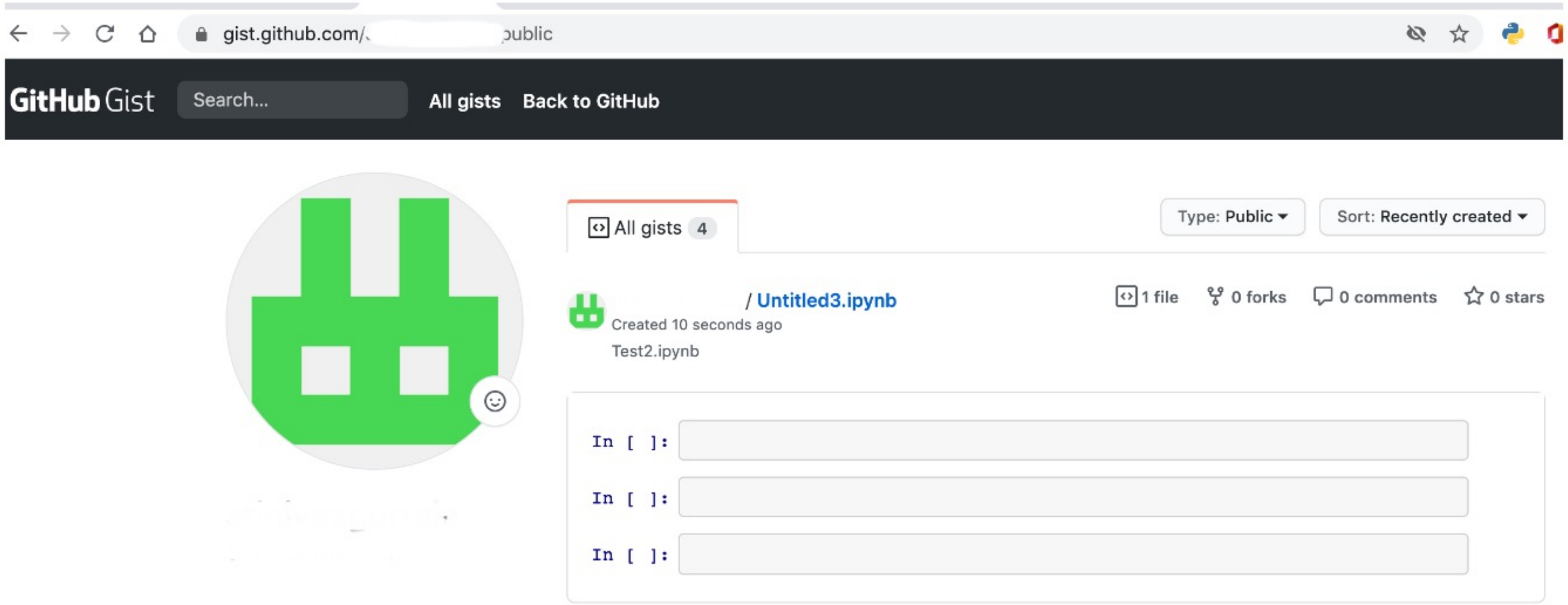
Set the gist id to update an existing gist, or leave blank to create a new one.

⊕ a new gist will be created

public: ☑ Make the gist public

description: Test2.ipynb


gist.github.com/[redacted]/public
GitHub Gist
Search...
All gists
Back to GitHub
Signed in as
Your gists
Starred gists
Help
Your GitHub profile
Sign out
All gists 4
Type: Public
Sort: Recently created
/ Untitled3.ipynb
Created 2 minutes ago
Test2.ipynb
1 file
0 forks
0 comments
0 s
In [ ]:
In [ ]:
In [ ]:


gist.github.com/[redacted]public
GitHub Gist
Search...
All gists
Back to GitHub
All gists 4
Type: Public
Sort: Recently created
/ Untitled3.ipynb
Created 10 seconds ago
Test2.ipynb
1 file
0 forks
0 comments
0 stars
In [ ]:
In [ ]:
In [ ]:

Numpy intro

NumPy is a Python package. It stands for 'Numerical Python'. It is a library consisting of multidimensional array objects and a collection of routines for processing of array.

Numeric, the ancestor of NumPy, was developed by Jim Hugunin.
Another package '**Numarray**' was also developed, having some additional functionalities.

In 2005, Travis Oliphant created NumPy package by incorporating the features of Numarray into Numeric package. There are many contributors to this open source project

Numpy

- NumPy contains a multi-dimensional array and matrix data structures.
- It can be utilised to perform a number of mathematical operations on arrays such as trigonometric, statistical, and linear algebra routines.
- NumPy is a wrapper around a library implemented in C.
- Pandas objects rely heavily on NumPy objects. Essentially, Pandas extends Numpy.

NumPy – A Replacement for MATLAB??

NumPy is often used along with packages like SciPy (Scientific Python) and Matplotlib(plotting library).

This combination is widely used as a replacement for MATLAB, a popular platform for technical computing.

However, Python alternative to MATLAB is now seen as a more modern and complete programming language.

NumPy’s main object is the homogeneous multi-dimensional array.
It is row/table/rectangular cuboid/etc. of elements (usually numbers), all of the same type, indexed by a tuple of positive integers.

As the name gives away, a NumPy array is a central data structure of the numpy library. The library’s name is short for “Numeric Python” or “Numerical Python”.

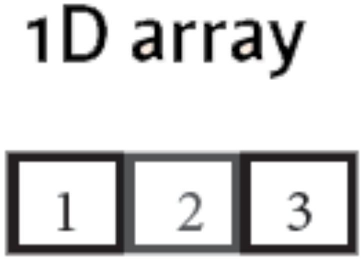
what is an array?

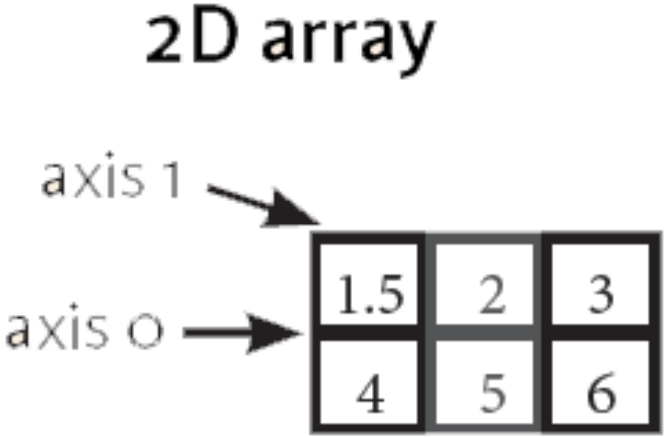
A numpy array is a grid of values, all of the same type, and is indexed by a tuple of nonnegative integers. The number of dimensions is the rank of the array; the shape of an array is a tuple of integers giving the size of the array along each dimension

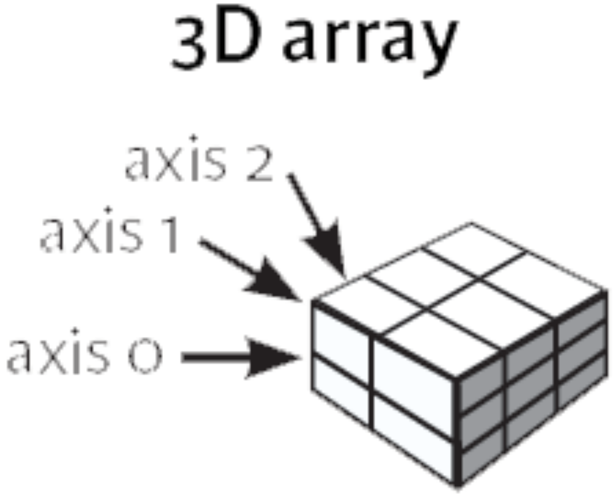
NumPy – ndarray Object

The most important object defined in NumPy is an N-dimensional array type called ndarray.

It describes the collection of items of the same type. Every item in an ndarray takes the same size of block in the memory







NumPy – Array Creation Routines

An instance of ndarray class can be constructed by different array creation routines. The basic ndarray is created using an array function in NumPy as follows:

```
numpy.array
```

Array from Existing Data

To make a numpy array, you can just use the np.array() function. All you need to do is pass a list to it, and optionally, you can also specify the data type of the data.

```
numpy.asarray(object, dtype=None)
```

Object: Input data in any form such as list, list of tuples, tuples, tuple of tuples or tuple of lists

dtype :By default, the data type of input data is applied to the resultant ndarray

How to do that

How To Load NumPy Arrays From Text

specific functions to load data from your files, such as loadtxt() or genfromtxt()

```
# This is your data in the text file
# Value1 Value2 Value3
# 0.2536 0.1008 0.3857
# 0.4839 0.4536 0.3561
# 0.1292 0.6875 0.5929
# 0.1781 0.3049 0.8928
# 0.6253 0.3486 0.8791
```

you use loadtxt() to load the data in your environment. You see that the first argument that both functions take is the text file data.txt. Next, there are some specific arguments for each: in the first statement, you skip the first row, and you return the columns as separate arrays with unpack=TRUE. This means that the values in column Value1 will be put in x, and so on.

```
# Import your data x, y, z = np.loadtxt('data.txt', skiprows=1, unpack=True)
```

Another function is genfromtxt()

Array from Numerical Ranges

This function returns an ndarray object containing evenly spaced values within a given range

```
numpy.arange(start, stop, step, dtype)
```

Start :The start of an interval. If omitted, defaults to 0
Stop :The end of an interval (not including this number)
Step :Spacing between values, default is 1
Dtype :Data type of resulting ndarray. If not given, data type of input is used

How to do that

Numpy data types

NumPy supports a much greater variety of numerical types than Python does

Data Types	Description
bool_	Boolean (True or False) stored as a byte
int8	Byte (-128 to 127)
int16	Integer (-32768 to 32767)
uint8	Unsigned integer (0 to 255)
float_	Shorthand for float64
complex_	Shorthand for complex128

Numpy Attributes

Numpy attributes gives us the info about the array

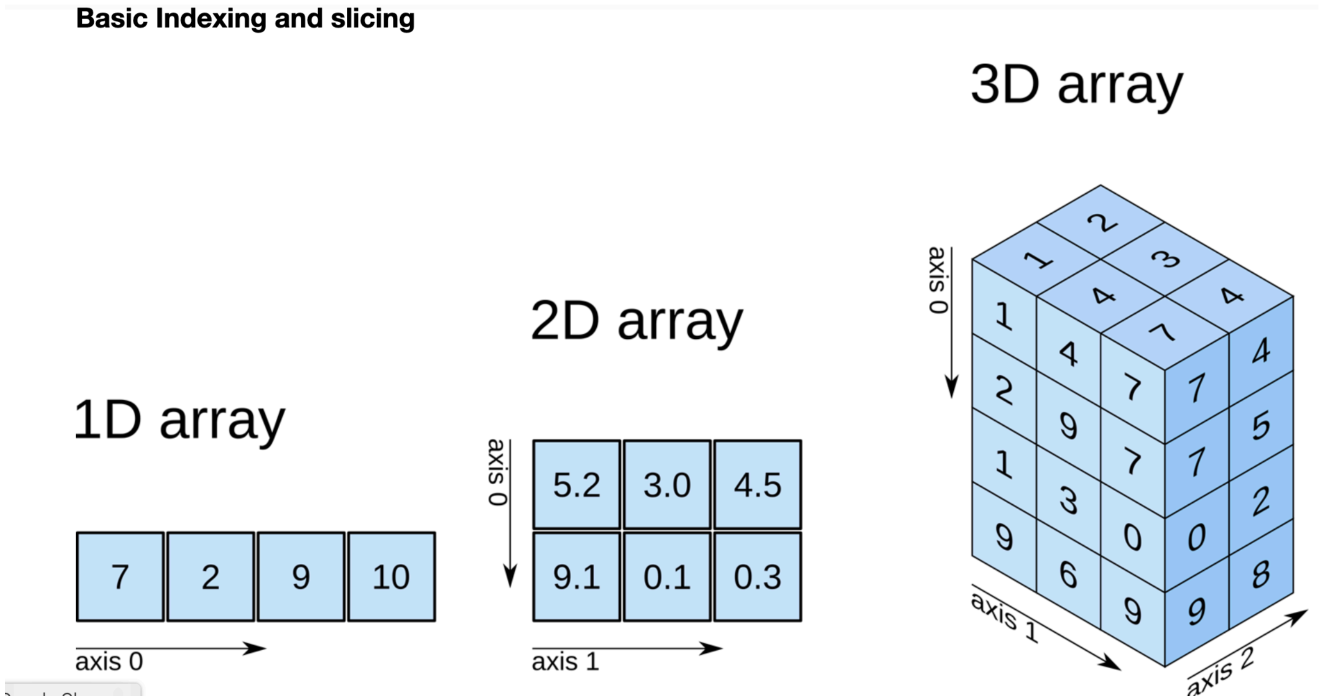
Each array from the defined values respectively have certain attributes which include
a) ndim (number of the dimensions) of the array.
b) shape(the size of each dimensions)
c) size(the total size of the array)
d) dtype(describes the data type of the array)

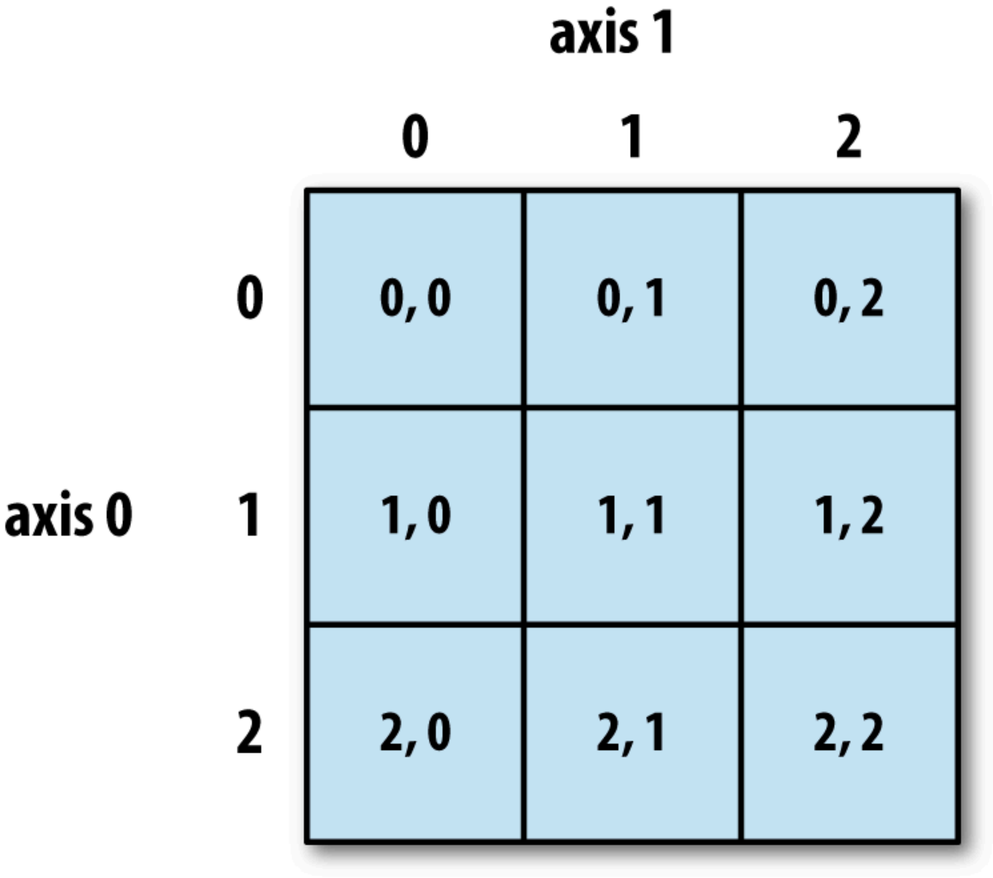
Indexing & Slicing

Items in ndarray object follows zero-based index.
Three types of indexing methods are available: **field access, basic slicing and advanced indexing.**

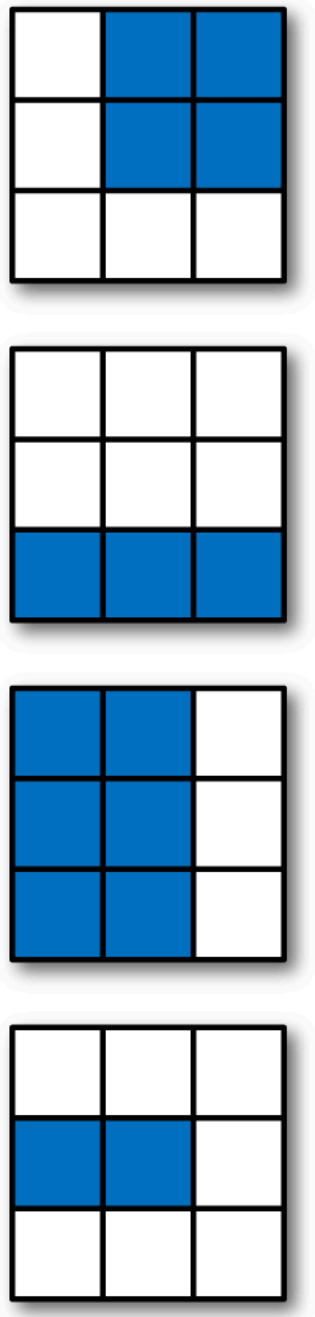
A Python slice object is constructed by giving start, stop, and step parameters to the built-in slice function. This slice object is passed to the array to extract a part of array

array[start:stop:step]





Indexing elements in a NumPy array



Expression	Shape
`arr[:2, 1:]`	`(2, 2)`
`arr[2]`	`(3,)`
`arr[2, :]`	`(3,)`
`arr[2:, :]`	`(1, 3)`
`arr[:, :2]`	`(3, 2)`
`arr[1, :2]`	`(2,)`
`arr[1:2, :2]`	`(1, 2)`

Advanced Indexing

It is possible to make a selection from ndarray that is a non-tuple sequence, ndarray object of integer or Boolean data type, or a tuple with at least one item being a sequence object.

There are two types of advanced indexing: Integer and Boolean.

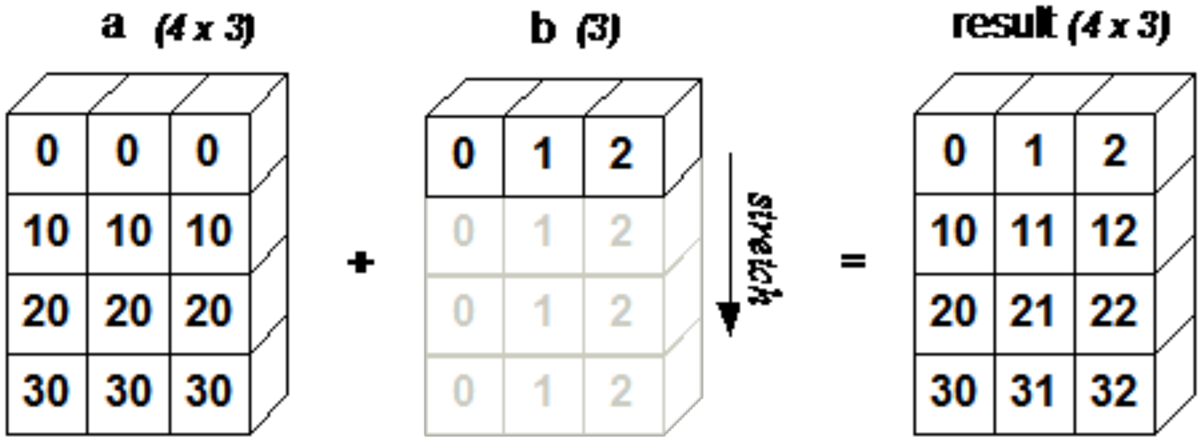
Integer :
This mechanism helps in selecting any arbitrary item in an array based on its Ndimensional index.

Boolean :
This type of advanced indexing is used when the resultant object is meant to be the result of Boolean operations, such as comparison operators

How to do that

Broadcasting

Broadcasting is the ability of NumPy to treat arrays of different shapes during arithmetic operations. Arithmetic operations on arrays are usually done on corresponding elements.



0	0	0
10	10	10
20	20	20
30	30	30

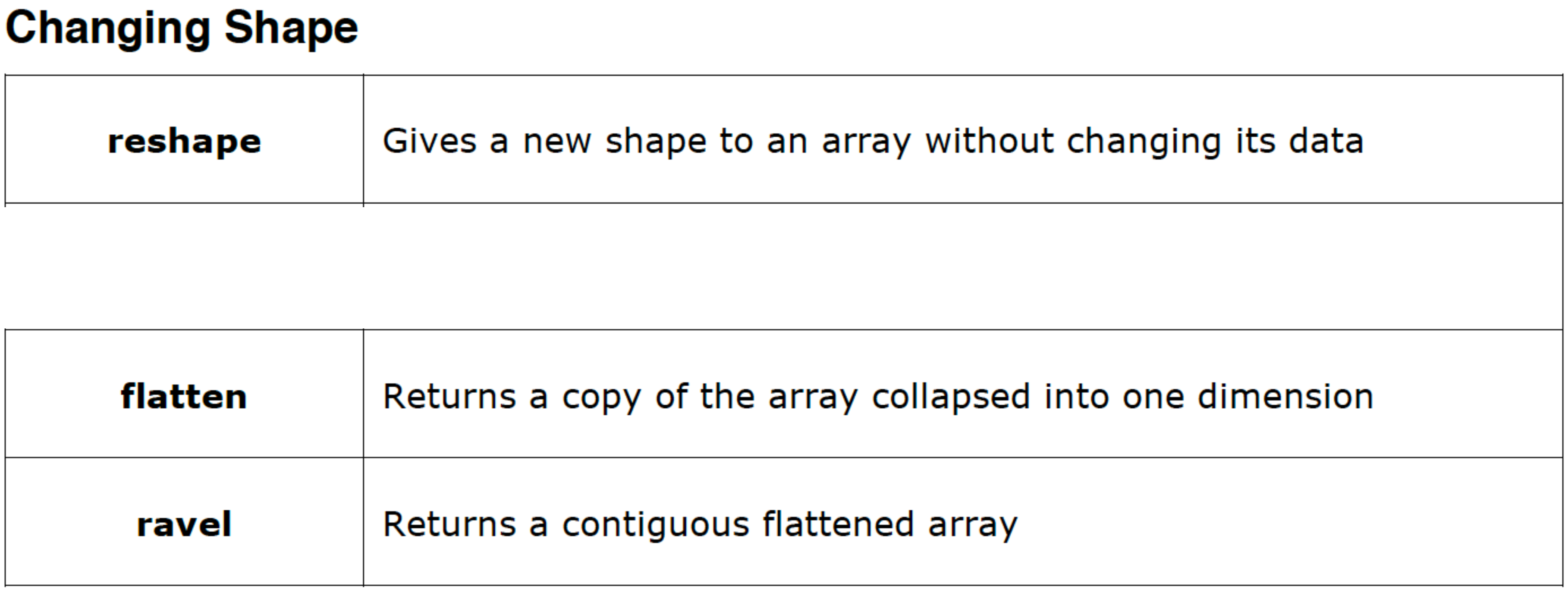
0	1	2
0	1	2
0	1	2
0	1	2

0	1	2
10	11	12
20	21	22
30	31	32

How to do that

Array Manipulation

Several routines are available in NumPy package for manipulation of elements in ndarray object. They can be classified into the following types



Changing Shape

reshape	Gives a new shape to an array without changing its data
flatten	Returns a copy of the array collapsed into one dimension
ravel	Returns a contiguous flattened array

How to do that

Joining Arrays

concatenate	Joins a sequence of arrays along an existing axis
stack	Joins a sequence of arrays along a new axis
hstack	Stacks arrays in sequence horizontally (column wise)
vstack	Stacks arrays in sequence vertically (row wise)

How to do that

Splitting Arrays

split	Splits an array into multiple sub-arrays
hsplit	Splits an array into multiple sub-arrays horizontally (column-wise)
vsplit	Splits an array into multiple sub-arrays vertically (row-wise)

Adding / Removing Elements

resize	Returns a new array with the specified shape
append	Appends the values to the end of an array
insert	Inserts the values along the given axis before the given indices
delete	Returns a new array with sub-arrays along an axis deleted
unique	Finds the unique elements of an array

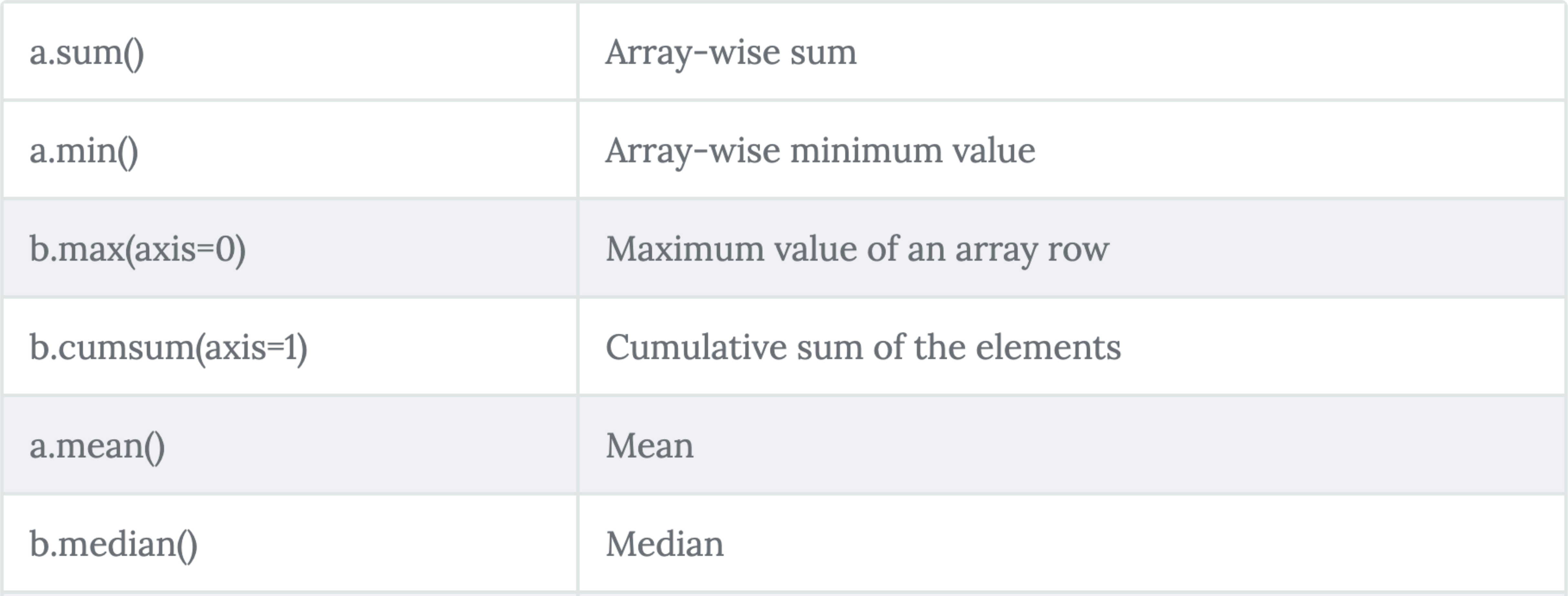
String Functions

add()	Concatenate two strings
capitalize()	first letter capitalized
title()	first letter of each word capitalized.
lower()	convert to lowercase
upper()	convert to upper case
split()	returns a list of words in the input string
strip()	strip of the specified characters leading and/or trailing in it
join()	returns a string in which the individual characters are joined by separator
replace()	string in which all occurrences of the sequence of characters is replaced by another given sequence

Arithmetic Operations

Arithmetic operations such as add(), subtract(), multiply(), power(), and divide()

Statistical Functions



a.sum()	Array-wise sum
a.min()	Array-wise minimum value
b.max(axis=0)	Maximum value of an array row
b.cumsum(axis=1)	Cumulative sum of the elements
a.mean()	Mean
b.median()	Median

Sort, Search & Counting

Numpy.sort()

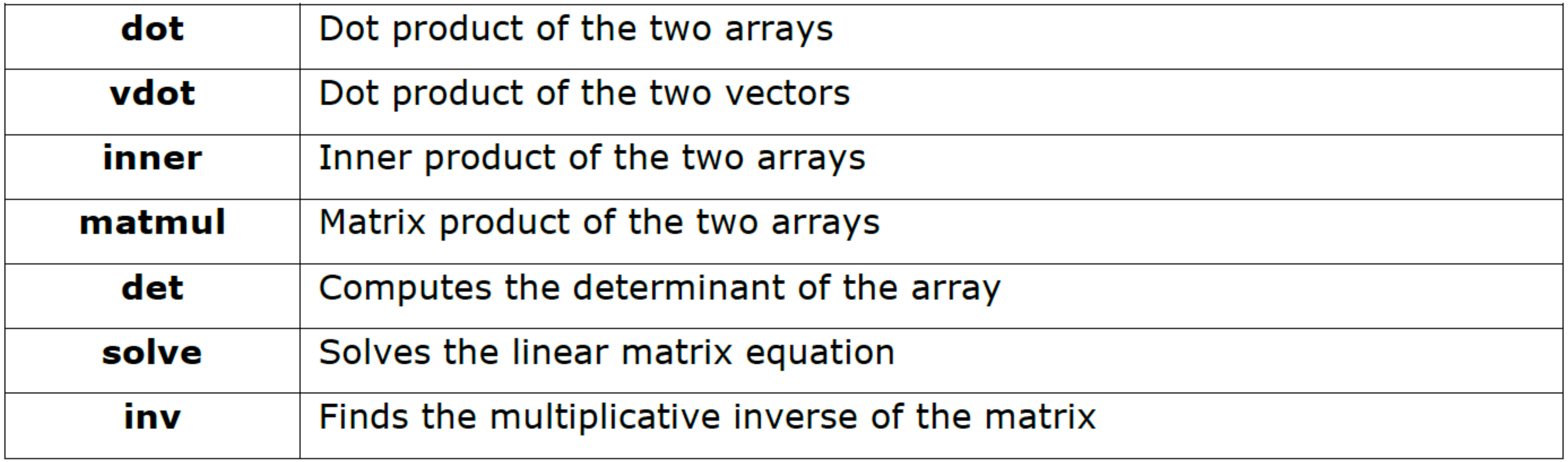
numpy.where()

Matrix operations

NumPy package contains a Matrix library numpy.matlib. This module has functions that return matrices instead of ndarray objects

Linear Algebra

NumPy package contains numpy.linalg module that provides all the functionality required for linear algebra



dot	Dot product of the two arrays
vdot	Dot product of the two vectors
inner	Inner product of the two arrays
matmul	Matrix product of the two arrays
det	Computes the determinant of the array
solve	Solves the linear matrix equation
inv	Finds the multiplicative inverse of the matrix

Thank you