IUM

(6 credits)

Javier Gonzalez Blanco
Sergio Berges Aguaviva

INDEX

1. INTRODUCTION

The project consists of a creation of a Jupyter Notebook in which the library pymongo is required to access a MongoDB database and get a specific dataset and to use it to draw graphics that explain concepts or ideas so that the user can get conclusions from those.

The data provided is about football, including information of players, games, clubs, competitions, appearances of players in games, lineups and events of the games, etc. According to this information, possible topics for the graphics could be: goals' evolution of a certain player during years, most valuable players in a certain league, stats about a club in a certain year, and many more options.

What we have done is to update in a MongoDB database the collection of CSV files that were provided by the professor, and after that we connect the Jupyter Notebook to that database in order to send queries so that the database returns subsets of data that are going to be analyzed in the Jupyter Notebook with Python functions and the use of related libraries (matplotlib, seaborn, etc.) that will transform the subsets to the appropriate format required to be visualized in different possible graphics.

2. JUPYTER NOTEBOOK

A. SOLUTION

The solution that we develop is composed of a variety of graphics all of them visualized in the same Jupyter Notebook, one after another, in which with the change of one or more parameters a graphic is drawn. We've submitted 10 graphics in which we have tried to give a consistent information of the context that we are analyzing, including as many different graphics as we considered in order to show good data information. But before all the graphics, the first cell of the Jupyter Notebook is used to define Python functions that establish a connection between the notebook and the MongoDB database.

For each graphic and the information that we want to visualize on it, it is necessary to put a value on one or more variables in order to query the database (with a Python function) and find the subset of the data required to the context of the graphic. Once the subset is ready, other Python functions will change the structure of the dataset in order to have a correct visualization required to the graphic used in each case.

Last, but not least, several markdown cells have been included in the Jupyter Notebook between the cell codes commenting what those cell codes do.

B. ISSUES

The main issue for the project was that we didn't have very clear what we had to do, especially at the beginning. This made us lose some time in the process of really starting to work on what was important. When we started to do the Jupyter Notebook we didn't organize the project well or what we had to do, so some more time was lost.

Another issue was with the managing of the repository of Git. We started with two branches besides the master one, one for Javier and the other one for Sergio in which each one of us could work

in the Jupyter Notebook and then put both works in the master branch. The issue was that we could work in the Notebook at the same time because it generated problems when we merged the branches. At the end, we decided to work in the master branch, working each member on its own part but at different moments.

Another main issue that made our work a bit more difficult was the comprehension of the Python syntax and the use of its related libraries. This is because in our home university we didn't really use Python as a programming language, so this is the first time that we are really working on a Python project so we didn't have an extensive knowledge of Python tools and how to use them.

It is precise to emphasize that we required in some aspects the use of AI to help us make our solution, in a way to help us to understand some Python concepts and provide us solutions in cases in which we didn't know how to continue.

C. REQUIREMENTS

Our solution satisfies all the requirements:

- There is a GitHub repository which contains the evolution of the project and the contributions of each team member in terms of commits and lines of codes written and deleted. The structure of our GitHub repository is the one that is described in the assignment document:
    - There is a folder with a Jupyter Notebook which sets a connection and accesses a database, selects a dataset and draws some graphics about the date set.
    - There is a folder with this report that explains how our journey was on the development of the project.
    - It has been included a folder with screenshots of some query examples.
    - It has also been completed the Self Assessment Form required by the teacher.
- Also the Self Assessment Form has been sent via Moodle.

D. LIMITATIONS

The first main limitation of our solution is that we couldn't charge into the Jupyter Notebook big subsets of the database. We don't have that much memory to upload big subsets of data as we were trying to do at the beginning of the project. Due to this, we need to provide a not that big dimension of the dataset, so we cannot analyze information in open contexts.

Another important limitation is related with the parameters that we send to the developed functions in the Jupyter Notebook in order to select subsets of data. We have approached the introduction of these parameters as if those were inserted in a form. This is, we include some variables with a determined value that will be the parameters of the Python functions that query the database, so that those variables will belong to a property of a document of a certain collection in the database.

3. CONCLUSIONS

In conclusion, our solution analyzes the data provided in the database and creates some graphics that makes the data more visible and understandable. We have acquired the Python knowledge necessary to develop the project of the data analytics part of the subject. We have also learnt that besides the powerful tool that is Jupyter Notebook it has its limits, which we have discovered in the part related to extracting data from the database.

Furthermore, we have learnt that Python is a very simple language to understand and to program in it due to their dynamic typing.

Also, the use of cells in Jupyter Notebook makes it easier and faster to work in this type of projects where you have to try some executions many times of a certain cell, without the need of executing also the previous cells that may not have any relation with those next cells.

Last, this has been the first time that we work in a project with a non relational database as MongoDB. We knew that MongoDB was a non relational database and the characteristics of these types of databases, but we never really worked with non relational databases. After this project, we know a bit more how MongoDB works and we open the possibility to use it in future projects.

4. DIVISION OF WORK

We divided the project as each one of us would make a data set and its equivalent graphic in base of analysis of the information in the database that could be interesting.

The creation of the project was divided in equal parts. However, we both work at the same time in the same space so we help each other in their parts and share the problems and solutions that we encounter, as we live together due to the fact that we are Erasmus students.

Also, as we used to work together in our home university in Spain, we know each other and we know what our teammate can do and the compromise he will put in this project. Despite the difficulties encountered in this work, we are both proud of the effort made by both members of the team.

5. EXTRA INFORMATION

In it there is a directory name solution with our solution of the project, a Jupyter Notebook.
For the right execution of the Jupyter Notebook there are some packages (Python related libraries) which are necessary to install:

- jupyter(1.0.0)
- matplotlib(3.8.5)
- pandas(2.2.0rc0)
- seaborn(0.13.1)
- pymongo(4.6.1)

The version of the packages is the one that we have used, other versions could have a successful execution or not. Due to this, the last version of those packages is the recommended one.

It is also important to emphasize that, as commented on previous sections, in the first cell of the Jupyter Notebook is where the connection to the database is made. The name that is included in our solution is the one that we had in our local database. For the right execution of the notebook you may need to change that name and put the name of your local MongoDB database.

6. BIBLIOGRAPHY

- Moodle Slides
- W3Schools Python: in order to understanding and discovering new Python tools
    - https://www.w3schools.com/python/default.asp
- Matplotlib page: to learn how to create different graphics showing all the necessary supporting information.
    - https://matplotlib.org/stable/plot_types/index.html