ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DIPARTIMENTO DI
INGEGNERIA DELL'ENERGIA ELETTRICA E
DELL'INFORMAZIONE
"Guglielmo Marconi" (DEI)

CORSO DI LAUREA MAGISTRALE IN
AUTOMATION ENGINEERING

TESI DI LAUREA

in *Optimization models and algorithms*

Optimization and Machine Learning techniques: An application to the Fantasy Premier League

CANDIDATO
Lorenzo Santoro

RELATORE
Prof.ssa Valentina Cacchiani

CORRELATORE/I
Prof. Omar El Housni
Prof. Andrea Lodi

Contents

Abstract

Fantasy sports have become a fertile ground for applying advanced analytics and optimization techniques. This thesis explores the development of a machine learning and operations research pipeline for optimizing team selection in the Fantasy Premier League (FPL). Using historical data, we first train predictive models to estimate player performances. These predictions are then embedded in Mixed Integer Programming (MIP) formulations to select optimal squads under realistic FPL constraints. We further refine our approach by incorporating weekly updates, market dynamics, and captain selection strategies. The results demonstrate that combining predictive analytics with mathematical optimization significantly improves final season scores compared to heuristic or greedy approaches. This study showcases the synergy between data-driven modeling and decision-making under uncertainty, with potential applications extending beyond fantasy sports.

1 Introduction

Fantasy Premier League (FPL) is one of the most popular fantasy sports games in the world, with over 10 million active users. The game requires participants, called managers, to build and manage a squad of real-life English Premier League players throughout the season. Points are awarded based on actual player performances in real matches, and managers must make weekly decisions such as transfers, starting lineups, and captain selections. Due to strict constraints on budget, player roles, and club quotas, FPL represents a challenging combinatorial optimization problem under uncertainty.

The goal of this thesis is to design and evaluate algorithmic strategies for improving decision-making in FPL using a combination of machine learning (ML) for player performance prediction and Mixed Integer Programming (MIP) for team optimization. The final aim is to construct a data-driven pipeline that supports both strategic planning (e.g., selecting the initial squad) and tactical decisions (e.g., weekly transfers and lineup choices), with the objective of maximizing cumulative fantasy points across the season. We start by collecting and preprocessing historical FPL datasets that include match-level player statistics, prices, ownership levels, and fixture details. These data are then used to build predictive models that estimate expected fantasy points for each player, either on average (season-wide) or gameweek-specific. Feature engineering plays a crucial role, incorporating lag-based indicators of player form, opponent difficulty, home/away effects, and team context. To ensure time-consistent predictions, we apply time-series cross-validation and use gradient boosting regressors for modeling.

Once reliable projections are obtained, we formulate MIP problems to select optimal squads and lineups. The initial model focuses on selecting the best set of 15 players before the season starts, subject to budget and positional constraints. Subsequently, we develop more advanced formulations that handle weekly re-optimization: one version assumes the squad is fixed and selects the optimal starting eleven each gameweek, while another allows player transfers and budget updates over time. An even more advanced version introduces the captaincy rule, where one player's score is doubled each week, requiring a separate decision variable in the model.

Each algorithm is evaluated on the 2023–2024 Premier League season, using simulations that apply the optimization at each gameweek. Comparative results show that data-driven MIP-based approaches outperform greedy and heuristic baselines by a substantial margin. For instance, static greedy methods yield roughly 850 points, while our most advanced model with ML predictions, market transfers, and captain selection achieves over 2100 points, closing the gap toward the 2799 points of the season's global winner.

The contributions of this thesis are threefold:

- A scalable framework for predicting fantasy football performance using machine learning with engineered features and time-series methods.
- Several MIP formulations that encode FPL rules and constraints for optimal squad selection, updated weekly or globally.
- Empirical validation of the proposed algorithms on real data, showing the effectiveness of combining predictive and prescriptive analytics.

In conclusion, this work demonstrates that a hybrid ML+MIP approach enables more rational, evidence-based fantasy football decision-making. Moreover, the methodology developed here could be generalized to other fantasy sports platforms or broader domains involving sequential resource allocation under uncertainty.

2 State of the Art: Machine Learning and Optimization in Fantasy Sports

Fantasy sports provide a rich context for research at the intersection of machine learning, behavioral science, and operations research. With the rise of platforms such as Fantasy Premier League (FPL), DraftKings, and FanDuel, scholars have investigated optimal team selection, behavioral patterns of participants, and the integration of predictive and prescriptive analytics to improve performance. This section reviews the literature across three key areas: performance modeling, behavioral biases, and optimization under uncertainty.

2.1 Performance Modeling and Player Evaluation

One of the foundational tasks in fantasy sports is to accurately assess the performance of individual athletes. Singal, Duru, and Reeb [1] explore the role of opponents in shaping fantasy performance, arguing that contextualizing a player's output by considering the strength of the opposing team leads to more accurate valuation. Their model, based on empirical data from the National Football League (NFL), integrates statistical features of opponents into predictive frameworks, thereby improving the evaluation process beyond traditional aggregate statistics. Similarly, Venter and Vuuren [2] apply performance modeling techniques to the Fantasy Premier League. They show that top-performing users leverage consistent heuristics when selecting players, often focusing on statistics such as recent form, upcoming fixtures, and market value. Their findings suggest that quantitative strategies outperform purely intuitive ones and emphasize the relevance of structured decision-making aided by data-driven tools. More recently, Gonzalez, Navarro, and Rodriguez [3] introduced individualized regression-based models to predict NBA player fantasy performance. By training specific models for each athlete and validating them with a two-stage process, the authors achieved high prediction accuracy and showed how player-specific forecasting can be integrated into lineup optimization. Additionally, Gurobi Optimization [4] demonstrated a practical framework combining machine learning (ML) for player projection with linear programming (LP) to recommend optimized lineups in fantasy basketball. Their work highlights the importance of end-to-end pipelines where predictive models directly feed into decision-making under constraints.

2.2 Behavioral Biases and Heuristic Approaches

Beyond technical modeling, several studies highlight the cognitive biases consistently exhibited by fantasy sports players, which can lead to systematic deviations from optimal decision-making. Bergman and Roychoudhury [5] identify a widespread behavioral pitfall termed *correlation neglect*, where users underestimate or ignore the statistical dependencies between players. A common manifestation of this is the tendency to select multiple players from the same real-world team, such as pairing a striker and midfielder from an offensively strong club, without accounting for the increased variance this introduces. If the team underperforms in a given Gameweek, all correlated players are likely to underperform simultaneously, compounding the loss. Their analysis shows that managers who diversify their player pool, analogous to diversifying an investment portfolio, tend to achieve more stable returns across the season.

This insight motivates the application of portfolio theory to fantasy sports, where player selection is reframed as a risk-return optimization problem. Rather than merely maximizing expected points, managers should consider the covariance structure among players, seeking lineups that balance upside potential with outcome robustness.

In parallel, Johnson and Patel [6] investigate a different set of behavioral distortions: confirmation bias, recency bias, and endowment bias. Their results, based on experimental setups with college students, showed that many participants tend to ignore updated statistical evidence in favor of emotional or historical attachments to players. For instance, recency bias leads users to irrationally favor players who have performed well in recent Gameweeks, even when upcoming fixtures or form indicators suggest otherwise. Similarly, endowment bias is reflected in the reluctance to transfer out underperforming players simply because they were part of the original squad, regardless of declining returns. These findings underscore the importance of integrating behavioral awareness into decision-support tools, which could help flag irrational decisions and promote evidence-based team management.

Another notable contribution by Kozlov and Wang [7] explores decision-making in the context of fantasy hockey (NHL). The study revealed that under time pressure or unfamiliar scenarios, users frequently rely on simplified heuristics such as selecting players from their favorite teams or choosing the highest scorers from the previous round, without accounting for contextual variables like fixture difficulty or player rotation risk. Such behavior exemplifies *availability bias* and overreliance on salient but potentially misleading cues.

Together, these findings illustrate that fantasy managers are not purely rational agents; instead, they are influenced by a range of cognitive shortcuts and emotional attachments. This opens opportunities for the development of decision-support systems that not only optimize under formal constraints, but also assist in mitigating cognitive biases, enhancing decision quality both algorithmically and psychologically.

2.3 Optimization under Uncertainty

Optimization models are essential for guiding decisions under the constraints typical of fantasy leagues (e.g., salary caps, team quotas, position requirements). While the reviewed papers focus more on modeling and behavior, they provide the theoretical basis for employing Mixed-Integer Programming (MIP), stochastic programming, and machine learning techniques to construct optimal fantasy lineups. Moreover, the use of predictive models to forecast player points (via regression, tree-based ensembles, or neural networks) feeds directly into the optimization pipeline, enabling the selection of the highest expected value team given the constraints. This synergy between prediction and prescriptive analytics is a growing area of interest in recent literature, opening the door for advanced ML/OR frameworks in fantasy sports.

Smith and Lee [8] proposed a hybrid method combining a neural network for performance prediction with an integer programming model to solve team selection problems for NFL fantasy contests. The paper demonstrates how such hybrid systems outperform greedy algorithms and manual strategies in multiple simulated tournaments. Furthermore, Deshmukh, Mehta, and Agarwal [9] employed reinforcement learning to model the decision process in fantasy cricket. Treating team selection as a sequential decision-making problem, their system adaptively updates player choices week by week using historical reward signals (i.e., past fantasy points). These methods illustrate how uncertainty can be addressed using a combination of ML-based forecasting, adaptive decision rules, and optimization models that consider long-term utility rather than single-round gains. It is worth highlighting that, concerning the algorithms developed in this thesis, uncertainty is captured entirely within the *objective function*, through the predicted player performances, while the constraints of the MIP model (e.g., squad size, budget, positional balance, club limits) remain deterministic and strictly enforced. This simplifies the formulation compared to full stochastic or robust optimization models, where uncertainty may also affect the constraints (e.g., player availability, budget volatility). By isolating the stochastic component in the objective, we maintain computational tractability while still accounting for the inherent unpredictability of player outcomes.

2.4 Summary

The recent literature reflects a rich interplay between statistical modeling, behavioral analysis, and optimization. Key contributions underscore the importance of contextual performance evaluation [1], the recognition and correction of cognitive biases [5], and the use of structured heuristics informed by data [2]. The integration of individualized forecasting [3], cognitive bias assessment [6], and prescriptive analytics frameworks [8, 9] broadens the scope of analytical research in this domain.

Future research will likely expand upon these foundations by integrating reinforcement learning, Bayesian optimization, and real-time decision systems tailored to dynamic fantasy sports environments. In particular, the development of transparent and explainable AI systems capable of justifying player recommendations may play a crucial role in bridging human intuition and algorithmic precision.

3 FPL rules

This section provides a brief overview of the Fantasy Premier League (FPL) rules ([2], [10]), which are essential for understanding how player scoring is calculated. After detailing the composition of the initial squad managed by participants (referred to as Managers), the rules governing player transfers into and out of the squad are summarized, the concept of FPL chips is explained and finally the deadline and players' scoring are reviewed.

3.1 Initial squad selection

Managers are required to select an initial FPL squad of 15 players (two goalkeepers, five defenders, five midfielders, and three forwards) from all EPL(English Premier League) players in the actual season (a pool of around 530 players). We will define those requirements as "role requirements" defined as r_d ($d \in \{\text{GK, DEF, MID, FWD}\}$):

- **Goalkeepers needed:** $r_{\text{GK}} = 2$
- **Defenders needed:** $r_{\text{DEF}} = 5$
- **Midfielders needed:** $r_{\text{MID}} = 5$
- **Forwards needed:** $r_{\text{FWD}} = 3$

Each player has a monetary value (fantasy currency) associated with him that typically ranges between £4 fantasy million and £14 fantasy million. The total value of the 15 players in the manager's initial squad may not exceed £100 fantasy million. In addition, managers may select only up to three players from any one of the twenty EPL teams. From their starting 11, participants nominate a captain and a vice-captain. The captain's score will be doubled. If a captain plays 0 minutes in the Gameweek, the captain will be changed to the vice-captain. If both captain and vice-captain play 0 minutes in a Gameweek, then no player's score will be doubled.

3.2 Squad composition

Among the fifteen players selected for the FPL squad, eleven players must be selected before each game week (GW) deadline, which typically is set 90 minutes before the start time of the first GW (EPL) match, to form a manager's FPL starting-eleven squad. All manager points for the GW are scored by these eleven players. In the case where some of the starting-eleven squad players do not play (in the EPL) during that GW, they will automatically be substituted by the remaining four players not included in the starting squad. Based on priorities specified by the manager and playing position, automatic substitutions are processed at the end of the GW, as follows:

- If the starting-eleven goalkeeper does not play during the GW, he is substituted by the replacement goalkeeper (if the replacement goalkeeper played during the GW).
- If any of the outfield players (who are not goalkeepers) do not play during the GW, they are substituted by the highest-priority outfield substitute who played during the GW, provided that this does not violate the squad formation rules. These rules specify that the starting-eleven squad can play in any formation, provided that one goalkeeper, at least three defenders, and at least one forward are selected at all times.

There are some constraints on the maximum and minimum number of players per role, we define those as max_d, min_d :maximum number of players allowed for role d (At most 1 GK, 5 DEF, 5 MID, 3 FWD), minimum number of players per role d (At min 1 GK, 3 DEF, 2 MID, 1 FWD). The manager is also required to select a captain and a vice-captain among the starting- eleven squad. The captain's score is doubled for the particular GW. In the case where the captain plays zero minutes during the GW, the vice-captain assumes the role of captain. If both the captain and the vice-captain play zero minutes during a GW, then no player's score is doubled.

3.3 Squad transfers

After the manager has selected an initial squad, he is allowed to change players in the squad by buying and selling players via a so-called transfer market. The player selling price may be less than the player's current purchase price as a sell-on fee of 50 percent (rounded up to the nearest £0.1m) will be applied on any profits made on that player. Unlimited transfers may be performed at no cost, until the first GW deadline. After the first GW deadline managers receive one free transfer per GW. In the case where a manager decides not to use the free transfer, an additional free transfer is allowed during the following GW. If a free transfer is not used in the following game, it will be carried over to the next one, and so on, until it is utilized. Four points are deducted from the manager's total score for each additional transfer. The maximum number of free transfers that a manager store in any gameweek is 5. Player costs change during the FPL season based on the popularity of the players in the transfer market. If many managers include a particular player in their squads, the player's popularity and hence his cost increases.

3.4 Fantasy Premier League Chips

In Fantasy Premier League (FPL), a *Chip* is a special advantage or boost that managers can use to improve their team scores and standings. Every manager has a limited set of chips each season; careful and strategic usage can significantly impact overall rank.

Types of Chips and Their Usage:

Wildcard: Allows unlimited transfers within the team's budget for a single Gameweek without any points deduction. Managers typically use it to restructure their squad entirely.

Triple Captain: Triples the points of the selected captain (traditionally doubled) for one Gameweek. Managers usually employ this on a player expected to score highly in a favorable fixture.

Bench Boost: Managers receive points from all squad players, including those on the bench, for a single Gameweek. This is typically activated when the entire squad has strong fixtures and expected performances.

Free Hit: Allows unlimited free transfers for a single Gameweek, reverting the squad to its original state the following Gameweek. This is ideal for temporarily navigating difficult or blank Gameweeks.

3.5 Fantasy Premier League Deadlines

The *FPL Deadline* refers to the fixed cut-off time by which managers must finalize all transfers, team selections, captaincy choices, and chip activations for each specific Gameweek.

Deadline Timing:

Typically, the deadline is set at 90 minutes before the scheduled kick-off time of the **first fixture of a Gameweek**. After this deadline, the game enters a lockout mode, and no further changes are allowed for the active Gameweek.

Implications of Missing the Deadline:

If a manager misses the deadline:

- Transfers made after the deadline will take effect from the next Gameweek.
- The current team configuration (players, bench order, captain, and vice-captain) from the previous selection will apply automatically.
- Managers cannot activate or deactivate chips for the active Gameweek after the deadline passes.

3.6 Fantasy Premier League Scoring System

In Fantasy Premier League, players earn points based on their real-life match performance. The scoring rules depend on player positions and actions performed during Premier League matches.

Points Assignment Overview:

The points awarded depend primarily on four player categories: **Goalkeepers**(GK), **Defenders**(DEF), **Midfielders**(MID), and **Forwards**(FWD). Points are calculated per match as shown in the table below:

Table 1: Fantasy Football Points by Role

Action	GK	DEF	MID	FWD
Playing up to 60 minutes	1	1	1	1
Playing 60 minutes or more (excluding stoppage time)	2	2	2	2
Goal scored	10	6	5	4
Goal assist	3	3	3	3
Clean sheet	4	4	1	0
Every 3 shots saved	1	-	-	-
Penalty save	5	-	-	-
Penalty miss	-2	-2	-2	-2
Bonus points (best players per match)	1 to 3	1 to 3	1 to 3	1 to 3
Every 2 goals conceded	-1	-1	0	0
Yellow card	-1	-1	-1	-1
Red card	-3	-3	-3	-3
Own goal	-2	-2	-2	-2

Additional Notes:

- **Clean sheets** are awarded to goalkeepers and defenders (4 points) and midfielders (1 point) only if they play at least 60 minutes and the team concedes no goals during their pitch time.
- Points for **saves** and **penalty saves** are exclusive to goalkeepers.
- **Bonus points** (1–3 points) are given via the Bonus Points System (BPS), rewarding the top three performing players in each match.
- Negative points from disciplinary actions (cards) and own goals apply immediately within the current Gameweek.

4 Algorithms implementation

4.1 Overview of Implemented Algorithms

This chapter presents a comprehensive progression of algorithmic strategies designed to address the Fantasy Premier League (FPL) optimization problem through a combination of predictive modeling and mathematical programming. As we move through the different methods, each introduces additional layers of modeling complexity, strategic depth, and real-world alignment. These enhancements aim to replicate the decisions of expert FPL managers more accurately, under the strict structural constraints imposed by the Fantasy Premier League.

The guiding principle behind this progression is to incrementally evaluate the benefits of integrating machine learning-based performance predictions with increasingly dynamic optimization frameworks. From static models that plan the season in advance to adaptive formulations that evolve weekly based on new information, each algorithm contributes to understanding what elements are essential for achieving high performance.

The chapter is structured around the following implementations:

- **Greedy Algorithm**: Serves as a baseline strategy. It selects the initial 15-player squad using a simple cost-efficiency metric, defined as the ratio between predicted season-long average points and player cost. Once selected, the starting 11 players are randomly chosen each Gameweek without further optimization.
- **Mean-MIP-Random Algorithm**: Introduces a Mixed Integer Programming (MIP) formulation to select the initial 15-player squad using season-averaged point predictions. The optimization respects all role, budget, and club constraints. The weekly lineup is still chosen randomly among the selected squad, allowing us to isolate the effect of optimal initial team construction.
- **Performance Prediction per Gameweek**: Before introducing more advanced decision models, we detail the machine learning pipeline developed to predict player performances for each Gameweek. Features include lagged statistics, contextual match data, and time-series cross-validation. These predictions are used in the upcoming models for dynamic optimization.
- **Mean-MIP-Rank Algorithm**: In this method, the starting 15-player squad remains fixed as in the previous approach. However, for each Gameweek, the starting 11 is selected by solving an MIP that uses adjusted per-GW player scores. These scores are refined through contextual bonuses that capture home advantage, recent form, and opponent difficulty based on dynamic team rankings.
- **GW1-MIP-Rank Experiment**: To understand the risk of short-term bias, this variant constructs the initial squad using only the predicted performances for Gameweek 1. It reveals how focusing optimization on a single week may lead to weaker long-term results.
- **Mean-MIP-Online Algorithm**: This formulation improves weekly responsiveness by re-estimating player performances using ML before each Gameweek and re-optimizing the starting 11 using these forecasts. The initial 15-player squad is fixed from the start, but the weekly decisions are fully data-driven and adapt to new information.
- **Market-MIP Algorithm**: Adds transfer dynamics to the model. Each week, players can be bought or sold while complying with FPL's market and budget rules. The algorithm now jointly selects the optimal starting 11, bench, and transfer strategy, balancing performance projections with transfer costs.
- **Market-MIP with Captain Selection**: The final enhancement includes the modeling of the captain selection rule. A new binary decision variable allows the optimization model to select one captain whose predicted points are doubled. The vice-captain is chosen heuristically. This model represents the most complete and realistic version of the FPL decision-making pipeline, accounting for predictions, transfers, lineup selection, and scoring multipliers.

Each method is rigorously evaluated on the 2023–2024 Premier League season using historical data. Performance is measured in terms of total points accumulated throughout the season, allowing for direct comparison across strategies. The results highlight the relative importance of various modeling components, revealing how accurate predictions, fine-grained optimization, dynamic transfer policies, and effective captaincy selection each contribute to maximizing FPL outcomes.

4.2 A Greedy algorithm

In fantasy football, the process of selecting an optimal starting lineup is a dynamic and data-driven challenge. The initial goal is to build an optimal Fantasy Premier League (FPL) team by reducing the problem to a simpler version. For now, we will focus on selecting 15 players without making any transfers across game weeks. The first algorithm we plan to implement is a greedy algorithm, serving as a baseline to evaluate the performance of more sophisticated approaches in future developments. The dataset used for the analysis is the one available online at: 'https://github.com/vaastav/Fantasy-Premier-League' [11]. To build up a greedy algorithm, the first thing we want to do is to have a prediction of the mean of fantasy points that the players will do, and then we will use those predictions as a metric to choose our starting 15 players. The procedure aims to forecast the total mean points of players for a given season by leveraging a time-series cross-validation method. This approach ensures that predictions account for temporal dependencies in the data, simulating real-world forecasting conditions where future events are predicted based on past data. The methodology can be summarized as follows.

1. **Data preparation:** Historical data containing features (player statistics) and the target variable (total fantasy points scored) are split into training and testing sets based on the season. The dataset is preprocessed to extract the season year and gameweek (GW) of each entry. Categorical features such as player position, team, and opponent team are encoded numerically using `LabelEncoder`. The dataset is sorted chronologically to preserve the temporal structure necessary for time series validation and lag feature extraction.

2. **Feature extraction:** A comprehensive set of engineered features is generated:

   - *Lag-based rolling averages* computed over the last three matches (e.g., `avg_pts_last_3`, `avg_goals_last_3`, `avg_assists_last_3`, etc.).
   - *Contextual performance indicators* that account for:
     - Home vs. away performance (`avg_pts_home_away`, `avg_ict_home_away`).
     - Opponent-specific history (`avg_pts_vs_opp`).
   - Static features including current `value`, `selected` percentage, team, opponent, and position.

   All missing values (e.g., due to insufficient history for rolling calculations) are imputed with zero to ensure model stability.

3. **Time-series cross-validation:** A k-fold time-series cross-validation strategy ($k = 7$) is applied:

   - Training and test sets are formed such that each test set chronologically follows the corresponding training set.
   - An XGBoost regression model (`XGBRegressor`) is trained for each fold.
   - Evaluation is performed using the Root Mean Squared Error (RMSE), with the goal of minimizing prediction error.

   Hyperparameter tuning is performed via grid search over `learning_rate`, `n_estimators`, and `max_depth`.

4. **Model training and seasonal weighting:** After the best hyperparameters are identified, the XGBoost model is retrained on the full training set. To prioritize recent seasons, a custom weighting scheme is applied:

   $$w_y = 1 + \frac{y - y_{\min}}{y_{\max} - y_{\min} + \varepsilon}$$

   where w_y is the weight of season y, and ε is a small constant added for numerical stability. These weights are normalized and used to compute weighted averages for all lag-based features, simulating player baselines.

5. **Final prediction:** For the target season:
    - The player features `value` and `selected` are initialized using values from gameweek 1.
    - Lag-based statistical features are replaced with the weighted averages computed across past seasons, simulating pre-season estimation in the absence of recent form data.
    - The trained model is used to assign a continuous `projected_performance` score to each player, representing their expected fantasy points.

At the end of this learning process, we will have a prediction of the mean of fantasy points scored by each player in the season of interest, defined for each player as $\hat{p}_c$, where $c \in C_\Omega$, and C_Ω is the set of players available in season Ω. This procedure combines robust feature engineering with a systematic time-series validation framework to ensure that the predictions are not only accurate but also reflective of the temporal structure of the data. By constructing a model based on prior seasons and validating it sequentially, this approach reduces the risk of overfitting and improves the reliability of the predictions. To build the greedy algorithm for the season Ω it is needed to compute for each player present in the season a "goodness-index" g_c ($c \in C_\Omega$). This index is computed as $g_c = \frac{\hat{p}_c}{k_{c1}}$, so dividing the predicted mean player c performance by its value(k_{c1}) at the first GW of the season Ω. Now it is possible to order all the players by their g_c, from the highest to the lowest, and obtain an ordered table of players T_Ω for season Ω. A sample of the table T_Ω could be appreciated in Fig.1. While building the team, role requirements r_d must be satisfied. The greedy algorithm implemented is defined as follows:

Algorithm 1 Greedy Algorithm for Team Selection

Dataset of players sorted by priority T_Ω,
Initial budget B,
Positional constraints r_d,
Max players per team $M_{team} = 3$,
Squad $\mathcal{S}$ with $|\mathcal{S}| = 15$,
$\mathcal{S} \leftarrow \emptyset$,
Initialize empty counters for the positions $Rolecount[d] \leftarrow 0, \forall d \in \{GK, DEF, MID, FWD\}$,
Initialize empty counters for each team $Teamcount[t] \leftarrow 0, \forall t \in \Omega$,

for player row c in ordered data frame T_Ω **do**
  if $|\mathcal{S}| = 15$ **then break**
    if $B - k_{c1} < 0$ **then continue**
      if Role of player c is d and $Rolecount[d] = \mathrm{r}_d$ **then continue**
        if Team of player c is t and $Teamcount[t] = M_t$ **then continue**
$\mathcal{S} \leftarrow \mathcal{S} \cup \{c\}$, $B \leftarrow B - k_{c1}$, $Rolecount[d] \leftarrow Rolecount[d] + 1$, $Teamcount[t] \leftarrow Teamcount[t] + 1$
return $\mathcal{S}$

---

name	gw1_value	position_corrected	team_x	mean_projected_performance	performance_ratio
Pervis Estupiñán	50.0	DEF	Brighton	2.7266169	0.05453233799999999
André Onana	50.0	GK	Man Utd	2.3309488	0.046618976
Gabriel dos Santos Magalhães	50.0	DEF	Arsenal	2.3188381	0.046376761999999995
Bukayo Saka	85.0	MID	Arsenal	3.8727903	0.045562238823529416
Aaron Ramsdale	50.0	GK	Arsenal	2.2565763	0.045131526
William Saliba	50.0	DEF	Arsenal	2.2223861	0.044447722
Emiliano Martínez Romero	50.0	GK	Aston Villa	2.205706	0.04411412000000001
David Raya Martin	50.0	GK	Brentford	2.1947846	0.04389569200000001
Marcus Rashford	90.0	MID	Man Utd	3.9266365	0.043629294444444446

Figure 1: Ordered table of players from season 2023-24

At this point, we have our 15 players. A random choice is then performed at each GW respecting the maximum number of players per role max_d and min_d. Below there are two plots (2,3) that show results for a simulation done on the 2023-24 Premier League season using this greedy algorithm for choosing the starting 15 players and randomly selecting among these our 11 player lineup for each GW.

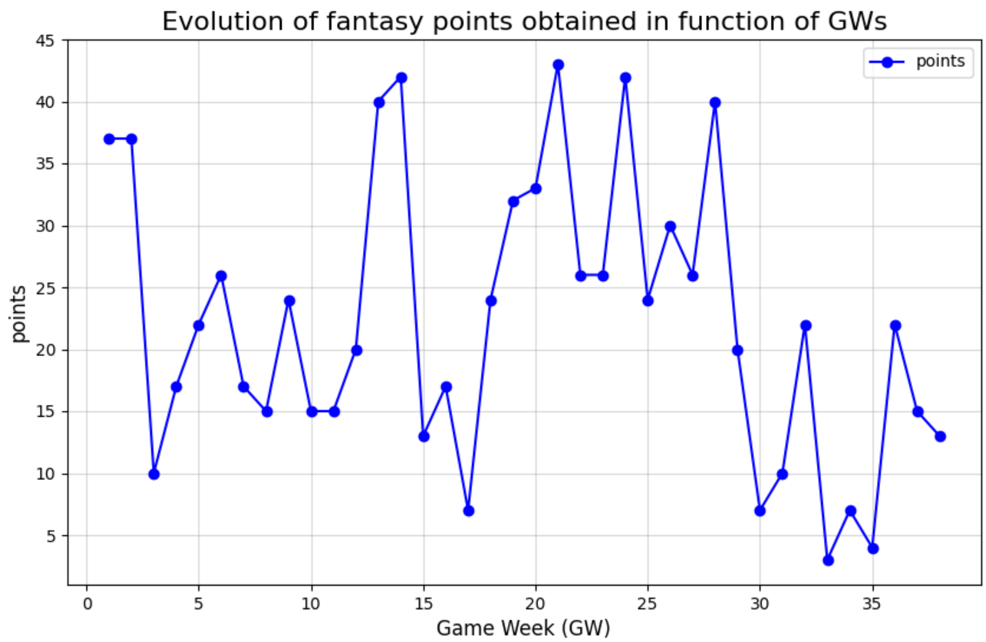

Figure 2: Evolution of points obtained over GWs by greedy alg

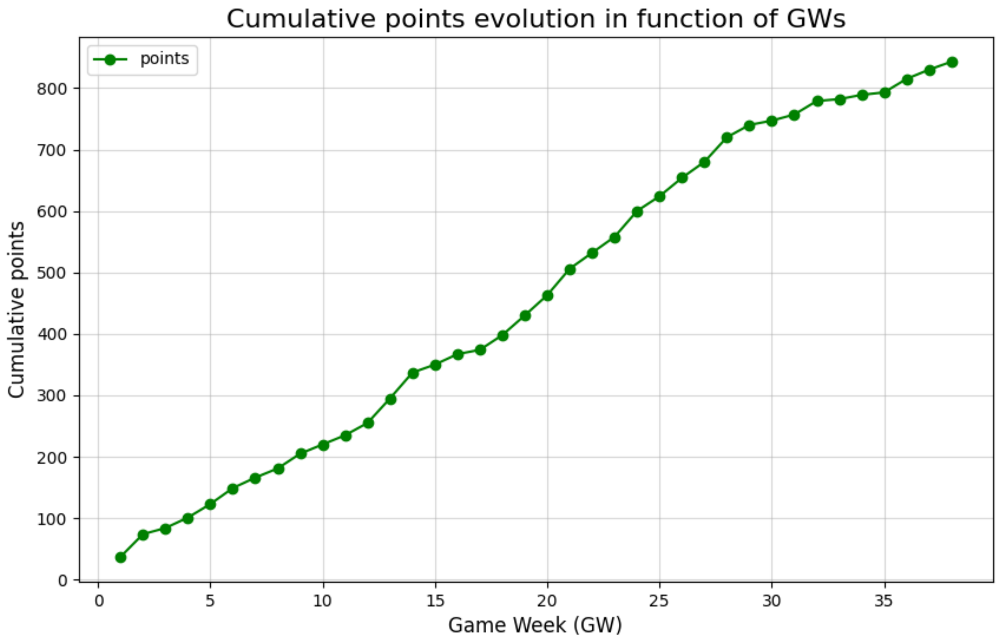

Figure 3: Cumulative points evolution over GWs using greedy alg

These results show that with this simple algorithm, the final score is quite far from the one that won the real FPL in the 2023-24 season. The algorithm scored around 850 points and the winner 2799.

4.3 Mean-MIP-random algorithm

One of the key challenges in Fantasy Premier League (FPL) is the initial selection of a strong 15-player squad that can generate consistent points throughout the season. While naive or greedy strategies may rely on current popularity or recent performances, such approaches often fail to consider long-term optimization under budget and team constraints. To overcome these limitations, we propose embedding machine learning predictions of player performance into a Mixed Integer Programming (MIP) model to perform the squad selection in a principled and data-driven way. In particular, we leverage the predicted mean performances $\hat{p}_c$, obtained via supervised learning algorithms trained on historical data, for each player c in season Ω. These estimates serve as a quantitative proxy for a player's expected contribution over the season and are used as the objective coefficients in the MIP formulation. The goal is to select the best possible squad of 15 players before the season starts, maximizing the projected cumulative performance while adhering to FPL constraints such as position quotas, budget limits, and maximum players per real-life team. Formally, we define the set of all players available in the season as $C_\Omega = \{1, 2, \ldots, n\}$, and the set of real-world teams as S_Ω. The optimization problem incorporates structural parameters (such as roles, teams, and costs) and is solved using Gurobi. This model-based selection process forms the basis of the algorithm we refer to as `mean-MIP-random`, where the 15-player squad is computed once at the beginning of the season through MIP, and the weekly selection of the starting 11 is performed randomly among them. Despite its simplicity in gameweek-level decisions, the strong initial selection achieved via optimization significantly outperforms greedy baselines. The effectiveness of this approach is illustrated in the figures below, where we track the evolution of points across GWs and compare the total season score against the baseline. To define the Mixed Integer

Programming (MIP) model for selecting the optimal squad of players, it is necessary to define the set of all players available in season Ω: $C_\Omega = \{1, 2, ..., n\}$ and the set S_Ω of all teams present in season Ω, and some other parameters:

- $\alpha_{cd} = \begin{cases} 1 & \text{if player } c \text{ has role } d \\ 0 & \text{otherwise} \end{cases}$
- $\beta_{cs} = \begin{cases} 1 & \text{if player } c \text{ belongs to team } s \\ 0 & \text{otherwise} \end{cases}$
- $\hat{p}_c$ = predicted performance of player c computed via machine learning
- k_{c1} = associated cost of player c at 1st GW.
- B = max available budget
- r_d = number of players required in role d
- Decision variables (binary):

$$x_c(c \in C_\Omega) = \begin{cases} 1 & \text{if player } c \text{ is selected} \\ 0 & \text{otherwise} \end{cases}$$

$$\begin{aligned}
\max_{\mathbf{x}} \quad & \sum_{c \in C_\Omega} \hat{p}_c \cdot x_c && && (1)\\
\text{s.t.} \quad & \sum_{c \in C_\Omega} x_c = 15, && \text{(squad size constraint)} && (2)\\
& \sum_{c \in C_\Omega} \alpha_{cd} \cdot x_c = r_d, \quad \forall d, && \text{(role requirements)} && (3)\\
& \sum_{c \in C_\Omega} \beta_{cs} \cdot x_c \leq 3, \quad \forall s \in S_\Omega, && \text{(max players per team)} && (4)\\
& \sum_{c \in C_\Omega} k_{c1} \cdot x_c \leq B && \text{(budget constraint)} && (5)\\
& \mathbf{x} \in \{0,1\}^{|C_\Omega|}. && \text{(binary decision variables)} && (6)\\
& d \in \{\text{GK, DEF, MID, FWD}\} && && (7)
\end{aligned}$$

After having solved the MIP problem via Gurobi, at each GW of the season Ω a random choice of 11 players is performed for the greedy algorithm. Here below it is possible to appreciate the evolution of gw points(4,5) and the improvement of using a MIP optimization model in the selection of the 15 starting players is obtained after a few GWs (6). As before, these simulation results refer to season 2023-24. The total points obtained by mean-MIP-random is 1250, while with the greedy, it was 850.

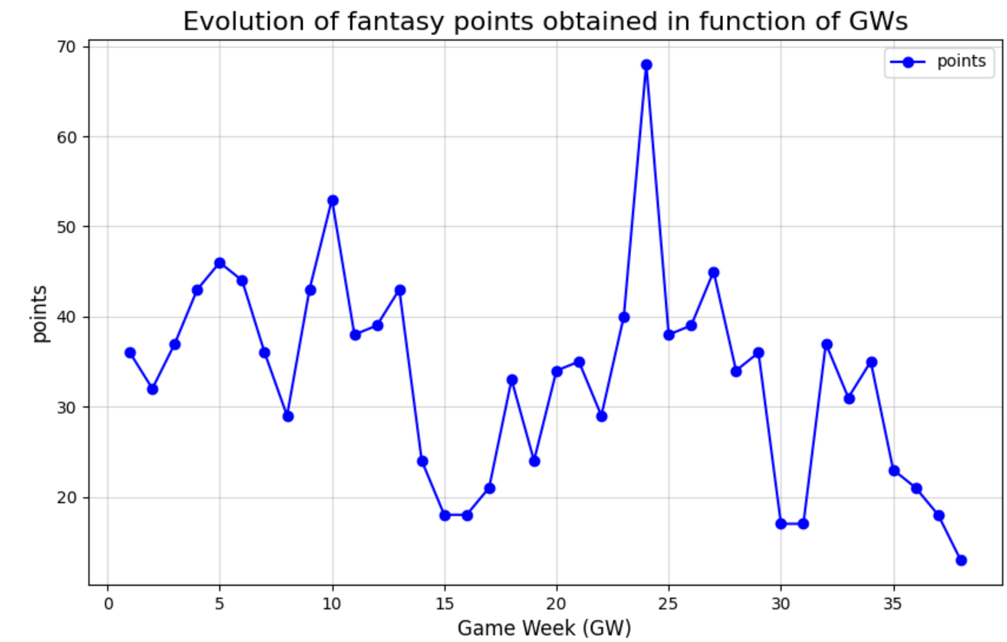

Figure 4: Evolution of points obtained over GWs by mean-MIP-random alg

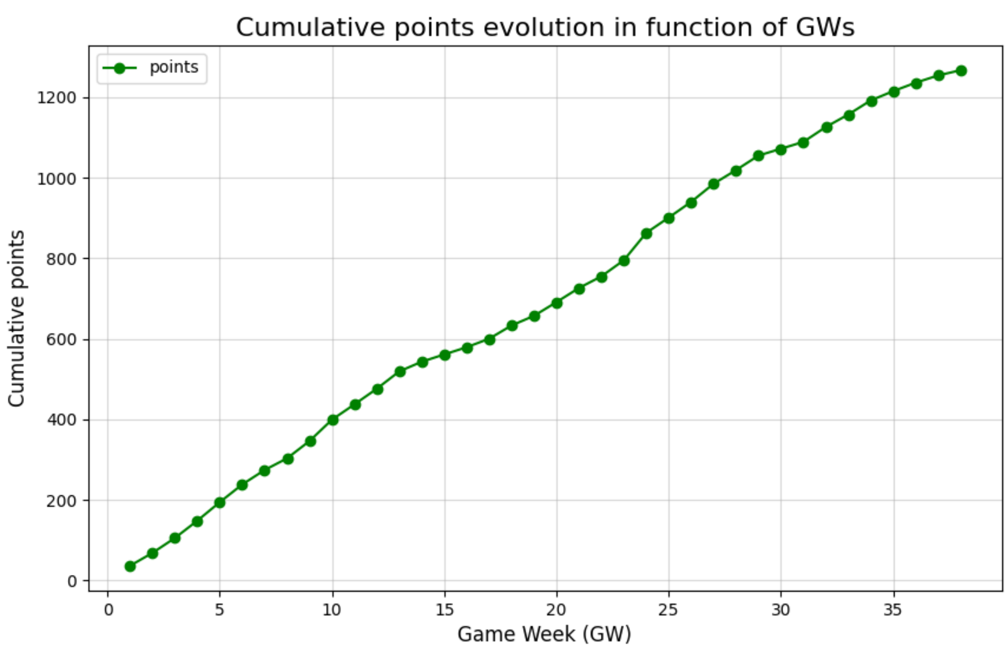

Figure 5: Cumulative points evolution over GWs using mean-MIP-random alg.

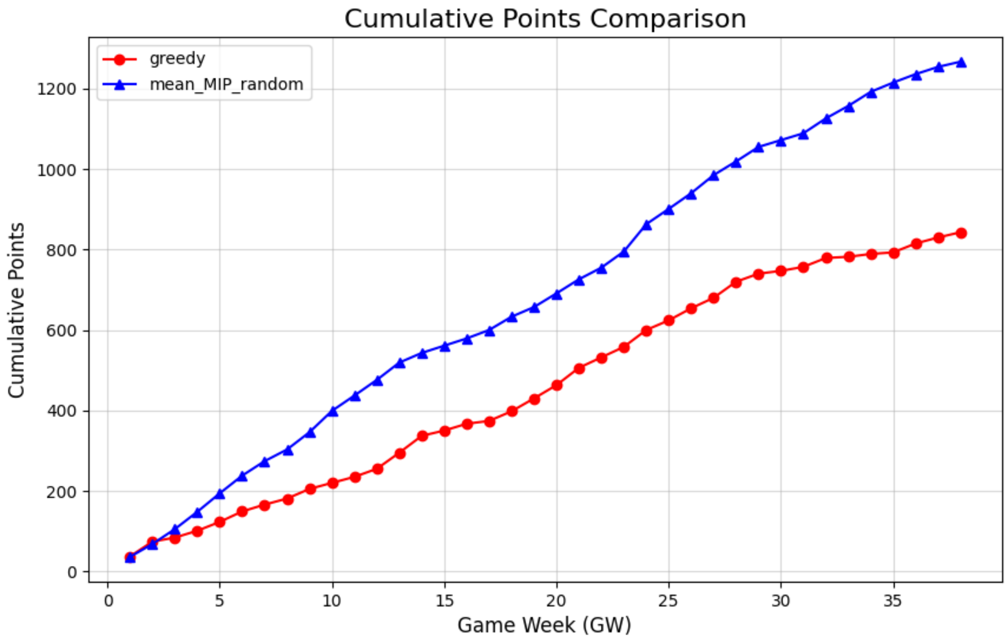

Figure 6: Comparison between greedy and mean-MIP-random (both with random selection of players over GWs)

At this point it is clear that the main improvement with respect to the greedy algorithm lies in using a MIP optimization problem in the selection of the 15 players. The score achieved is around 1220 points, while the winner of the Fantasy Premier League season 2023-24 had 2799.

4.4 Prediction of performances for each GW

At this stage of the project, in order to further improve our final score, we transition from static mean performance predictions ($\hat{p}_c$) to dynamic, Gameweek-specific predictions. The goal is to forecast the fantasy points that each player will score in each individual Gameweek, capturing form trajectories, match-specific conditions, and short-term trends. The prediction process is structured into the following steps:

1. Data Preparation Relevant features are selected from both static player attributes and dynamic, lag-based performance metrics. The model relies on a rich set of features including:

- **Player-specific static features:** *value*, *times selected*, *position*, and *team*.
- **Match-specific contextual features:** home/away status (*was_home*) and opponent team (numerically encoded).
- **Lag-based features:** for each of the past 5 Gameweeks, we include time-lagged statistics such as:
  - fantasy points, minutes, assists, goals, clean sheets
  - influence, threat, creativity, ICT index, BPS
  - penalties, cards, own goals, goals conceded, saves

– market signals like *transfers in/out* and *value changes*

These features enable the model to learn individual player trajectories and dynamic form signals.

Categorical variables are numerically encoded using `LabelEncoder`, ensuring compatibility with gradient-based machine learning models.

2. Training and Validation We adopt a Time Series Cross-Validation (TSCV) strategy with 6 splits to preserve the chronological nature of the data. A `GradientBoostingRegressor` is trained for each GW prediction, with hyperparameters optimized via Grid Search. The following results are obtained for the model trained to predict GW1:

- **Training RMSE:** 2.03
- **Test RMSE:** 2.12
- **Cross-validated RMSE:** 2.04
- **Best Parameters:** learning rate = 0.05, number of estimators = 100

On average, the training time per Gameweek model is approximately **12 minutes**, making the approach computationally feasible for seasonal simulation.

3. Model Interpretation Feature importance analysis reveals that the most predictive features are those related to recent playing time and form. The top-ranked features for GW1 include:

- `minutes_lag_1` (importance score: 0.63)
- `ict_index_lag_1`
- `selected_lag_1`
- `value`
- `minutes_lag_2`
- `total_points_lag_1`

This confirms that players who consistently play and are influential in recent matches tend to have higher predictive power.

4. Prediction Procedure After training, the model is used to forecast fantasy points for all future Gameweeks in the season. To simulate a realistic setting where only past data is available, features such as lag-based statistics, player value, and selection percentage are *frozen* at their values in Gameweek g, and used for all future predictions beyond g.

The outcome of this step is, for each player c, a performance vector:

$$\mathbf{p_c} = (p_{c1}, p_{c2}, \ldots, p_{c38}) \in Z^{38}$$

where p_{cg} represents the predicted fantasy points for player c in Gameweek g (where p_{c1} is the predicted performance in terms of fantasy points of player c at GW 1 of a specific season Ω, p_{c2} the predicted performance for GW 2, and so on).

These forecasts are first exploited in the **mean-MIP-rank** algorithm, where they guide weekly player selection through a contextual scoring system. More importantly, they serve as the backbone of the objective functions in the **mean-MIP-online** and more advanced models, enabling fully dynamic re-optimization of the starting lineup as updated predictions become available each Gameweek. In the following, we will compute other parameters to help us define an objective function for a new MIP problem that would be solved online for each GW.

4.5 The Mean-MIP-rank

After having computed the vector $\mathbf{p_c}$ for each player c of the season Ω, a way to compute a ranking of teams S ($\forall S \in S_\Omega$) is defined.

The process of computing team rankings involves the integration of data from both the previous seasons and the current season up to a certain point in time. It is designed to weigh recent performances more heavily than older ones and builds a ranking system that reflects the relative strength of teams over time. The following is a detailed description of the methodology.

Step 1: Data preparation The process begins with collecting match data, which includes the results of games played over multiple seasons. Additional match details, such as the season, game, teams involved, and their scores, are also considered. The data is then filtered to retain:

- Matches from all past seasons.
- Matches from the current season, but only up to a specified point in time (e.g., a specific gameweek), since the ranking will be computed many times in the algorithm depending on the GW for which GW we have to select 11 players.

This ensures that the dataset reflects both historic and recent performances.

Step 2: Determining Match Outcomes Each match is assigned a result based on the performance of the teams. A simple scoring system is used:

- **3 points** are awarded for a win.
- **1 point** is awarded for a draw.
- **0 points** are awarded for a loss.

This system standardizes the evaluation of match outcomes and creates a baseline for further calculations.

Step 3: Applying Season and Gameweek Weighting To account for the relative importance of matches from different times, the following weighting mechanisms are applied:

- **Season Weights:** Matches from more recent seasons are assigned higher importance. By contrast, older seasons are weighted less, reflecting their reduced relevance to a team's current strength.
- **Gameweek Weights:** Within each season, matches from more recent game weeks are given a higher significance. Matches from earlier game weeks are weighted less, as they are further removed from the team's most recent performance.

These weights are calculated such that the importance of a match decreases exponentially as it becomes older, either in terms of the season or the game.

Step 4: Calculating Weighted Performance For each match, a weighted performance score is calculated by multiplying the match result points (win, draw, loss) by the combined weights derived from the season and game. This produces a score that reflects not only the outcome of the match but also its recency and relevance.

Step 5: Aggregating Team Scores Once weighted performance scores have been calculated for all matches, they are joined to each team S. This involves summing the scores for each team across all matches in the dataset. A key value is computed for each team: The total weighted score ($rank_s$, $s \in S_\Omega$), which takes into account both match outcomes and their weights.

Step 6: Normalization and Ranking To ensure comparability across teams, the aggregated weighted score for each team is normalized. The highest weighted score is set as the benchmark (i.e., normalized to 1), and the scores of all other teams are expressed as a proportion of this value. This produces a normalized performance index for each team, ranging from 0 to 1.

Teams are then sorted in descending order of their performance index. The highest-ranking team is assigned the first position, and the ranking continues in order until all teams have been ranked. This process ensures a thorough and fair evaluation of team performance by considering both recent and historical data while weighting recent performances more strongly to give an accurate assessment of current team form. The aim of this new algorithm is to improve previous results by implementing an online player choice each GW based on predicitons and recent performances. First, 15 players are selected as in the previous MIP formulation, using predictions $\hat{p_c}$ for each player. After having obtained the 15 players, now the innovation lies in selecting in an online fashion the 11 players for each GW, not just randomly as done before, using vectors $\mathbf{p_c}$ for each player c $\in C_\Omega$, team ranking, and other parameters. At the core of the selection process lies the objective function, which ranks each player in the squad by computing a performance score (or weight). This score reflects the likelihood that the player will maximize the overall team points and is derived from:

- **Home/Away Adjustments:** Players' predicted performances are given a different bonus whether their upcoming match is at home or away, as performance at home typically tends to be stronger for most players or teams.
- **Recent Performance:** Points scored by players in recent game weeks are weighted more heavily to capture the current form. An exponential decay is often used to give progressively less importance to older performances.
- **Fixture Difficulty:** The expected difficulty of the opponent affects the projected score. Players facing weaker opponents are weighted more favorably.

The combined score for each player at GW g is formalized as: $P_{cg} = p_{cg} \cdot w_p + h_b + opp_b + recent_c \cdot w_r$
Where:

- P_{cg} is the final predicted performance for player c at GW g.
- p_{cg} is the previously predicted performance by the ML model for player c at GW g.
- w_p is the weight given to the ML performance prediction.
- h_b is the bonus given to players playing the upcoming match at home.

- opp_b is the bonus given to players based on the ranking position of the opponent team. This bonus is inversely proportional to the ranking (e.g. a player playing the upcoming match against a good-ranked team will receive a small bonus).
- $recent_c$ is a mean of points of player c in recent matches.
- w_r is the weight given to recent performances.

The MIP to solve at each GW g of season Ω now looks like this:

$$\max_{\mathbf{x}} \sum_{c \in C_{\Omega_1}} P_{cg} \cdot x_c \tag{8}$$

$$\text{s.t.} \quad \sum_{c \in C_{\Omega_1}} x_c = 11, \qquad \text{(starting lineup size)} \tag{9}$$

$$\sum_{c \in C_{\Omega_1}} \alpha_{cd} \cdot x_c \leq max_d, \quad \forall d \in \{GK, DEF, MID, FWD\}, \qquad \text{(max players per role in lineup)} \tag{10}$$

$$\sum_{c \in C_{\Omega 1}} \alpha_{cd} \cdot x_c \geq min_d, \quad \forall d \in \{GK, DEF, MID, FWD\}, \qquad \text{(min players per role in lineup)} \tag{11}$$

$$\mathbf{x} \in \{0,1\}^{|C_{\Omega_1}|}, \qquad \text{(binary decision variables)} \tag{12}$$

Where:

- C_{Ω_1} is the set of (15) players chosen before GW 1.
- $$x_c(c \in C_{\Omega_1}) = \begin{cases} 1 & \text{if player } c \text{ is selected in the lineup} \\ 0 & \text{if player is on the bench} \end{cases}$$

This dynamic selection process ensures that the starting 11 players maximize the team's potential score for each gameweek while maintaining flexibility to respond to injuries, form, or other external factors. Below are shown results for using this algorithm in season 2023-24(7,8) and the improvement obtained concerning other algorithms(9). The total points obtained using this algorithm is 1450, much closer to the points of the winner (2799).

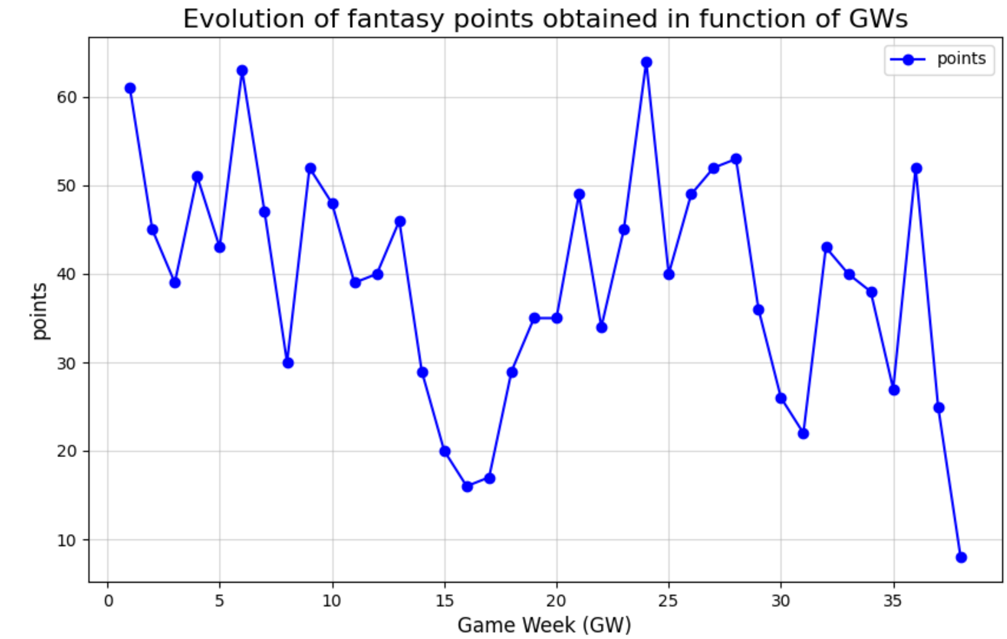

Figure 7: Evolution of points obtained over GWs with mean-MIP-rank

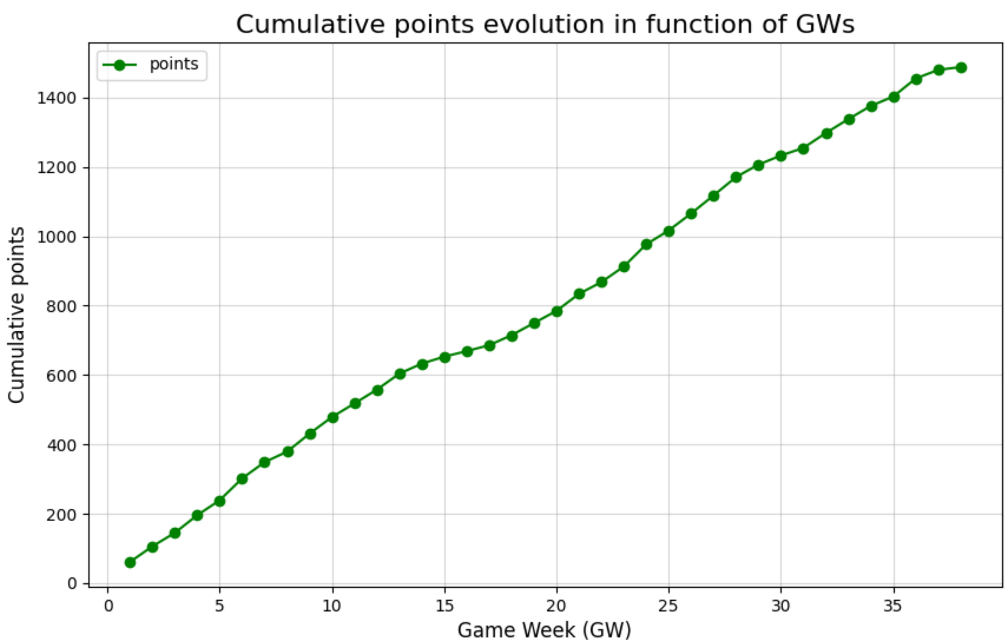

Figure 8: Cumulative points evolution over Gws with mean-MIP-rank

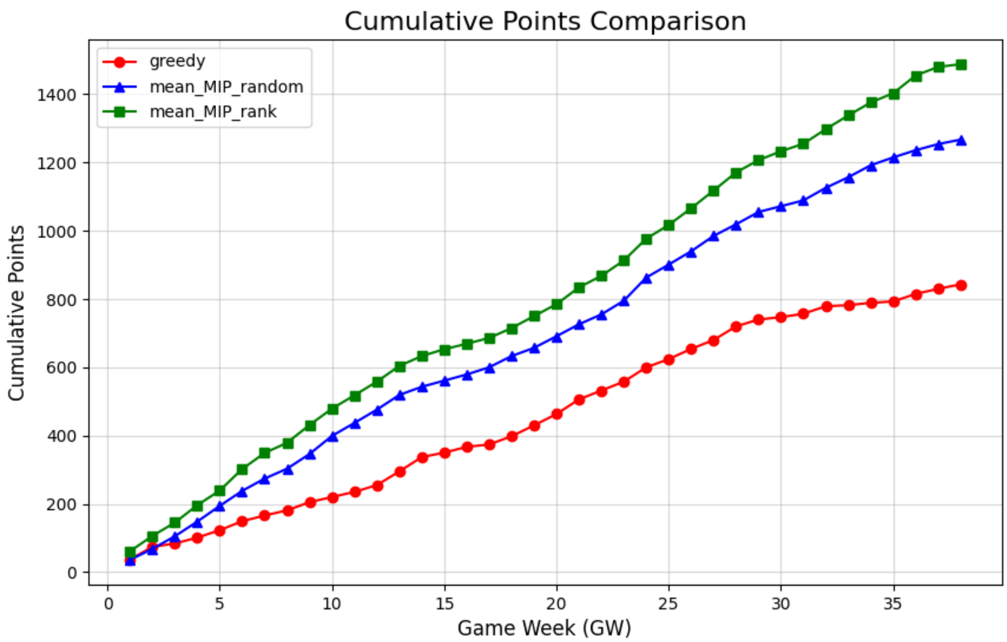

Figure 9: Comparison between greedy, mean-MIP-random and mean-MIP-rank

With this new algorithm around 1450 points are scored, better with respet to the previous 1220 score by the mean-MIP-random algorithm, but still far from the 2799 points of the winner.

4.6 An experiment: the gw1-MIP-rank

Now, it may also be interesting to look at the performances of the mean-MIP-rank on players that have been chosen only based on the predicted first GW performances. The MIP used to select the 15 players is:

$$\max_{\mathbf{x}} \sum_{c \in C_\Omega} p_{c1} \cdot x_c \tag{13}$$

$$\text{s.t.} \quad \sum_{c \in C_\Omega} x_c = 15, \qquad \text{(squad size constraint)} \tag{14}$$

$$\sum_{c \in C_\Omega} \alpha_{cd} \cdot x_c = r_d, \quad \forall d, \qquad \text{(role requirements)} \tag{15}$$

$$\sum_{c \in C_\Omega} \beta_{cs} \cdot x_c \leq 3, \quad \forall s \in S_\Omega, \qquad \text{(max players per team)} \tag{16}$$

$$\sum_{c \in C_\Omega} k_{c1} \cdot x_c \leq B \qquad \text{(budget constraint)} \tag{17}$$

$$\mathbf{x} \in \{0,1\}^{|C_\Omega|}. \qquad \text{(binary decision variables)} \tag{18}$$

$$d \in \{\text{GK, DEF, MID, FWD}\} \tag{19}$$

Note that concerning the MIP that has been used to select 15 players using predicted mean performances $\hat{p_c}$ for each player c, now in the objective function there is p_{c1}, the predicted performances for GW1 of season Ω. Here a plot comparing all the 4 algorithms' performance on season 2023-24 is shown(10).

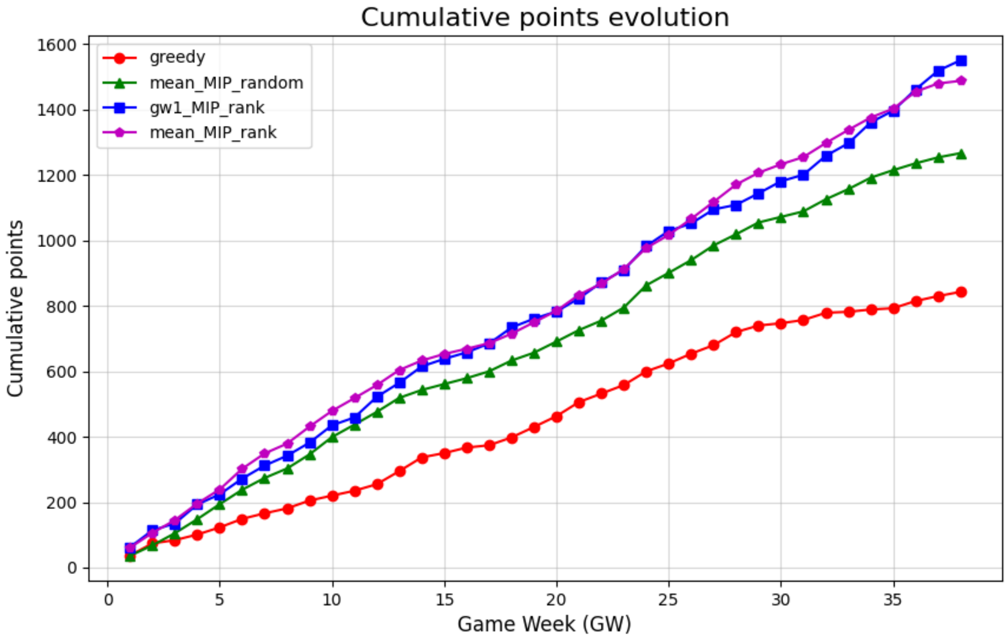

Figure 10: Comparison of all 4 alg.

The mean-MIP-rank overall performs better, and it is reasonable since the starting 15 choice has been performed looking to the all upcoming season and not only to the the first season GW.

4.7 The Mean-MIP-online

At this stage of the project, the strategy shifts from manually constructing the objective function to leveraging machine learning (ML) techniques to estimate player performance in a data-driven and adaptive manner. The core idea is to improve the estimation of the *goodness index* (denoted as $\mathbf{p}_c$), which quantifies each player's expected contribution to the total points in a given Gameweek (GW). This index is then used in the Mixed-Integer Programming (MIP) model to optimize the team selection process. Previously, in models such as the `mean-MIP-rank`, recent player performance data was aggregated manually using weighted averages, with a higher emphasis on more recent GWs to capture current form. While this approach provided a basic form of adaptability, it lacked flexibility and failed to incorporate contextual factors such as temporal trends across the season and player specific performance trajectories. To overcome these limitations, we now adopt a machine learning approach that allows us to dynamically learn patterns from historical data and make informed predictions about future performance. Specifically, before each GW g, we train an ML model using a variety of features, including recent form, historical performance, team context, and opponent strength, to estimate the projected performance vector $\mathbf{p}_c$ for all players in the squad. This vector then serves as input to the MIP, guiding the selection of the optimal starting 11.

The starting 15 squad remains fixed as in the `mean-MIP-random` formulation, but the starting 11 lineup is re-optimized before every GW based on the updated ML predictions. Let us refer to this new algorithm as `mean-MIP-online`. The MIP model solved at each GW g is:

$$
\begin{aligned}
\max_{\mathbf{x}} \quad & \sum_{c \in C_{\Omega_1}} p_{cg} \cdot x_c && && (20) \\
\text{s.t.} \quad & \sum_{c \in C_{\Omega_1}} x_c = 11, && \text{(squad size constraint)} && (21) \\
& \sum_{c \in C_{\Omega_1}} \alpha_{cd} \cdot x_c \leq max_d, \quad \forall d, && \text{(max players per role in lineup)} && (22) \\
& \sum_{c \in C_{\Omega_1}} \alpha_{cd} \cdot x_c \geq min_d, \quad \forall d, && \text{(min players per role in lineup)} && (23) \\
& \mathbf{x} \in \{0,1\}^{|C_{\Omega_1}|} && \text{(binary decision variables)} && (24) \\
& d \in \{\text{GK, DEF, MID, FWD}\} && && (25)
\end{aligned}
$$

By combining machine learning predictions with optimization, we enable a more responsive and accurate player selection strategy. The learning model adapts over time as new data becomes available, improving the robustness of the forecasted p_{cg} values and allowing the algorithm to reflect changes in player form, injuries, or favorable fixtures.

As shown in Figure 12, this ML-based approach achieves a modest but consistent improvement in total points over the course of the season compared to previous heuristics or static weighting schemes, validating the benefit of integrating data-driven forecasting into the optimization pipeline. In addition, Figure 11 shows the evolution of fantasy points across all GWs in the 2023–24 Premier League, which is noticeably more regular compared to previous algorithms, suggesting that we are moving towards a more robust approach.

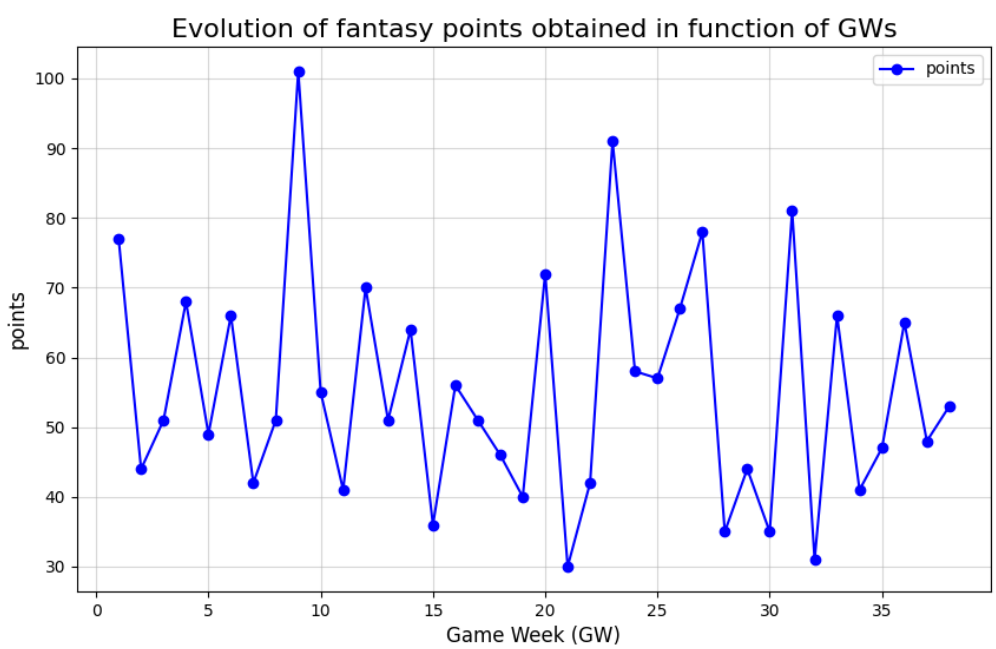

Figure 11: Evolution of fantasy points with mean-MIP-online

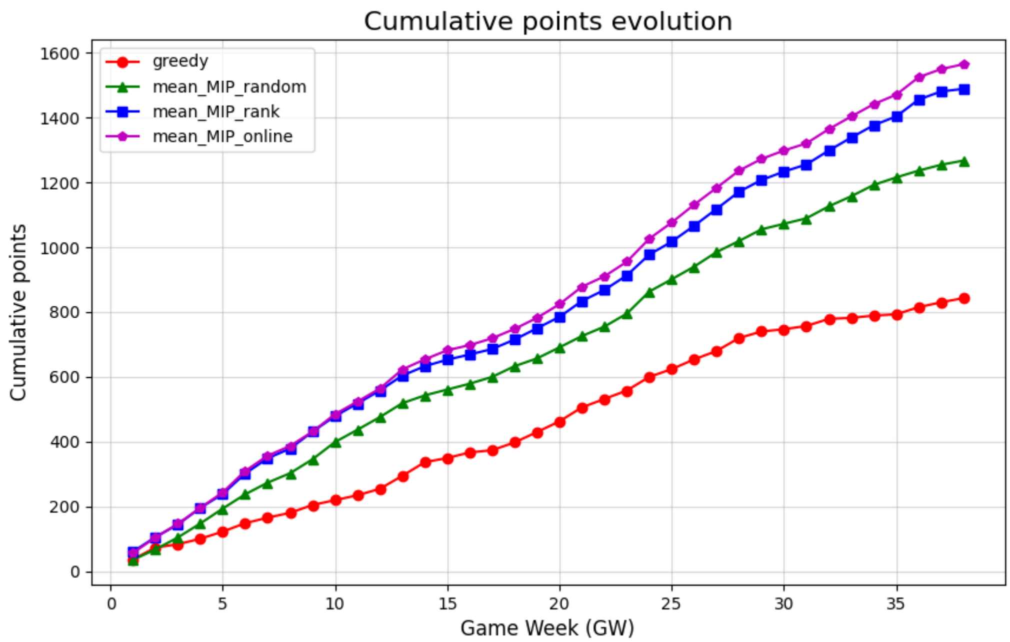

Figure 12: Comparison of mean-MIP-online with other alg

4.8 MIP formulation with market

Introducing now the possibility of performing **strategic player transfers**, we move toward a more realistic and powerful formulation that better reflects the actual dynamics of the Fantasy Premier League (FPL). Unlike static team selection models, which optimize based on a fixed pool of players for the entire season, this enhanced model allows the team composition to evolve over time by making substitutions at each gameweek (GW). In practice, FPL managers are allowed to alter their squad by buying and selling players between gameweeks, subject to constraints on budget, team size, and a limited number of free transfers. Capturing this real-world mechanism requires a richer formulation in our Mixed-Integer Programming (MIP) model. In this section, we present a **dynamic MIP optimization framework** that simultaneously selects the optimal 15-player squad for a given GW and determines which players to transfer in and out from the market. The model incorporates *player performance predictions*, transfer penalties, role constraints, team quotas, and budget flow across weeks. The objective function balances **maximizing projected points** based on predicted performance while **minimizing the cost of extra transfers**, thus enabling a trade-off between potential gains and managerial decisions.

To formalize this problem, we define a set of binary decision variables representing the status of each player (starter, bench, transferred in, or transferred out), and we introduce variables to model free transfer logic and budget transitions. Moreover, player performance predictions p_{cg}, estimated via machine learning, are computed not only based on recent form but also by forecasting the player's expected contributions across the remaining matches of the season. This forward-looking perspective helps the algorithm evaluate each player's overall strategic value beyond the next gameweek.

In particular, considering future matchups and the long-term expected output of each player leads to more stable decision-making over time. By incorporating future projections in the preference score p_{cg}, the model naturally avoids excessive squad turnover between gameweeks, focusing instead on transfers that offer sustained benefit over multiple rounds. The resulting formulation, referred to as the **market-MIP model**, is resolved independently at each gameweek g, updating team composition in accordance with the latest player projections and historical squad configuration.

The additional modeling of captain and vice-captain selection, which further influences the final score, will be addressed in a later section. The aim of the new objective function is to maximize projected performance for the next GW g and minimize transfer cost. The variables whose value we want to optimize are :

- $\mathbf{x}$ that is the stack of all variables for each player c that will be in the 11 lineup at each GW g : $\mathbf{x} = (x_{11}, x_{21}, ..., x_{|C_\Omega|1}, ..., x_{|C_\Omega|38})$
- $\mathbf{s}$ that is the stack of all variables for each player c that will be in the 5 bench lineup at each GW g : $\mathbf{s} = (s_{11}, s_{21}, ..., s_{|C_\Omega|1}, ..., s_{|C_\Omega|38})$
- $\mathbf{y}$ that is the stack of all variables for each player c that will be transferred, from the pool of the players available in the market, in to the team at each GW g : $\mathbf{y} = (y_{11}, y_{21}, ..., y_{|C_\Omega|1}, ..., y_{|C_\Omega|38})$
- The set $C_{\Omega g-1}$ is the set of all players in season Ω that at GW g-1 were in the squad.
- B_{g-1} is the budget that is stored from GW g-1.
- $\mathbf{t}$ that is the stack of all variables for each player c that in the GW previous of g was either in the starting 11 lineup or previous bench and now is sold: $\mathbf{t} = (t_{11}, t_{21}, ..., t_{|C_{\Omega g-1}|1}, ..., t_{|C_{\Omega g-1}|38})$
- z is a binary variable that assumes a value 1 if at least 1 transfer is done at GW g, so it is a way to model the first free transfer given at each GW.

In the objective function we have parameters p_{cg} that are obtained as in the mean-MIP-online, so running a ML algorithm to predict players' performances before each GW g, taking into account also recent data from last GWs. The gw-MIP-market model that we want to solve at each GW g is :

$$\max_{\mathbf{x},\mathbf{s},\mathbf{y},z,\mathbf{t}} \sum_{c\in C_\Omega} p_{cg}\cdot x_{cg} - 4\sum_{c\in C_\Omega} y_{cg} + 4z \tag{26}$$

$$\text{s.t.} \quad \sum_{c\in C_\Omega} x_{cg} = 11, \qquad \text{(starting 11 constraint)} \tag{27}$$

$$\sum_{c\in C_\Omega} s_{cg} = 4, \qquad \text{(bench constraint)} \tag{28}$$

$$z \leq \sum_{c\in C_\Omega} y_{cg}, \qquad \text{(free transfer condition)} \tag{29}$$

$$\sum_{c\in C_\Omega} k_{cg}\cdot y_{cg} \leq B_{g-1} + \sum_{c\in C_{\Omega g-1}} k_{cg}\cdot t_{cg}, \qquad \text{(budget constraint for g > 1)} \tag{30}$$

$$\sum_{c\in C_\Omega} k_{c1}\cdot(x_{c1}+s_{c1}) \leq B_1, \qquad \text{(budget constraint for g = 1)} \tag{31}$$

$$\sum_{c\in C_\Omega} y_{cg} = \sum_{c\in C_{\Omega g-1}} t_{cg}, \qquad \text{(transfer balance)} \tag{32}$$

$$\sum_{c\in C_\Omega} y_{cg} = \sum_{c\in C_\Omega} (x_{cg}+s_{cg}), \qquad \text{(lineup-transfer consistency)} \tag{33}$$

$$\sum_{c\in C_\Omega} \alpha_{cd}(x_{cg}+s_{cg}) = r_d, \quad \forall d, \qquad \text{(exact players per role in squad)} \tag{34}$$

$$\sum_{c\in C_\Omega} \alpha_{cd}(x_{cg}) \leq max_d, \quad \forall d, \qquad \text{(max players per role in lineup)} \tag{35}$$

$$\sum_{c\in C_\Omega} \alpha_{cd}(x_{cg}) \geq min_d, \quad \forall d, \qquad \text{(min players per role in lineup)} \tag{36}$$

$$\sum_{c\in C_\Omega} \beta_{cs}(x_{cg}+s_{cg}) \leq 3, \quad \forall s\in S_\Omega, \qquad \text{(max players per team in squad)} \tag{37}$$

$$t_{cg} = 1-(x_{cg}+s_{cg}), \quad \forall c\in C_{\Omega g-1} \qquad \text{(Previous starters transfer constraint)} \tag{38}$$

$$x_{cg}, s_{cg}, y_{cg} \in \{0,1\}, \quad \forall c\in C_\Omega, \qquad \text{(binary variables)} \tag{39}$$

$$t_{cg} \in \{0,1\}, \quad \forall c\in C_{\Omega g-1}, \qquad \text{(binary variables)} \tag{40}$$

$$z \in 0,1 \tag{41}$$

The budget B is updated at each iteration and is computed as $B = previousBudget + gainedBudget - spentBudget$, where previousBudget ($B_{g-1}$) is the budget that remains from the previous GW g-1, gainedBudget is the amount of fantasy coins gained at GW g by selling players from the current team, and spentBudget is the budget spent at GW g for players bought from the market. In the figure 13 can be seen the huge improvement obtained by combining the possibility of doing some player transfer and the ML prediction of values p_{cg} computed before each game week using historical data as a time series.

Explanation of Constraints

The model includes several constraints to ensure the feasibility and realism of the team selection and transfer process in accordance with Fantasy Premier League rules:

- **(26) Starting 11 constraint:**

$$\sum_{c\in C_\Omega} x_{cg} = 11$$

Exactly 11 players must be selected for the starting lineup at each gameweek g.

- **(27) Bench constraint:**

$$\sum_{c \in C_\Omega} s_{cg} = 4$$

Ensures that 4 players are selected as substitutes on the bench at each GW.

- **(28) Free transfer condition:**

$$z \leq \sum_{c \in C_\Omega} y_{cg}$$

If no transfers are made, $z = 0$. If at least one transfer occurs, $z = 1$ and allows the model to apply the free transfer rule.

- **(29) Budget constraint for $g > 1$:**

$$\sum_{c \in C_\Omega} k_{cg} \cdot y_{cg} \leq B_{g-1} + \sum_{c \in C_{\Omega g-1}} k_{cg} \cdot t_{cg}$$

The total value spent on new players at GW g must be less than or equal to the available budget plus proceeds from sold players.

- **(30) Initial budget constraint for $g = 1$:**

$$\sum_{c \in C_\Omega} k_{c1} \cdot (x_{c1} + s_{c1}) \leq B_1$$

The total cost of the initial 15-player squad must be within the initial budget.

- **(31) Transfer balance:**

$$\sum_{c \in C_\Omega} y_{cg} = \sum_{c \in C_{\Omega g-1}} t_{cg}$$

The number of players bought must equal the number of players sold to maintain a constant squad size.

- **(32) Lineup-transfer consistency:**

$$\sum_{c \in C_\Omega} y_{cg} = \sum_{c \in C_\Omega} (x_{cg} + s_{cg})$$

Ensures that the new squad composition corresponds to the players selected to play or be benched.

- **(33) Role-based squad structure:**

$$\sum_{c \in C_\Omega} \alpha_{cd}(x_{cg} + s_{cg}) = r_d \quad \forall d$$

The squad must include an exact number of players per position (e.g., 2 GK, 5 DEF, etc.).

- **(34) Maximum players per role in lineup:**

$$\sum_{c \in C_\Omega} \alpha_{cd} x_{cg} \leq max_d \quad \forall d$$

Role-based upper bounds for players in the starting 11.

- **(35) Minimum players per role in lineup:**

$$\sum_{c \in C_\Omega} \alpha_{cd} x_{cg} \geq min_d \quad \forall d$$

Role-based lower bounds to ensure formation feasibility.

- **(36) Team quota constraint:**

$$\sum_{c \in C_\Omega} \beta_{cs}(x_{cg} + s_{cg}) \leq 3 \quad \forall s \in S_\Omega$$

No more than 3 players can be selected from the same real-world team.

- **(37) Transfer-out condition:**

$$t_{cg} = 1 - (x_{cg} + s_{cg}) \quad \forall c \in C_{\Omega g-1}$$

A player from the previous squad is considered transferred out if they are not in the current lineup or bench.

- **(38–41) Binary constraints:**

$$x_{cg}, s_{cg}, y_{cg}, t_{cg}, z \in \{0, 1\}$$

All decision variables are binary, reflecting the discrete nature of player selection and transfers. In the following plot it can be appreciated the score improvement obtained by implementing the market dynamics in the MIP formulation.

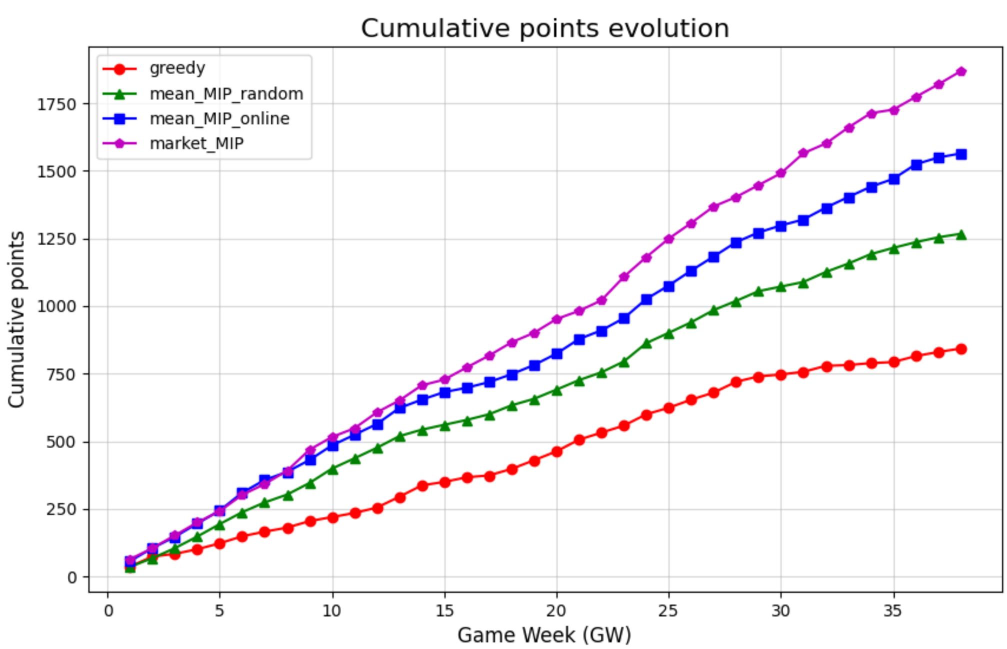

Figure 13: Comparison between market-MIP and other algorithms

The figure 13 clearly shows the significant improvement achieved by introducing market dynamics into the MIP formulation. By allowing the squad to evolve over time through strategic player transfers, the model can react to changes in player form, injuries, or favorable fixtures. This adaptability, combined with machine learning predictions that anticipate not only the next gameweek but also future performances, enables smarter, longer-term decisions. As a result, the total points accumulated over the season increase substantially compared to both static optimization strategies and simpler models that do not account for transfer opportunities. The inclusion of market behavior adds flexibility and realism, aligning the optimization more closely with how actual FPL managers operate throughout the season. The score achieved with this market-MIP is 1878 points, recalling that the winner has 2799 for 2023-24 Premier League season.

4.9 The MIP-market with captain selection

To further enhance the quality of the optimal lineup, we extend our market-aware MIP model to explicitly model the Fantasy Premier League (FPL) captain rule. In the FPL ruleset, one player in the lineup can be designated as the captain, whose fantasy points will be doubled for that Gameweek (GW). If the selected captain does not participate (i.e., plays 0 minutes), the vice-captain will inherit the captaincy and thus have their score doubled. Let us refer to the following improved version of the **market − MIP** as **market − MIP − cap**.

To reflect this rule in our optimization model, we introduce a new binary decision variable:

- $cap_{cg} \in \{0, 1\}$: Equals 1 if player c is selected as captain in gameweek g, and 0 otherwise.

This extension leads to a modified objective function:

$$\max \sum_{c \in C_\Omega} p_{cg} \cdot x_{cg} + \sum_{c \in C_\Omega} p_{cg} \cdot cap_{cg} - 4 \sum_{c \in C_\Omega} y_{cg} + 4z$$

The captain's predicted score is added again in the objective, effectively doubling their contribution. The remaining terms correspond to transfer penalties and the free-transfer bonus, as described in the baseline market-aware formulation. Let us recall that all the predicted scores p_{cg} are computed as in the mean-MIP-online, so retraining the ML model before each GW.

Additional Constraints for Captain Selection

Several constraints are added to ensure the correctness and realism of the captain selection mechanism:

- **(42) Unique captain constraint:**

$$\sum_{c \in C_\Omega} cap_{cg} = 1$$

  Exactly one player must be designated as the captain each gameweek.

- **(43) Captain must be in the starting lineup:**

$$cap_{cg} \leq x_{cg}, \quad \forall c \in C_\Omega$$

  A player can only be appointed captain if they are also selected as a starter.

- **(44) Bench constraints for expensive players:** To avoid budget inefficiencies, a soft heuristic is applied:

$$s_{cg} = 0 \quad \text{if } k_{cg} > 50$$

  Players with a high price are prevented from sitting on the bench unless strongly justified by the optimization.

- **(45) Vice-captain heuristic (post-optimization):** After the optimal solution is found, the vice-captain is selected as the highest-scoring starter (according to predicted performance) excluding the captain. This step is done heuristically and not encoded as a variable in the MIP, since vice-captaincy only takes effect in the rare case where the captain does not play.

Prediction Consistency Adjustment

In order to mitigate the risk of overvaluing players with uncertain availability, we introduce a penalty to the projected performance of players who played less than 10 minutes in the previous gameweek. This adjustment forces the model to favor consistent starters when selecting the captain or the rest of the squad.

Effectiveness and Results

This enhancement leads to better alignment with actual fantasy scoring rules and improves the realism of the generated optimal lineups. Empirical evaluations show that captaincy can lead to large score boosts (up to 20+ points in a single gameweek), and this model consistently identifies players with both high upside and strong likelihood of playing.

Figure 14 shows the empirical improvements when adding captain modeling compared to a baseline market-aware optimization.

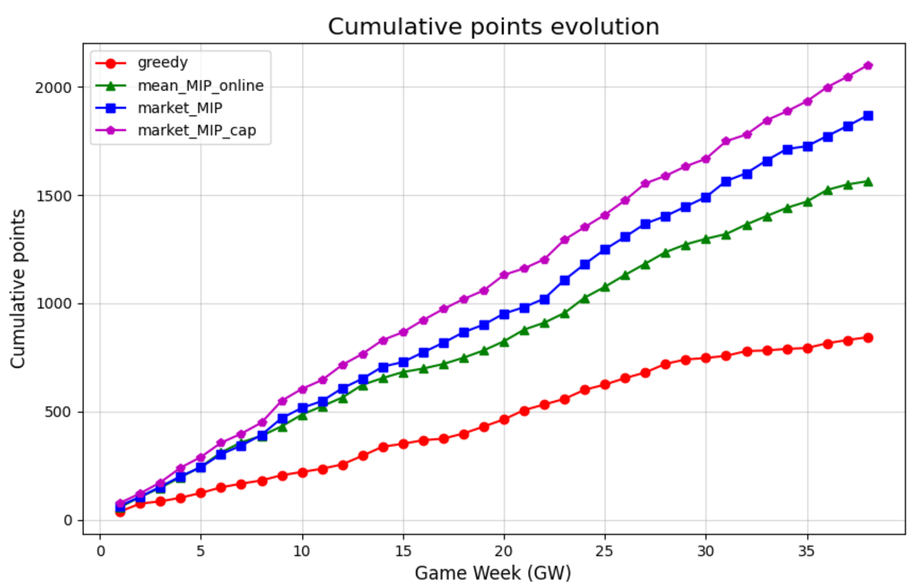

Figure 14: Impact of captain selection in market-MIP formulation

From the Figure 14 it can be appreciated the strategy improvement. The total achieved points are now 2140 with respect to the winner points 2799.

5 Model Evaluation: Real vs Predicted Distributions

To evaluate the quality of our performance prediction model, we plot the density functions of both the actual and predicted total points for all players across in a sample GW (the trend is overall the same over all season). Figure 15 shows the distribution comparison.

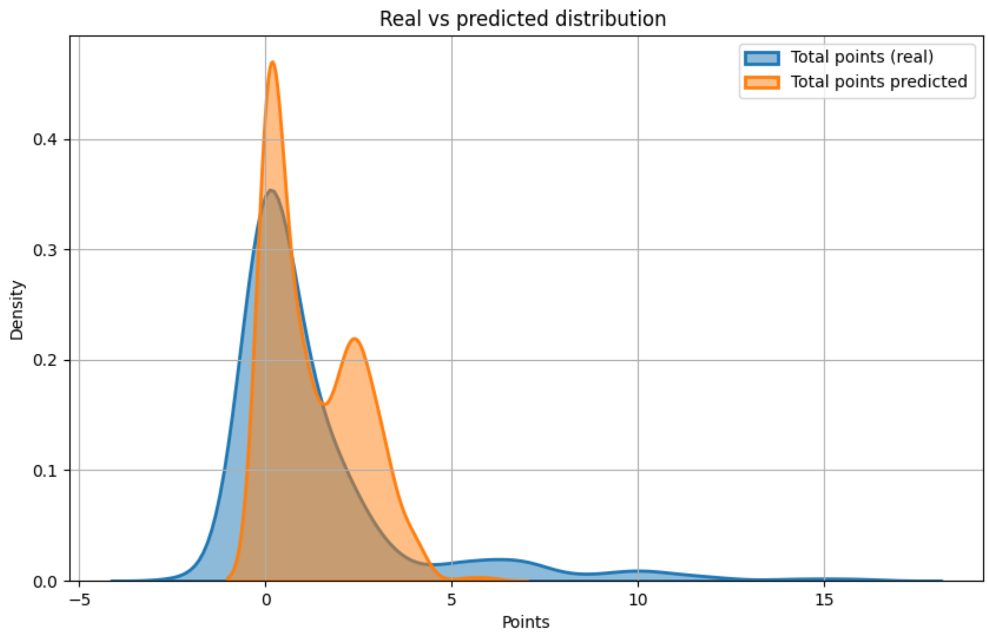

Figure 15: Density of real vs predicted fantasy points.

As observed, the predicted distribution is more concentrated around the mean with a sharper peak, suggesting the model tends to regress to the mean and underestimates the presence of outliers (i.e., players who greatly overperform). In contrast, the real distribution is slightly more spread out and includes a longer right tail, reflecting the fact that some players (e.g., top scorers like Haaland or Salah) accumulate substantially more points than the majority of the pool.

This mismatch highlights a limitation of the current prediction model: while it is effective at capturing the general trend and correctly predicting the majority of player outcomes, it lacks the capacity to consistently identify those few explosive performers who disproportionately influence fantasy outcomes. These are typically the players that successful FPL managers captain frequently or transfer in early based on form and fixtures.

To further compare the predictive approach with a theoretical optimum, we analyze the distribution of total points scored per Gameweek by our Market-based strategy versus the Perfect strategy (which has access to real GW data). Figure 16 shows that the Perfect strategy consistently achieves higher median scores and lower variance, while the Market strategy exhibits a wider spread and occasional underperformance. This confirms that while our model provides structure and stability, it does not fully capitalize on high-scoring opportunities.

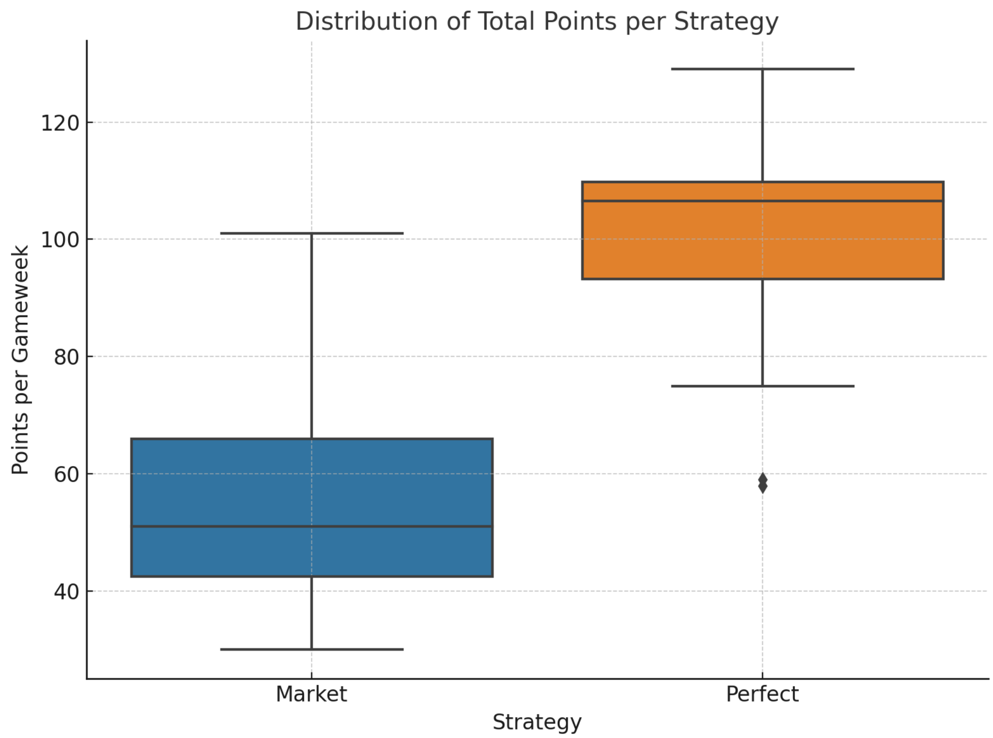

Figure 16: Distribution of total points per Gameweek across strategies

A complementary analysis in Figure 17 reveals the number of overlapping players (out of 11) in the weekly starting lineups between the two strategies. On average, only about 4 to 6 players are shared each week, indicating that even though our predictions are competitive, they diverge considerably from the optimal choices. This suggests that differences in projected performance value can lead to materially different selections and that the Perfect strategy identifies key performers that our model overlooks.

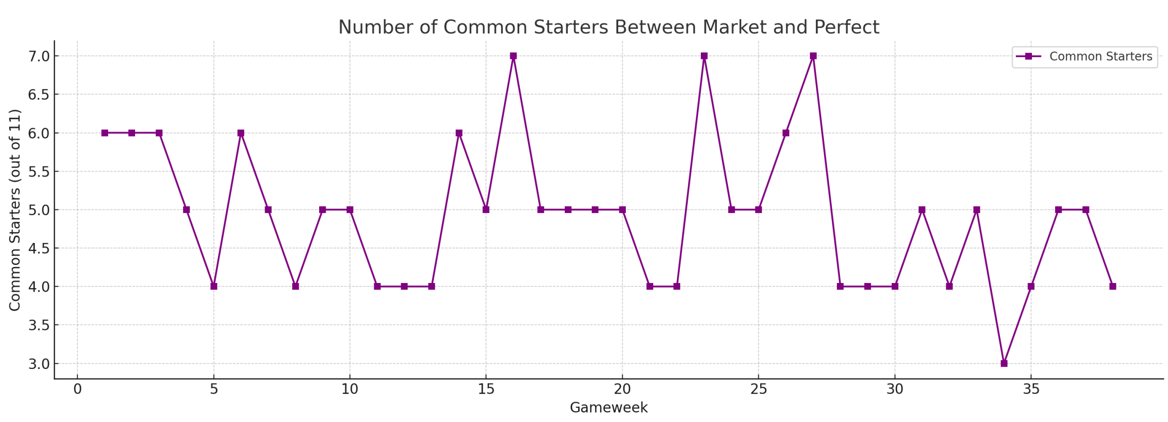

Figure 17: Number of common starters between Market and Perfect strategy across Gameweeks.

What is missing to win? Despite the notable performance of our optimization algorithms (e.g., reaching 2140 points with Market-MIP-Captain), the gap with the winning score of 2799 points remains significant. We identify several critical factors that need further exploration:

- **Outlier detection and exploitation**: Our model currently treats player performance as a relatively smooth, continuous signal. Winning FPL managers tend to anticipate outlier performances through intuition, tactical fixture analysis, and trend spotting. Integrating an anomaly detection model or a high-variance player classifier may help us to capture these turning points.

- **Adaptive captaincy strategy**: The captain rule can lead to weekly differences of 20+ points. While our MIP formulation models this explicitly, further gain could be achieved by combining predictions with upside potential (e.g., variance, ceiling outcomes), not just expected value.

- **Injury/news modeling**: Real-life FPL players follow Twitter, Reddit, or injury reports in real time. Our model does not react to last-minute changes. Introducing a real-time feed or probability of playing could improve our predictive reliability.

- **Psychological and behavioral modeling**: Many top managers adopt a long-term strategic perspective, planning chip usage and transfers several Gameweeks in advance. Integrating such foresight into our framework, through multistage optimization or reinforcement learning techniques, could help reduce the performance gap.

- **Risk-based decision making**: The Market strategy often avoids risky players with high upside due to uncertainty in the predictions. A robust decision-making pipeline should explicitly incorporate risk-reward trade-offs, potentially using stochastic or robust optimization models that allow for controlled variance in weekly scores.

- **Implement the chips dynamics**: Our current implementation does not account for the powerful mechanism of "chips" available to FPL managers, such as Free Hit, Bench Boost, Triple Captain, and Wildcard, which can drastically alter weekly and seasonal performance. Modeling chip usage within the optimization pipeline, especially through scenario analysis or multistage decision frameworks, could significantly enhance team selection flexibility and overall point maximization across the season.

6 Conclusions and Future Work

This thesis has explored a data-driven approach to decision-making in Fantasy Premier League (FPL) by combining predictive modeling with optimization techniques. Starting from simple greedy strategies and evolving toward more refined Mixed Integer Programming (MIP) models enriched with machine learning (ML) predictions and market dynamics, we developed a pipeline capable of simulating realistic FPL seasons with progressively higher performance.

The best-performing strategy, *market-MIP-cap*, which incorporates weekly ML predictions, market transfers, and captain selection, achieved a total of **2,140 points** in the 2023–24 Premier League season simulation, substantially outperforming static or heuristic baselines. However, this score still falls short of the top real-world manager performance (2,799 points), indicating that the proposed method, while effective, may still suffer from structural limitations.

To better contextualize the quality of our results, it is helpful to consider historical data from past FPL seasons. For example, during the 2020/2021 season, a score of 2,307 points corresponded to a global rank of 336,193 out of 8.24 million participants, placing that model in the top 4.08% of all players (Venter and Vuuren [2]). Based on this reference, and considering that approximately 11.5 million managers participated in the 2023–24 season, our score of 2,140 would likely correspond to a placement in the **top 6–7% globally**. This highlights the relative strength and competitiveness of our optimization framework, despite the existing gap from the absolute top scores.

To investigate the potential ceiling of our framework, we ran a set of offline experiments in which a single global optimization is performed using perfect foresight: the model is given access to the actual fantasy points each player will score across the season. This idealized scenario yields results that outperform the best online method by more than 250 points, demonstrating a remarkable performance gain when all decisions are taken holistically rather than sequentially. Specifically, the offline method using true future performance data achieved a total of 4885 points, compared to 4607 points obtained by the best online method when applied week-by-week on the same true data.

Importantly, we repeated these offline experiments not only using actual future performance data, but also using our machine learning-based predictions of future performance (i.e., predicted fantasy points for each gameweek). The result was consistent: the offline global optimization still outperformed the best online methods by approximately the same margin. In this case, the offline optimization using predicted data yielded 2491 points, again exceeding the best online predictive strategy (2140 as said before).

Notably, in both online approaches (based on real and predicted data), the score obtained in the first gameweek was higher than that of the offline strategy. This supports the correctness and realism of the experimental setup, as it illustrates that the offline method does not simply dominate at every step, but rather achieves superior cumulative performance by making globally optimized, long-term decisions.

This finding is critical and highlights a deeper insight: the key limitation is not only the quality of the predictions, but also the structure of the weekly optimization framework.

In other words, even when predictions are accurate and available for all future gameweeks, optimizing one gameweek at a time leads to suboptimal global outcomes. This suggests that short-sighted strategies, regardless of how informed they are, fail to capture important long-term trade-offs, such as the benefit of holding a player through a difficult fixture due to strong upcoming matches, or planning transfers to exploit favorable price dynamics.

This analysis motivates a shift in perspective: rather than treating each gameweek as an isolated problem, future approaches should model FPL as a sequential decision-making process over time. This opens the door to dynamic programming (DP), where decisions are optimized over a time horizon considering future states and transitions (e.g., available budget, player form, remaining chips). A DP-based formulation could, for example, optimize an entire season by precomputing or approximating optimal transfer paths, captain schedules, and chip usage plans.

Such a framework could be further extended by integrating:

- Stochastic programming, to explicitly model uncertainty in player performances or injuries.
- Reinforcement learning, to learn policies from past FPL seasons.
- Robust optimization, to make decisions that perform well across a range of possible scenarios, not just the expected one.

In conclusion, this thesis has demonstrated the strength of combining machine learning with optimization in FPL decision-making. However, the consistent performance gains observed in offline, globally optimized settings, both with real and predicted data, strongly suggest that future methods must move toward holistic, anticipatory, and sequential models. These will be essential to closing the performance gap with expert human managers and approaching the true optimal performance achievable within the FPL rules. This work was carried out as part of a collaborative effort between the University of Bologna and Cornell Tech University in New York City.

Acknowledgements

I express my sincere gratitude to my advisor, Prof. Valentina Cacchiani, and to my co-advisors, Prof. Andrea Lodi and Omar El Housni. Thank you for your invaluable guidance, encouragement, and support throughout this project. Your insights have been fundamental to my work, and you gave me the incredible opportunity to fulfill a lifelong dream: experiencing work and daily life in New York City.

This achievement is the result not only of my dedication and perseverance, but also of the unwavering support of my friends and family, who shared with me the highs and lows of this long academic journey.

To my family: thank you for showing me the right way to approach the challenges of everyday life. This master's degree was made possible thanks to your positivity and constant love.

To Ale: thank you for being my everyday accomplice, for all the small things that made a difference. Someone once said that "the devil is in the details", but I prefer to see in every detail the priceless love you have given me throughout our relationship. I love you.

To all my friends: I consider myself extremely lucky to have so many wonderful people in my life. Some of you are in Naples, some in Rome and Bologna, and others across Europe, the US, and around the world. Whether near or far, you have all played a fundamental role in my life. Over the years, I have learned that physical distance does not matter, your laughter and empathy have always been close to my heart.

I am very proud of this achievement, and I'm still countin'.

References

[1] Vijay Singal, Augustine Duru, and David M. Reeb. "The role of opponents in fantasy sports performance evaluation". In: *Journal of Economics and Business* 70 (2013), pp. 36–49.

[2] Van Zyl Venter and Jan H van Vuuren. "An optimisation approach towards soccer Fantasy Premiere League team selection". In: *Department of Industrial Engineering, Stellenbosch University* (2020). DOI: 10.5784/40-1-753. URL: https://doi.org/10.5784/40-1-753.

[3] Jesus Gonzalez, Felipe Navarro, and Marta Rodriguez. "Personalized player performance prediction for daily fantasy sports: An NBA case study". In: *Journal of Sports Analytics* 10.1 (2024), pp. 45–63. DOI: 10.1007/s41060-024-00523-y. URL: https://link.springer.com/article/10.1007/s41060-024-00523-y.

[4] LLC Gurobi Optimization. *Combining Machine Learning and Optimization Modeling in Fantasy Basketball*. https://www.gurobi.com/jupyter_models/combining-machine-learning-and-optimization-modeling-in-fantasy-basketball/. Accessed April 2025. 2023.

[5] Nittai Bergman and Dilip Roychoudhury. "Correlation Neglect in Fantasy Sports". In: *Management Science* (2022).

[6] Erica Johnson and Suraj Patel. "How cognitive biases shape fantasy football decisions". In: *Dominican University Research News* (2023). URL: https://www.dominican.edu/news/news-listing/professor-students-examine-how-cognitive-biases-shape-fantasy-football.

[7] Sergey Kozlov and Lina Wang. "Heuristics in highly competitive games: Evidence from fantasy NHL". In: *Games and Economic Behavior* (2024). Forthcoming. URL: https://www.researchgate.net/publication/380021883_Heuristics_in_highly_competitive_game_evidence_from_the_fantasy_National_Hockey_League.

[8] Jonathan Smith and Daniel Lee. "Optimal lineup selection in fantasy football using machine learning and integer programming". In: *ArXiv preprint* (2023). arXiv: 2309.15253 [cs.LG]. URL: https://arxiv.org/abs/2309.15253.

[9] A. Deshmukh, V. Mehta, and T. Agarwal. "Reinforcement Learning for Fantasy Sports: A Sequential Decision Framework for Cricket Team Selection". In: *ArXiv preprint* (2024). arXiv: 2412.19215 [cs.AI]. URL: https://arxiv.org/html/2412.19215v1.

[10] Premier League. *Fantasy Premier League*. Accessed: 2025-03-12. 2025. URL: https://fplchallenge.premierleague.com/home.

[11] Vaastav Anand. *FPL Historical Dataset*. Retrieved August 2022 from https://github.com/vaastav/Fantasy-Premier-League/. 2022.