**CSCI B550 Penalty Shootout Simulator**

**Shourish Kothawale**

**Abstract**

The penalty shootout simulator is a project that recreates the experience of a soccer penalty shootout. It uses real-world data from the FC 25 dataset on Kaggle, which includes information about player accuracy, goalkeeper save rates, and other performance statistics. This data helps make the simulation more realistic by reflecting how actual players and goalkeepers perform.

The simulation alternates between a shooter taking a penalty and a goalkeeper trying to save it, mimicking the structure of a real penalty shootout. Users can try out different lineups or strategies and see how they play out in the simulation. It’s a simple way to explore the dynamics of penalty shootouts while learning about how data can be used to predict outcomes.

**Introduction**

This project focuses on developing a penalty shootout simulation model that uses probability distributions based on real-world player metrics from the Kaggle FC 25 dataset. The goal is to create a realistic and data-driven system that can analyze and simulate penalty shootouts under professional soccer rules. By relying on player-specific data, such as shooting accuracy and tendencies, and goalkeeper save probabilities, the simulation provides a more accurate representation of outcomes.

The system simulates penalty shots using coordinates to represent the ball's location and the goalkeeper's save attempt. Each player's probability distribution is dynamic, incorporating multiple factors such as their historical performance, tendencies, and even situational variables, enhancing the realism of the model. The shootout follows standard penalty shootout rules, including early termination when a winner is determined before all rounds are complete and sudden death rounds if the game is tied after the initial series.

The purpose of the simulation is to explore two key questions:

1. What is the best lineup of players for maximizing a team's chances of winning penalty shootouts?
2. Could hiring a different goalkeeper with a higher save rate significantly increase the team’s success?

To answer these questions, we ran experiments with multiple trials, testing various team configurations and goalkeeper options. The results from these trials aim to provide insights into optimal strategies and team-building decisions for penalty shootouts. We will be restricting the team selection two specific teams that are already present as we want to maximize their outputs. The two teams chosen would be “Man Utd” and the best team possible based on their ratings which would be manually set. The additional keeper would be made up, with statistics for that manually set up. But we must make sure that the player we pick will be in budget, so we cannot just simply max out the ratings for a player, as transferring that would be very expensive. So, we Are looking for a player that would give you decent increase in save rate.

**Model Description**

The penalty shootout simulation consists of two primary components: **Shot Taker** and **Goalkeeper**. The model evaluates penalty attempts by determining the trajectory of the shot and the goalkeeper’s ability to block it. The interaction of these two elements is governed by player-specific attributes from the dataset, ensuring a realistic outcome based on real-world data.

**Shot Taker**

The Shot Taker is responsible for generating the shot’s coordinates based on multiple factors:

* **Shot Targeting:**  
  The likelihood of the shot being on target is determined using the player's *Penalties* attribute from the dataset, which represents their historical accuracy.
* **Shot Side Selection:**  
  The side of the goal (left or right) is based on the player’s *Preferred Foot* and *Overall Rating (OVR)*. The probability of shooting to either side is modeled using linear interpolation between these attributes. Players are more likely to aim opposite their preferred foot, but higher OVR players exhibit greater variability, representing their strategic versatility.
* **Shot Power:**  
  Shot power is derived from the dataset and modeled using an empirical distribution. The mean corresponds to the player's *Shot Power* attribute, and the distribution becomes more skewed left for lower-rated players, reflecting their inconsistency in delivering powerful shots. Additionally, higher shot power impacts the goalkeeper’s reflexes, making it more difficult for the goalkeeper to respond effectively to a fast-moving shot.
* **Shot Coordinates:**  
  Coordinates are determined based on the shot side, accuracy, and power. The accuracy is modeled as a normal distribution, with the standard deviation controlling the variability. The resulting shot lies within a circle centered on the chosen side, with more accurate players having smaller radii, reflecting their precision.

Players are selected cyclically from the team lineup, starting with the first shooter and looping back as necessary.

**Goalkeeper**

The Goalkeeper attempts to save the shot based on metrics such as reflexes, diving ability, and positioning.

* **Dive Decision:**  
  The side to which the goalkeeper dives is determined by their *Reflexes*. A custom probability distribution function (PDF) is generated for each goalkeeper, where higher reflex ratings increase the likelihood of diving towards the correct side. Lower-rated goalkeepers exhibit a uniform probability of diving in any direction.
* **Movement Before the Shot:**  
  The goalkeeper's lateral movements before the shot are modeled to mimic their real-life behavior of side-stepping to distract the shooter. This movement is influenced by the *Positioning* attribute, with better-positioned goalkeepers exhibiting larger, more controlled lateral movements.
* **Block Area Calculation:**  
  The block area represents the region the goalkeeper can effectively cover and is calculated using the unit circle. A square approximation with a side length of 0.5 meters is used for simplicity.
  + **Diving Angle:**  
    Using the slope between the keeper's starting position and the shot trajectory, an angle is calculated. A conical range around this angle represents the keeper’s potential diving direction. Goalkeepers with higher *Diving* statistics have smaller deviations, indicating more precise dives.
  + **Diving Distance:**  
    The keeper’s reach is determined by their *Diving* rating, with better ratings allowing for longer dives. Variability is introduced to reflect realistic differences in reach across goalkeepers.

**Shot Outcome**

To determine the outcome of a shot:

1. Check if the shot is on target (determined by the *Penalties* attribute).
2. Compare the shot coordinates with the goalkeeper’s block area.
3. If the coordinates overlap with the block area, the shot is saved. Otherwise, it results in a goal.

This calculation is intuitive, as a shot is typically blocked when it intersects with the goalkeeper's hands. Edge cases, such as deflections, are acknowledged as areas for future refinement.

**Simulation Description**

The penalty shootout simulation was implemented using a combination of VBA, SQL, and Python to integrate and process the model components effectively. Each tool served a specific purpose in the simulation pipeline, allowing for a comprehensive system that emulates real-world penalty shootouts based on player data.

**Development Process**

* **VBA Implementation:**  
  The primary simulation logic was written in VBA due to its seamless integration with Excel, where the dataset was stored and manipulated. VBA was used to create functions for calculating shot trajectories, goalkeeper dives, and determining the outcome of each shot. However, the lack of prior knowledge of VBA syntax presented a significant obstacle, as it is less intuitive compared to other programming languages. Despite this, VBA's close interaction with Excel made it a suitable choice for real-time calculations and updates.
* **SQL for Data Wrangling:**  
  SQL was utilized for preprocessing the dataset, including filtering, sorting, and tailoring team lineups. SQL allowed for efficient handling of the 17,000+ players in the dataset, enabling the creation of custom teams with specific attributes.
* **Python for Data Manipulation:**  
  Python was used to manipulate and process data, particularly for creating player-specific probability distributions and refining empirical data. Python's robust libraries and simplicity in handling large datasets complemented the SQL and VBA components.

**Major Challenges and Decisions**

1. **VBA Syntax Complexity:**  
   Transitioning to VBA for the simulation posed an initial hurdle due to its verbose syntax and limited documentation compared to Python. Despite this, VBA’s compatibility with Excel made it the preferred choice, and learning the language proved worthwhile for direct data manipulation and visualization.
2. **Empirical Distribution Creation:**  
   Tailoring probability distributions for over 17,000 players was computationally expensive and impractical. This challenge was addressed by leveraging mathematical estimations to generate accurate, tailored PDFs for shot power, accuracy, and other metrics. Simplifications were made where necessary to balance realism and computational efficiency.
3. **Mathematical Dependencies:**  
   The simulation heavily relied on mathematics, particularly for creating PDFs and determining shot and dive coordinates. Analytical approaches ensured that results were consistent and aligned with observed player behaviors.

**Verification and Validation**

To validate the model:

* Initial test runs were compared with real-world penalty statistics, such as the observed save rate of approximately 15%. Early results deviated from this benchmark, with save rates ranging between 20% to 25%.
* Adjustments were made to goalkeeper block areas and shot accuracy distributions, reducing the overestimation of saves. After these changes, the save rates fell within an acceptable range (9% to 13%).

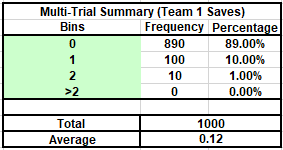


Figure 1: A result from an experiment

* We also Validated the model by using debug statements to print out the results and see if they made sense.

Verification ensured that:

* Shot trajectories aligned with player accuracy and shot power attributes.
* Goalkeeper dives adhered to realistic reflex and reach values.
* The simulation adhered to professional soccer penalty shootout rules, including sudden death and early termination scenarios.
* We also looked at previous results from a very recent game and got very close results to the game. The goal difference is of 2 from the actual game but we expect that level of error as the simulation is estimated and never 100% accurate.

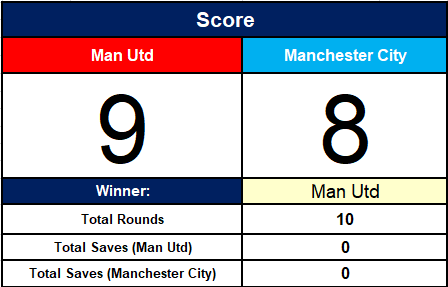


Figure 2: Result of One Simulation

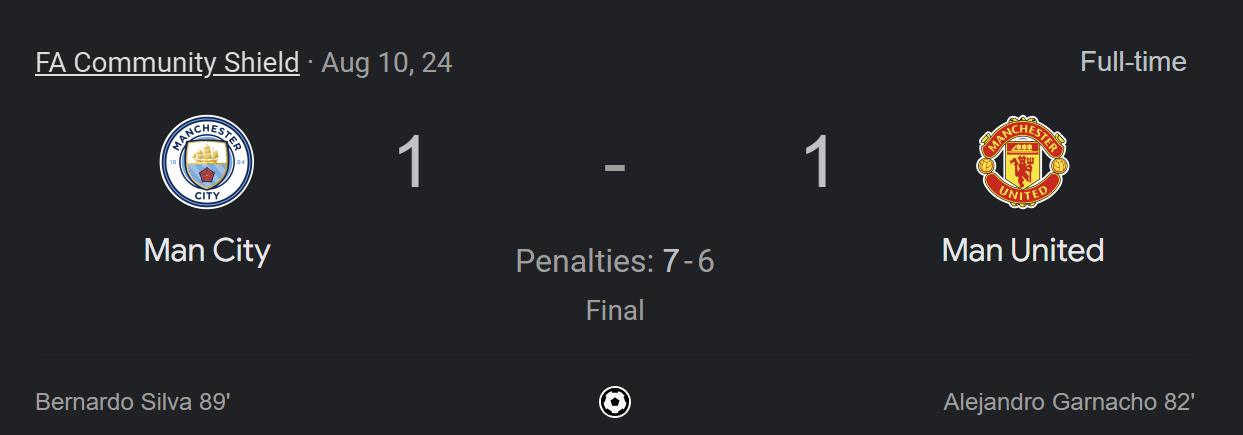


Figure 3: Real life result

* Most of the other simulations also are close to this result in penalties.
* The simulation was also set up such that the teams were aligned to the actual line up. Another validation was that Manchester City had actually missed a shot which caused them to lose, which is reflected in the simulation too.

**Major Decisions**

Several important decisions shaped the design and implementation of the penalty shootout simulation. These decisions balanced the need for realism, efficiency, and practicality in achieving accurate and meaningful results.

1. **Reducing the Complexity of Goalkeeper Diving:**  
   Initially, the goalkeeper’s dive mechanics were designed to include a highly detailed simulation of body movements and angles, accounting for fine-grained differences in reflexes and reach. However, this approach proved computationally intensive with negligible improvement in accuracy.
   * **Decision:** Simplify the goalkeeper’s diving model to a square block area and conical angle range based on reflexes and diving ratings.
   * **Reason:** The simpler model delivered realistic results with less computational overhead and aligned with observed save percentages in real-world penalty shootouts.
2. **Using SQL for Data Handling:**  
   Handling the large FC 25 dataset, which includes over 17,000 player records, was challenging within VBA. Manipulating such extensive data directly in VBA was slow and cumbersome, especially for tasks like filtering and sorting.
   * **Decision:** Employ SQL for data wrangling, such as filtering players by attributes, setting up custom team lineups, and organizing data.
   * **Reason:** SQL’s structured query language is highly efficient for dealing with large datasets, providing better performance and ease of use than VBA for data handling.
3. **Using FC 25 Game Data:**  
   Penalty shootout-specific datasets were unavailable, necessitating a decision about what source data to use for the simulation.
   * **Decision:** Utilize the FC 25 dataset from Kaggle, which provides detailed player metrics derived from FIFA, a well-known soccer simulation game.
   * **Reason:** The FC 25 data accurately represents real-world soccer player attributes, such as penalties, shot power, and reflexes. FIFA’s game engine is highly regarded for its realistic modeling, making this dataset a reliable substitute for direct penalty-specific data.

By making these decisions, the simulation achieved a balance between complexity, practicality, and accuracy, ensuring a model that is both computationally efficient and realistic.

Figure 4: Win Probability matching actual Head-to-Head of the matches

**Hypothetical I**

**Hypothetical I: What is the Best Lineup of Players for Maximizing a Team's Chances of Winning Penalty Shootouts?**

**Description**

This hypothetical explores the optimal lineup of players to maximize a team's success in penalty shootouts. The simulation was configured to test various combinations of players, focusing on attributes such as:

* **Penalties Rating:** Determines the likelihood of shots being on target.
* **Shot Power:** Influences shot speed and its impact on goalkeeper reflexes.
* **Preferred Foot and Overall Rating (OVR):** Affects shot placement and variety.

The lineup selection was based on tailoring these attributes for a balanced team composition, including players with strong penalty-taking abilities and a reliable goalkeeper with high reflex and diving statistics. After Running the experiment with 1000 trials we found the following Line up for Man Utd to be the best against Liverpool.

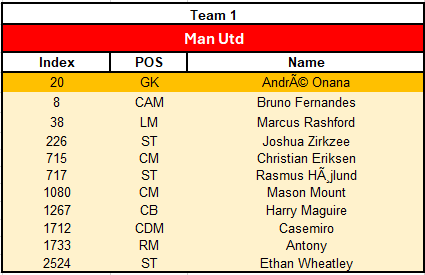


Figure 5: Best Man Utd team

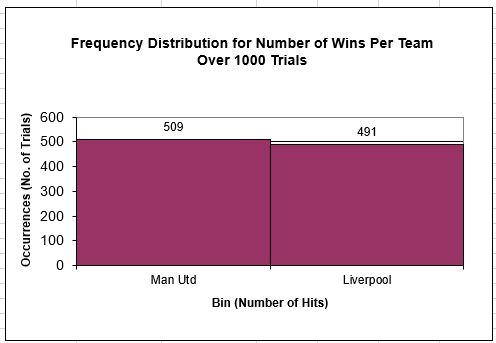


Figure 6: Win Rate Just above Liverpool

We chose Man Utd and Liverpool for this task since their players almost have similar rating as a team.

The main reason this team works is because this team is the team with the highest accuracy. The more accurate the team is the less likely they are to miss. Note we are testing this with the Best Liverpool team possible. The second most accurate team we could find after was the following asserting our result.

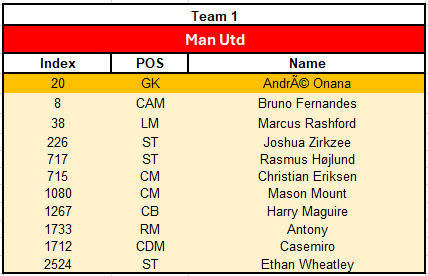


Figure 7: Second Best Team for Man Utd

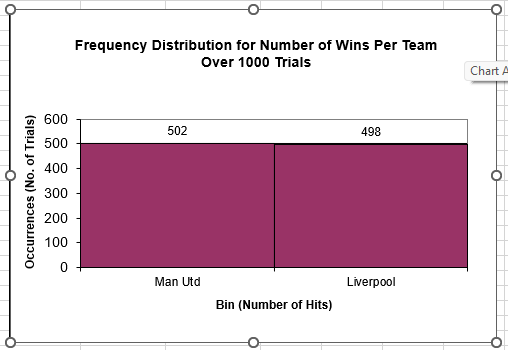


Figure 8: Second Best Result

The Second team consists of the same players but they go in different order. This clearly shows order matters in Penalties, as there are rules for early exit. The most optimal strategy that come out of these is you should players with the highest penalty statistics. This falls through as these are the players that have the lowest chance to miss. This means that they will miss less allowing the possibility of more goals to be scored. If the players miss more it’s going to make it worse as the keeper doesn’t have to save as the goal won’t be counted in the first place.

**Hypothetical II**

**Hypothetical II: Could Hiring a Different Goalkeeper with a Higher Save Rate Significantly Increase the Team’s Success?**

**Description**

This hypothetical examines the impact of goalkeeper performance on a team's success in penalty shootouts. Specifically, the question asks whether replacing the current goalkeeper with one who has a higher save rate (e.g., better reflexes, diving skills, and positioning) can significantly improve the team’s overall win rate.

**Methodology**

We can use a search approach like binary search to find the player needed. We would be comparing our team against the best possible team which is players with the highest chance of shot on target. They are the best team since they are less likely to miss and the following chart also proves it.

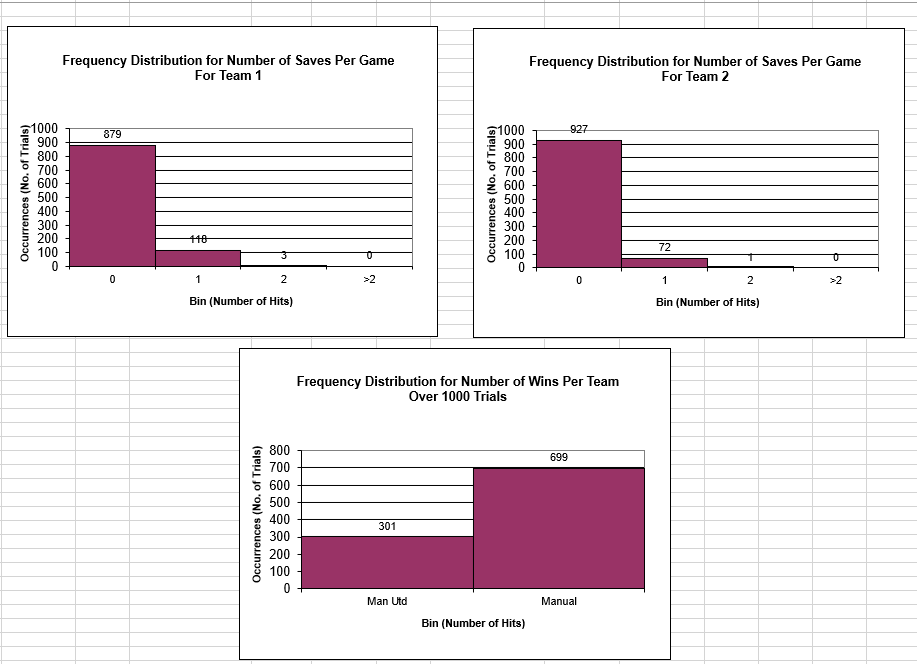


Figure 9: Comparing graphs of teams

We can clearly see that the Best XI keeper is much better and that there is a significant difference in how many shots are blocked. We can also see the significant difference in the Win rates of the teams. The Binary search like approach will be first seeing how much difference the best possible keeper would make (i.e. all stats = Max (stat points) except for shooting stats as that would not make sense). Then we will try to find a keeper between the current keeper’s stats and the best keepers stats. We want to make the least possible changes to the current keepers’ statistics so that we can reduce costs. Remember the better the player the higher their transfer fee. To make sure not much is changed from the current keeper we keep their shot statistics the same.

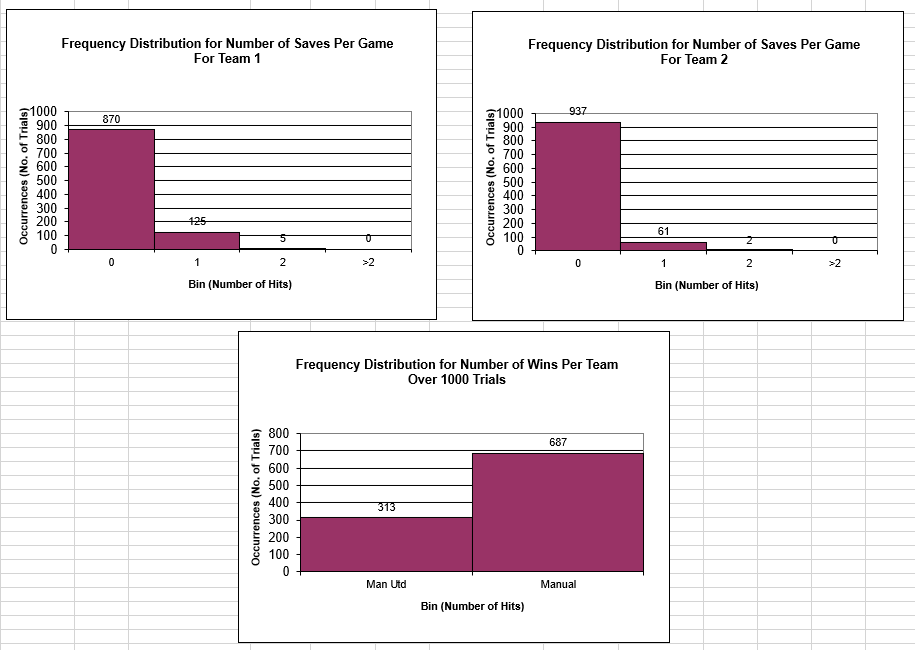


Figure 10: Better Keeper Graph

After Trying out the strategy the player with the GK statistics as the following, those statistics are based on the fact that we have to stay in budget so we can’t go on crazy changes:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GK Diving | GK Handling | GK Kicking | GK Positioning | GK Reflexes |
| 87 | 86 | 91 | 85 | 89 |

We can clearly see that the increase of the number of saves is not too much, and the win rate does not go up by much, so we can say it is not highly recommended that Manchester United changes their keeper for penalties. This clearly shows us that in a penalty match the keeper statistics play a lesser part compared to the shot takers. This further supports the Statistics that only a 15% chance of seeing a save in a penalty shootout, which is not many shots in the long run.

**Future Works**

While the current simulation provides valuable insights into penalty shootouts, there are several areas for improvement and further exploration:

1. **Integration of Computer Vision:**
   * Implementing computer vision could simulate the goalkeeper’s ability to "see" the ball. This would enhance the model's realism, enabling the goalkeeper to make more informed decisions based on shot trajectory, shot taker body movement, and other visual cues.
   * Similarly, computer vision for the shot taker could help simulate pre-shot adjustments based on the goalkeeper's positioning and movement.
2. **Artificial Intelligence for Goalkeepers:**
   * Developing an AI model for the goalkeeper could further refine decision-making processes, such as selecting the optimal dive direction and timing. This would mimic real-world goalkeeper strategies and improve the save rate in the simulation.
3. **Incorporating Physics:**
   * Adding physics elements like gravity, air resistance, and wind effects would provide a deeper layer of realism. This would help simulate shot dynamics more accurately, including variations in ball spin and speed.
4. **Additional Hypothetical Questions:**
   * Investigate the psychological effects of penalty shootouts, such as fatigue or pressure, on player performance
   * See how changing shooting stats affects the game
   * Would training of players improve performance considering it would improve their statistics?

The **CSCI B550 Penalty Shootout Simulator** modeled realistic penalty shootouts using data-driven insights from the **FC 25 dataset**, addressing optimal lineup selection and goalkeeper performance. Results showed that focusing on players with high penalty accuracy and strategic lineup ordering significantly boosts success, while hiring a better goalkeeper has only a marginal impact due to the inherently low save rate (~15%). The project bridges sports analytics and computational modeling, providing actionable insights for team strategy. Future enhancements, such as integrating computer vision, AI decision-making, and physics-based simulations, could further refine its accuracy and expand its applications in sports analysis.