PYTHON FOR DATA SCIENCE AND MACHINE LEARNING BOOTCAMP VIA UDEMY

Section 3: Jupyter Overview

6. Jupyter Notebooks

★ when opening a new file:

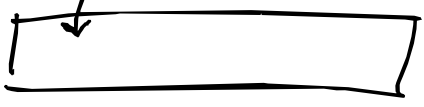
New > Notebook

Python [conda env: base]

★ to put a title on notebook:

→ just click on Untitled

★ Code cell

In [ ]: 

★ shift + Enter = run code/cell

★ alt (windows) + Enter = puts a new cell after a cell that already had input

option (mac)

★ save/download → straightforward

★ infinite while loop? Or error?

Kernel > Restart Kernel...

★ Help?

Help > Notebook Help

★ text that you see is called Markdown

★ to enable Markdown:

Code > Markdown

→ all become a markdown

→ like for comments

Section 3: Jupyter Overview

7. Optional: Virtual Environments

- Sometimes you'll want to program in different versions of a library
  - e.g. ☆ you locally a program with Scikit-Learn 0.17
    - ☆ Scikit-Learn 0.18 is released
    - ☆ you want to explore 0.18 but don't want your old code to break

- Sometimes you'll want to make sure your library installations are in the correct location
- E.g.:
  - ☆ you want multiple versions of Python on your computer
  - ☆ you want one environment with Py 2.7 and another with Py 3.5

! There is the virtual environment library for normal Python distributions

☆ Anaconda has a built-in virtual environment manager that makes the whole process really easy

☆ Check out the resource link for the official documentation that we will go over now

→ http://conda.pydata.org/docs/using/envs.html

☆ To create an environment: (do this using terminal)

conda create --name <my-env>

e.g. conda create --name fluffy numpy

Proceed ([y]/n)? y

Linking packages
COMPLETE

☆ To activate this environment, use:

activate fluffy

☆ To deactivate this environment, use:

deactivate

☆ >>> import numpy as np
☆ >>> import pandas as pd
☆ >>> quit()

→ (fluffy) C:\Users\Marisol>python

>>> import numpy as np
>>> import pandas as pd → ERROR!

→ so (fluffy) C:\Users\Marisol> conda install pandas

or

→ ☆ if deactivation

(fluffy) C:\Users\Marisol>deactivate

→ C:\Users\Marisol>conda create --name
mypython3version python=3.5 numpy
or
3.5 anaconda

Section 4: Python Crash Course

9. Introduction to Python Crash Course

Topics Covered

- Data Types
  - Numbers
  - Strings
  - Print Formatting
  - Lists
  - Dictionaries
  - Booleans
  - Tuples and Sets
- Comparison Operators
- if, else if, and else Statements
- For Loops
- While Loops
- range()
- List Comprehension
- Functions
- Lambda Expressions
- Map and Filter

Section 4: Python Crash Course

10. Python Crash Course - Part 1 of 6

Variable Assignment

* for python use _ (underscore)
  - e.g. name_of_var = 12

Strings Example:

In [36]:
```
num = 12
name = 'Sam'
```

* alt/option + enter:

In [37]:
```
print('My number is {} and my name is {}'.format(num, name))
```

* → after shift + enter:

Out[37]: My number is 12 and my name is Sam

G

[77]:
```
print('My number is {one} and my name is {two}, more {one}'.format(one=num, two=name))
```

*result after shift + Enter:

My number is 12 and my name is Sam, more 12

Indexing Strings

In [41]:
```
s = 'hello'
```

In [43]:
```
s[4]
```

Out[43]: 'o'

or

In [44]:
```
s = 'abcdefghijk'
```

In [46]:
```
s[0:]
```

Out[46]: 'abcdefghijk'

In [47]:
```
s[:3]
```

Out[47]: 'abc'

Lists

- sequence of elements in a set of square brackets separated by commas

[97]:
```
[1, 2, 3]
```

[97]:
```
[1, 2, 3]
```

[99]:
```
['a', 'b', 'c']
```

[99]:
```
['a', 'b', 'c']
```

[115]:
```
my_list = ['a', 'b', 'c']
```

[107]:
```
my_list.append('d')
```

[109]:
```
my_list
```

[109]:
```
['a', 'b', 'c', 'd']
```

[111]:
```
my_list[0]
```

[111]:
```
'a'
```