Data Visualization using KDE Heat-maps

Adam Mengistu (*Author)*
Sam Houston State University
Huntsville, TX
adambmengistu@gmail.com

I. Introduction

In the realm of football/soccer, the power of data can be used to the fullest through techniques such as web scraping and data visualization. This has become paramount in understanding player performance and strategic patterns for players, coaches, fans, etc. With the wealth of information available today, extracting and visualizing data not only allows many to gain insights into the past but also empowers those to predict and plan for the future. This paper goes in depth to unravel the careers of two iconic players, Lionel Messi and Cristiano Ronaldo, by using extracted data from all their matches. Through the use of Kernel Density Estimation (KDE) heat maps, the aim is to not only provide a visual representation of their careers but also to engage in a comprehensive comparison and contrast of these two players.

The world of fans and analysts have been captivated by the rivalry between Messi and Ronaldo. Both players have showcased unparalleled skill, determination, and consistency over the years, setting numerous records and accumulating an abundance of data in the process. By leveraging web scraping techniques, we have collected data on their career actions, thus opening the door to a deeper understanding of their respective playing styles and contributions to their teams.

By using KDE-generated heat maps, we not only aim to depict the careers of Messi and Ronaldo in a visually appealing manner but also to uncover insights into their strengths and weaknesses. Through our analysis, we aspire to offer a unique perspective on the contributions of these two legends to the game and shed light on how their careers may influence future matches.

The following sections will provide a detailed account of the methods used, the data extracted, and the insights gained, allowing readers to embark on a data-driven journey into the remarkable careers of Lionel Messi and Cristiano Ronaldo.

II. Preparation & Data Collection

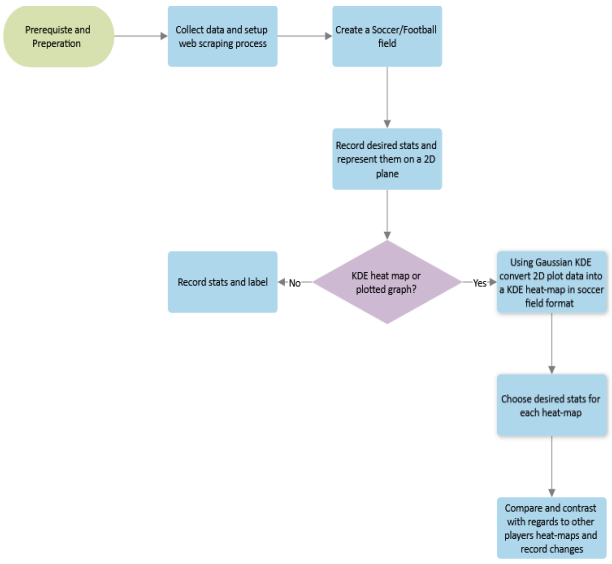


Fig. 1 Flow chart of the entire project process.

To begin our goal we must prepare and preprocess useful and sufficient data to be used for visual representation. The method I chose was basic web scraping which involves the use of gathering large amounts of unformatted HTML data from a website and converting it to be of use and JSON data extraction. Of course one must see if it's allowed by the website of choice as sometimes this can result in being banned from the domain. I chose understat.com, a sports analytical site that records player data and stats free to the public. To implement this in code one must first send a "*request.get(link)*", this code portion is to begin access to a site. When conducting web scraping users are practically simulating the actions of a user by retrieving data but to save time using a computer to accumulate large amounts of data. In the code we will act like a user but in reality we are simulating the task of maybe multiple users with how much data we are collecting. After that task has been completed we then need to retrieve all JSON data and trim it accordingly to the format of one's choice to then understand the website's data and variables that can be used in our python code implementation. One can find this info on any website when right clicking and selecting "inspect" as this shows all HTML code used for any site. For this implementation we are going to focus on a player's shots, goals, xG (expected goals), assists, shots on target and blocked shots. These metrics are chosen to see a players attacking abilities and how threatening they are to an opposition defense. Primarily, I chose two of the most efficient attackers of the modern game for this implementation, Cristiano Ronaldo and Lionel Messi as they are a great fit for showcasing the differences.

Once the data is gathered and trimmed a simple task will be converted for usage. I converted the data to floating point (float64) to properly display the data and info on a plane/graph to later use when we showcase the points of interest in a 2D plane. To create the 2D plane we will need to create a mock football/soccer field to mimic an actual cartesian plane so we can properly showcase the actions of these two players. This can be done manually by creating numerous lines in matplot or importing the "mplsoccer" python library. Once complete it's time to gather info for each player and properly display the results in separate points first before any heat map is created.

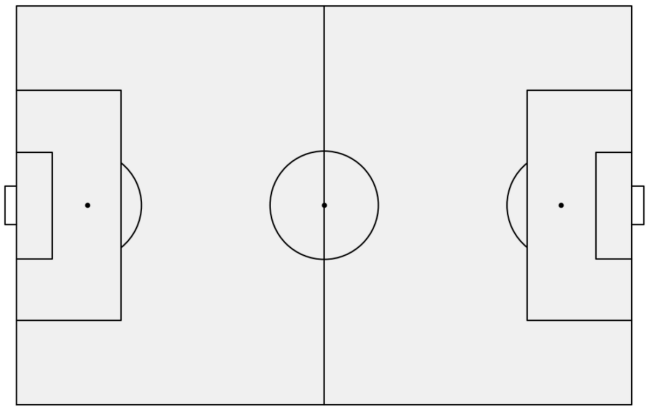


Fig 2. Soccer field created in matplot using lines relative to the cartesian plane.

III. Implementation

My implementation shows career statistics for each player and each metric and uses the xG or expected goal percentage to determine the size of each point with regards to how easy each attacking move was. To calculate the xG many factors come into play such as shot location, shot history, defensive shape, shot type, etc. Typically for each player's shots, the xG is more when they are closer and center to the goal compared to if they were more far away and outward.

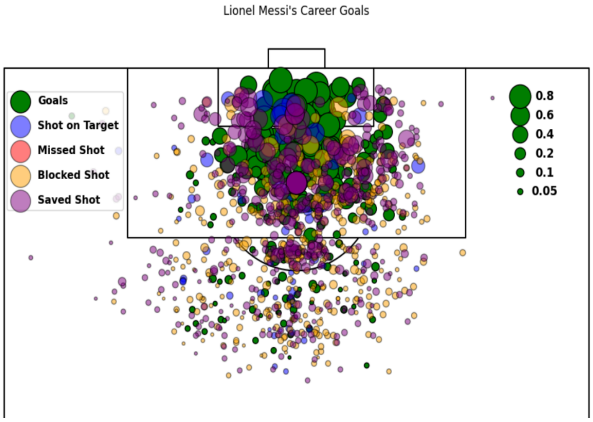


Fig 3. Lionel Messi's career goals shot map with recorded xG for each shot.

Once every metric is recorded in dot form we begin the conversion into KDE heatmaps form. To accomplish that one must import “gaussian_kde”, “numpy” and ”scipy.stats” first before anything. Once downloaded you would need to initialize a meshgrid and linspace to overlay the dimensions of the entire representation of the field that you desire. After that is completed the y and x values need to be inserted into a values and positions variable to be read. Finally to be represented as a heatmap a color format is needed as I mostly recommend inferno for a more traditional heatmap look. This process will be repeated as many times for each metric or recorded stat needed as well as changing of many preferences like the alpha which will tinker the color representation and heat map severity.

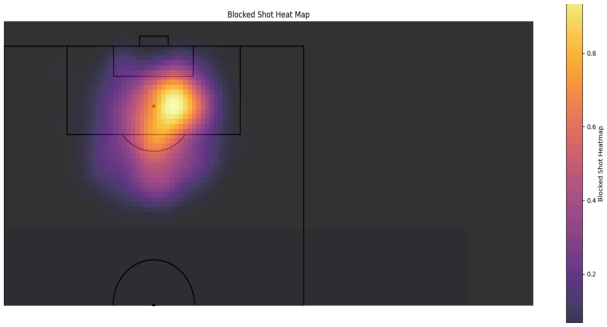


Fig 4. Blocked shot heat map for Lionel Messi.

IV. Results

After concluding the pre-processing, data collection and implementation, we are given numerous amounts of visualized data in KDE heatmap format for both players for their career actions. Now they can be compared and contrasted with regards to the intensity factor in each heatmap. Using the above example, areas that are closer to 1.0 are the most successful and most traversed areas and as they get lower down the bar ones that are represented as purple and black are areas of the least success and least traversed spots of the field. We can display them in comparison using matplot or manually as shown.

V. Previous Experiments

To further evaluate the comparison between the two heatmaps generated, one can overlay the two players' heatmaps and use the difference of each area as a new represented color in python’s “opencv” library. This is not the most visually appealing way of conducting the comparison but it does offer a unique way for comparison between the two players and can offer help for areas that often are hard to spot at a first glance. To achieve this, you would need to store the first player's heatmap in a numpy array and subtract that by the second player of choice heat map array, then store the difference in a variable and represent that variable as a KDE heat map.

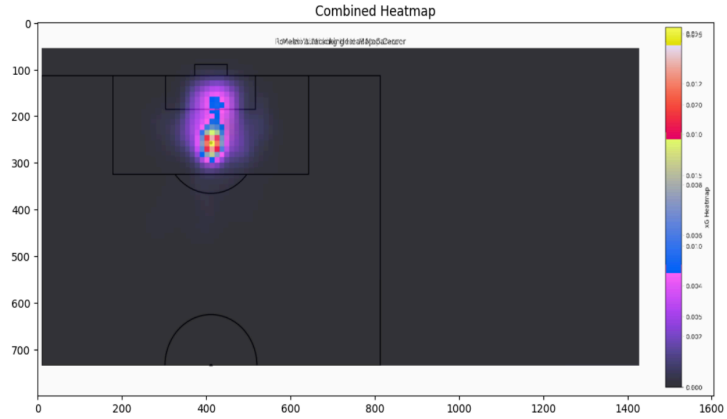


Fig 5. Combined attacking heat map for both Lionel Messi and Cristiano Ronaldo.

The result is a combination of each image pixel and identifies areas of similarities and differences with regards to what color it represents, similar to the original heatmap that was created before. Previously this project was planned to orchestrate both players in a more comparative format as the thought of using computer vision and openCV was in the works however this is often very computationally demanding and expensive to fund the resources of full uncut length football matches. Previous runs were halted due to numerous other reasons such as the video not fully recording players fully, access to matches against the two players, and mainly camera angles not being able to support each player on the pitch simultaneously. A mock run of what it would look like was created but lacked the

true computer vision implementation incorporated into it.

Fig 6. Ronaldo's recorded Heat Maps

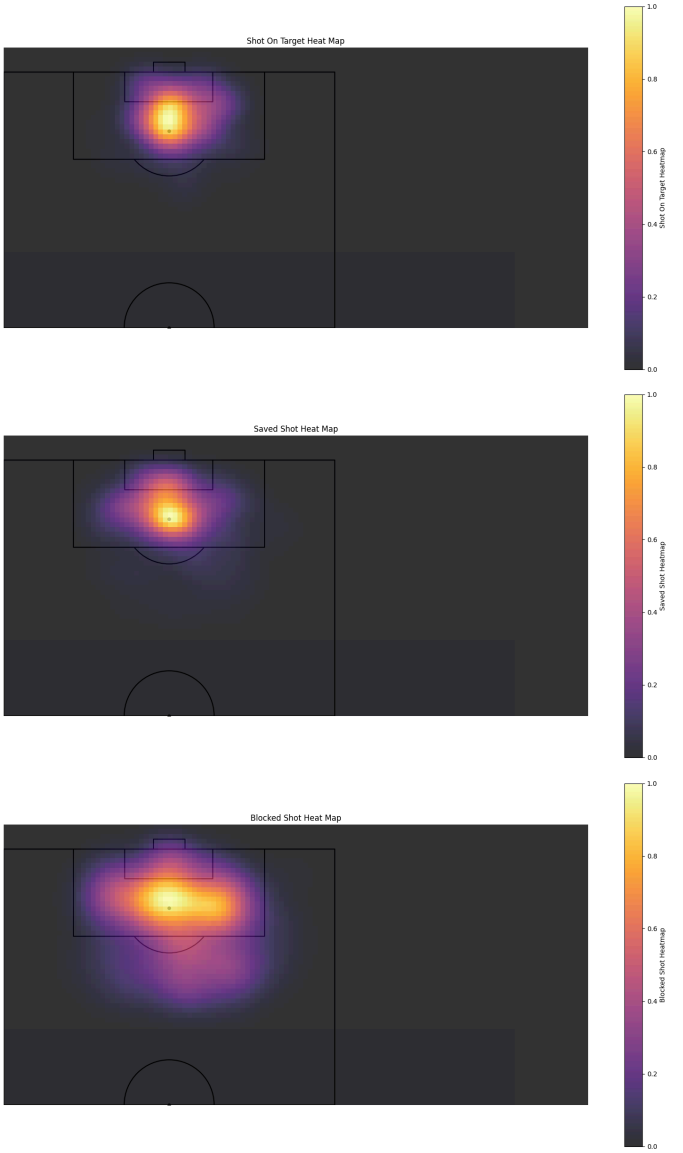
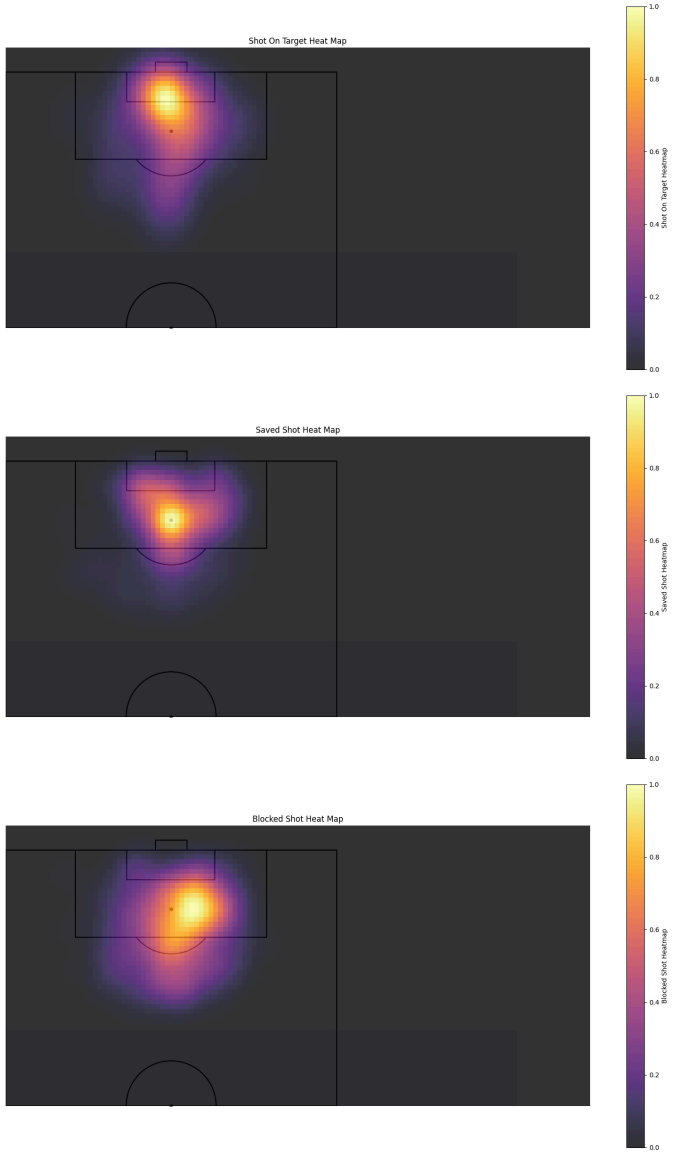
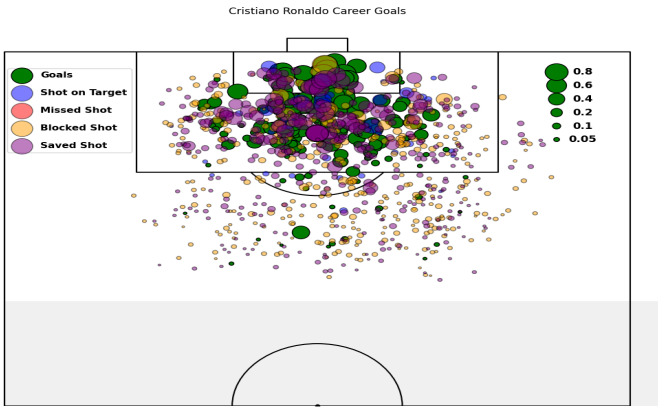
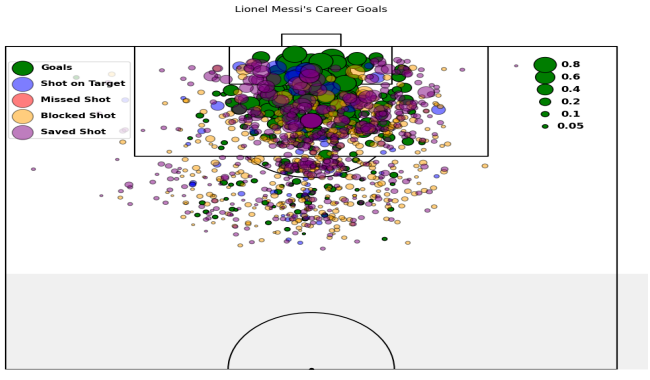


Fig 7. Messi's recorded Heat Maps







VII. Compare and Contrast

In order to gain a comprehensive understanding of the project's underlying objectives, a thorough comparison and contrast of the generated data is needed. Examining the heat maps of Cristiano Ronaldo reveals a notable trend: Ronaldo exhibits a heightened level of effectiveness when operating on the right side of the pitch. Proof that his play style is best when shifting towards the right is reflected in his impressive success rate in right-sided shots compared to those on the left. Notably, Ronaldo's proficiency in penalty kicks is remarkable, boasting a success rate of 83.6%, having scored 139 out of 166 penalties (Biswas 2021). This statistic underscores the significance of the penalty spot as his prime scoring location. As a listed Left Winger (LW), Ronaldo's preferred playstyle involves driving down the wing and exploiting opportunities to cut right. Note that this aggressive manner causes many of his blocked and saved shots to be located when slightly left or right form the goal post as this manner isn't a guarantee in the modern game with the abundance of skilled keepers.

Conversely, Lionel Messi's attacking heat maps unveil distinctive insights into his preferred approach. Unlike Ronaldo, Messi excels in driving down the right wing, showcasing his proficiency on the left side. His scoring tendencies lean towards the area near the goal, with the penalty spot being a focal point. Messi's prowess is evident in his 77% success rate from the penalty spot, having converted 109 out of 144 shots (Brischetto 2022). However, Messi encounters challenges on the right side near the goal, where tricky angles prove to be a challenging task as keepers are tracking the ball easily. As seen with Ronaldo, typically wing players tend to see several blocked and saved shots when they are slightly leaning to the left or right of a goalpost due to the new age goalkeeper’s positioning.

VIII. Conclusion

It becomes evident that the heat maps generated through Kernel Density Estimation (KDE) provide a powerful tool for analyzing and comparing the playing styles of iconic footballers such as Lionel Messi and Cristiano Ronaldo and the utilization of web scraping techniques has enabled a detailed examination of each player's career actions, shedding light on their attacking prowess, preferred areas on the pitch, and goal-scoring tendencies, etc. By converting the recorded metrics into dot form and subsequently into KDE heat maps, the analysis becomes more nuanced, allowing for a deeper understanding of the players' strengths, weaknesses and giving ultimate insight to anyone who utilizes it.

References

Lionel Messi Recorded Stats
[1] (“Lionel Messi | Paris Saint Germain | XG | Shot Map | Goal Stats | Understat.com,” n.d.)

Cristaino Ronaldo Recorded Stats
[2] “Cristiano Ronaldo | Juventus | XG | Shot Map | Goal Stats | Understat.com.” n.d. Understat.com. https://understat.com/player/2371.

MPL Soccer Field Documentation
[3] Rowlinson, Andrew. n.d. “Mplsoccer: Football Pitch Plotting Library for Matplotlib.” PyPI. Accessed November 29, 2023. https://pypi.org/project/mplsoccer/.

Messi Penalty Success Rate
[4] Brischetto, Patrick. 2022. “Lionel Messi penalty kick history: Argentina captain's record on penalties.” Sporting News. https://www.sportingnews.com/ca/soccer/news/lionel-messi-penalty-history-record-argentina/wm5uvkvdpi4bp85trqxvsbom.

Ronaldo Penalty Success Rate
[5] Biswas, Koushik. 2021. “Penalty stats: Ronaldo vs Bruno, who is better in success rate?” Sportz Point. https://sportzpoint.com/football/penalty-stats-ronaldo-vs-bruno-who-is-better-in-success-rate.

SciPy Documentation
[5] “scipy.stats.gaussian_kde — SciPy v1.10.1 Manual,” *docs.scipy.org*. https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.gaussian_kde.html