



Programación Estadística con Python

Sesiones 1 y 2

Alberto Sanz, Ph.D

asanz@edem.es

MASTER EN DATA ANALYTICS PARA LA EMPRESA

□ Objectives

□ Substantive:

- **Empower** the students in contexts in which data is relevant.
 - Make you **fluent** and **comfortable** in data management with Python.
 - Make you **competent** in data-based decision making.
 - Learning to **learn Python autonomously**

□ Procedures:

- **Case oriented** methodology.
 - Theory is to serve us (not us to serve theory)
- **Practical** approach: Course based on a real life case dataset. Examples, exercises and problem.

□ Objectives



- Make you **fluent** and **comfortable** in data management with Python.



Evaluation (percentage)

- | | | |
|-------------------------|----------------|----|
| □ Group challenge | (by Session 7) | 30 |
| □ Group Final Challenge | (Sessions 1 6) | 70 |
-

- Overview of the **Spyder environment** and of the **Python language**.
- Python as an object oriented language: Console examples.
- Our first dataset: an object of objects.
- Learning to learn Python. Some order in the internet Galaxy:
 - ▣ Valuable tutorials
 - ▣ Valuable forums
- Loading external packages: Reading external data with Python.
- Enriching our dataset: merging data.
- Our first plots in Python

- Overview of the **Spyder** environment and of the **Python** language.
- Python as an object oriented language: Console examples.
- Our first dataset: an object of objects.

Our first dataset: an object of objects

7

```
# Alberto Sanz.  
# 2019 09 01  
# Our first dataset  
  
import pandas as pd  
  
# Define variables.  
name = ['Bianca', 'Pedro', 'Alberto']  
gender = ['Female', 'Male', 'Male']  
age = [20, 35, 46]  
  
#create a dataframe  
class2019 = pd.DataFrame({'name': name, 'gender':  
                           gender, 'age':age})  
  
class2019.shape  
class2019.head()  
#QC OK  
  
#Clean up  
del (name, gender, age)  
  
# Export dataframe to Excel  
class2019.to_excel("class2019.xlsx")
```

Reading external data (I)

```
# Created on Thu Jul 25 11:37:07 2019
# @author: Alberto Sanz
# Reading external data in CSV

import os
import pandas as pd

# Change working directory
os.chdir('C:\carp_alb\EDEM\PEP\code_and_data')
os.getcwd()

#Reads data from CSV file and stores it in a dataframe called rentals_2011
#Pay attention to the specific format of your CSV data (; , or , .)

rentals_2011 = pd.read_csv ("washington_bike_rentals_2011.csv", sep=';',
decimal=',')
rentals_2011.shape
rentals_2011.head()
#QC OK
```


BREAK

Reading external data (II)

10

```
# Created on Thu Jul 25 11:37:07 2019
# @author: Alberto Sanz
# Reading external data from EXCEL

import os
import pandas as pd

# Change working directory
os.chdir('C:\carp_alb\EDEM\PEP\code_and_data')
os.getcwd()

#Reads data from EXCEL and stores it in a dataframe named rentals_2011

rentals_2011 = pd.read_excel ("washington_bike_rentals_2011.xlsx")
rentals_2011.shape
rentals_2011.head()

#Our first plot
```

Learning to learn Python

Some order in the internet Galaxy:

▣ Valuable forums:

- <https://stackoverflow.com>
- <https://www.datacamp.com>

▣ Valuable tutorials:

- <https://datatofish.com/python-tutorials/>
- <https://pbpython.com/>
- <https://matplotlib.org/tutorials/index.html>

Expanding our dataset (I)

12

```
# Load weather data in a new dataframe
weather_2011 = pd.read_csv ("weather_washington_2011.csv", sep=';',
                             decimal=',')

weather_2011.shape
weather_2011.head()
#QC OK

# Merge the two dataframes(rentals & weather) into a new single dataframe

rentals_weather_2011 =pd.merge(weather_2011, rentals_2011, on="day")
rentals_weather_2011.shape
rentals_weather_2011.head()
#QC OK

#####
## Extra TIP: explore the merge parameter how='' #
## how='outer' / how='inner' / how='left' / how='right' #
## to learn how to manage merge when the number of observations is not #
## the same in both files to be merged #
#####
```

Expanding our dataset (II)

13

```
# ADD NEW CASES (Rows) TO DATAFRAME
# Read cases from another year (2012) in a new dataframe
rentals_weather_2012 = pd.read_csv ("rentals_weather_2012.csv", sep=';',
                                     decimal=',')

rentals_weather_2012.shape
rentals_weather_2012.head()
# QC OK
# Check dimensionality of both dataframes
print (rentals_weather_2011.shape)
print (rentals_weather_2012.shape)
# QC OK

#WE CAN MERGE THE TWO DATA FRAMES IN A NEW ONE CONTAINING SAME
#VARIABLES (COLUMNS) BUT MORE CASES (ROWS)
rentals_weather_11_12 = rentals_weather_2011.append(rentals_weather_2012,
                                                    ignore_index=True)

print (rentals_weather_11_12.shape)
print (rentals_weather_11_12.head())
print (rentals_weather_11_12.tail())

#####
# Tricks of the trade: Column order is set alphabetically while merging
# You can restore it by doing:
rentals_weather_11_12 = rentals_weather_11_12[rentals_weather_2011.columns]
```

Questions?

Thank you !

Alberto Sanz
asanz@edem.es