Data Visualization

Professor: Laurent Vuillon

Milestone 3
Lebron Jenkins Process Book

Paul Juillard (288519)
Mathieu Desponds (283229)
Antoine Magron (288416)

June 3, 2022

1 Why NBA is important

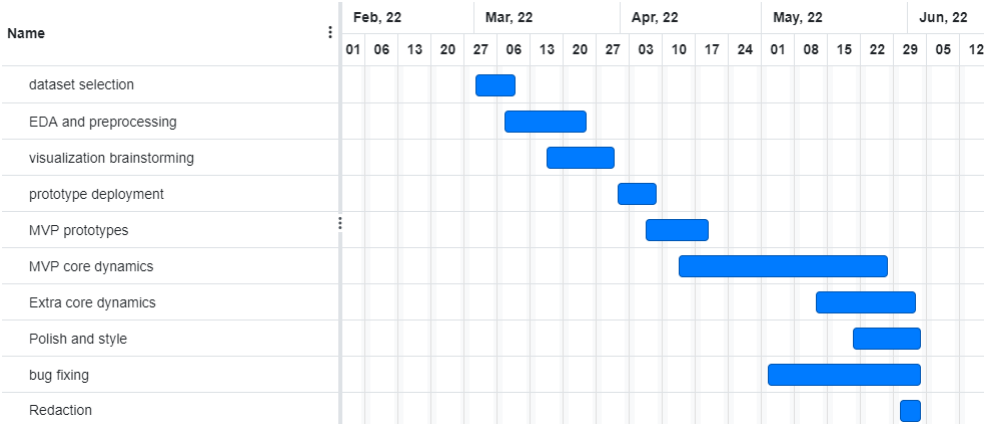
In a world of hardship and division, sport achieves an incredible goal: it brings everyone together! In the United States, the basketball league, the NBA, is very popular. Teams play all year long in intense seasons, and fans get to witness careers and even legends being made. Some names such as Lebron James or Kobe Bryant are known even to non-believers as international sports icons!

The competition for the top is fierce, and very well documented! For our project, we use data distributed by kaggle.com and made available by the NBA itself on their official website. Details about every match, team and season are available, going back all the way to 2003! Faced with the richness of this data, how can we resist looking for unforeseen success factors in our own sciency way?

With some prior data manipulation, we expect to allow users to discover peculiar insights on their favorite sport. Using interactive data selection in our two first visualizations, users can focus their understanding and investigate on their own. In our third interactive tool, the users have complete freedom to construct their own team and visualize its effectiveness!

2 Development process

We used an iterative design development principle, going through working prototype phases before moving on. This allowed us to build a solid basis before adding extras, and to ensure a functional MVP rather than multiple kind-of-working visualizations. Here is a chart of the process. Further information can be found in the peer breakdown section.



3 Achievements

3.1 Map visualization

The goal of the visualization is to show the travels of the different teams during the season and see the number of kilometers has impact on their results.

For this, we let the user choose up to three teams and a season that will be put on the map and will move following the journeys they did during the season. If you want to go forward or backward in the season you can move the slider on the bottom and the travels and the statistics will be changed accordingly.

On the side, we have a table with the statistics of these teams that show the number of kilometers and the percentage of wins until now.

However, with our visualization it is hard to see if there is a real link between the wins and the number of kilometers. The financial aspect, how much the teams invest and therefore the strength of their team has certainly a much more important impact on the results. What we could have done is first compare teams with the same budget. And secondly we could have looked at different moments in the season where a chosen team had long travel or small ones and compare the number of points they made during this period. However, for this we would have needed the drawings on the map by date more than by matches which was an important choice as we will explain below.

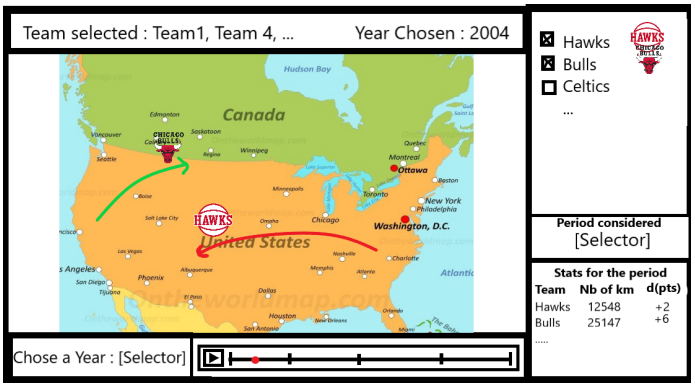


Figure 1: The sketch of the first visualization made for the second Milestone

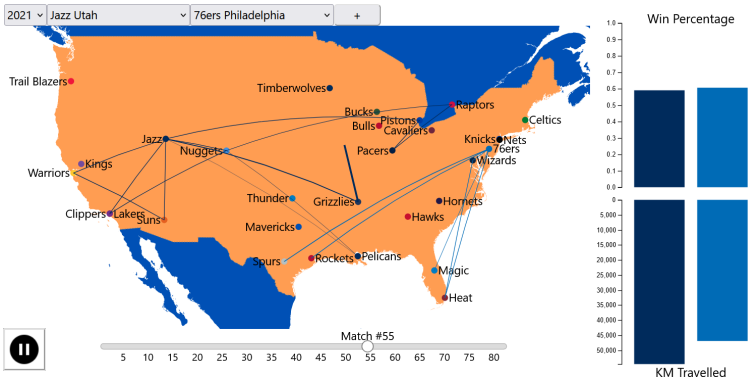


Figure 2: The final visualization

3.1.0.1 D3-geo and and the Map

The main challenge of this visualization was to understand well how we can draw a map and understand how to use the projections on the paths and on the points. First we had to find the data for a North American map which we did not find until we found out with this tutorial that we could take a world map and take only the countries of interest. After that we were able to project a path on the map.

3.1.0.2 Execution order and event loop

We also struggled with the order of the execution of the rendering. The map was sometimes drawn after the cities. To overcome this issue, we tried to use the wait/async function of Javascript and the Promises but we were not able to get the desired result. Therefore we use the callback method which we think is not the fastest one but the rendering is made in the right order.

3.1.0.3 Team selector

Another challenge we faced was to have global variables so that we could share information between the files. In fact, we needed to get the values of the selected teams and season that were in the *viz1-selector.js*

in the *viz1-slider.js* and in *viz1-map.js*. The way we managed it was to create a class for all the selectors - the teams selected and the chosen season - and export an instance of this class. Therefore, we can import this variable in other files and use it as a global variable.

3.1.0.4 Readability and max teams

We had to find a compromise on the number of teams we wanted to display at the same time. On one hand, putting a lot of teams on the map and seeing them move showed how busy and dynamic a season is for players. But on the other hand, the map was less readable and as it became harder to distinguish between teams. In fact, we found it interesting to keep the track of the paths of the last period, but this implied to reduce the number of teams displayed. We finally decided to keep only three teams on the map to value the readability over the liveliness of the plot, and emphasize on the comparison between a few teams rather than intuition on the big picture.

3.1.0.5 Steps per match rather then by date

In the beginning, we wanted to draw the travels by date. However, what we wanted to show was the number of kilometers and the impact on the results. Therefore the dynamism given by the fact that all the teams are moving at the same time was more important than the real time travels.

3.2 Rankings

Rankings are easy to understand, and the agreed upon abstraction to know which team is the best. But rankings can be more complicated than expected. If too volatile, they may not mean much. If too predictable, then the sport might not even be fun to watch.

With a bit of preprocessing, we computed the evolution of rankings during seasons, and show each team's evolution in the ranking. The user can select a season, which can yield a mess for some volatile seasons. To ease readability, teams are color-coded as in the map visualization and are highlighted on hover.

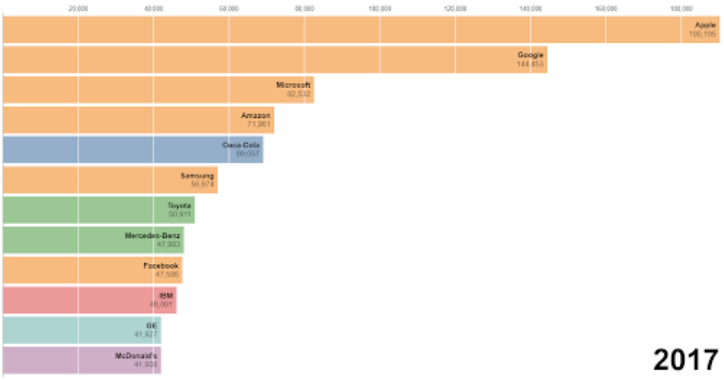


Figure 3: Sketch of the second visualization made for the second Milestone

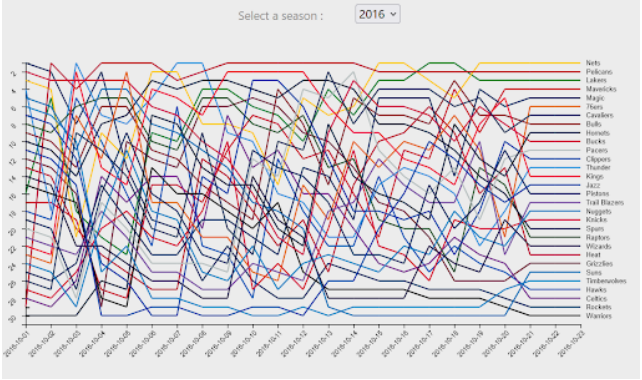


Figure 4: Final ranking visualization

3.2.1 Sketches and achievements

3.2.1.1 Readability

As briefly stated above, readability was a challenge for this visualization. Showing only part of the data was not an option, as we want to hint at the volatility of the rankings. Colors were the go-to answer, but they were not sufficient. They eased focus, but did not remove any ink from the plot. In order to do so, we chose to let the user reduce the ink ratio on interaction, and let them select the data points they want to follow more precisely.

3.2.1.2 Hovering and axis

Moreover, when we hover a team, we wanted to highlight its name. To this end, we needed to give an id to the different teams. However, this ended up being more tricky then expected. We were not able to find how to assign ids to elements of the axis because we did not find how to use the data used in the call function of the axis.

3.2.1.3 Season irregularity and sparseness

NBA seasons have a peculiar repartition. The regular season spans almost the whole year with relatively big gaps in the middle. The range and density of data to show for a regular season visualization was hard to manage, even with the above readability adaptations. We decided to reduce our selection to playoffs, which spans a much shorter period and already gives a good intuition about what we hope for the user to investigate.

3.3 Fantasy Basketball

All the prior visualizations were regarding teams, but we also have astonishing details on player characteristics. To highlight this diversity, we wanted to design a tool incorporating a lot of freedom for the user to play with. The goal is to motivate the user to search and query the data playfully.

With inspiration from fantasy sport platforms, we let the user discover players and arrange them in teams: this is their query. We selected a subset of statistics to display as the overview of the user's query. This arbitrary selection could be improved through user testing, and the scripts were made in a modular way to keep arrange for these possible future modification.

Figure 5: The sketch of the fantasy visualization made for the second Milestone

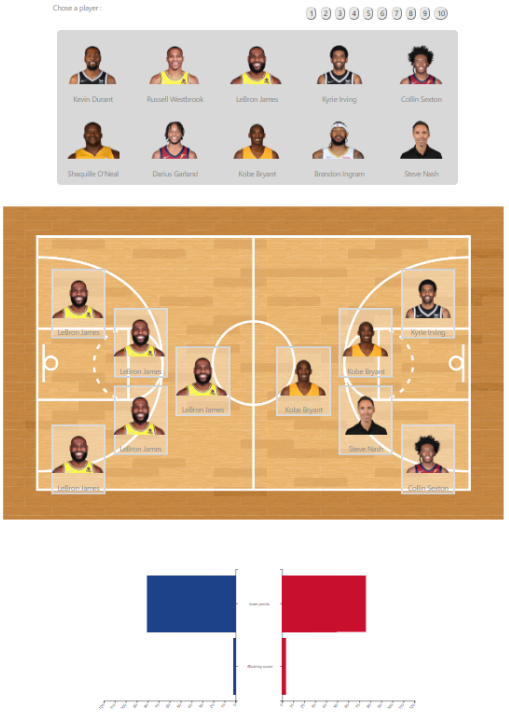


Figure 6: The final visualization

3.3.0.1 Player selector and presentation

We needed a way to present players to the user, and select them. To answer our need for interactivity, we chose drag-and-drop as a selection mechanism. Dragging the players themselves forces more involvement from the user in its interaction with the visualization, and is less serious and abstract than list selectors or checkboxes. To go further in the direction of this familiarity and away from seriousness, we chose to

show player faces, available from the NBA website, in addition to their names.

3.3.0.2 Player pool limit

To not overwhelm users, we decided to limit the available players for selection to a 100. As a measure of their popularity, we used their efficiency score, which we hope to be correlated. The 100 bound could also be fine-tuned with respect to user feedback if conducted.

3.3.0.3 Choice of statistics

Compiling an overview of a team from player statistics was now as evident as expected. With our limited domain knowledge of basketball, we decided to refrain from unrealistic constructions and kept to safe compilations:

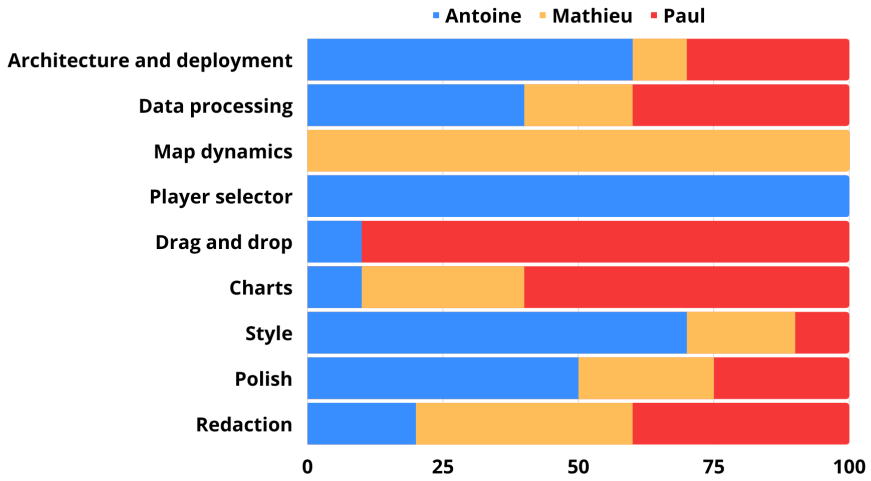
a) Sum of average player points per game

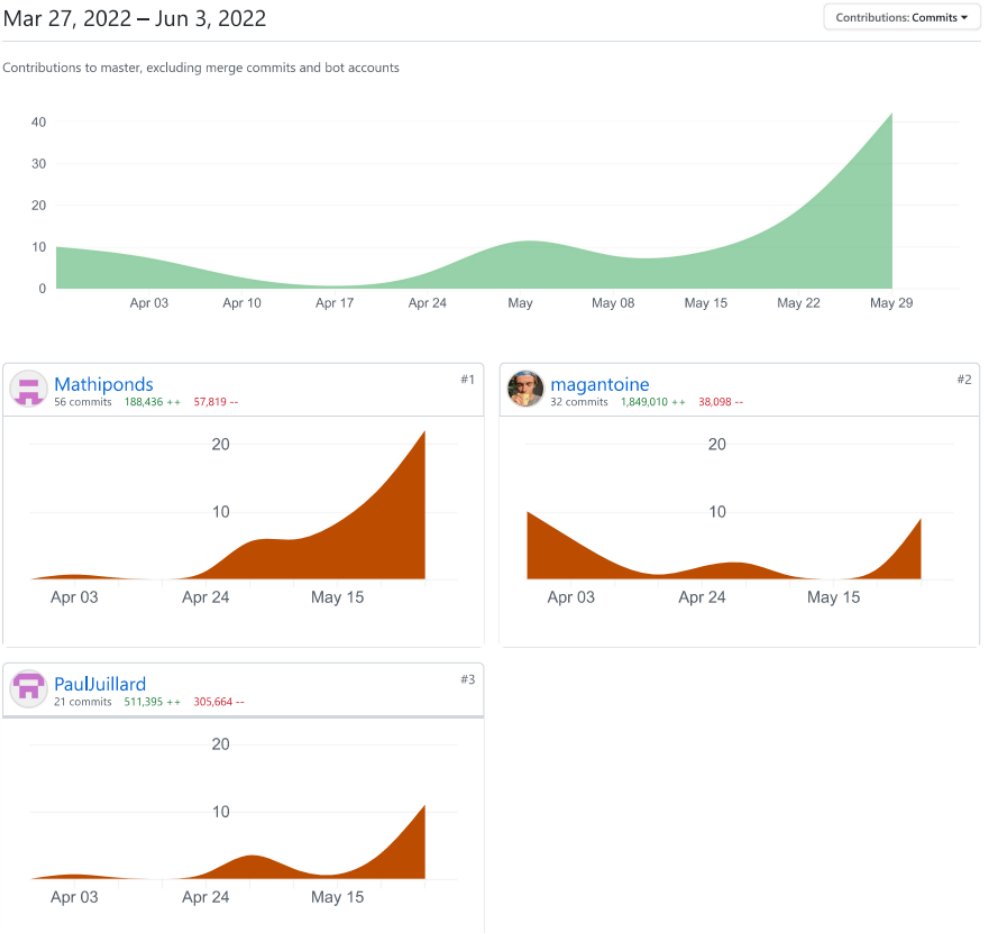
b) Mean player efficiency score.
The player efficiency score is a compilation of all the player's characteristics in a representative 0 to 1 score. This compilation is done by the NBA officials. We scaled this score to fit the axis to not overcomplicate the graph, considering the scale of this score is irrelevant as it has no unit, only comparisons stand.

As said earlier, this selection of statistics would in practice not be final, and be enhanced through user-feedback loops and/or creative extensions.

4 Peer Breakdown

The following diagram and graphs breakdown our team member's respective contributions. The first is an agreed upon participation graph in core development steps. The second is Github's insights report a few hours before the submission deadline.





5 Conclusion

We wished to construct a highly interactive and dynamic visualization to let our user investigate, at the pace of their curiosity, the ins and outs of NBA rankings and success factors through a few selected perspectives, like travelling and ranking. In order to do so we had to process our data, done mostly in python with pandas, and we extensively used d3 to build our visualizations. We also had to implement some JavaScript behaviors like drag-and-drop or the slider tick update event loop.

At this point in our website's life, a user-feedback-loop would be, if not needed, a great potential for improvement and adaptation.

We are overall happy with our development process and our team's involvement in the project (like not giving up in hard times when the map would not behave).