Data Exploration and Preparation with Python

Exercise Sheet #2

Exercise 2.0: Install the Anaconda Distribution

Go to https://www.anaconda.com/distribution/ and download the Anaconda Distribution for Python 3.

As soon as you have installed the Anaconda Distribution, you can launch the Anaconda Navigator or a terminal (using the command "*conda*") to manage your python environments and tools. Now you are ready to explore Python!

Exercise 2.1: *General Python Capabilities*

Open a Jupyter Notebook by using the Anaconda navigator or by simply writing "jupyter notebook" into your terminal. Create a new notebook file and try to solve the following tasks to explore some general python capabilities:

1. A very useful feature of Jupyter notebooks are the "*magic commands*". You can view a list of these commands by typing *"%lsmagic*" into a cell.
    a. Write the *Pythagorian Theorem* using latex code.
    b. What is the difference between *"%magic*" and *"%%magic*"?
    c. There are a lot of very useful magic commands. You can easily access lists of the most useful ones in the internet. These magic commands also often come with possible options. You can try out, e.g.:
        - *%who*, which gets output as a list as %who_ls
        - *%time* in various versions to calculate the time your code needs to run
    d. Additionally, to some magic commands you should get familiar with some shortcuts in Jupyter notebooks. You can easily get an overview by clicking the "*help*" *button*.
2. Import the frameworks *pandas*, *numpy*, and *matplotlib* into your Jupiter notebook file
    a. Create a python tuple with the following input:
    {spicy soup, spicy chicken, spicy salad, simple soup, simple chicken, raw chicken}
    → Why is it usually not practical to use tuples?
    b. Transform the tuple into a pandas list. Then, transform it into a pandas dataframe with the two columns "*adjective*" and "*noun*".
    c. Replace "spicy" with "peppery" for all dishes.
3. Classes in Python
    a. Create a class "*child*" having the attributes *name*, *age*, *height*, and *weight*
    b. Now, create a class "*adult*" that inherits the attributes of "*child*" and has the additional attribute *occupation*.
4. Pandas dataframes
    a. Generate a pandas dataframe with size *10,000x3* filled with random integers between *0-1,000* and the column labels "*A*", "*B*", and "*C*".
    b. Measure how long the creation of that dataframe takes and how much memory it uses
    c. Provide some basic statistical values of the dataframe's data
    d. Sort the first column and visualize the data of that column using some visualization of your choice.

Exercise 2.2: *Titanic*

This time, your boy-/girlfriend forced you to watch the blockbuster movie "*Titanic*" again with him/her. While you were watching, you wondered how you would have realized your investigations when you would have used Python instead of RapidMiner. After finishing the movie, your significant other is sleeping on the couch. Thus, it's the perfect time to revisit your past investigations. You go to your computer, locate your gathered titanic data file, and open a new Jupyter notebook.[1]

a. First, you import the csv file into a dataframe.
b. You select the columns with rather high explanatory power for "survived" by using your personal logic. Then, you evaluate your logic by having a look at a correlation table. What does it tell you?
c. Now, you are trying to comprehend the quality of the data again. How many missing attribute values are there? You decide to fill missing values with the respective column average values.
d. You create a pivot chart indicating the means of "Age" and "Sex" depending on survival.
e. Lastly, you perform binning for "Age" again by defining the intervals manually (use the bin sizes of Exercise 1.3). Add the binned values as a new column to your dataset. Compute the correlation with "survived" and compare the binned age with the numeric age.

Exercise 2.3: *Wine Quality*

You have just realized that your wine-loving mother in law has birthday tomorrow. Obviously, in order to bypass the anger of your fiancé/e, you will buy some wine. However, almost simultaneously, you stumble across a dataset describing wine of different quality. Of course, you decide to use this data to choose the wine and start to wonder: *"Wouldn't an AI that always chooses the perfect wine constitute the better present for your mother in law?"*[2]

I. Data Preprocessing

1. You import the data and start exploring to gather a better understanding of it by checking, e.g., data types, number of columns and rows, number of missing values, etc.
2. You decide to bin the "quality" column into two categories "mediocre" and "excellent". However, before you do, you have to get an overview of the quality values and their distribution by visualizing them in a fitting plot. How many data points did you place in each category?
3. Afterwards, you set the "quality" column as the label.
4. Then, you decide to scale the data using the scikit-learn function "*StandardScaler*". What does it do?
5. To finalize your preprocessing, you split the data into a training and test set. You decide to set the size of the test set to 33% of the original data.

II. Modeling

6. You decide to train several models using the following algorithms:
    - Random Forest
    - Linear Regression
    - Support Vector Machine
    - Decision Tree
    - Neural Network

III. Model Evaluation

7. To evaluate the models, you make use of the classification report and the respective confusion matrices. How do you interpret the results?

---

[1] You can find the titanic dataset in our git repository.

[2] You can find the wine quality dataset in our git repository.