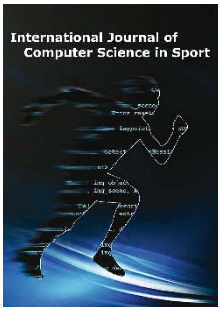
International Journal of Computer Science in Sport

Volume 19, Issue 2, 2020

Journal homepage: http://iacss.org/index.php?id=30

sciendo

DOI: 10.2478/ijcss-2020-0008



Optimising Daily Fantasy Sports Teams with Artificial Intelligence

Ryan Beal, Timothy J. Norman and Sarvapali D. Ramchurn

University of Southampton, University Rd, Southampton, SO17 1BJ

Abstract

This paper outlines a novel approach to optimising teams for Daily Fantasy Sports (DFS) contests. To this end, we propose a number of new models and algorithms to solve the team formation problems posed by DFS. Specifically, we focus on the National Football League (NFL) and predict the performance of real-world players to form the optimal fantasy team using mixed-integer programming. We test our solutions using real-world data-sets from across four seasons (2014-2017). We highlight the advantage that can be gained from using our machine-based methods and show that our solutions outperform existing benchmarks, turning a profit in up to 81.3% of DFS game-weeks over a season.

KEYWORDS: OPTIMISATION, MACHINE LEARNING, FANTASY FOOTBALL, NFL

Introduction

It is estimated that over 50 million people play fantasy sports in the US while over 5 million people regularly play the Fantasy Premier League in the UK alone. While in the UK the game is free, in the US, the average spend by each player is $467 per season, representing $15billion per year (Forbes, 2013). Fantasy sports impact on peoples lives as a recreation activity where people play in leagues against friends and have been shown to have helped grow the sports markets by bringing more television viewers to events (Nesbit & King, 2010). In recent years, Daily Fantasy Sports (DFS) have seen significant growth where players form teams for single game-weeks to compete in competitions to win money. Therefore, tools to help players form optimal teams would be of significant interest for many sports fans both to gain a competitive advantage against friends or win money in DFS leagues. Fantasy sports games require players to make predictions about the performance of real-world teams and individual members of these teams, and then optimise the set of team members that earn them points and win prizes. As such fantasy sports present significant computational challenges involving prediction and optimisation, where information about the teams and their players is typically limited to their historical performance data.

In this paper, we propose solutions to both the prediction and optimisation problems posed by fantasy sports. Building upon these, we aim to form the optimal teams and compare this approach to human based approaches. More specifically, we focus on the fantasy games based on the National Football League (NFL) in the US, where the market is dominated by DFS. In such games, given a set of rules (e.g., budget constraint, limits on members from the same team) players can create fantasy teams (of real NFL players) that earn points based on real-world performance. Prizes can be won on a daily basis based on the rules of the DFS game played (i.e., prize size, number of winners).[1] In particular, the optimal selection of team members is a typical combinatorial optimisation problem such as the work shown in (Matthews, Ramchurn, & Chalkiadakis, 2012).

Previous work on artificial intelligence for fantasy sports is rather limited, (Matthews, Ramchurn, & Chalkiadakis, 2012) developed algorithms to predict the performance of soccer teams and players in the Fantasy Premier League (FPL) for football. They employed reinforcement learning techniques to generate points predictions for teams and then derived individual player performance predictions. Moreover, they employed Mixed-Integer Programming (MIP) techniques to optimise the selection of player transfers over many weeks. Their approach was shown to outperform humans 99% of the time both in simulations and in the real-world.[2] In turn, in relation to DFS for NFL, (Sugar & Swenson, 2015) propose a number of regression-based approaches to make points predictions and demonstrate that points can be better predicted for some player positions than others. They show that these predictions can enable algorithms to outperform humans at the game. In general, while a number of prediction algorithms have been proposed for DFS for NFL, none of these address the optimisation of teams every game-week. Furthermore, it is as yet unclear which prediction technique works best for the NFL, given limited time series data about individual players.

Against this background, in this paper, we develop new algorithms to predict player performance as well as optimisation models for DFS that attempt to find the optimal team

[1] The English Premier League is another example of real-world leagues with associated fantasy sports but is not as significant in revenues as the DFS for the NFL.

[2] https://www.bbc.co.uk/news/av/technology-44070530/can-ai-pick-the-perfect-fantasy-football-team.

while accounting for limited human computation capabilities. We compare our results to those shown by the heuristics used by typical fantasy players against optimal algorithms. In more detail, this paper advances the state of the art in the following ways:

1. We propose a range of time series prediction techniques to predict player performance and benchmark them on a real-world data-set.
2. We propose a knapsack packing formulation for the DFS problem and solve it optimally using MIP techniques.
3. We model and evaluate human prediction and optimisation approaches against our techniques and a series of benchmarks from the literature. In doing this, we determine how prediction and optimisation, individually and in concert, influence outcomes.
4. We evaluate all our models on data from over four seasons from a well known DFS game and show that our approach is able to return a profit 81% of the time.

Our results show that the ability to model uncertainty and predict outcomes in DFS games has less of an impact on cash wins than the quality of the optimisation model. More importantly, this machine advantage can be, and probably is already being, exploited to win games against the majority of players. Thus, our work exposes interesting challenges for the industry, and establishes a baseline for the development of future games where machines may be able to perform all the legwork for human players.

The rest of this paper is structured as follows. Section 2 gives a background to the paper while Section 3 focuses on player points prediction. Then Section 4 focuses on the team formation problem, Section 5 describes the experiments we have run, Section 6 discusses our findings. Finally, Section 7 concludes.

Background

In this section, we first provide a brief overview of related work in AI applied to fantasy sports, as well as highlighting the papers that we aim to improve on with our methods. We then proceed to explain the key principles and constructs of a typical DFS game.

Related Work

The work by (Matthews, Ramchurn, & Chalkiadakis, 2012) established a set of machine learning and optimisation approaches to the computational challenges presented by fantasy sports. Their work focused on the Fantasy Premier League (FPL) for English football and extended the statistical model from (Dixon & Coles, 1997) and multi-knapsack packing optimisation techniques. Specifically, this work established baselines for sequential team optimisation problems using real-world data-sets. Building upon such work, and focusing on the NFL, work in (Sugar & Swenson, 2015) uses ridge regression, Bayesian ridge regression and elastic net algorithms for making point predictions. Moreover, (Landers & Duperrouzel, 2018) uses perceptron and boosted decision tree methods to predict player fantasy points. They use a brute force approach on a filtered data-set to select the optimal team each game-week. Our work differs as we develop machine learning algorithms for time series prediction specifically and also propose the first optimisation model for DFS. Furthermore, we compare and contrast against human performance. Finally, (Beal, Norman, & Ramchurn, 2019) discusses applications of artificial intelligence in sports, including a section on fantasy sports. This paper also exposes DFS and American football as open research areas on the sports analytics field.

Another area of DFS research has been focused around a discussion of the legality of DFS which has been widely debated across the US and historically gambling on games of chance has been illegal (Easton & Newell, 2019). Therefore, to aid the debate of the legality of DFS there has been a discussion on whether it is a game of skill in (Boswell, 2008) and (Meehan, 2015). These papers aim to show that DFS is a game of skill and should be legal in the US. We agree with these statements and the results in this paper would further support that conclusion.

Daily Fantasy Sports

This paper focuses on Daily Fantasy Sports where a team is selected with 9 positions, including 1 quarterback (QB), 2 running backs (RB), 3 wide receivers (WR), 1 tight end (TE), 1 kicker (K) and 1 team defence (DEF). Each team member has a price (in virtual money). Each week, A DFS player is given a positive budget B (e.g., \$60,000 in the FanDuel DFS game) to purchase a set of players, one for each position so that the total cost of players fits within B. Typically, the better the team member selected, the higher the salary value this team member will have. Once a team is selected, it is entered into a league. Points are awarded for how well the team members perform in the real-world and cash can be won in these leagues. Points are given to members for a number of different performance metrics.[3] In what follows, we use the term ‘players’ to denote members of the fantasy team for simplicity (not to confuse with participants of the DFS game).

Points Prediction

Points prediction is the first computational challenge addressed in this paper. We model the points prediction to focus only on the time series of individual players' fantasy scores from a given number of prior weeks. Note that this is an untested approach in this domain. We decide to use time series due to the importance of a players form when predicting their individual performance. This section discusses the features that are used in our model, defines the problem and outlines time series methods we use to solve the problem.

Features

The feature set, X, is formed using a number of prior weeks' fantasy scores.[4] The target, y, is the fantasy score that is being predicted and is in the corresponding row to the features in X. An example of the design matrices are shown in the equation below. This represents 4 previous games, where n is the game-week and x_n is the player points we aim to predict.

$$\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x_1} & \boldsymbol{x_2} & \boldsymbol{x_3} & \boldsymbol{x_4} \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{x_{n-4}} & \boldsymbol{x_{n-3}} & \boldsymbol{x_{n-2}} & \boldsymbol{x_{n-1}} \end{pmatrix} \boldsymbol{y} = \begin{pmatrix} \boldsymbol{x_5} \\ \vdots \\ \boldsymbol{x_n} \end{pmatrix} \tag{1}$$

Problem Definition

In order to predict the points that a player will obtain in a given game-week, machine learning techniques (ϕ) are trained using the feature set X and target set y. This will output the predictions of y using the features in corresponding rows of X. This gives us $\phi(x_i) = y_i$. In this paper we test a number of different techniques for the function ϕ. This allows us to find the best solutions for each fantasy position. We also test feature sets of different lengths and

[3] Full pointing system can be found on https://www.fanduel.com.

[4] We test to find the optimal number of prior weeks in each position (shown in section 3).

different machine learning techniques for ϕ in each DFS position for the final points prediction model. In what follows we outline the methods of ϕ that are tested.

Time Series Machine Learning Methods

Here we discuss the details of methods of ϕ that are tested to make the player point predictions. These methods are used with the feature set X and target set y as we have discussed. These techniques are applied using the scikit-learn[5]. package and results from their testing are shown in Section 5.

Linear Regression

The linear regression model (Seber & Lee, 2012) is shown in the equation below, an Ordinary Least Squares method is used to find the parameters.[6] Once the regression weightings (α, β) have been learned when training the model, they can be used to make the point predictions.

$$f(x) = \alpha + \beta x \tag{2}$$

where, $f(x)$ is our points prediction, x is our feature row consisting of previous game-week scores, α and β are the weightings learned by the machine learning method.

Radial Basis Functions (RBF)

RBF as discussed in (Park & Sandberg, 1993), makes predictions based on the distance a point is from a number of centres. The formula for the RBF model is shown in the equation below.

$$f(x) \ = \sum_{j=1}^{M} \lambda_j \phi(|x - m_j|) \tag{3}$$

where, $f(x)$ is our points prediction, M is the number of centres, λ is the weight for each centre, m_j is the centre point and x is the player's prior game-week points. Training data is used with K-means clustering to find the centres and the lambda weights.

Recurrent Neural Networks (RNNs)

RNNs and in particular Long Short Term Memory (LSTM) methods have found success for predicting time series based data in the past, as shown in (Gers, Schmidhuber, & Cummins, 1999). As LSTMs have a particular architecture, it allows the learning of long-term dependencies. The architecture is shown below in Figure 1.

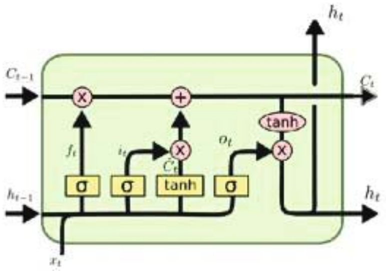


Figure 1: LSTM Archtecture[7]

[5] https://scikit-learn.org.

[6] Other methods such as Ridge and Lasso regression were tested but performed no better.

[7] https://colah.github.io/posts/2015-08-Understanding-LSTMs/.

We use this LSTM approach for the points prediction regression problem that we are trying to solve. In an LSTM network, there are three gates:

- **Forget Gate Layer:** The forget gate layer (f_t) decides what information must be discarded. It will consider the inputs h_{t-1} and x_t, and for each element present in the previous cell state (C_{t-1}), will decide to keep or discard that element.
- **Input Gate Layer:** The input gate layer (i_t) decides what information will be stored in the current cell state (C_t). The current cell state is calculated using a $tanh$ layer that creates a vector containing the new values C_t.
- **Output Gate Layer:** The output gate layer (o_t) decides the output (h_t) to transmit to the next copy of the network. This output is calculated using a sigmoid layer, which decides which parts of the current cell state will be kept. The output of the sigmoid gate is multiplied by the cell state, in this way the output will be formed by the only parts that the algorithm has decided to keep.

The final output points prediction (which in this case is h_t), will be calculated using the following equation:

$$h_t = o_t * tanh(C_t) \tag{4}$$

Random Forest

A random forest model from (Breiman, 2001) is formed with a collection of different tree predictors where x is the prior game-weeks player points, $h(x, \Theta)$ is the individual tree's output and Θ is a random vector generated, independent of the past random vectors but with the same distribution.

$$f(x) = \frac{\sum_{k=1}^{F} h(x, \Theta_k)}{F} \tag{5}$$

The point prediction $f(x)$ is given by taking an average of the collection of tree predictor outputs and F is the number of trees in the forest.

Team Formation Optimisation

The following section focuses on the second computational challenge: the team formation optimisation. We will discuss the constraints that exist and define the problem as a mixed-integer-program (MIP). Assuming we have high quality predictions, the challenge of optimisation is key to succeeding in DFS.

This is a challenging problem as there are a number of constraints and issues that must be addressed, these include the following: generating selections that fit constraints, evaluating the selections against a budget, and generating a set of players that maximises the points. We can also look at the uncertainty in our teams/predictions so we can form teams of different risk levels. As shown in this section, we develop an MIP algorithm for this problem (which is a novel application of these techniques in this domain). Our MIP solution finds a team which guarantees that we select the optimal players within the given constraints based on the player points predictions set out in the previous section.

Problem Definition

There are a full set of possible players[8], Z, to select from which is of length L. The team formation problem has a number of constraints - we are aiming to form a team that contains 9 players each filling one of the fantasy positions while keeping the total salary of the selected players below a budget, B. Our objective is that the selected team will maximise the number of predicted points. We also consider that any team would need to contain players from 1 or more unique NFL teams. To meet the constraints of the problem, we need 3 properties for each player:

- Player positions - to ensure we fill the slots in a team. The DFS positions the team needs to fill are: 1 quarterback, 3 wide-receivers, 2 running-backs, 1 tight-end, 1 kicker and 1 defence.
- Player NFL teams - to ensure we have more than 1 different NFL team represented in the fantasy team.
- Player salary - to ensure that the team's total salary is below the given budget (B).

We also need the player's points prediction which is the value we are aiming to maximise to give the optimal predicted team under the given constraints.

Mixed-Integer Programming Approach

The mixed-integer programming approach has proved to be successful in (Matthews, Ramchurn, & Chalkiadakis, 2012) when being used for fantasy sports in football/soccer. We base our MIP solution on the 1-0 knapsack packing problem (Chu & Beasley, 1998). We use a yes/no decision variable to represent if a player is selected or not. Each player in Z is represented by $z = \{1,0\}$, 1 meaning selected in the fantasy team and 0 meaning not selected. These decision variables can be used to set the constraints in a numeric format. We formulate this as an MIP and the problem is defined as (for a set of players of length L):

$$\begin{aligned} maximise & \sum_{i=1}^{L} (y_i . z_i) \\ s.t. & \sum_{i=1}^{L} (w_i . z_i) \le B \\ & \sum_{i=1}^{L} z_i = 9 \\ & \sum_{i=1}^{L} (qb_i . z_i) = 1 \\ & \sum_{i=1}^{L} (wr_i . z_i) = 3 \\ & \sum_{i=1}^{L} (rb_i . z_i) = 2 \\ & \sum_{i=1}^{L} (te_i . z_i) = 1 \\ & \sum_{i=1}^{L} (k_i . z_i) = 1 \\ & \sum_{i=1}^{L} (d_i . z_i) = 1 \\ & |C_t| > 1 \end{aligned} \tag{6}$$

[8] Injured players are removed so a subset of players who are not on the injury list can be created.

As well as z there are four variables for each player so the constraints in the above equation can be met:

- Player salary: $w \in \mathbb{Z}$ which represents player value.
- Player position: $p \in \{QB, WR, RB, TE, K, DEF\}$. Each position will have its own integer set, where the set contains a 1 if a player is that position and a 0 if not. This means we have 6 new sets $qb, wr, rb, te, k, d \in \{1,0\}$.
- Player team: $t \in \{arz, atl, bal \ldots ten, was\}$. Where t represents one of the 32 NFL teams. C_t is the set of different real world teams in the selected fantasy team.
- Player points prediction: $y \in \Re_0^+$, predicted number of points the player will obtain.

The "subject to" (s.t.) part of the equation represents the use of the decision variable z_i to meet the constraints of the DFS problem. The objective function is the sum of player points (y_i) which are summed if $z_i = 1$ and not summed if $z_i = 0$. The first constraint represents the sum of the selected players' salaries (w_i) should be less than or equal to the given budget (B). The second constraint is the total number of players in the team. The following six constraints represent the number of players allowed for each fantasy position where the variables qb_i, wr_i etc are also binary values representing if a player is that position or not (e.g., number of WRs = 3). Finally, the team will be checked to ensure that the number of real-world NFL teams represented in the fantasy team (C_t) is greater than 1.

Memory usage and solution-time rise exponentially as more integer variables are added. In an average NFL week there are between 400-450 players to be considered in the data set meaning that there are an estimated 3.5×10^{10} possible line-ups. The MIP model that is created will be able to handle these in a reasonable time frame and find the optimal solution.

Experiements

This section outlines the experiments which tests the performance of our DFS model and compares our machine-based model to human performance in Experiment 5.[9]

Experiment 1: Points Prediction Methods

The first test runs the machine learning methods (discussed in Section 3) on each position to obtain root mean squared errors (RMSE). The 2014-17 FanDuel data-set is used, with 70% of the data being the training set and 30% being test set (10-fold cross validatation). Different numbers of prior game-weeks features are tested for each position. To create the final model used in this paper, we select the best performing algorithm in each position.

Positions Performance

We break down the RMSE result for each of the DFS positions. We test linear regression (LR), LSTMs, radial basis functions (RBF) and random forest (RF) using a different number of weeks as an input feature (between 2-6 weeks)[10]. Table 1, shows the top results for each position for the data that is tested. Only the best result for each of the methods is shown with

[9] Experiments run using Pythons Scikit-Learn framework and with FanDuel data for 4 seasons (2014-17) scraped from - http://rotoguru1.com. For reference, the highest score a player achived in each posistion in a single gameweek was QB=37.8, WR=30.4, RB=32.1, TE= 17, K=17, DEF=21.

[10] We tested 2-6 weeks for feature data as 1 game week would not be enough features for the model and anything more than 6 weeks we found the results dropped off significantly due to changes in form/playing time.

the number of previous games data that was used in the feature set (shown in Table 2).

The results show how we are able to be much more accurate for tight-ends (TE), kickers (K) and defence (DEF) than what we can be for the quarterback (QB) and running backs (RB). This is most likely due to the swing in points that QBs and RBs can have each week as they are the most likely to obtain more points when they perform well, whereas a kicker is limited to how many opportunities he has to produce points.

Final Model Selection

We design the final points prediction model by selecting the best performing algorithm and number of input weeks for each position. The final selection is shown in Table 2 with the number of prior games used in the feature set of that model (# Games).

Table1: RMSE evaluation of method by position (where lower is better).

Position	**LR**	**LSTM**	**RBF**	**RF**
QB	**6.89**	7.62	7.82	6.91
WR	**4.39**	7.45	7.38	4.51
RB	7.09	6.58	7.34	**6.07**
TE	**4.10**	4.51	6.49	4.12
DEF	**3.85**	3.91	5.43	4.00
K	3.5	4.04	3.62	**3.28**

Table 2: Final Algorithm Selection.

	QB	**WR**	**RB**	**TE**	**DEF**	**K**
Algorithm	LR	LR	RF	LR	LR	RF
# Games	4	3	5	5	3	4

Experiment 2: Comparison to Player Points Prediction Benchmarks

We compare our results to the results shown in (Sugar & Swenson, 2015) using the same data which is used to test their model from the 2015 NFL season. Their results are also compared to Yahoo's fantasy point prediction for the same dataset. Figure 2 shows the comparison of the RMSEs for each of the positions against the (Sugar & Swenson, 2015) and Yahoo models.

We found that the total average of the models discussed in this paper could be used to improve the previous work. As shown in Figure 2, we found our model achieves a lower RMSE than those shown in (Sugar & Swenson, 2015), with the average being 16.9% lower than the Yahoo average and 15.9% lower than the average found by their model.

Our models produce a lower error rate for all positions apart from running back where we perform 0.5% behind Sugar and Swenson's model (labelled S&S in Figure 2) and 5% behind the Yahoo model. This may be due to running backs achieving a high number of points on a more sporadic basis in comparison to the other positions meaning that the time series forecast is not as accurate.

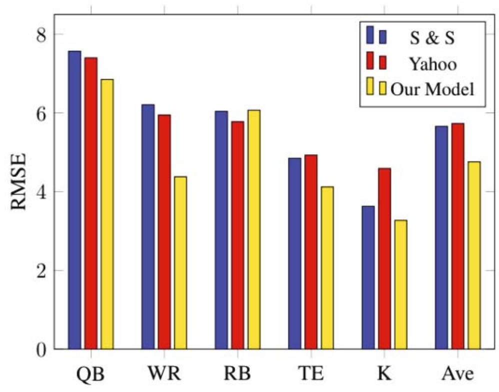


Figure 2: Comparison Against Results Sugar and Swenson [2015] where lower is better.

Experiment 3: Team Optimisation Methods

To compare the MIP and the filtered brute force approach that is used in (Landers & Duperrouzel, 2018), both were implemented and tested to find a team in every normal season game-week across 4 seasons between 2014-17. The MIP problem is solved using an Integer Programming solver - IBM ILOG's CPLEX 12.3. For each week 100 teams were found using each of the approaches. This test was run using the actual FanDuel points scores along with the player salaries for the corresponding week. We record both the run-time of the method and total points that the selected team obtains for each of the game weeks.

The average run-time for the MIP approach was 18.5ms and the brute force approach was 396.0ms. A student t-test was conducted to compare the run-times of the two approaches. There was a significant difference in the times for the MIP optimiser ($\mu = 22.00$, $\sigma = 3.52$) and brute force ($\mu = 1143.3$, $\sigma = 52.27$) conditions; $t = -88.2437, p \ = 8.4e^{-40}$. This highlights that unsurprisingly the MIP team-optimiser runs significantly faster than the brute force approach using the filtered set of players. In terms of the points that the algorithms would have selected each week, they are equal on most occasions. However, there are a number of occasions where the MIP algorithm scores more points as it ensures that the team is optimal by considering all possible players and not just a filtered set.

Experiment 4: Team Optimiser using Point Predictions on FanDuel

This test evaluates the model's performance as if it was used in an actual DFS league contest. This combines the two computational challenges of DFS, bringing together the points prediction and team optimisation models. In this test, predictions are calculated for every player in every week using the final model discussed in Section 5. We train the model using 2015 FanDuel data. This produces a set of players who all have a corresponding points prediction and salary value for each week. We use this to form a predicted optimal team for each week and the actual FanDuel total is calculated, this is shown in Figure 3. A threshold for making a profit is set at 111.21 for the 2016 season, this is our "cash line".11 This value is taken from (Landers & Duperrouzel, 2018) where the cash-line value is used to evaluate their models, we can then compare our results in the same game-weeks as the models outlined in that paper.

[11] We are unable to find exact FanDuel league data, the average threshold is used instead.

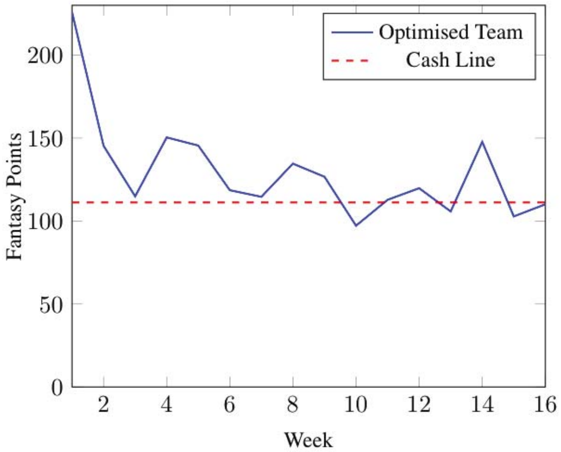


Figure 3: 2016 Week by Week Points.

We compare this with the work in (Sugar & Swenson, 2015; Landers & Duperrouzel, 2018). These models are tested on weeks 3-9 (to give a fair comparison to the test performed in the previous papers) and compared using the percentage of weeks where the model achieved points above above the cash line. Sugar and Swenson [2015] achieves a percentage of 71.4% and (Landers & Duperrouzel, 2018) achieves a percentage of 82.0%. Our model in this paper is able to better both of those as it achieves 100% success rate for those weeks. On average, across the whole season (weeks 1-16) the number of points each week scored on average was 124.94 and the percentage success rate is 81.3% showing that if we were to have run out models in the 2016 season we would have expected to produce a profit in 81.3% of the game-weeks.

Experiment 5: Modelling Human DFS Players

We test 4 models using AI and human based methods to asses the complex tasks within the DFS problem. This allows us to compare how an average human performs in DFS tasks with our models, we can then extrapolate the key skills that are needed for successful DFS teams and show what humans/computers are good/bad at in terms of DFS.

Human DFS Players

We model human approaches to optimisation and predictions. This is due to the absence of individuals' data from FanDuel and the large variation in prediction tactics. We do this by referring to principles proposed by human experts in the field.[12]

- **Optimisation:** We simulate a human approach to optimisation by selecting teams using the top 10 predicted high scoring players for each position and then selecting a random team that fits the constraints from the filtered set of players. This is a typical approach recommended by DFS experts. We do this 1000 times for each game-week and take an average of the performance. This is how a human would pick a team if given predictions, randomly slotting high performing players into positions until they find a team that fits the budget constraint.
- **Prediction:** To simulate a human's approach to points prediction we base our model on what competitors typically use on FanDuel. They provide an average metric on their

[12] 10 Things DFS Players Should Never, Ever Do (Point 7) - https://bleacherreport.com/articles/2408489-10-things-dfs-players-should-never-ever-do. The uncertaintiy of this is a limitation to this experiement, it would be of value to test against real-world data on DFS player team selections.

site "Fantasy Points per Game (FFPG)". Therefore, we use this metric to give an idea of how an average human would select high performing players.

Human-Machine Models

Using the assessment of how humans create DFS teams we test the following models:

- **All AI:** Using the MIP team optimisation model with the machine-learning player points prediction model.
- **AI Optimisation:** Using the MIP team optimisation model with human approach to points prediction
- **AI Predictions:** Using the machine learning prediction model with the human approach to optimisation.
- **No AI:** Using the human points prediction and team optimisation approaches. This represents a humans performance with an average understanding of the game.

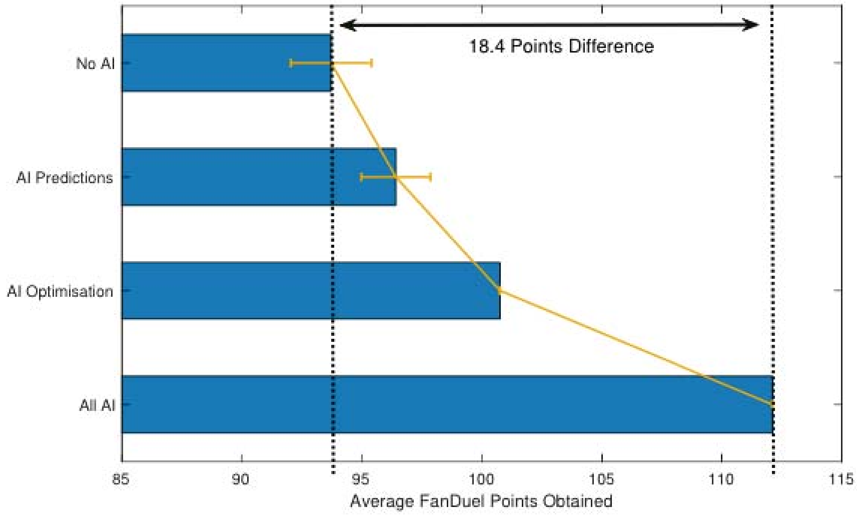


Figure 4: Average Points Comparison Between AI and Human Models over 4 seasons.

We test these methods across 4 NFL seasons of FanDuel data (2014-2017). The results are presented in Figure 4, and show that on average across the 4 seasons the "All AI" model performs better than the other models which we would expect. The error bars indicate 95% confidence intervals for the random team formations, therefore these are not plotted for the models that use the MIP team formation. We also find that when using the 2016 season cash line of 111.21 points the All AI model produces a profit 81.3% of the time, AI optimisation 47.1%, AI predictions 23.5% and No AI 11.8%. Interestingly, the model using good optimisation (with average predictions) outperforms the the models that use the "human" style of optimisation. From this, we conclude that humans perform particularly poorly in optimising team selection in DFS. This would suggest that if an expert human was to make points predictions and use the MIP optimisation algorithm to select a team, they could improve their DFS performance.

Discussion

In general, our results show that an MIP team formation model can be used to outperform the simulated human resulrs, using even the simplest prediction techniques. Particularly, the models ability to select the optimial team within the constraints of the game means that it would make DFS a game of performance prediction rather than optimisiation. We have identified that the optimisation player selection within the DFS constraints is the main skill associated with successful DFS teams. The player points prediction though has more luck associated with it. Therefore assuming, in the future, such an optimiser as presented in this paper were available to all DFS participants, the winners would be those who provide the best player predictions. However, due to uncertainties in player performance, if all participants have access to the same data-sets and optimiser, they would still be unlikely to predict the same outcomes.

We have also shown that our model results provides the baseline of academic research in this area as we have shown evidence of our methods outperforming the ones that were available for comparison. We were able to outperform the results shown in both (Sugar & Swenson, 2015) and (Kapania, 2012). It is worth noting though that there are many online optimisers and prediction tools for DFS, however as their methods and results are not published we are not able to compare. However, in future work this is something that could be explored. We would also like to run our models over an entire NFL season week-by-week to obtain a better understanding of how the models would perform in the real-world rather than just by back testing.

Our work also has broader applicability beyond the domain of fantasy sports. Indeed, others (Dang et Al., 2006; Ramchurn et Al., 2010; Chalkiadakis & Boutilier, 2012) have shown how such techniques could be extended to solve problems in choosing the best sensors to surveil an area, dispatching optimal teams of emergency responders and selecting appropriate sets of agents to work within a coalition formation problem. Due to the nature of the problem and data provided by fantasy football, it presents a live test-bed environment that can be utilised to test further machine learning and optimisation algorithms for sequential team formation in the real-world. This is due to the wealth of historic data that is available over a number of seasons which is being constantly updated as every new NFL game-week passes and because of the real world feedback (in the form of points and actual player performance) which can be used to validate the performance new methods.

Future work will focus on exploring approaches that improve the accuracy of our prediction model. In particular, the use of NFL match outcome predictions to aid our player points prediction, as players in a winning team are more likely to achieve a higher number of points and this would factor in difficulty of the players opposition. There are also a number of open research questions remaining in the DFS space. Such as exploring the affect of adding levels of risk/uncertainty to DFS sports team. As multiple DFS teams can be entered into leagues, by having the uncertainties this would present an interesting portfolio optimisation problem where by entering a number of different teams at different risk levels we can maximise the chance of returning profits. Further challenges are discussed in (Beal, Norman, & Ramchurn, 2019).

Conclusion

In this paper, we developed a number of machine-based models to solve the problems posed by DFS games. We prove that we are able to improve on the previous published models in this space by 15.9% on average for points prediction and by 377.5ms for the optimisation run time. We also show that we can improve on an average human approach by 2.8% when adding machine learning predictions, by 7.5% when using a MIP optimisation and by 19.7% when using both together. When using both machine-based models together (“All AI model”), we found that in 2016 a profit would have been made in 81.3% of the game-weeks.

The machine-based approach demonstrates an increase in performance in comparison to the average human performance over a number of years worth of data from FanDuel. This is particularly the case for the team optimisation challenge in DFS, as this is where the machine-based model makes the greatest improvement to the performance. Overall, this paper shows baseline models for the used of AI for DFS prediction and optimisation in the NFL, improving on previous published results.

Acknowledgements

We would like to thank the reviewers for their comments. This research is supported by the AXA Research Fund and the EPSRC NPIF doctoral training grant number EP/S515590/1.

References

Beal, R., Norman, T. J., & Ramchurn, S. D. (2019). Artificial intelligence for team sports: a survey. *The Knowledge Engineering Review*.

Boswell, J. (2008). Fantasy Sports: A Game of Skill That Is Implicitly Legal under State Law, and Now Explicitly Legal under Federal Law. *Cardozo Arts & Entertainment Law Journal*, 1257-1278.

Breiman, L. (2001). Random Forests. *Machine Learning.*, 5-32.

Chalkiadakis, G., & Boutilier, C. (2012). Sequentially optimal repeated coalition formation under uncertainty. *Autonomous Agents and Multi-Agent Systems*, 441-484.

Chu, P. C., & Beasley, J. E. (1998). A Genetic Algorithm for the Multidimensional Knapsack Problem. *Journal of Heuristics*, 63-86.

Dang et Al., V. (2006). Overlapping coalition formation for efficient data fusion in multi-sensor networks. *Proceedings of the 21st National Conference on Artificial Intelligence*, (S. 635-640).

Dixon, M., & Coles, S. (1997). Modelling association football scores and inefficiencies in the football betting market. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 265-280.

Easton, T., & Newell, S. (2019). Are daily fantasy sports gambling? *Journal of Sports Analytics*, 35-43.

Forbes. (2013). *Forbes*. Von https://www.forbes.com/sites/briangoff/2013/08/20/the-70-billion-fantasy-football-market/#1c340385755c. (accessed on: 20/06/2020).

Gers, F., Schmidhuber, J., & Cummins, F. (1999). *Learning to forget: Continual prediction with LSTM.* IET.

Kapania, N. (2012). *Predicting Fantasy Football Performance with Machine Learning Techniques*. Von http://cs229.stanford.edu/proj2012/Kapania-FantasyFootballAndMachineLearning.pdf (accessed on: 20/06/2020).

Landers, J. R., & Duperrouzel, B. (2018). Machine Learning Approaches to Competing in Fantasy Leagues for the NFL. *IEEE Transactions on Games*.

Matthews, T., Ramchurn, S., & Chalkiadakis, G. (2012). Competing with Humans at Fantasy Football: Team Formation in Large Partially-observable Domains. *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, (S. 1394-1400).

Meehan, J. C. (2015). The Predominate Goliath: Why Pay-to-Play Daily Fantasy Sports are Games of Skill under the Dominant Factor Test. *Marquette Sports Law Review*, 5.

Nesbit, T., & King, K. (2010). The impact of fantasy sports on television viewership. *Journal of Media Economics*, 24-41.

Park, J., & Sandberg, I. W. (1993). Approximation and Radial-Basis-Function Networks. *Neural Computation*, 305-316.

Ramchurn et Al., S. (2010). Coalition formation with spatial and temporal constraints. *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems*, (S. 1181-1188).

Seber, G. A., & Lee, A. J. (2012). *Linear regression analysis.* John Wiley & Sons.

Sugar, G., & Swenson, T. (2015). *Predicting Optimal Game Day Fantasy Football Teams*. Von http://cs229.stanford.edu/proj2015/115_report.pdf (accessed on: 20/06/2020).