

FormulaTour

Gianluca Capozzi

"La Sapienza" University of Rome

capozzi.1693255@studenti.uniroma1.it

Marco Costa

"La Sapienza" University of Rome

costa.1691388@studenti.uniroma1.it

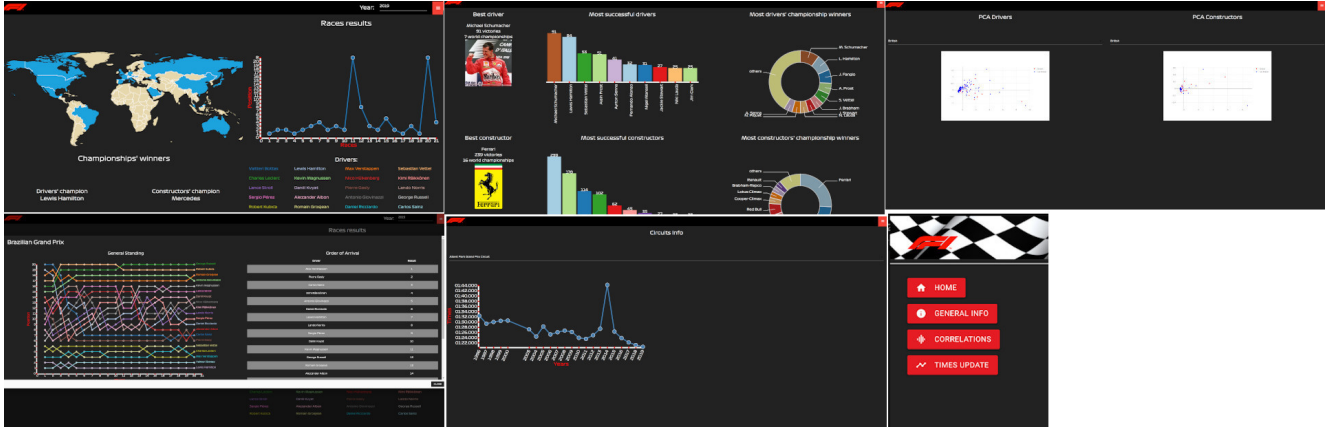


Figure 1: Framework Views

Abstract

1. Introduction

Formula 1 has a very long history, as it has reached the 70th season. During all these years many drivers and constructors took part to the races, which took place in different circuits located in different countries. This means that it is very difficult to navigate this history, since it is characterized by many components, each related to the other. The goal of this project is to provide a way to navigate the history of the Formula 1 by highlighting each component and their relations. Our framework is divided into four main views each of which contains multiple sub-views. An initial view summarizes the races of the selected season by showing the countries and the exact location of the circuit in which they took place. A general view summarizes the most important results achieved by the drivers and the constructors during the Formula 1 history. A correlation view that shows if it exists a correlation between nationalities and number of victories/podiums. The last view is a times update views that summarizes the evolution of the lap times on a selected circuit during the years. So, our project is

composed by:

- A visual analytics framework that supports the examination and exploration of Formula 1 history;
- A visual representation of the features selected by the users.

The project is a web application and it is developed using the d3.js [1] library and the python3 language for performing PCA.

2. Related works

There are other frameworks that show the evolution of Formula 1 through its different elements, so races, drivers and constructors. An example is The history of Formula 1 framework, based on the same dataset used for this project. The main difference is that in that project there is not a division of the races with respect to the years and also the user is not able to locate geographically the position of the circuits.

3. Dataset

The dataset used for this project is obtained from the Ergast Developer API [2] in .csv format. It is composed by 13 tables: circuits, constructorResults, constructorStandings,

constructors, driverStandings, drivers, lapTimes, pitStops, qualifying, races, results, seasons and status. The tables considered for this project are:

- **races:** a table that contains informations about each race, as the year and the date in which it was played, the circuitId (which identifies the circuit in which it took place), the name of the Grand Prix (such as Italian Grand Prix) and so on;
- **circuits:** a table that contains informations about each circuit, such as the name of the circuit, its coordinates, the country and so on;
- **drivers:** a table that contains informations about each driver, such as his name and surname, his date of birth, his nationality and so on;
- **results:** a table that, for each race and driver who took part in that race, shows the result of that driver in that race and other infos;
- **driverStandings:** a table that, for each race and driver who took part in that race, shows the driver's point into the drivers' standing after the conclusion of that race;
- **qualifying:** a table that, for each race and driver who took part in that race, shows the results that the given driver achieves during the qualifying for that race;
- **constructors;**
- **constructorStandings.**

The .csv files are manipulated using both d3.js and python. Some of the queries performed over the dataset needs the interaction of the user, while others are performed without parameters chosen by the users.

4. Visualization

The visualization is composed by 4 main views, accessible from a simple side menu. Each of these views provides some processed information about the dataset.

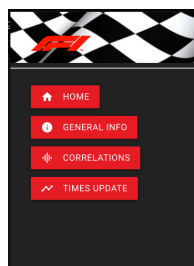


Figure 2: Menu

4.1. Home view

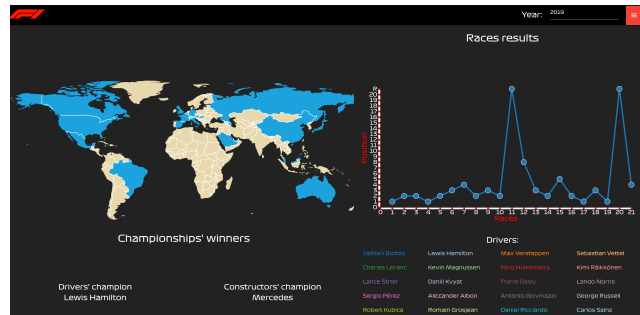


Figure 3: Home view

When click on a circuit

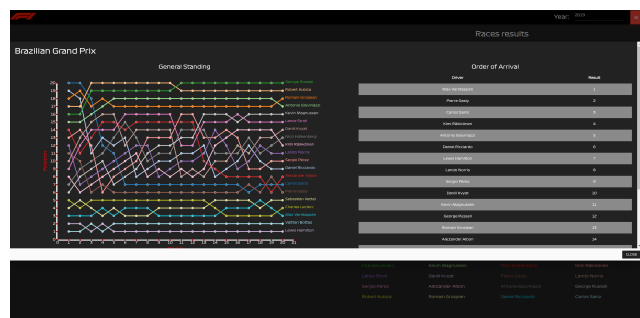


Figure 4: Home view

4.2. General info view

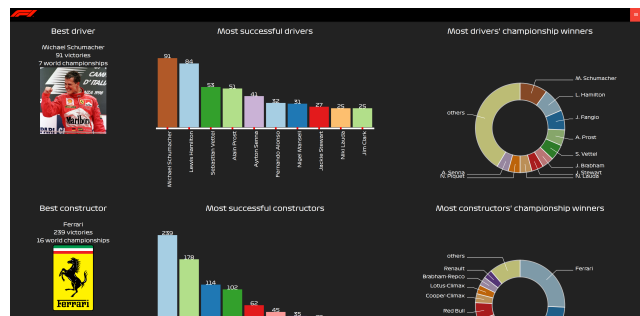


Figure 5: General info view

4.3. Correlations view

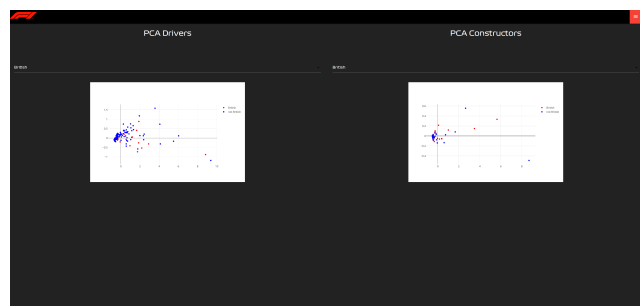


Figure 6: Correlations view

4.4. Times update view

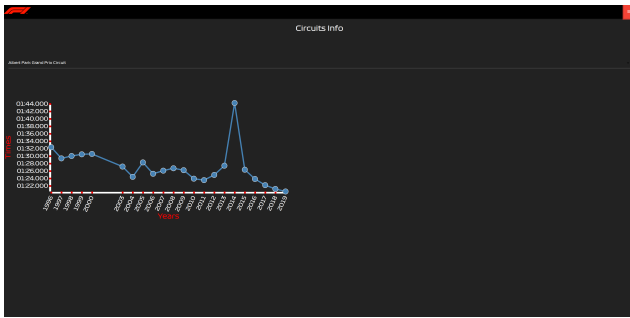


Figure 7: Times update view

5. Analytics

6. Future Works

7. Conclusion

Conclusion text

References

- [1] Data-Driven Documents: <https://d3js.org/>
- [2] Ergast Developer API: <http://ergast.com/mrd/>
- [3] Plotly JavaScript Open Source Graphing Library: <https://plot.ly/javascript/>