UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Posgrados

Modeling a Fantasy Football Team Using Machine Learning Algorithms and Linear Programming

Proyecto de Titulación

Santiago Viteri Puyol

Alejandro Proaño Lozada, Ph.D.

Director de Trabajo de Titulación

Trabajo de titulación de posgrado presentado como requisito para la obtención del título de Magíster en Inteligencia Artificial

Quito, 19 de diciembre de 2024

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ
COLEGIO DE POSGRADOS

HOJA DE APROBACIÓN DE TRABAJO DE TITULACIÓN

Modeling a Fantasy Football Team Using Machine Learning Algorithms and Linear Programming

Santiago Viteri Puyol

Nombre del Director del Programa:	Felipe Grijalva
Título académico:	Ph.D. en Ingeniería Eléctrica
Director del programa de:	Inteligencia Artificial
Nombre del Decano del colegio Académico:	Eduardo Alba
Título académico:	Doctor en Ciencias Matemáticas
Decano del Colegio:	Ciencias e Ingenierías
Nombre del Decano del Colegio de Posgrados:	Dario Niebieskikwiat
Título académico:	Doctor en Física

Quito, diciembre 2024

© DERECHOS DE AUTOR

Nombre del estudiante: Santiago Viteri Puyol

Código de estudiante: 00338823

C.I.: 0603503731

Lugar y fecha: Quito, 19 de septiembre de 2024.

ACLARACIÓN PARA PUBLICACIÓN

UNPUBLISHED DOCUMENT

DEDICATORIA

Para Banana, Papas, y Neno con todo mi cariño.

AGRADECIMIENTOS

Expreso mi agradecimiento a Vaastav por su labor en la recopilación de información sobre los jugadores de la liga de fantasía de la Premier League, la cual resultó fundamental para el desarrollo de este proyecto. Asimismo, extiendo mi gratitud a mi tutor, Alejandro Proaño, por su paciencia y apoyo a lo largo de todo el proceso.

RESUMEN

La formación de un equipo de Fantasia de la Premier League requiere analizar el desempeño de los jugadores a lo largo del tiempo y optimizar la selección de aquellos con mayor probabilidad de destacar. Este proceso demanda un entendimiento profundo del juego y del desempeño de los jugadores, lo cual resulta desafiante debido a la complejidad inherente del deporte. Las técnicas de aprendizaje de maquina ofrecen una posible solución a este problema de toma de decisiones. Por ejemplo, modelos supervisados y no supervisados, como las redes LSTM y el agrupamiento K-means, permiten capturar patrones temporales y revelar estructuras ocultas en los datos de los jugadores. En este estudio, aplicamos y contrastamos el desempeño de dos modelos de aprendizaje de maquina. El primer modelo utiliza redes LSTM para predecir los puntos futuros de los jugadores, combinando estas predicciones con un algoritmo de optimización para seleccionar la alineación óptima para las siguientes cuatro jornadas. El segundo modelo emplea un algoritmo K-means para agrupar a los jugadores en clústeres basados en métricas de desempeño. Las transferencias de jugadores se evalúan en función del clúster al que pertenecen y de sus puntuaciones con respecto al indice ICT (Influencia, Creatividad, Amenaza). Al comparar estas dos estrategias—optimización basada en LSTM y selección de jugadores basada en clústeres—buscamos determinar cuál enfoque genera mejores resultados en términos de puntos totales acumulados durante la temporada de la Premier League

Palabras clave: Aprendizaje de máquina, programación lineal, redes LSTM, agrupamiento K-means, Fútbol de Fantasía

ABSTRACT

Building a Fantasy Premier League team requires analyzing player performance over time and optimizing the selection of those most likely to excel. This process demands a deep understanding of the game and player development, which is challenging due to the sport's inherent complexity. Machine learning techniques offer a potential solution to this decision-making problem. For example, supervised and unsupervised models like LSTM networks and K-means clustering capture temporal patterns and reveal hidden structures in player data. In this study, we apply and contrast the performance of two machine learning models. The first model uses LSTMs to forecast players' future points, combining these predictions with an optimization algorithm to select the optimal lineup for the next four game weeks. The second model applies a K-means algorithm to group players into clusters based on performance metrics. Player transfers are then evaluated based on the cluster they belong to and their ICT (Influence, Creativity, Threat) scores. By comparing these two strategies—LSTM-based optimization and cluster-based player selection—we aim to determine which approach yields better outcomes regarding total points accumulated over the Premier League season.

Key words: machine learning, linear programming, LSTM networks, K-means clustering, Fantasy Soccer

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Modeling a Fantasy Football Team Using Machine Learning Algorithms and Linear Programming

Santiago Viteri Puyol, Alejandro Proaño Ph.D

***Abstract*—Building a Fantasy Premier League team requires analyzing player performance over time and optimizing the selection of those most likely to excel. This process demands a deep understanding of the game and player development, which is challenging due to the sport's inherent complexity. Machine learning techniques offer a potential solution to this decision-making problem. For example, supervised and unsupervised models like LSTM networks and K-means clustering capture temporal patterns and reveal hidden structures in player data.**

In this study, we apply and contrast the performance of two machine learning models. The first model uses LSTMs to forecast players' future points, combining these predictions with an optimization algorithm to select the optimal lineup for the next four game weeks. The second model applies a K-means algorithm to group players into clusters based on performance metrics. Player transfers are then evaluated based on the cluster they belong to and their ICT (Influence, Creativity, Threat) scores.

By comparing these two strategies—LSTM-based optimization and cluster-based player selection—we aim to determine which approach yields better outcomes regarding total points accumulated over the Premier League season.

I. Introduction

FANTASY football is a popular game where participants build virtual teams composed of real professional players, with their in-game statistical performances translating into fantasy points. Managers compete by assembling teams based on the actions of the selected players in actual matches. Like in real-world sports, fantasy football involves strategic decisions about player selection, transfers, and lineup changes.

A. Fantasy Premier League Rules

Fantasy Premier League (FPL) is structured to simulate the experience of managing a football team, with each manager tasked with assembling a squad of 15 players under a fixed budget of £100 million. The squad composition includes 2 goalkeepers, 5 defenders, 5 midfielders, and 3 forwards. However, only 11 players are active each game week, while the remaining 4 are substitutes. Each week, a lineup must feature one goalkeeper, and at least: three defenders, three midfielders, and one forward, with a maximum of 3 players from any real-life team.

Points are awarded based on player performances in real matches, as outlined in Table I. For example, they earn points for contributions such as goals, assists, clean sheets, and penalty saves, while points are deducted for red and yellow cards, own goals, and other events like missed penalties.

Table I: Point System of Fantasy Premier League (FPL)

Position	Category	Points
Goalkeepers	For playing up to 60 minutes	1
	For playing 60 minutes or more	1
	Each goal scored	6
	each assist	6
	Keeping a clean sheet	4
	Every 3 saves made	3
	Each penalty saved	3
	Penalty missed	-2
	Every two goals conceded	-1
	A yellow card	-1
	A red card	-3
	Own goal	-2
Defenders	Playing up to 60 minutes	1
	Playing 60 minutes or more	1
	Each goal scored	6
	Each assist	6
	Keeping a clean sheet	4
	Penalty missed	-2
	Every two goals conceded	-1
	A yellow card	-1
	A red card	-3
	Own goal	-2
Midfielders	Playing up to 60	2
	Playing 60 minutes or more	2
	Each goal scored	5
	Each assist	5
	Penalty missed	-2
	Every two goals conceded	-1
	A yellow card	-1
	A red card	-3
	Own goal	-2
Forwards	Playing up to 60 minutes	2
	Playing 60 minutes or more	2
	Each goal scored	4
	Each assist	4
	Each penalty missed	-2
	Every two goals conceded	-1
	A yellow card	-1
	A red card	-3
	Own goal	-2

Managers are permitted one free transfer each game week, with additional transfers costing 4 points each. Furthermore, FPL provides managers with strategic "chips" that can be used throughout the season:

- **Wildcard**: Allows unlimited transfers within one game week, usable twice per season.
- **Triple Captain**: Triples the score of the selected captain, usable once per season.
- **Bench Boost**: Activates points from all 15 players,

counting those on the bench as well, usable once per season.
- **Free Hit**: Allows managers to completely change their team for one game week before reverting to the previous lineup, usable once per season.

These rules introduce layers of strategy, as managers must consider both player form and matchups when making weekly lineup and transfer decisions.

B. Machine Learning for Fantasy Football

Machine learning techniques can provide valuable insights to assist in making effective fantasy football decisions. Long-short-term memory (LSTM) networks, widely used for time series analysis, excel in capturing temporal dependencies and recognizing long-term patterns. Unlike traditional statistical methods, which often struggle with non-linear data, LSTMs can model complex patterns by retaining memory over time. This ability has proven beneficial in sports analytics, finance, and weather forecasting [7].

At the same time, K-Means clustering has gained prominence in identifying natural clusters within datasets, proving especially useful for segmentation and classification tasks. K-Means effectively groups data points with similar characteristics, enabling insights that may not be immediately apparent. This method has been applied across domains like marketing, biology, and sports analytics to create more targeted strategies and enhance decision-making [8, 9].

As with other machine learning methods, the effectiveness of LSTMs and K-Means relies on several factors [10], including model architecture, feature selection, and data pre-processing. While LSTMs require substantial data to capture temporal trends accurately, K-Means is highly sensitive to the choice of features and the number of clusters. However, when implemented effectively, these models can provide deeper insights and more precise predictions than traditional methods, revolutionizing how we approach data-driven decision-making in time series analysis and segmentation tasks.

C. Objective of this Study

In light of this, fantasy football players, like regular football managers, require accurate forecasts of player performance and effective segmentation to build robust squads. In this paper, we contrast two models that rely on machine learning techniques—one focused on predicting total player points and the other on clustering players for selection and transfer decisions— to determine which enhances Fantasy Premier League team selection.

II. Related Work

Research in fantasy football team selection is still in its early stages, but significant advancements are being made due to the increasing availability of detailed player data. Bonomo et al. [10] were among the first to present two mathematical models related to fantasy football. Their research aligns with the broader trends in sports analytics, particularly inspired by the "Moneyball" approach, which emphasizes data-driven strategies to enhance team performance. The first model, utilizes predictions to select lineups, while the second model employs actual outcomes to identify the optimal squad.

Building on this foundation, Eilertsen [7] developed a forecast-based optimization model that used traditional statistical methods—such as recent player performance, regression on explanatory variables, and bookmakers' odds—to improve team selection strategies. Their rolling horizon heuristic allowed for weekly updates based on dynamic forecasts, while game chips were incorporated to assess their impact on specific game weeks.

Similarly, Beal Norman and Ramchun [8] demonstrated the potential of AI in optimizing fantasy sports teams from the NFL, showcasing how machine learning algorithms can improve decision-making for fantasy managers. Their work emphasized using three different methods combined with a mixed integer programming approach to search for the best selection of players across the NFL season.

More recently Bangdiwala et al. [5] utilized machine learning models to predict points in the Fantasy Premier League (FPL), comparing the effectiveness of linear regression, decision trees, and random forests. Their findings demonstrated a clear advantage of machine learning techniques over traditional regression models. However, a key limitation of their study was the absence of a proposed method for squad selection.

In its study, Gupta [1], aimed to forecast players' performance by employing a hybrid model combining ARIMA models and Recurrent Neural Networks (RNNs). The method predicted player points based on historical data from three FPL seasons. Linear Programming (LPP) was applied to maximize total points while considering constraints such as player types (goalkeepers, defenders, midfielders, and forwards) and budget limits. Nevertheless, Gupta's approach focused on optimizing a single team for the entire season, without considering player transfers or the strategic use of game chips that are key elements of the Fantasy Premier League.

Lombu et al. [13] applied LSTM to predict Fantasy Premier League (FPL) points, showcasing its ability to capture historical sequences and outperform CNN in predictive tasks. Additionally, Lindemann et al [12]. highlighted the ability of LSTMs and CNN-LSTMs to capture temporal dependencies.

Akhanli and Hennig explored in [14] the clustering of football players based on performance data from the 2014-15 season of eight European leagues. Their analysis resulted in two different clustering approaches: one that grouped players into major categories, and another that formed smaller clusters to identify players with similar profiles.

Similarly, Wijngaard [15] compared K-means and Expectation Maximization on match data from La Liga spanning 2004 to 2019. The analysis found that K-means outperformed Expectation Maximization, concluding that 8 clusters represented the best separability of player types.

A. Problem Statement

Despite the advancements, this study is the first to integrate an LSTM with an optimization model for the English Premier League (EPL). Additionally, we propose a pipeline that combines LSTM-based forecasting for player performance with clustering techniques to group players based on key metrics for further team selection.

III. Materials & Methodology

A. Dataset Description

The dataset in this study comprises historical player statistics from the 2016-2017 to 2023-2024 seasons sourced from Vaastav's GitHub repository [6]. For each season two datasets are retrieved:

1) Merged GW: The first database captures player statistics for each game across 56 columns. Key metrics include assists, goals scored, clean sheets, expected points (xP), bonus point system (bps), and total points. Playtime data provide insights into the players' abilities such as completed passes, minutes played and key passes. Advanced statistics like ict index and influence highlight a player's impact. Finally, transfer and value details (transfers in, transfers out, value) reflect market activity.

2) Players Raw: The second database focuses on player attributes throughout the season. It includes player-specific information like its full name, team, and position (element type). Additionally, cumulative performance statistics such as goals scored, assists, and clean sheets provide a comprehensive view of the player's contributions throughout the season.

The final dataset is created by combining "Merged GW" with "Players Raw" to ensure that player details such as name, element type, and team are included. A column indicating the season year is included.

B. Data Pre-processing

The final dataset is then organized into separate dictionaries for goalkeepers, defenders, midfielders, and forwards. Each player has an associated list of total points and ICT index scores, initialized to zero for all game weeks, to maintain consistency in the length of time series data across all players. These lists are later populated with actual points from the original dataset. Similarly, this process is applied to the value variable but is initialized to 4, since it represents the lowest possible value.

The first model, the LSTM-LP model, updates its predictions every four-game weeks. Eilerstein et al. [7] found that planning horizons between 3 and 5 weeks was acceptable, with 3 weeks being optimal. However, FPL forums recommend planning 4 to 6 weeks ahead when using forecasts. Consequently, we chose a 4-week time step, as the root mean square error (RMSE) showed minimal variation between 3-week and 4-week forecasts. For each update, the final dictionaries incorporate data from the 2016/17 season up to the recent game.

In contrast, the LSTM-K-Means, updates its predictions after every game, starting from the 2019/20 season. Unlike the LSTM-LP model, it does not use an optimization algorithm to select a team. Instead, it continuously evaluates players based on their ICT Index and clustering. These weekly updates allow the model to stay responsive to changes in player form and performance, even if it doesn't provide a structured, fully optimized team selection.

C. LSTM-LP Model

The methodology is shown in Figure 1. First, the initial input consists of a dictionary containing the points and values of each player. This data is processed using an LSTM model to predict the players' future points and values for the next four weeks. These predictions are then adapted to fit into a linear programming model, which selects the optimal squad for the next four game weeks. This process is repeated by updating the data with actual results, forecasting the players' new values and points for the subsequent four games, and reapplying the model until the season concludes. The following sections describe this pipeline in detail.

*1) **LSTM**:* Receives a time-series dictionary of fantasy football metrics, such as total points and player values, and processes them into sequences organized by a data module at a defined time step of one. These sequences are then fed to an LSTM that has one hidden layer with a size of 100, followed by a fully connected layer that maps the predictions to a real number.

*2) **Linear Programming Optimizer**:* Following the problem stated in [7], the linear programming optimizer is designed to maximize the total expected points of a fantasy football squad while adhering to constraints on team composition, budget, transfers, and the use of special "chips" (such as wildcard, free hit, bench boost, and triple captain). Below is a detailed explanation of the model components:

Sets:

- $\boldsymbol{T}$ - Gameweeks.
- $\boldsymbol{P}$ - Players.
- $\boldsymbol{C}$ - Teams.
- $\boldsymbol{L}$ - Substitution priorities, where 1 is first priority.

Key Constants and Parameters:

- **Expected Points per Gameweek ($pt(p, t)$):** Represents the expected points for player p in gameweek t.
- **Transfer Constant (i_t):** with a value $i_t \ll 1$ equal to .01.
- **Positional Adjustment Constants (κ_l):** These constants (κ_1, κ_2, and κ_3) are used to prioritize substitutions based on the order of players on the bench.

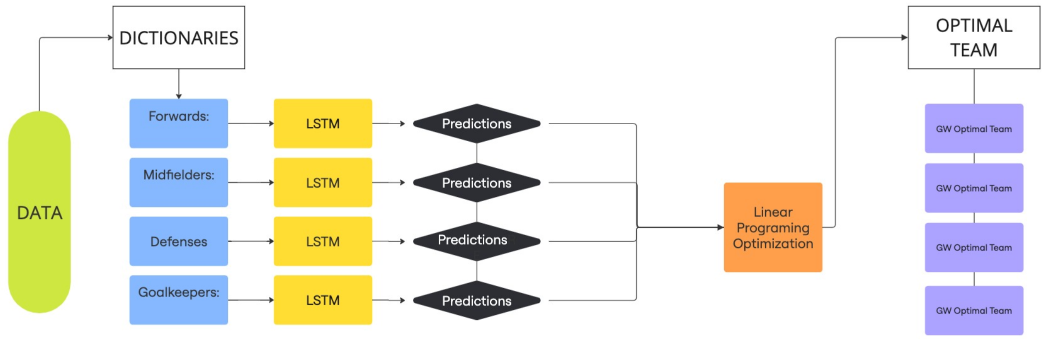

Figure 1: Process flow for the LSTM-LP model.

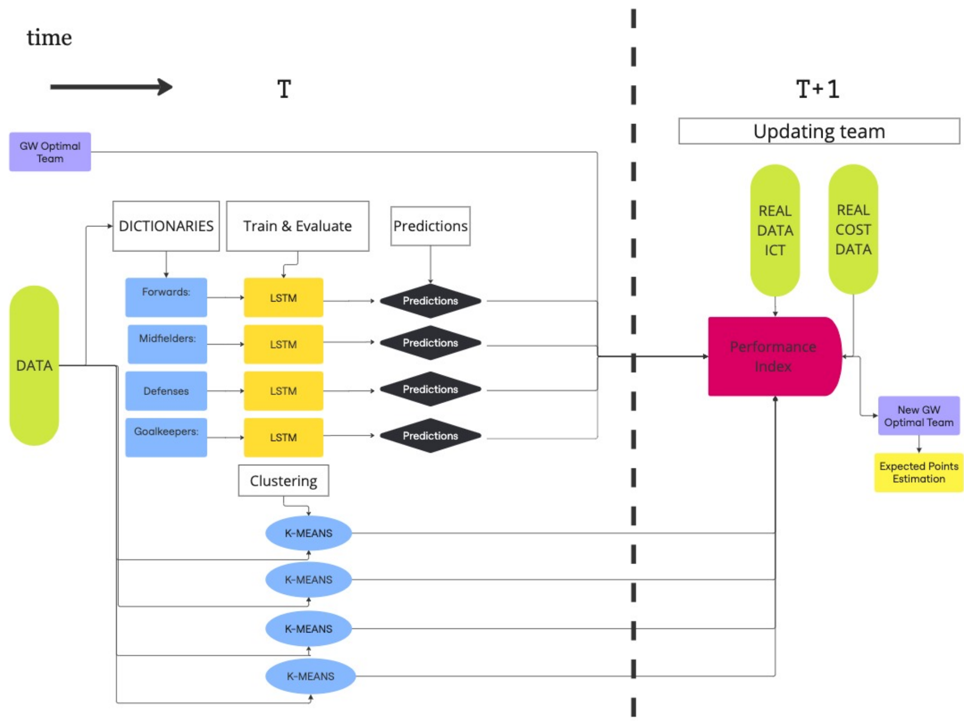

Figure 2: Process flow for the LSTM-K-Means model.

Here, $l \in \boldsymbol{L}$ represents the substitution priorities. The constants are set such that $\kappa_l \ll i_t$ for all κ_l, ensuring they only influence the decision when a substitution is needed. Additionally, $\kappa_1 > \kappa_2 > \kappa_3$, reflecting the priority order of the substitutes.
For example, if the substitutes on the bench are Haaland, Caicedo, and Díaz, with Haaland having the highest expected points followed by Caicedo and Díaz, then:

- if κ_1 is assigned to Haaland, he is the first-priority substitute (e.g., replacing an injured forward).
- if κ_2 is assigned to Díaz, as he is the second-priority substitute.
- if κ_3 is assigned to Caicedo, as he is the third-priority substitute.

- **Number of Players:** Constants G, D, M, and F specify the number of goalkeepers ($G = 2$), defenders ($D = 5$), midfielders ($M = 5$), and forwards ($F = 3$) required in the squad.
- **Club Limit (MC):** Maximum number of players allowed from the same club ($MC = 3$).
- **Starting Lineup Rules (E, EK, ED, EM, EF):** Define total players in the starting lineup ($E = 11$) and requirements for each position such as goalkeeper ($EK = 1$), defender ($ED \geq 3$), midfielder ($EM \geq 3$), and forward ($EF \geq 1$).
- **Cost/Budget Constraints (BS, CB, CS):** Define initial budget ($BS = 1000$), player sell value (CS) and player acquisition cost (CB), to ensure the squad remains within budget.

Decision Variables:

- **Player Selection ($xpt(p,t)$):** Binary variable to determine whether player p is selected in gameweek t.

- **Gameweek Selection ($ypt(p,t)$):** Binary variable indicating if player p is in the starting lineup for gameweek t.
- **Gamechip Activation ($wildcard$, $free-hit$, etc.):** Binary variables indicating if specific gamechips are activated in a gameweek.
- **Captain, Vice-Captain and Triple Captain (c, vc, tc):** Binary variables to assign captain and vice-captain roles and triple captain.
- **Number of Penalties** (α): Variable that counts the number of penalized transfers per game.

Objective Function: The model aims to maximize the total expected points across the season by summing the expected points of selected players ($pt(p,t)$) over four-week intervals (z), where the range of weeks is t_z to t_z+3. [1] The function presented below is optimized using the GLPK solver, which applies the Simplex method for linear programming (LP) and a branch-and-bound algorithm with cutting planes for mixed integer programming (MIP) problems.

$$tpt_k = \sum_{t=t_z}^{t_k+3} \sum_{p\in P} (pt(p,t)\cdot(ypt(p,t)+c(p,t)+i_t\cdot vc(p,t)+2\cdot tc(p,t)))$$

$$\max_{p\in P}(tpt_k + \sum_{t=t_z}^{t_z+3}\sum_{p\in P}\sum_{l\in L} pt(p,t)\cdot\kappa_l\cdot gptl(p,t,l) - \sum_{t\in T} R\cdot\alpha(t))$$

Where:

- t_z: Initial week of the current interval z.
- t_z+3: Final week of interval z.
- $pt(p,t)$: Expected points for player p in week t.
- $ypt(p,t)$, $c(p,t)$, $vc(p,t)$, $tc(p,t)$: Binary variables for selecting players in the starting lineup, captaincy, vice-captaincy, and triple captaincy, respectively.
- κ_l: Constant adjusting points based on substitution priority.
- $gptl(p,t,l)$: Points adjusted according to substitution priority.
- $R\cdot\alpha(t)$: Penalty for additional transfers in week t.

Constraints: Constraints ensure the optimizer adheres to fantasy football game rules, including squad composition, budget, and weekly transfers. Key constraints are as follows, and are applied independently to each interval z:

- **Gamechip Constraints:** Ensure that each chip (e.g., wildcard, free hit) can only be used once per season, with no chips used in gameweek 1.
- **Squad Composition:** The squad must include exactly 15 players, specifically 2 goalkeepers, 5 defenders, 5 midfielders, and 3 forwards.
- **Starting Lineup:** Requires exactly 11 players in the starting lineup each gameweek, with position-specific minimums for defenders, midfielders, and forwards.
- **Budget Constraints:** Enforces that the total cost of selected players remains within budget.
- **Transfer Constraints:** Applies transfer limits and penalizes excess transfers.

Examples of Constraints:

- **Gamechip Activation:** The constraint below ensures that the wildcard chip is activated only once for the first half of the season and once for the second half:

$$\text{Wildcard}(t) = \begin{cases} 0, & \text{if } t = t_1, \\ \text{Binary variable (0 or 1)}, & \text{if } t > t_z, \end{cases}$$

Subject to:

$$\sum_{t=t_z}^{t_z+3} \text{Wildcard}(t) \leq 1 \quad \text{if} \quad \forall z \in \{1,2,3,4,5\},$$

and

$$\sum_{t=t_z+1}^{t_z+3} \text{Wildcard}(t) \leq 1, \quad \forall k > 5.$$

where:
– t_z: Initial week of the interval z.
– t_z+3: Final week of the interval z.

- **Squad Composition:** Ensures exactly 15 players are in the squad:

selected squad constraints rule 1:

$$\sum_{p\in P} xpt(p,t) = 15 \quad \forall t\in[t_z, t_z+3]$$

- **Starting Lineup:** Ensures 11 players in the starting lineup (15 with bench boost):

starting lineup constraints rule 1:

$$\sum_{p\in P} ypt(p,t) = E + 4\cdot\text{bench-boost}(t) \quad \forall t\in[t_z,t_z+3]$$

- **Budget Constraints:** Ensures the squad remains within budget:

budget constraints rule 1:

$$BS - \sum_{p\in P} CB(p,1)\cdot xpt(p,1) = vt(1)$$

budget constraints rule 2:

$$vt(t-1) + \sum_{p\in \boldsymbol{P}} CS(p,t)\cdot upt(p,t) - \sum_{p\in \boldsymbol{P}} CB(p,t)\cdot ept(p,t) = vt(t), \quad \forall t\in[t_z,t_z+3]$$

budget constraints rule 3:

$$xpt(p,t-1) + ept(p,t) - upt(p,t) = xpt(p,t), \quad \forall p\in P, \forall t\in[t_z,t_z+3]$$

budget constraints rule 4:

$$ept(p,t) + upt(p,t) \leq 1, \quad \forall p\in P, \forall t\in[t_z,t_z+3]$$

[1] An example of the intervals is as follows:
- Interval 1 ($z=1$): Weeks $t=1$ to $t=4$.
- Interval 2 ($z=2$): Weeks $t=5$ to $t=8$.
- Interval 3 ($z=3$): Weeks $t=9$ to $t=12$.
- And so on until the end of the season.

D. LSTM-K-Means Model

The second model, shown in Figure 2, begins with data processing. Then, the LSTM model predicts each player's ICT index. The players are clustered using k-means, which are expected to be grouped according to their performance level. The initial team selection follows the same choices as in the first model. At the end of each game week, the predicted ICT index is compared to the actual values, and players are replaced based on the minimum relative error, calculated as:

$$\text{Relative Error} = \frac{\text{ICT}_{\text{real}} - \text{ICT}_{\text{predicted}}}{\text{ICT}_{\text{real}}} \tag{1}$$

In the case of a tie, the player from the lowest cluster is chosen for replacement. Replacement players are selected based on equal or lower value while belonging to the highest cluster possible. This process is repeated weekly until the season concludes. A detailed illustration is shown in Algorithm 1.

Algorithm 1 LSTM-K-Means Team Selection

Begin Procedure
Initialize: Initial team selection based on Model 1 gameweek 1 choice.
for each game week **do**
At time t: Before a game occurs.

- **Input:** Processed data into their dictionaries for each game week
- **Step 1:** Predict ICT index for each player at the start of the game week using the LSTM model.
- **Step 2:** Cluster players based on their most important features during the game week using k-means.

At time t+1: After the game:

- **Step 3:** Compare the predicted ICT index with actual values and calculate the **Relative Error** for each player (using Equation 1).
- **Step 4:** Replace players based on the minimum **Relative Error**.
  - **If there is a tie in Relative Error:**
    - Choose the player from the lowest cluster for replacement.
- **Step 5:** Update the team by choosing a replacement player based on:
  - Equal or lower value.
  - Belonging to the highest possible cluster.

end for
End Procedure =0

*1) **LSTM**:* The model begins by organizing the ICT index dictionary into sequences with a time step of one using a data module. These sequences are then processed by an LSTM, which includes a hidden layer of 100 units and a fully connected layer that outputs predictions as real numbers.

*2) **K-Means**:* This pipeline begins by dividing the received dataset into four, one for each player position. Missing values are handled, and relevant features for clustering are selected. The data is then standardized to have a zero mean and unit variance before being fed into the clustering algorithm. The optimal number of clusters is determined using two methods: the *Elbow Method*, which calculates inertia for various cluster numbers, and the *Silhouette Score*, which assesses cluster separation and compactness.

The scores from both methods are plotted, and the number of clusters is determined by averaging the results, as combining these approaches enhances clustering effectiveness [16]. This final cluster count is then used as the input parameter for the K-Means algorithm, which assigns cluster labels to each data point. These labels are added to the entry data, providing each player with a cluster identity that groups individuals with similar performance. For instance, a player like Julian Alvarez would be expected to be assigned to the same cluster as Erling Haaland since they exhibit comparable performance levels.

*3) **Expected Points**:* As seen in Figure 2 the model does not predict the total players point. Instead, the expected points for each player in a given week are estimated based on their position and the cluster to which they belong, as follows:

Sets:

- $\boldsymbol{P}$ - Set of players available for selection in a gameweek. Where:
  - $p_i \in \boldsymbol{P}$ - Represents each player within the set $\boldsymbol{P}$.
- $\boldsymbol{S} \subset \boldsymbol{P}$ - Set of players in the selected lineup for a game.
- $C_{\text{captain}} \subset \boldsymbol{P}$ - Set of players chosen as captain.
- $C_{\text{triple-captain}} \subset \boldsymbol{P}$ - Set of players selected with a triple captain boost.

Key Mappings:

- **Player Position** ($\text{pos}(p_i)$): Maps player p_i to their playing position (e.g., forward, midfielder, defender, goalkeeper).
- **Player Cluster** ($C(pos(p_i))$): Maps the player p_i to the cluster to which they belong based on their position. For example, if p_i is Moisés Caicedo, then $pos(p_i)$ is "midfielder," and $C(pos(p_i))$ represents the cluster assigned to Caicedo considering that he is a midfielder.
- **Player Expected Points** ($\mu_{C(pos(p_i))}$): Represents the mean predicted points for the cluster $C(pos(p_i))$ to which player p_i belongs. For example, if p_i is Moisés Caicedo, and he belongs to the cluster C_3 within the "midfielder" position, then $\mu_{C(pos(p_i))}$ corresponds to the mean predicted points for all players in cluster C_3 of midfielders.

Expected Points: The total expected points for a gameweek (GW) are calculated as follows:

$$\sum_{p_i \in S} \mu_{C(pos(p_i))} + \sum_{p_j \in C_{\text{captain}}} \mu_{C(pos(p_j))} + 2 \cdot \sum_{p_k \in C_{\text{triple-captain}}} \mu_{C(pos(p_k))}$$

where:

- $\mu_{C(pos(p_i))}$ - Mean predicted points for a player based on their cluster.
- $\mu_{C(pos(p_j))}$ - Additional mean points for captains, accounting for the doubling of their points.
- $2\mu_{C(pos(p_k))}$ - Additional mean points for players with a triple captain boost (double the captain's bonus).

E. Experimental Setup

Metric	Maximun number of Epochs
Total Points	100
Value	800
ICT	200

Table II: Metrics and corresponding number of epochs

1) The datasets for the LSTM model are divided into training and validation sets, with 99% of earlier data allocated for training and the remaining 1% for validation. The model is optimized using the Adam optimizer, with a learning rate of 0.001 and a batch size of 32. Table II presents the maximum number of training epochs for each model. To prevent over-fitting we apply early stopping on validation loss with patience of 5 epochs. Model checkpoints are saved to retrieve the best-performing model. [2]

2) The linear programming optimizer follows the same setup described by Eilersten in [7]. The model is built using the Pyomo library, and the GLPK solver is employed with a maximum execution time of 1000 seconds.

3) For the clustering, feature selection is applied to identify three relevant features for each dataset type, considering a maximum of seven clusters.

The LSTM forecasts were run using a T4 NVIDIA GPU provided by Google Colab. The following GitHub repository contains detailed instructions on how to run both models.

- https://github.com/santiagoviteri01/final_mmia_project

F. Metrics

Similar to Bangdiwala et al. [5] and Beal et al. [8], the root mean square error (RMSE) is used as the evaluation metric for the LSTM models, as it quantifies the difference between predicted and actual statistics.

In fantasy football, which depends on predicting time-series statistics for players (such as total points or the ICT index), the RMSE measures how well the model captures the underlying patterns in the data. A lower RMSE indicates that the model's predictions closely align with actual performance. Consequently leading to improved team optimization due to an accurate player selection and better evaluation of player momentum.

[2]The decision to use only 1% for validation, follows Makriddakis in [2], it allows more recent data to be included in the training set. Since recent data is crucial for capturing seasonality and trends, not considering it in the training process would diminish the accuracy of the model's predictions. By keeping the validation set small, the model uses the most historical data to detect recurring patterns, while still leveraging early stopping to prevent overfitting and ensure generalization.

IV. Results

A. LSTM-LP

Metric	Root Mean Squared Error (RMSE)
Value	6.8254
Total Points	1.3065

Table III: Root Mean Squared Error (RMSE) for Value and Total Points

*1) **RMSE**:* The RMSE for the total points and value is shown in table III. In this analysis, we calculate the accuracy of the predictions against their actual values. The forecasts are iterative compared to the real outcomes over different periods (or game weeks). The RMSE obtained for Value is 6.8254 and for Total Points is 1.3065.

Gameweek	Starting Players	Captain	Substitutes
1	Gabriel Jesus-4, Erling Haaland-4, James Ward-Prowse-3, Mohamed Salah-3, Bukayo Saka-3, Jack Harrison-3, Martin Odegaard-3, Max Kilman-2, Rico Henry-2, Sven Botman-2, Alisson Becker-1	Erling Haaland-4	Brennan Johnson-4, Joel Veltman-2, Ethan Pinnock-2, Jordan Pickford-1
19	Ollie Watkins-4, Erling Haaland-4, Mohamed Salah-3, Bukayo Saka-3, Douglas Luiz-3, Bruno Fernandes-3, Son Heung-min-3, Kieran Trippier-2, Trent Alexander-Arnold-2, Dan Burn-2, David Raya-1	Erling Haaland-4	Julian Alvarez-4, Joachim Andersen-2, Marc Guehi-2, Alisson Becker-1
37	Jarrod Bowen-3, Ollie Watkins-4, Erling Haaland-4, Phil Foden-3, Bukayo Saka-3, Cole Palmer-3, Son Heung-min-3, Kieran Trippier-2, Virgil van Dijk-2, William Saliba-2, Jordan Pickford-1	Bukayo Saka-3	Julian Alvarez-4, Trent Alexander-Arnold-2, Benjamin White-2, Bernd Leno-1

Table IV: LSTM-LP Summary of Selected Players, Captains, and Substitutes for Gameweeks 1, 19, and 37

*2) **Selected Players**:* Table IV briefly summarises the selected starting players, captains, and substitutes for Gameweeks 1, 19, and 37. The starting lineup shows the players with their respective positions. Erling Haaland (for all gameweeks) was the most frequently chosen captain, follow by Mohamed Salah and Ollie Watkins.

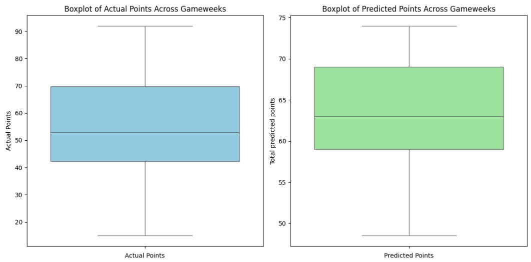

Figure 3: Boxplot of predicted vs actual points from the second model squad.

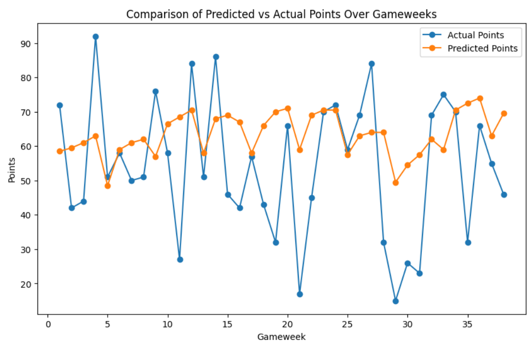

Figure 4: First Model Squad: predicted vs actual points.

*3) **Expected vs. Actual Team Results:*** Figures 3 and 4 present the boxplot and time series for the predicted and actual points. The graphs highlight a discrepancy between expected and actual outcomes, with the average predicted points at 63.4, compared to a mean of 54.4 for the actual points. This indicates that the model tends to overestimate player performance on average. Additionally, the distribution of actual points is wider, ranging from 15 to 92 points, while the predicted points are constrained to a narrower range of 48.5 to 74 points. This difference, particularly evident in Figure 4 -where the forecast series demonstrates lower variability- suggests that the model struggles to account for the full range of potential outcomes. Such limitations arise from the inherent unpredictability of player performance, where factors like injuries, team dynamics, or unexpected match conditions introduce uncertainty.

B. LSTM-K-Means

Metric	Root Mean Squared Error (RMSE)
ICT Index	1.6098

Table V: Root Mean Squared Error (RMSE) for ICT Index

*1) **RMSE:*** The RMSE for the results of the LSTM applied to ICT is shown in table V. The predictions are compared to the real outcomes at the end of each week, leading to a final RMSE of 1.6098.

*2) **Feature Selection:*** Table VI summarizes the features selected for each player position (forwards, midfielders, defenders, and goalkeepers). These features are the three most relevant in determining player performance and are consistent across all 37 game weeks of the season.

Position	Selected Features
Forwards	'bps', 'ict_index', 'influence'
Midfielders	bps', 'ict_index', 'influence'
Defenders	'xP', 'bps', 'clean_sheets'
Goalkeepers	'xP', 'bps', 'minutes'

Table VI: Selected Features for Each Position

The feature bps (Bonus Points System) is found to be relevant across all positions. Forwards and midfielders relied on influence and the ICT index. On the other hand, expected points (xP) played a significant role in defensive positions, such as defenders and goalkeepers. Finnaly, forwards, and midfielders shared the same feature set, while defenders and goalkeepers had two common features.

*3) **K-Means:*** Clustering was applied to group players based on their performance metrics. Across the entire season, each position has four clusters. This clustering helps identify distinct performance groups within each position, providing insights into player categorization and trends.

Position	Cluster	Parameter	GW 1	GW 19	GW 37
Forwards	**Cluster 0**	**mean**	0.29	0.19	0.22
		std	0.44	0.30	0.33
	Cluster 1	**mean**	2.05	1.70	1.81
		std	0.55	0.43	0.43
	Cluster 2	**mean**	3.82	2.99	3.10
		std	0.88	0.47	0.49
	Cluster 3	**mean**	7.38	5.20	5.11
		std	1.15	0.96	0.98
Midfielders	**Cluster 0**	**mean**	0.22	0.20	0.17
		std	0.35	0.29	0.25
	Cluster 1	**mean**	1.69	1.56	1.40
		std	0.52	0.45	0.40
	Cluster 2	**mean**	3.09	2.90	2.68
		std	0.72	0.54	0.54
	Cluster 3	**mean**	6.01	4.63	4.48
		std	1.79	0.95	0.87
Defenders	**Cluster 0**	**mean**	0.08	0.10	0.10
		std	0.17	0.18	0.18
	Cluster 1	**mean**	1.10	1.15	1.16
		std	0.38	0.32	0.32
	Cluster 2	**mean**	2.26	2.28	2.26
		std	0.56	0.42	0.41
	Cluster 3	**mean**	4.15	3.67	3.65
		std	1.13	0.59	0.60
Goalkeepers	**Cluster 0**	**mean**	0.03	0.04	0.04
		std	0.09	0.09	0.10
	Cluster 1	**mean**	0.26	0.92	0.99
		std	0.38	0.32	0.36
	Cluster 2	**mean**	2.24	2.63	2.43
		std	0.49	0.44	0.38
	Cluster 3	**mean**	3.94	3.59	3.61
		std	0.93	0.50	0.45

Table VII: Points Summary for Forwards, Midfielders, Defenders, and Goalkeepers for Clusters in Gameweek 1, Gameweek 19, and Gameweek 37

Table VII presents the clustering results for forwards, midfielders, defenders, and goalkeepers for weeks 1,19, and 37. Each cluster summarizes the main statistics of total points.

The cluster analysis reveals that different clusters have distinct means, with minimal overlap when considering one standard deviation. For example, for forwards, Cluster 0 in gameweek 1 has a mean of 0.29 and a standard deviation of 0.44, while Cluster 1 has a mean of 2.05 with a standard

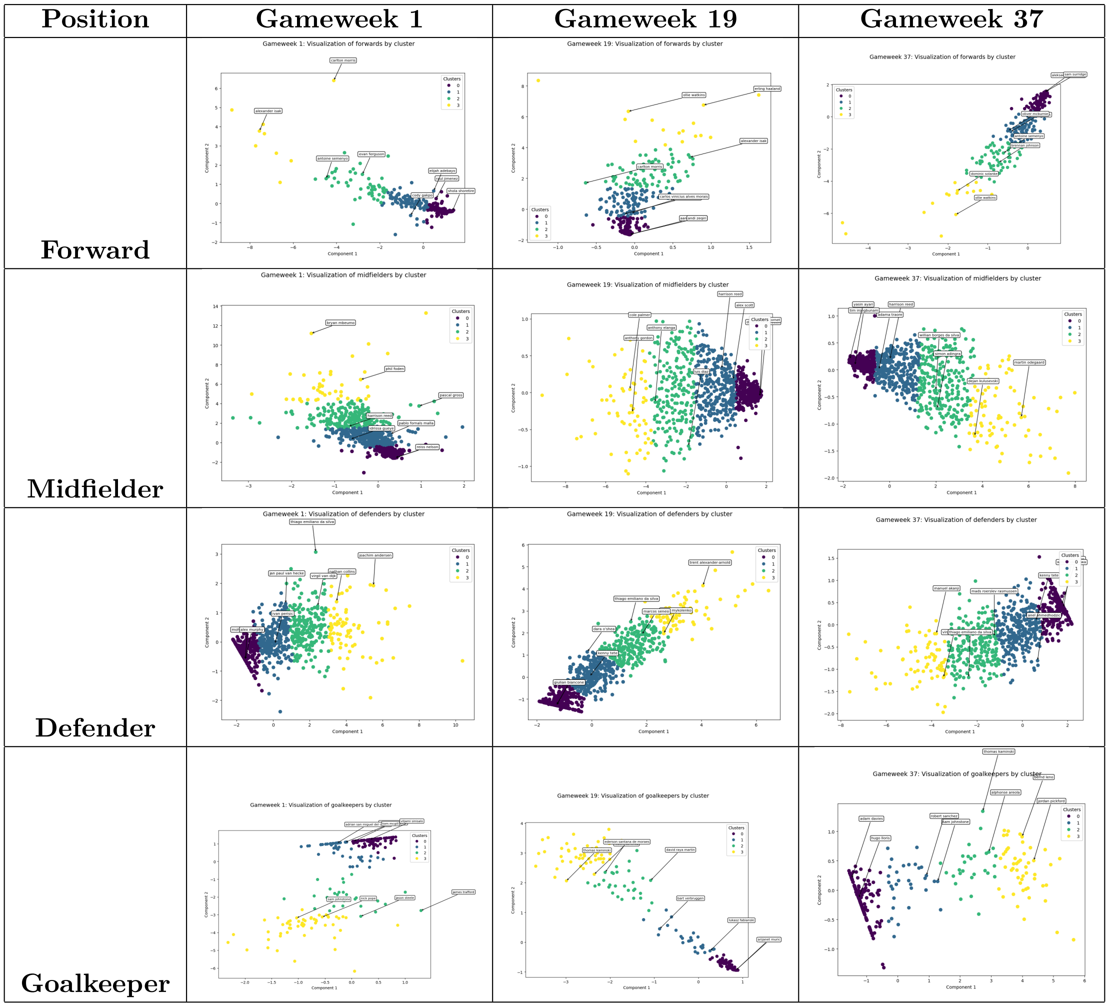

Figure 5: MDS Clustering results across different gameweeks

deviation of 0.55. Even when accounting for one standard deviation, the clusters tend to not overlap.

Changes in the cluster mean provide‘ insight into performance trends. Lower-performing clusters (e.g., Cluster 0 and 1) remain relatively stable. In contrast, the highest-performing clusters (e.g., Cluster 3 and 2) show a decline in their mean performance by Gameweek 37. For example, forwards in Cluster 3 drop from a mean of 7.38 in Gameweek 1 to 5.11 by Gameweek 37, indicating a performance decline.

The clustering analysis in Figure 5, reveals distinct performance trends across different player positions—Forwards, Midfielders, Defenders, and Goalkeepers—throughout Gameweeks 1, 19, and 37. The clusters are color-coded to represent varying performance levels: yellow (high performance), green (medium-high performance), blue (medium-low performance), and violet (low performance). Across all positions, there is a slight tendency for the number of high-performing players to increase as the season progresses. It is also evident that, throughout the season, low-performing players are the most numerous, followed by medium-low performers, medium-high performers, and high performers are the least represented.

*4) **Selected Players**:* The starting players, captains, and substitutes for Gameweeks 1, 19, and 37 is shown in VIII. It details the players chosen for each game along with their positions. Mohamed Salah was the most frequently selected captain throughout the season, with Rodrigo Muniz and Alexis Mac-Callister also commonly chosen.

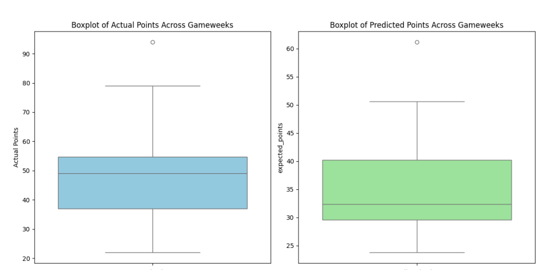

Figure 6: Boxplot of predicted vs actual points from the second model squad.

*5) **Expected vs. Actual Team Results**:* Figures 6 and 7 highlight significant limitations of the second model. Although the model captures general trends, it consistently underestimates points, as shown in Figure 7. This issue

Gameweek	Starting Players	Captain	Substitutes
1	Gabriel Jesus-4, Erling Haaland-4, James Ward-Prowse-3, Mohamed Salah-3, Bukayo Saka-3, Jack Harrison-3, Martin Odegaard-3, Max Kilman-2, Rico Henry-2, Sven Botman-2, Alisson Becker-1	Erling Haaland-4	Brennan Johnson-4, Joel Veltman-2, Ethan Pinnock-2, Jordan Pickford-1
19	Jason Steele-1, Ben Davies-2, Issa Kabore-2, Joel Veltman-2, Lewis Miley-3, Mohamed Salah-3, James Ward-Prowse-3, Elijah Adebayo-4, Dominic Calvert-Lewin-4, Brennan Johnson-4, Ethan Pinnock-2	Ben Davies-2	Max Kilman-2, Jordan Pickford-1, Wilson Odobert-3, Josh Brownhill-3
37	Thomas Kaminski-1, Max Kilman-2, Tyrick Mitchell-2, Teden Mengi-2, Moises Caicedo Corozo-3, Sander Berge-3, Alexis Mac Allister-3, Brennan Johnson-4, Dominic Calvert-Lewin-4, James Ward-Prowse-3, Rodrigo Muniz Carvalho-4	Brennan Johnson-4	Jarrad Branthwaite-2, Martin Dubravka-1, Wilson Odobert-3, Calum Chambers-2

Table VIII: LSTM-K Means Summary of Selected Players, Captains, and Substitutes for Gameweeks 1, 19, and 37

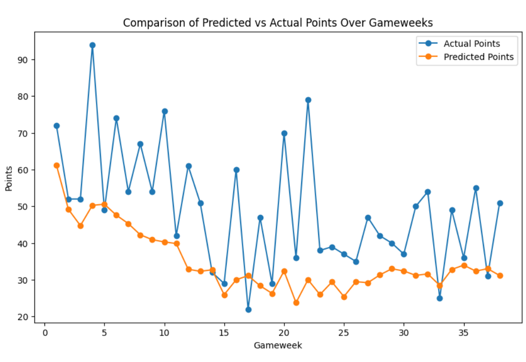

Figure 7: Second Team Squad: predicted vs actual points

arises from relying on cluster averages ($\mu_{C(pos(p_i))}$) to calculate expected points, which assumes uniform performance within each cluster and ignores individual variability. Consequently, the model struggles to account for high-or-low-performing outliers who deviate from the cluster mean. Another issue is the model's heavy reliance on the ICT index for all positions, which makes it less effective at identifying the best replacements. Since the ICT index is most relevant for forwards and midfielders, it doesn't work as well for defenders and goalkeepers. To improve, the model needs to better account for individual player differences and use metrics specific to each position.

V. Discussion

The two models developed for fantasy football team selection, LSTM-LP, and LSTM-K-means, showed distinct performance levels and strengths. Their effectiveness was assessed through the fantasy league rankings, each reflecting the models' contributions to team formulation and player performance prediction.

The predictions for the total points in the LSTM-LP model achieved a relatively low RMSE of 1.3065. This reflects the LSTM's strong capability in estimating players' performance over the season, which is critical for optimizing team selection. However, predicting player values proved more challenging, as evidenced by a higher RMSE of 6.8254, emphasizing the inherent difficulty in modeling market dynamics.

In practical applications, the LSTM-LP model achieved an average fantasy score ranking in the top 10% of global players during the evaluated seasons. Additionally, its results were comparable to Eilertsen's state-of-the-art model, reinforcing its robustness and adaptability across seasons.

The K-means pipeline effectively identified position-specific factors and seasonal performance trends through feature selection and clustering. For instance, forwards' total points were strongly influenced by ICT index metrics, while the top cluster across all positions exhibited a decline in mean points, suggesting that fatigue may impact top-performing players due to their higher workload.

Despite achieving a slightly lower average ranking—placing in the top 30%-40%—the LSTM-K-means model provided valuable insights into player attributes and performance trends.

Current Team: Building on the success of the LSTM-LP model, this pipeline was selected to construct the team for the 2024-2025 season. Up to Gameweek 8, the team has averaged 63 points per game, with highs exceeding 100 points, as illustrated in the following figure

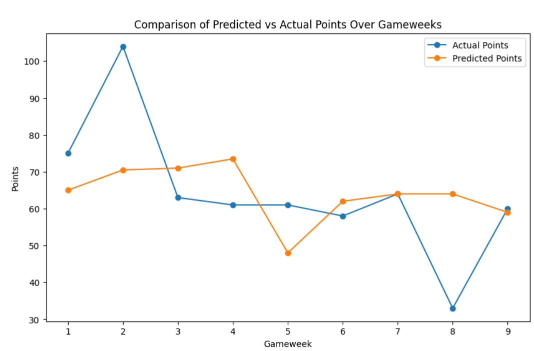

Figure 8: Current Team Squad: predicted vs actual points.

VI. Conclusion

The analysis demonstrates that the LSTM-LP model achieved state-of-the-art performance due to the LSTM neural network's effectiveness in predicting players' total points. This success can be attributed to its capacity to capture temporal patterns in player performance. However, future improvements could be made by integrating additional features to capture other traits of player volatility.

Clustering techniques provided valuable insights into seasonal performance trends. The results indicated that players in high-performing clusters tend to experience a noticeable decline toward the end of the season, likely due to fatigue. This suggests that incorporating fatigue as a feature, or developing models that account for a player's likelihood of being rested, could further enhance prediction accuracy and decision-making.

Finally, while the LSTM-K-Means model captures general trends, its performance remains slightly above average, underscoring the need for position-specific metrics to enhance its results. Since feature importance varies by player position, incorporating performance indices tailored to each position could enable the model to make more informed replacement decisions.

References

[1] Gupta, A. (2017). Time Series Modeling for Dream Team in Fantasy Premier League. International Conference on Sports Engineering ICSE-2017, 23-25 October 2017, Jaipur, India.

[2] S. Makridakis, R. M. Hogarth, and A. Gaba, "Forecasting and uncertainty in the economic and business world," *International Journal of Forecasting*, vol. 25, no. 4, pp. 794–812, 2009.

[3] B. Lindemann, T. Müller, H. Vietz, N. Jazdi, and M. Weyrich, "A survey on long short-term memory networks for time series prediction," *Journal of Time Series Prediction*, vol. 35, no. 4, pp. 1-23, 2021.

[4] A. Yadav, C. K. Jha, and A. Shara, "Optimizing LSTM for time series prediction in Indian stock market," *International Conference on Computational Intelligence and Data Science (ICCIDS 2019)*, 2019.

[5] M. Bangdiwala, R. Choudhari, A. Hegde, and A. Salunke, "Using ML Models to Predict Points in Fantasy Premier League," in *Proceedings of the 2022 Asian Conference on Intelligent Systems and Applications (ASIANCON)*, 2022, doi:

[6] V. Anand, "Fantasy-Premier-League," github.com. Available: ht tps://github.com/vaastav/Fantasy-Premier-League.

[7] W. Eilertsen, A. Gupta, and B. K. Kristiansen, "Developing a Forecast-Based Optimization Model for Fantasy Premier League," Industrial Economics and Technology Management, June 2018. Supervisor: M. Stålhane, Co-supervisor: X. Wang, Giovanni Pantuso, University of Copenhagen.

[8] R. Beal, T. J. Norman, and S. D. Ramchurn, "Optimising Daily Fantasy Sports Teams with Artificial Intelligence," *International Journal of Computer Science in Sport*, vol. 19, no. 2, 2020. doi: 10.2478/ijcss-2020-0008.

[9] G. Sugar and T. Swenson, "Predicting Optimal Game Day Fantasy Football Teams," Department of Aeronautics and Astronautics, Stanford University. Available: http://gsugar@stanford.edu, http://swens558@stanford.edu.

[10] Bonomo, Flavia & Duran, Guillermo & Marenco, Javier. (2013). Mathematical programming as a tool for virtual soccer coaches: A case study of a fantasy sport game. International Transactions in Operational Research. 21. 10.1111/itor.12068.

[11] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "The Performance of LSTM and BiLSTM in Forecasting Time Series," 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 2019, pp. 3285-3292.

[12] J. P. Lindemann et al., "Hybrid CNN-LSTM Architectures for Time Series Prediction," 2021 International Joint Conference on Neural Networks (IJCNN), 2021, pp. 1-8.

[13] A. S. Lombu, I. V. Paputungan, and C. K. Dewa, "Predicting Fantasy Premier League Points Using Convolutional Neural Network and Long Short Term Memory," Jurnal Teknik Informatika, vol. 5, no. 1, 2024, doi: https://doi.org/10.52436/1.jutif.2024.5.1.1792.

[14] Akhanli, S. E., & Hennig, C. (2023). Clustering of football players based on performance data and aggregated clustering validity indexes. Journal of Quantitative Analysis in Sports. https://doi.org/10.1515/jqas-2022-0037

[15] Wijngaard, G. (2023). Clustering soccer players: investigating unsupervised learning on player positions.

[16] K. Soufiane and M. T. Khadir, "A multiple clustering combination approach based on iterative voting process," Journal of King Saud University - Computer and Information Sciences, vol. 34, no. 1, pp. 1370-1380, Jan. 2022. doi: 10.1016/j.jksuci.2019.09.013.