FIFA 18 DATA VISUALIZATION

Raúl Moldes Castillo, Álvaro Alonso Lancho, Sergio Marín Sánchez

1. Problem characterization in the application domain.

This step involves talking to the target client in order to identify the problem to be solved in the application domain.
In our project, the target clients are the FIFA video game developers. The main goal of the project is to provide tools to visually analyze the data provided by the 2018 version of the game in order to ensure this data is consistent and reliable. Furthermore we would like to explore relationships between different variables involved in the dataset (like the nationality, position and wage of the players), in order to detect correlations and outliers in the different stats established by the developers.

2. Data and task abstractions

DATA ABSTRACTIONS

The dataset being used in this project is obtained from the platform Kaggle(FIFA 18 Complete Player Dataset (kaggle.com)). It is published under an Open Source License, and has a tabular structure with a single key. The table contains information about every player included in the FIFA 18 video game. Each player is identified by a numeric (integer) unique ID. The main variables which we are going to consider for our tasks are the following:

- Player Attribute variables(Aggression, Crossing, Balance, Speed, Overall, Potential…)
    - Type of variable: Ordered quantitative sequential variable.
    - Variable Range: from 0 to 100.
    - Semantics: The performance of the player in that particular statistic.
- Player Positional variables(ST, GK , CB, CM…)
    - Type of variable: Ordered quantitative sequential variable.
    - Variable Range: from 0 to 100.
    - Semantics: The performance of the player in that particular position.
- Preferred Positions:
    - Type of variable: Categorical.
    - Semantics: The preferred positions of that player.
- Wage:
    - Type of variable: Ordered quantitative sequential variable.
    - Semantics: The wage range the player belongs to.
- Value
    - Type of variable: Ordered quantitative sequential variable.
    - Semantics: The value range the player belongs to.
- Club:

    - Type of variable: Categorical.
    - Semantics: The club that the player belonged to in 2018.
- Nationality:
    - Type of variable: Categorical.
    - Semantics: The nationality of the player.
- Name:
    - Type of variable: Categorical.
    - Semantics: the name of the player.
- Age:
    - Type of variable: Ordered quantitative sequential variable.
    - Semantics: the age of the player in 2018.

TASK ABSTRACTIONS

TASK 1: Identifying correlations in the player's statistics between different positions.

- Why is this visualization being used: players which belong to the same positions should present some kind of correlation between the attribute variables. For instance, defenders may all have a low level of finishing but a high level of interception; and midfielders may have a high level of speed but a low level of goalkeeper positioning. Analyzing if this is the case in the data provided by the FIFA can help us detect if there are anomalies in the data from the 2018 version of the videogame. Therefore, the objective with this visualization is to present the information to third parties, that is, the FIFA 18 developers.
- What kind of search is performed on the dataset: Exploring the data to understand the distribution of the statistics across players belonging to different positions.
- What kind of query is made based on the results of the previous question: the target is to identify trends, outliers and correlations between player statistics across different positions. To achieve that, a multidimensional query is performed. We filter players by position and retrieve only the statistics that we want to analyze.

TASK 2: Cluster analysis between statistics and different characteristics of the players.

- Why is the visualization being used: the objective of this visualization is to discover clusters of players based on similarity on the value of the statistics. We want to identify different groups of players based on the value of its attributes, and to explore how these groups relate to other attributes; mainly the wage, market value, player position, the country and the division where the player plays.
- What kind of search is performed on the dataset: This is also a kind of exploration search as we are visualizing multiple dimensions (three different statistics and player wages) to understand the relationships and patterns among them. This type of search is exploratory because we are not looking for a specific item or value, but rather trying to understand the structure and distribution of the data.
- What kind of query is made based on the result of the previous question: this is again a multidimensional query, as we are retrieving players based on multiple dimensions.

The target is again to explore similarities between players according to their values on the different statistics.

TASK 3: Bubble Chart Analysis: Exploration of Player Performance and Market metrics.

- Why is the visualization being used: the visualization is being used to explore and analyze relationships between football player characteristics, such as overall rating, market value, wage, and nationality. It provides an interactive platform for users to understand how these factors interact, making it easy to understand.
- What kind of search is performed on the dataset: As mentioned in the previous point, we need to obtain the overall rating (Overall), nationality (Nationality), value of the player in the market (Value) and the wage of the football player (Wage). It is a type of locate search, as for each point we know the target, but not the location.
- What kind of query is made based on the result of the previous question: It is a type of filter query which involves selecting among the players of a specific nationality or nationalities to locate a certain player based on its overall and market value.

TASK 4: Histogram Chart: Analyzing the relationship between player potential, age and position.

- Why is this visualization being used: player potential is an important attribute that reflects how much a player can improve his skills in the future. It is influenced by factors such as age and position. For example, younger players may have higher potential than older ones, and some positions may require more potential than others. By using a histogram plot, we can compare the frequency distribution of player potential across different age groups and positions. This can help us understand how potential varies with age and position, and identify the best players for each position based on their potential.
- What kind of search is performed on the dataset: exploring the data to understand the relationship between player potential, age and position.
- What kind of query is made based on the results of the previous question: the target is to analyze the distribution of the most promising players between different ages or positions.

3. Interaction and visual encoding.

IDIOM 1: Parallel coordinates chart.

In this Idiom, the marks are the lines that encode the relationships between attributes. The different attribute names are encoded with spatial regions on the x axis. Furthermore, the

magnitude of each quantitative attribute(statistic) is encoded using spatial position on the y axis.
A multiple filter selector is implemented for the user to be able to select among the possible stats that can be displayed. The user can also select among the possible player preferred positions to analyze the relationships between the variables for each position.
In Figure 1. The parallel coordinates plot for position *RW and GK,* and attributes: *Acceleration, Aggression, Agility, Balance* and *Ball Control* are shown.

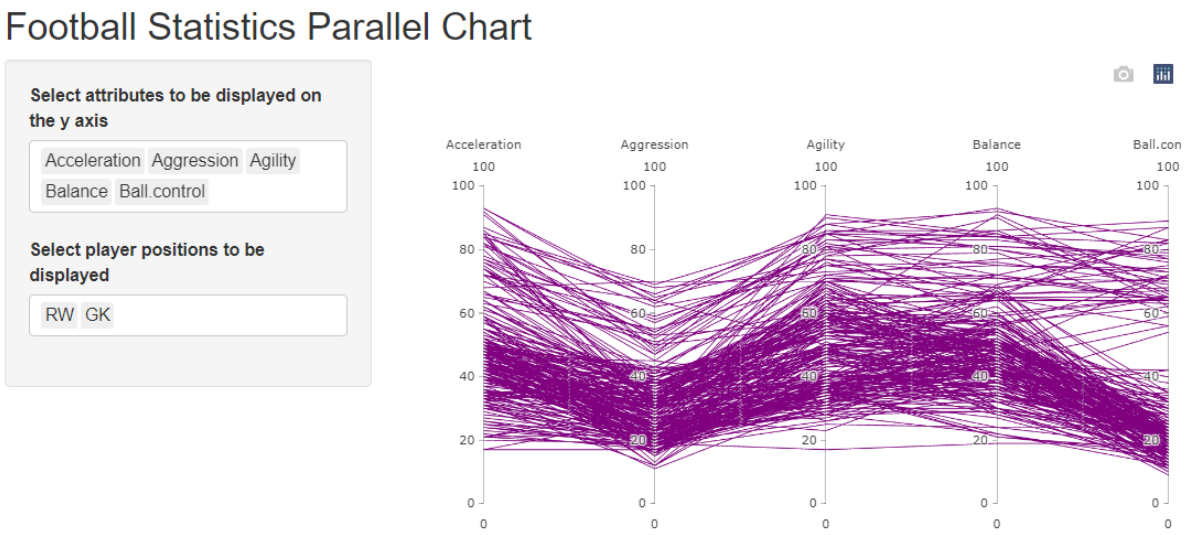

Figure 1. Parallel coordinates plot for position RW and GK.

IDIOM 2: Cluster analysis based on player attributes:

In this idiom, the marks are the points which represent each player. Spatial position across the x, y and z axis is used to encode the value of each player across the distribution of statistics.
The color channel is used to encode the attribute used to perform the cluster analysis. A filter is added in order for the user to be able to select on which attribute to perform the cluster analysis.
Depending on the user selection, that attribute will be encoded with a different color scale. For ordered attributes(like player wage and player value), we have chosen a sequential colormap using hue and luminance to encode the attribute. For categorical attributes, like division, country or player position, we have chosen a categorical colormap, using hue as the main channel to encode the attribute.
In Figure 1, the cluster analysis based on player position is shown. As can be seen, two main groups are identified: one for the goalkeepers(blue) and one for the rest of the players(green, orange and pink).

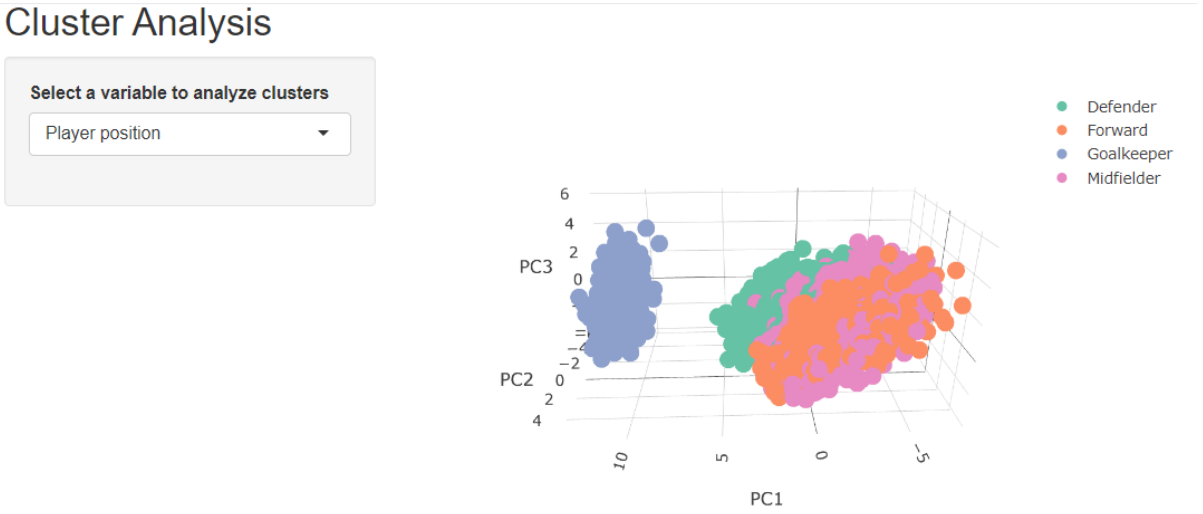

Figure 2. Cluster analysis based on player position

IDIOM 3: Bubble chart analysis based on overall rating, nationality and market:

In this idiom, the Bubble Chart Analysis represents every football player's data. Each player is represented as a bubble on the chart, with spatial positions along the x and y axes it encodes the overall rating of the player and the value in the market respectively. The size of the bubbles represents player wages, providing an additional dimension to the analysis.

The color channel is utilized to encode the nationality of players, offering insights into the distribution of players from different countries. Users can interact with the app by selecting specific nationalities, resulting in a filtered view that allows for a focused analysis of player groups based on their country of origin.

Figure 1 of our analysis displays the Bubble Chart based on 2 nationalities, Portuguese and Argentinian. Users can observe patterns and relationships between player overall ratings, market values, wages, and nationalities. The visual representation facilitates the identification of groups and trends within the dataset.

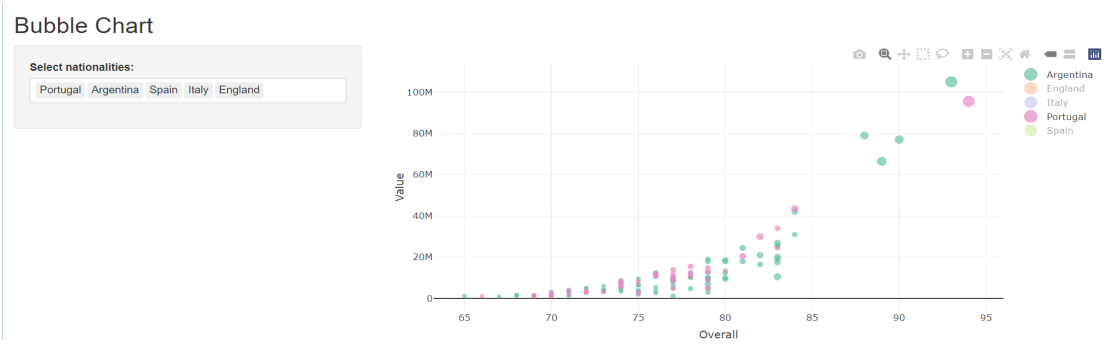

Figure 3. Bubble chart analysis of football players from Portugal and Argentina.

IDIOM 4: Histogram chart:

In this idiom, the histogram represents in each bar the amount of players who have a potential score within the same range.

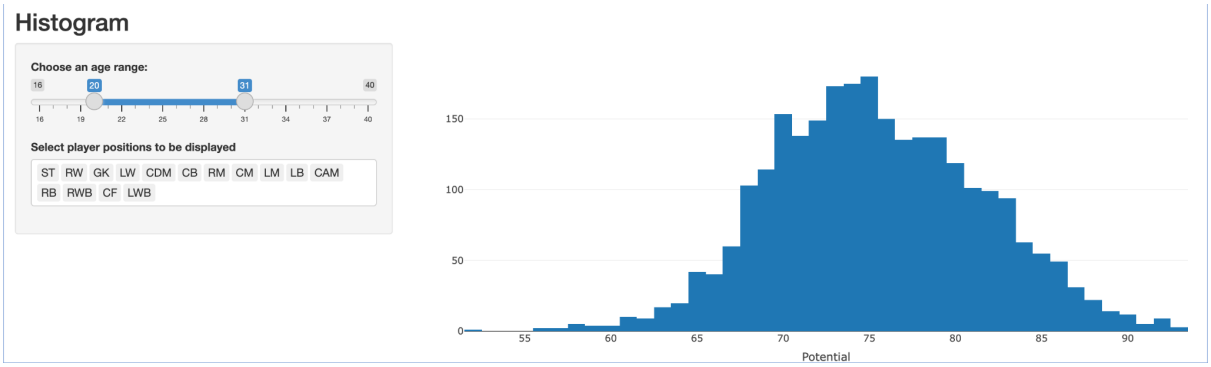

Figure 4.Histogram of players with ages between 20 and 31 years old.

4. Algorithmic implementation.

4.1. Software and technologies used.

In this project, R and Rstudio have been used to develop the charts. Libraries ‘shiny’ and ‘plotly’ have been key for us to be able to develop the final application.

4.2. Data cleaning and preprocessing.

At the beginning, the dataset was too large for R to process it properly. Therefore we filtered it to keep only players from Spanish, English and Italian first and second division, and store the result on a separate file. For this part we have used pandas in python and we have used an external webpage to check the teams that were certainly in these leagues. After that, we had to match some teams as some names had little discrepancies.

4.3. Feature engineering.

In order to be able to make the charts, some variables needed to be modified, and new variables had to be created:

- Variable *PreferredPositions* contained, for each player, a list of preferable positions in sorted order. We only wanted to analyze the most preferred position for each player, so we kept the first element of each list. Moreover, we added a new variable called

Position, which contains broader information about positions. This variable is of type categorical and has levels: *Goalkeeper*, *Defender*, *Forward* and *Midfielder*.

- Variable *Club* contained the club that each player belonged to. From this column two new categorical variables were created:
    - Variable *Country* contains the country where that player plays. It can be *Spain, England* or *Italy.*
    - Variable *Division* contains the division in which the club of that player played in 2018. It has levels *First* and *Second.*
- Finally, for the cluster analysis we have performed PCA over the subset of the data that contains the player stats. This is done using the function *prcomp.*

4.4. IDIOM 1 Implementation.

The parallel coordinates plot consists of three functions:

- UI input function: this function is implemented on the ui shiny object, and is used to display the chart to the user. Moreover,two input filters are implemented in the ui:
    - A filter to select the attributes that are displayed in the x axis of the parallel coordinates plot. This helps to ensure scalability.
    - Another filter to filter players by position and reduce the number of results.
- Reactive expressions: this is implemented in the server object of shiny. It is used to reactively filter the dataset according to the position selected by the user.
- Server output function: this function is used to display the parallel coords chart. It is implemented using *plotly* library's function plot_ly; with attribute type set to '*parcoords*'.

4.5. IDIOM 2 Implementation.

The tridimensional scatter plot for cluster analysis is implemented using two functions:

- UI input function: this function is implemented on the ui shiny object, and is used to display the scatter plot to the user. Moreover, an input filter is implemented in the ui for the user to select the cluster analysis variable.
- Server output function: this function is used to display the scatter plot chart. It is implemented using *plotly* library's function plot_ly. The attributes x, y and z of this function are set to the three principal components. Attribute color is set to the variable selected by the user to perform the cluster analysis.

4.6. IDIOM 3 Implementation.

The Bubble Chart Analysis is implemented using two functions:

- UI input function: Is responsible for defining the layout of the Shiny app, including the user interface components. For this analysis the key components are:

    1. fluidPage: This function creates the layout of the app, where all the data is visualized into the different representations.

2. titlePanel: It sets the title of the app, in this case “Buble Chart”.

3. sidebarPanel: It contains the input elements for user interaction. In this case, it includes all the countries contained in the dataset, and for the initialization only five are chosen (Spain, Italy, Argentina, Portugal and England).

4. mainPanel: The main panel contains the output elements, in this case the plotlyOutput for displaying the bubble chart.

- Server output function: Is responsible for defining the server logic of the Shiny app. For this analysis the key components are:

    1. renderPlotly: It is the function where all the logic is defined. It is associated with the plotlyOutput("bubble_chart") in the UI.

    2. Filtering Logic: Inside the renderPlotly function, the dataset is filtered based on the user's selection of nationalities and the categorical variables (value and wage) are converted into numerical.

    3. Plot_ly: It specifies the data, x-axis (overall), y-axis (value), size (wage), color (nationality) and text (name of the player and wage to be shown once the user is on the plot).

    4. layout: Basically it sets the titles for the x and y axes, providing a chart title, and configuring other layout options.

4.7. IDIOM 4 Implementation.

The histogram analysis is implemented using two functions:

- UI input function: this function is implemented on the ui shiny object, and is used to display the chart to the user. In addition, two input filters are implemented in the ui:
    - An age slider to filter the number of players that we want to show.
    - Another filter to filter players by position and reduce the number of results.
- Server output function: this function is used to display the histogram. It is implemented using *plotly* library’s function plot_ly. The x axis is set to the *Potential* variable.

5. Application deployment and how to run.

The developed application has been deployed with all dependencies to shinyapps.io.
The url of the app is: https://raulmoldes.shinyapps.io/shinyapp/.

To run the source code of the app, navigate to the folder of the project and open it on a console. After that open a console and type: ***runApp().***

6. References.

1. Dataset: FIFA 18 Complete Player Dataset (kaggle.com).
2. Source code: RaulMoldes/DataVisualization: Data Visualization Assignment (github.com)
3. Deployed application: https://raulmoldes.shinyapps.io/shinyapp/.
4. R documentation: R: Documentation (r-project.org).
5. Shiny documentation: Shiny - RStudio.
6. Plotly documentation: Plotly r graphing library in R.
7. Teams in every league: https://www.fichajes.com/.