Abstract

Contents

1 Introduction **1**
1.1 Motivation 1
1.2 Aims 1
1.3 Methodology 1

2 Foundations **2**
2.1 Fantasy Leagues 2
2.2 SPITCH 5

3 Literature Review **9**

4 Implementation **19**
4.1 Business Context 19
4.2 Data 22
4.2.1 Procurement 22
4.2.2 Processment 25
4.2.3 Exploration 29
4.3 Models 40
4.4 Application 40

5 Conclusion **41**

List of Figures **A**

List of Tables **B**

Bibliography **C**
A.1 Diagrams . . . . . H

A.2 Tables . . . . . H
A.3 Screenshots . . . . . H
A.4 Graphs . . . . . H

Decleration of Authenticity **I**

Chapter 1

Introduction

1.1 Motivation

1.2 Aims

1.3 Methodology

Chapter 2

Foundations

2.1 Fantasy Leagues

Fantasy Leagues can look back on a history of over 60 years. Wilfred Winkenbach, a sports entrepreneur and enthusiast from the USA, designed a fantasy golf game in the 1950s. In this game, a team was made up of several golfers, and the team with the lowest swings in total won. Building on the success of this game, Winkenbach developed the first fantasy football league in 1962, which is similar to today's fantasy leagues. (cf. Green, 2014) This league consisted of 8 participants, friends or co-workers of Winkenbach, who met in a restaurant and wrote down their line-up for the coming season. The scoring system was kept very simple and was limited to the main events in a football game: touchdowns, field goals and interceptions. The simple reason was that each event had to be counted by hand by the game master. (cf. Fabiano, 2007) From this game, leagues quickly developed in other sports, such as baseball. One of the reasons this type of game first spread in the USA is the ease of assigning points to individual actions in the popular represented sports. For example, during an attack in American football, there are several plays, separated by pauses in which it can be assessed relatively clearly, for example, by the yards gained or lost, whether the play was successful. In soccer, on the other hand, there are fewer interruptions, plus unlike in baseball or American football, there are no intermediate milestones that can be reached between moves. These missing pauses lead to a more wild game, with difficult to evaluate actions. In addition, although there are roles within a soccer team, these roles, except the goalkeeper, are more strategic and do

not restrict the players in their playing actions. A defender can score goals or intercept passes just as well as a striker. In contrast, in American football or baseball, each player often has one single task per turn that can either succeed or not. All these factors did sports like soccer challenging, if not impossible, to implement as fantasy leagues.

However, with the advancement of modern image recognition technologies and player tracking devices such as two high-resolution cameras per playing side, these times are a thing of the past. (cf. Hoffmann, 2014) Nowadays, every event on a soccer pitch is automatically trackable and therefore offers the possibility to evaluate the performance of different players much more accurately. These advances allow fantasy soccer leagues to exist, as they can build their game on this data basis.

Nevertheless, the main goal of fantasy leagues is always the same for all sports: assemble a team that performs best. However, there are differences between the fantasy leagues in how this performance is evaluated. One major factor for this difference originates in the differences in the sport disciplines themself. Despite that, even leagues in the same discipline can differ to create a unique selling point. Since this work is primarily focused on soccer, the following considerations are limited to fantasy soccer leagues. Furthermore, the decision was made to use the provider SPITCH, which only offered the first Bundesliga at the time of writing. That is why all comparisons are made concerning this game system.

The list of differences between the individual fantasy soccer leagues is long. Each provider of a league wants to have its unique selling point and highlights different tactical elements. It is not the aim of this thesis to show all the differences. This section merely serves to give a rough overview of the world of Fantasy Soccer Leagues and, in addition, to show that each game must be approached strategically differently and, as a result, different questions must be asked. The research in this paper, therefore, applies primarily to the game SPITCH. Nevertheless, this thesis aims to shed light on problems in as general a manner as possible and to be useful for research in similar areas.
A major difference in fantasy soccer is the national league in which the fantasy league is located. For example, there are providers for the English *Barclays Premier League*, the Italian *Serie A* and for the German *Bundesliga*, the latter being observed in this work. However, it is not only the selected national league that differentiates the various providers. Furthermore, a differentiation can be made between the availability of players. For example, one popular game mode exists where 20 fantasy managers compete against

each other, similar to the actual real-world competition. Each fantasy manager is assigned a team of random players. These own players can be traded with the other 19 teams for other players or money on a virtual transfer market. It is important to note that this game mode creates a segregated space, where the participants compete against each other continuously over an entire season. On top of that, each player exists **only once** and can only play for one fantasy manager's team at the time. In contrast, in SPITCH, any player can be bought and used by any fantasy manager. Furthermore, the participants do not play in a segregated space but with unlimited opponents. Further details and the general regulations are described in the following chapter.

2.2 SPITCH

This section intends to provide the necessary rules from SPITCH needed to understand the optimization problem at hand. Additionally, this section aims to outline the first approaches to a possible solution.

As already mentioned in the previous chapter, SPITCH is a provider for fantasy soccer leagues. (cf. SPITCH, 2021c) At the beginning of writing, SPITCH only provided competitions for the German *Bundesliga*. Until now, numerous national football associations from different countries joined. Furthermore, football managers can these days compete in various other game modes. This thesis solely deals with the traditional game mode for the German *Bundesliga*.
To counteract confusion that may arise, the following terms and their meaning in the context of this work, such as player or manager, are explained in more detail. Furthermore, each word is assigned a variable that will help understand the calculation of scores more quickly.

Table 1: SPITCH Glossary

Term	**Variable**	**Meaning**
Manager	M	Participants of SPITCH
Player	P	Real soccer player, i.e. Manuel Neuer
Value	V	Transfer market value of a Player P
Event	E	In-game events such as Goal, Pass, Unsuccessfull Pass etc.
Points	p	Points according to SPITCH points catalogue
Score	S	Sum of points p
Round	R	Game-Round, e.g. matchday
Line-up	L	Line-up consisting of 11 players P

Like most fantasy leagues, the aim in the traditional game mode is to line up a team that performs best. Unlike most fantasy leagues, the managers M in a SPITCH competition only assemble a line-up L for the upcoming match day. So when planning the line-up, it is not necessary to think long-term for the entire season. A new line-up consisting of different players can be created for each round R. Each line-up consists of 11 out of 711 possible players P. Each player P_i, $\{\, i \mid i \in \{1, 2, ..., 711\} \,\}$ has a score S_{PR} for each of the 34 rounds R_j, $\{\, j \mid j \in \{1, 2, ..., 34\} \,\}$. For simplicity, as the rounds are separated and thus do not influence each other, the following declarations are all round-specific. The final line-up score S_L is the sum of 11 individual player scores S_{P_i} :

$$S_{\mathrm{L}} = \sum_{i=1}^{11} S_{\mathrm{P_i}} \tag{1}$$

This score S_L is used to create a ranking of managers M and therefore decides if the manager wins a prize or not. The individual player score S_{P_i} is calculated using the occurred events O during a match multiplied with their corresponding points p given by the SPITCH points catalogue. It exists a number of 33 different event types, such as pass, goal or tackle, therefore E_k, $\{\, k \mid k \in \{1, 2, ..., 33\} \,\}$ applies. For example, a pass is granted two and a goal 200 points. For negative event types, such as a missed chance, negative points can also be awarded. (cf. SPITCH, 2021a) Hence, a player can have a negative score. The individual player score can be calculated using the following equation:

$$S_{\mathrm{P_i}} = \sum_{k=1}^{33} O_{\mathrm{k}} * p_{\mathrm{k}} \tag{2}$$

Given equations (1) and (2), the final line-up score S_L can be calculated using:

$$S_{\mathrm{L}} = \sum_{i=1}^{11} \sum_{k=1}^{33} O_{\mathrm{ik}} * p_{\mathrm{ik}} \tag{3}$$

The line-up allows nine players P per real-life club. (cf. SPITCH, 2021b) Furthermore, each player P_i has one of the following simplified positions: goalkeeper, defender, midfielder, attacker. As a result, for example, four players who, in reality, all play as right defenders can be lined up in SPITCH without any disadvantages. Players can not be lined up for

another position as their by SPITCH assigned simplified position. There is a selection of ten different formations that can be used to vary the number of defenders, midfielders, and attackers. However, this selection is limited to the relevant formations in reality, so there are only formations with a maximum of 5 players in one position form, except the goalkeeper position.

Each player P_i has a transfer market value V_i. As already explained in the previous chapter, in SPITCH, any player can be fielded by any manager. For this reason, the prices of the players are based on various factors, which, however, are not publicly available. These factors include how many managers draft this player, his historical performance, and his level of fame in reality. These values V exist to constrain the managers in their player choices. Since each manager only has a budget of €150m, he cannot exclusively field star players but must at the same time resort to more unknown players. This restriction turns the problem into a so-called **knapsack problem**. If the manager does not spend the budget completely, for example, by buying only inexpensive players, he will start the round with bonus points. The same applies vice versa if the budget is exceeded. This positive or negative score is called manager score S_M. The relation between the budget deviation Δ_{Budget} and manager score S_M is represented by the linear function:

$$S_{\mathrm{M}} = \frac{\Delta_{\mathrm{Budget}} \cdot 0.8}{100{,}000} = \Delta_{\mathrm{Budget}} \cdot 0.8 \cdot 10^{-5} \tag{4}$$

Thereby, the factor $0.8 \cdot 10^{-5}$ is used by SPITCH as a balancing method. (cf. SPITCH, 2021b) For example, if the budget exceeds €10m, i.e., a budget of -€10m, the manager starts with $-€10\mathrm{m} \cdot 0.8 \cdot 10^{-5} = -80$ manager score. Consequently, manager points do not exponentially increase or decrease the further one moves away from the budget threshold. For this reason, the transfer market value V of a player P can be converted and taken into account to his points p. For instance, a player P_1 with a transfer market value V_1 of €10m must therefore first score 80 points p to achieve a total positive score for the team. Since a linear relation can be established between the target value, the final line-up score S_L, and the weight of the transfer market values V, there is **no typical knapsack problem at hand**.

Since the transfer market value of a player V_i can be counted towards a player's individual score $\mathrm{S_{P_i}}$, equation (2) can be supplemented by equation (4) to create the

adjusted player score:

$$S_{\mathrm{PiM}} = -V_{\mathrm{i}} \cdot 0.8 \cdot 10^{-5} + \sum_{k=1}^{33} O_{\mathrm{k}} * p_{\mathrm{k}} \tag{5}$$

resulting in the following equation to calculate the **adjusted** final line-up score S_{LM}:

$$S_{\mathrm{LM}} = S_{\mathrm{L}} + S_{\mathrm{M}} = \sum_{i=1}^{11} \sum_{k=1}^{33} O_{\mathrm{ik}} * p_{\mathrm{ik}} + \Delta_{\mathrm{Budget}} \cdot 0.8 \cdot 10^{-5} \tag{6}$$

Each round, one player of the line-up can be appointed as captain, which results in his score getting doubled. The managers can participate for free or with a stake. The higher the stake, the higher the prize. The stake is graded according to so-called *fields*, such as the *€2 field* or the *€30 field*. Only the participants of the individual fields compete against each other. In the *free field*, the top 10 managers in the ranking, i.e., the ten managers with the highest adjusted final line-up scores $\mathrm{S_{LM}}$, win. In all other fields, the top 25% of managers, i.e., the upper quartile, win. Within these winning zones, the percentage of the prize won decreases exponentially. SPITCH does not publish the exact calculation of this decrease. The calculation of the price won will be addressed later in this work.

Chapter 3

Literature Review

This chapter presents the current state of research in two different domains. The first is about predicting sporting events using machine learning. The latter examines sports betting with a particular focus on betting odds and how these can help to predict events in the future.

The established guidelines of Brocke et al. (2015) and Webster and Watson (2002), are used to determine the current state of research and respectively document the literature search process. As stated by Webster and Watson (2002), two types of literature reviews exist. This literature review belongs to the second type, which is, according to Webster and Watson, in general, shorter and where *'authors [...] tackle an emerging issue that would benefit from exposure to potential theoretical foundations'* (Webster and Watson, 2002, p. 14). First, as recommended by Brocke et al. (2015), the literature search process is documented as accurately as possible to facilitate future research on this topic. Then, the literature found is summarised in a concept matrix according to Webster and Watson (2002) and examined according to specially selected criteria. Based on this examination, research gaps are identified, and finally, the research question for this thesis is formulated.

According to Brocke et al. (2015), in order to find relevant literature on the research areas dealt with, the topic is divided into separate concepts. These concepts help to find literature in scholarly databases using keyword search. The keywords searched for in this thesis were *'fantasy football'*, *'machine learning'*, *'prediction'* and *'betting odds'*. The keywords were entered in every existing combination to find articles that do not correspond to all keywords. Based on the research of Gusenbauer (2019), *Google Scholar* and *Microsoft*

Academic, the most extensive academic search engines, were used for the literature search. When selecting the results from this search, attention is paid to the currently awarded VHB journal rankings (see V., 2015) for the sub-field of business informatics to ensure that the literature researched is of high quality. This ranking is chosen because it is well-known and accepted in the German research area. One journal that would be less considered following this ranking, but seems extremely relevant to the research in this thesis, is the *Journal of Quantitative Analysis in Sports* (JQAS). This journal gets published by the American Statistical Association (ASA), which according to themselves, *'is the world's largest community of statisticians'* (see *About ASA* 2021). Using the papers from the JQAS and journals highly ranked by the VHB, the remaining literature is found using backward search and forward search suggested by Webster and Watson (2002).

In the process mentioned above, 22 papers were examined and compared in a concept matrix (see Figure 2 on page 12) as required by Webster and Watson (2002). The concepts used to examine the papers will be briefly discussed from left to right in this paragraph.

The year of publication, the VHB ranking, and the distinction in which form the paper was published serve to evaluate the quality of the literature. That is to ensure that primarily the most recent papers in renowned peer-reviewed journals were analyzed. The sport discipline helps to notice similar approaches in different sports. While sports differ, some are more related than others. The main idea behind this is that there may be viable approaches from a similar sport that would have been unconsidered otherwise.

During the research, to the best of my knowledge, no publication was found which deals precisely with the problem at hand. For this reason, the research had to focus on similar approaches, objectives, or tasks. The solving approaches vary from more straightforward approaches such as mixed integer programming to more complex multi-hierarchical Bayesian models. Some publications used a combination of several methodologies, which are strongly dependent on the task to be solved. A distinction was therefore made between optimization and prediction tasks. Although almost all papers unanimously had the goal of setting up a team that would score as many points as possible, they came at the solution differently. The matrix distinguishes between publications that optimized only the team performance as a whole and those that predicted the performance for each individual player and then combined the best players into a team. At the same time, it investigated which papers relied on betting odds or another form of prediction markets. Lastly, the data used in each publication was analyzed. Due to the always different data,

a generalized view was applied, which examines whether time-series data is used, whether the home advantage was taken into account and whether betting odds were used.

The articles are sorted by criteria in the following order: *'VBA Rank', 'Machine Learning Approach', 'Neural Network', 'Individual Performance', 'Betting Odds'*. This sorting ensures that the papers that are most similar to the thesis at hand and at the same time have a high VBA Rank are displayed first. For comparison purposes, the thesis at hand can be found at the bottom of the matrix. In this way, it can be quickly recognized that no publication deals precisely with the problem of the thesis. The paper that is closest to the topic is the paper by Landers and Duperrouzel (2017), even if it investigates football instead of soccer.

Table 2: Concept Matrix

	Published In					Solution Approach				Task		Solution Objective			Key Features		
Paper	**Journal**	**CP**	**Other**	**VBA-Rank**	**Sport**	**MIP**	**ML-Approach**	**NN**	**Bayesian**	**Optimisation**	**Prediction**	**Individual**	**Team**	**Betting**	**Time-Series Data**	**Home-Advantage**	**Betting Odds**
Landers and Duperrouzel (2017)	X			B	football		X			X	X	X		X	X	X	X
Lutz (2015)			X	n.R.	football		X	X			X	X			X		X
Deng and Zhong (2020)	X			n.R.	soccer		X	X		X	X		X	X	X	X	X
Wheatcroft (2020)	X			JQAS	soccer		X			X	X		X	X	X	X	X
Shah et al. (2021)		X		n.R.	soccer		X			X	X			X	X	X	X
Spann and Skiera (2009)	X			n.R.	soccer					X	X		X	X	X		X
Anik et al. (2018)		X		A	cricket		X			X	X	X			X	X	
Becker and Sun (2016)	X			JQAS	football	X	X			X	X	X			X		
Goldstein et al. (2014)		X		n.R.	soccer	X	X			X	X	X			X		
Egidi and Garby (2018)	X			JQAS	soccer				X	X	X	X			X		
Bonomo et al. (2014)	X			n.R.	soccer	X				X	X	X			X	X	
Matthews et al. (2012)		X		n.R.	soccer				X	X	X	X			X		
Skinner and Guy (2015)	X			n.R.	basketball			X			X	X			X		
Pappalardo et al. (2019)	X			B	soccer		X	X		X		X			X		
Yurko et al. (2019)	X			JQAS	football		X		X	X		X			X		
Demediuk et al. (2021)		X		n.R.	e-sports		X			X		X			X		
Edwards (2018)		X		n.R.	football	X				X		X	X		X		
Karthik et al. (2021)		X		C	cricket		X	X	X		X		X		X		
Beliën et al. (2017)	X			n.R.	cycling	X				X			X		X		
Rein and Memmert (2016)	X			n.R.	soccer		X		X				X			X	
Nevill and Holder (1999)	X			n.R.	soccer										X	X	
Thesis at Hand (2021)			**X**	**n.R.**	**soccer**		**X**	**X**		**X**	**X**	**X**		**X**	**X**	**X**	**X**

CP = Conference Proceeding, MIP = Mixed Integer Programming, ML-Approach = Machine Learning Approach, NN = Neural Network, Individual = Individual Perforamance, Team = Team Performance

The following paragraph summarises various concepts that have been frequently discussed in the presented literature. These topics are presented in alignment with the data mining process.

The first concept discussed is the *preprocessing of the data*. In order to achieve optimal results, the data mining process must adjust the data in advance without compromising its validity. Many of the authors tackle the problem that few exceptional players outperform the average players. These outliers are firstly hard to predict and secondly degrade the prediction accuracy. To solve this problem, the authors used various techniques to boost their prediction accuracy. For example, Landers and Duperrouzel (2017) developed a calculated threshold that players must reach at least to be included in the analysis. All players below this threshold are sorted out. At this point, it should be mentioned that it is crucial to choose a threshold instead of a point range, as in this case, the players with the highest points are not omitted. This approach can only be applied if data from previous games are available. Lutz (2015), Egidi and Gabry (2018), and Yurko, Ventura, and Horowitz (2019) focus in their papers on what to do if this data is not available. Yurko, Ventura, and Horowitz (2019) state that one major problem they could not solve that negatively influences the team performance is the uncertainty of players appearing in the lineup due to unpredictable events with no evidential data like injuries. Lutz (2015) investigates the case of new players who joined at the beginning of the season ('new joiners'), as these players naturally do not have previous game data. He suggests taking the mean points of all players on a similar position in this case. (cf. Lutz, 2015, p. 3) In contrast to that, Egidi and Gabry (2018) take a different approach. In their paper, they compare two different solutions to this problem. On the first try, they put the expected points to zero, and on the second try, they guessed the points in a calculated range. In both cases, the processed points from the player often were too low to be considered for their starting lineup. Nevertheless, they find out that the second approach is more precise and improves their models overall. Furthermore, Egidi and Gabry (2018) discover that simplifying the data, if more details do not add value, increase their models' accuracy as well. This is similar to the approach Deng and Zhong (2020) take. In their studies, they use the *Kaggle European Soccer Database*, a table with a total of 144 attributes, wherefrom they only carefully select 28 attributes to improve the model.

This links to the second concept, the *feature selection*. As mentioned, Deng and Zhong (2020) and Egidi and Gabry (2018) reduce the attributes fed to their model to increase accuracy. In his paper, Lutz (2015) examines precisely the question of if and how far

the number of attributes must be limited. He proceeds in three different ways. First, he does not exclude any features, figuring out that this approach is the least accurate. Secondly, he selects the features manually according to his assessment. Lastly, he chooses a more analytical path: *Recursive Feature Elimination with Cross Validation* (RFECV). This method *'recursively eliminates features and checks if the regression method's results improve by cross-validating.'* (Lutz, 2015, p. 4) This calculated approach yields the highest prediction accuracy. This method in combination with *univariate selection* was used by Anik et al. (2018) as well. One key feature that is discovered in this way is the position of the players. Lutz (2015), Demediuk et al. (2021), and Egidi and Gabry (2018) all increased their accuracy by modeling each position separately. Similar to Lutz' second manual approach, Deng and Zhong also select their features based on their perception and note that *'sufficient background knowledge of the practical application is essential.'* (Deng and Zhong, 2020, p. 4). That confirms the discovery of Rein and Memmert, who claims that at the current state of research, *'most [Machine Learning] soccer analyses are performed by computer scientist research groups with little apparent involvement by sports scientists.'* (Rein and Memmert, 2016, p. 6).

From these researches could be inferred that it is beneficial to interview sports experts on their opinion on essential features if manual feature selection is used. However, if this is not possible, feature selection algorithms should be applied. In addition, the models could be even further improved by omitting players and features that offer little added value for the predictions. Each position should thereby be modelled separately. Finally, missing data can be dealt with in three ways: setting it to zero, giving it a mean value from similar players, and estimating it accurately. The latter is promising the most success.

Once the feature selection process is complete, the next step, respectively the third concept, is to select the right *machine learning approach*. First, as in the concept matrix, a distinction must be made between optimization and prediction tasks. Only two different methodologies were chosen for the optimisation task: either brute force optimisation (Landers and Duperrouzel, 2017) or mixed-integer programming (Becker and Sun, 2016; Edwards, 2018; Beliën, Goossens, and Reeth, 2017; Bonomo, Durán, and Marenco, 2014; Matthews, Ramchurn, and Chalkiadakis, 2012). Which of these two methods is used depends primarily on how much computing power is required for the previous task.

For the prediction task, a variety of methods are used that range from simple linear regression to more complex feed-forward deep neural networks. In the following, the

focus is limited to the three most frequently used and most promising methods: *Gradient Boosted Decision Trees (GBDT)* (Landers and Duperrouzel, 2017; Deng and Zhong, 2020), *Random Forest* (Deng and Zhong, 2020; Shah, Hyman, and Samangy, 2021; Demediuk et al., 2021; Karthik et al., 2021) and *Deep Neural Networks (DNN)* (Karthik et al., 2021; Skinner and Guy, 2015; Deng and Zhong, 2020; Lutz, 2015; Landers and Duperrouzel, 2017). In the paper of Deng and Zhong (2020), all of the previously mentioned methods are used and compared. However, only a 'simple' Decision Tree model is used instead of a GBDT model. As a criterion for their Decision Tree, they use the 'information entropy', which is *'a mathematical measure of the degree of randomness in a set of data, with greater randomness implying higher entropy and greater predictability implying lower entropy.'* (Deng and Zhong, 2020, p. 4) They note that while Decision Tree models compute faster and require fewer data processing than Random Forest models, for this reason, they are more prone to over-fitting as the number of data increases, making them less accurate in general. The DNN they create consists of *'5 fully-[connected] dense layers, five activation layers, two dropout layers and one batch normalization layer.'* (Deng and Zhong, 2020, p. 4) They apply the *Softmax* function to transform the data and use *sparse categorical cross entropy* as base for the model's loss. The batch size is set to 32, and the model trains 500 epochs. According to their research regarding prediction accuracy, the DNN is the most accurate (0.99), followed by the Decision Tree model (0.91) and the Random Forest model (0.84), with the DNN probably being over-fitted with an accuracy of 0.99.

Landers and Duperrouzel (2017) use a GBDT model in their studies in which they want to predict the individual player performance for each player in the *National Football League (NFL)*. They test the team their model predicts against 300.000 randomly selected teams and achieve the highest scores in five of eleven weeks. In addition, they let their model predict the 100 best team constellations and thereby manage to get into the profit range of the 20th percentile in 68% of the cases. Unfortunately, they do not provide further information on their model but again emphasize the variety of well-thought and self-engineered features they use. Furthermore, like Deng and Zhong, they also agree to the straightforward implementation of Decision Trees, as it is not necessary to normalize or scale the features. (cf. Landers and Duperrouzel, 2017, p. 6)

In the researches from Shah, Hyman, and Samangy (2021), they attest the Random Forest model to produce the best results for their problem. They compare four different approaches to calculate the expected rate of goals. The calculations are based on: previous

goals, expected goals from prediction markets, Linear Regression and Random Forest. To compare their models, they use the *Brier Score*, *'a score function that helps determine the accuracy of any probabilistic model. '* (Shah, Hyman, and Samangy, 2021, p. 7) Another implementation of the Random Forest algorithm is used by Demediuk et al. (2021), who calculate a so-called *'Performance Index (PI)'* for each player during a e-sport game of Dota2. Here, the algorithm is used to predict the chance of winning the game based on real-time in-game data. This, later on, helps the final calculation of the PI. Depending on the length of the game, the Random Forest model predicts the correct winner with an accuracy of 0.55 to 0.8.
Of all the methods used in the literature reviewed, DNNs are the most commonly used. Similar to Deng and Zhong (2020), Karthik et al. (2021) benchmark their feed-forward DNN against Machine Learning algorithms like K-Nearest Neighbours (KNN) or Random Forest. Although their DNN, with an accuracy between 0.88 and 0.94, does not appear to be over-fitted, it also outperforms all Machine Learning models by a margin of at least 0.08. Their DNNs input layer has one neuron for each feature fed to the classifier. *'The model consisted of three hidden layers, each with 64, 32 and 16 neurons, respectively. Finally, the output layer consisted of 7 neurons [...]. A learning rate of 0.3 is used for training 500 epochs. A categorical cross-entropy loss function with sigmoid activation functions in hidden layers and a softmax activation function in the output layer is used for training the classifier. The basic hyperparameters [...] were empirically optimized using the grid search approach.'* (Karthik et al., 2021, p. 7) In contrast to this more complex DNN, Lutz (2015) uses a DNN with only one hidden layer and compares it to his results with *Support Vector Regression (SVR)*. The DNN with the best accuracy trained 50 epochs has 50 hidden units and uses the Sigmoid squashing function. This straightforward DNN already outperforms his SVR model slightly. In his conclusion, he states that DNNs with multiple hidden layers could provide increased accuracy. (cf. Lutz, 2015, p. 5)

In summary, it can be concluded from the methods analyzed that there is no algorithm in this area of research that predominantly offers the best prediction accuracies. Instead, various methods must be experimented with and adapted precisely to the problem at hand. Nevertheless, the relevant literature shows methods that promise more success than others, which should be specially addressed for this reason. These methods include Decision Trees and Deep Neuronal Networks.

The final concept discussed in this chapter is the influence of *betting odds* in current

researches in this area. The impact of values that want to predict the future can already be observed in the literature reviewed. Landers and Duperrouzel (2017) for example, want to predict the winning team against the spread, based on historical spread betting data. Shah, Hyman, and Samangy (2021) use a metric called 'expected goals', which indicates how many goals a soccer team will score according to the participating bettors. Although Deng and Zhong (2020) do not explain any further how they use or process the betting odds in their dataset, they claim to feed them to their models. In his paper, Wheatcroft (2020) investigates the overreaction of soccer betting odds in mismatch to the underlying reality. He, therefore, explains ubiquitous biases in sports betting, like the home-underdog bias and the contrary away-favourite bias. Although studies from Nevill and Holder (1999) show that the home advantage does exist, many of the papers discuss how many influences the home advantage has. (Bonomo, Durán, and Marenco, 2014; Landers and Duperrouzel, 2017; Shah, Hyman, and Samangy, 2021; Deng and Zhong, 2020) In his studies, Wheatcroft (2020) shows that there is a tendency to overestimate the influence of the home advantage. Furthermore, he defines a nominal statistic called *'combined odds distribution'* (COD) which indicates *'the performance relative to expectations of a team in its previous matches'* (Wheatcroft, 2020, p. 4). If the COD is above 0.5, the team performed better than predicted by the odds and vice versa. From this statistic, he infers that the hot-hand bias exists in soccer bettings odds, where an event seems to have a higher probability if it occurred recently in the past. In addition, he explains that even though algorithms are generally considered not to be biased prone, they still are created by biased humans.

In their studies, Spann and Skiera (2009) compare three different forecast methods, namely prediction markets, tipsters and betting odds. They discover that prediction markets and betting odds are more accurate than expert opinion. This discovery is in contrast to the results of Goldstein, McAfee, and Suri (2014), who figured out that the prediction accuracy of smaller, smarter crowds tends to be higher than the general swarm intelligence. However, to obtain the highest prediction accuracy according to Spann and Skiera (2009), all three forecasting methods should be combined.

In the end, future predicting values such as spread bets, betting odds or prediction markets have already been used in the literature but not yet as planned in this work. Furthermore, it could be shown that almost every form of using these values promises general success. In addition, it was proven that even though these values are not free of biases, they still offer added value in forecasting.

In the following concluding paragraph of this chapter, as requested by Webster and Watson (2002), the research gap found and the research question resulting from it will be addressed. Some of the authors in the presented literature state that *'investigations into these [fantasy sports] games in the academic literature are virtually nonexistent'* (Landers and Duperrouzel, 2017, p. 1) or *'the prediction of Fantasy Football has barely been studied'* (Lutz, 2015, p. 1). Rein and Memmert (2016) even claim that all main characteristics of big data implementations are highly relevant and provide specific solutions to address tactical analytics in elite soccer. In addition to that, as already described in the previous paragraph, future predicting values, especially betting odds, seem to offer a significant value in increasing prediction accuracy. Although Deng and Zhong (2020) include betting odds in their studies, it is only one of many features, and they only aim to predict team performance instead of individual player performances. In contrast, this thesis closely observes this influence, taking the results of Wheatcroft (2020) and Goldstein, McAfee, and Suri (2014) into account.

The central research question is first divided into two sub-questions. These questions are *'How accurately can individual soccer player performances be predicted using historical data?'* and *'How accurately can individual soccer player performances be predicted using betting odds?'* After these two questions are answered respectively; the central research question can be answered, which reads as follows:

'How accurately can individual soccer player performances be predicted using historical data and betting odds?'

Chapter 4

Implementation

As already mentioned in chapter 2, the task Fantasy League players face is to set up an optimal line-up. To find out this optimal set-up, indirect anticipation of the future is always useful. As can be seen from the literature review in chapter 3, many attempts have already been made to make this anticipation no longer manual and based on random factors. Quite the contrary, through the combination of big data and sport (Rein and Memmert, 2016, cf.), it is now possible to use the data collected in the past to make automated predictions for the future. A promising approach to this type of challenge is machine learning. In order to approach this machine learning problem methodically, the book *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* by Géron is used as a guideline. The implemented methodology results in a *Python Jupyter Notebook*, in which it is analysed whether and with which models this problem can be solved. In the first step, Géron advises to think fundamentally and to answer basic questions about the problem. This is done in the following section.

4.1 Business Context

Based on the rules explained in section 2.2, two main findings emerge. The first point is the main objective of the game: assemble a team of 11 players that will score as many points as possible on the upcoming match day. The second point is that the constraint placed on the players, the budget, can be converted to a player's total score. Regarding the first point, it is necessary to distinguish at what point the main goal of the game is

achieved. There are three different angles to approach this: if only the problem itself is considered, the goal would be to put together the team that scores highest. Secondly, from a game perspective, it would be enough to field the best team of all competitors, i.e., to place first in the final ranking. Lastly, a purely economic objective would be to put together a team that makes a profit by ending in the profit zone at the end of the game. Past rankings show that even the first place of a match day never achieves the whole number of points. Instead, the top places only reach 80 to 85% of the maximum possible score. Furthermore, it is important to notice that there are no well-known managers among these top places who always achieve top positions. According to these observations, it is challenging to field the best possible team, let alone to achieve first place regularly. A probable explanation for this is the strong influence of luck due to things that are not predictable, such as injuries. Therefore, it seems much more realistic to pursue the latter, purely economic goal, to end in the winning zone regularly. The previous rankings reveal that the points needed to reach the winning zone in the staked fields are lower, between 60 and 65% of the maximum possible score. For these reasons, this work aims to write a model that regularly achieves a score of 65% of the best possible score.

Regarding the second point, the best possible score is achieved by the line-up with the highest adjusted final line-up score S_{LM}. To create this line-up, the eleven players with the highest adjusted player score S_{P_iM}, which can be placed in one of the available formations, must be found. In addition, the player with the highest player score S_{P_i} must be appointed as captain. As the score of a player of the upcoming match day can be seen as a label for the prediction, this is **supervised machine learning**. Since the transfer market values, as well as all possible line-ups, are already known in advance, only the final score of each player has to be predicted. From another point of view, if the prediction of the final points of each player had an accuracy of 100%, the best line-up could be calculated automatically. Therefore, it can be concluded that the model, which is supposed to achieve a regular 65% of the best possible score, can be divided into two parts. In the first step, a machine learning model predicts the score for each player. In the second step, the best line-up is calculated based on these scores. Since no machine learning model is needed for the second step, the goal of the work can be specified further. The updated goal is to write a model that predicts upcoming individual player performances as accurately as possible to enable the system to achieve a line-up score of 65% of the best possible score.

The literature review shows that in the context of predicting individual player perfor-

mances, future predicting metrics such as betting odds have not yet been investigated. In other contexts, these kinds of metrics have already helped to improve predictions. (cf. Landers and Duperrouzel, 2017) Moreover, these values are generally considered to have a high potential in this regard. (cf. Wheatcroft, 2020; Goldstein, McAfee, and Suri, 2014) For this reason, the influence of betting odds will be examined more closely in this thesis. To investigate this influence, the machine learning models will first predict the players' final scores without the betting odds features. These models are called **baseline** models. Then, the models will again predict the final scores of the players with the betting odds features. These models are called **treatment** models. Finally, it can be examined which models provided the more accurate predictions and consequently which of the models compiled the better line-up as a result.

From these investigations now presented, a machine learning model will emerge that provides the most accurate predictions. This model is then implemented in a tool that, prior to a matchday, collects the current and required data and feeds it to the model. Based on the model's predictions, the best line-up for the upcoming match day is created and provided to the end-user. The creation of the tool is therefore divided into two phases: an online and an offline phase. First, the offline phase takes place, in which the model with the most accurate predictions is found. This requires a lot of historical data. The tool can then be used in the online phase. In this phase, the next predicted line-up can be queried. Current data is used for the prediction. In *Big Data* terms, the models in the offline phase are so-called batch processing models, while the model in the online phase is a stream processing model.

As already mentioned at the beginning of this section, SPITCH is a competition in which, on the one hand, even the best-placed players take turns and, on the other, almost never achieve the highest possible score. These two attributes are characteristics of gambling. Furthermore, football is complex and has many influencing factors that cannot all be taken into account. The betting odds try to take some of these factors into consideration, which is why they should probably achieve better results. However, it may also be that the bettors' assessment of bias is inferior. For all these reasons, it can be assumed that it will probably be difficult to achieve the set target.

4.2 Data

The cornerstone of any machine learning project is the underlying data. In the following chapter, the three essential steps in dealing with data that took place during the thesis are highlighted. First, the data was obtained from various sources. Then the data was processed so that it could be brought into relation with each other. Finally, the data was analysed to gain insight into the characteristics of the data and to draw implications for the machine learning models.

4.2.1 Procurement

As mentioned in the previous section, two types of models will be compared: baseline and treatment models. While the baseline models only use the data provided by SPITCH, the treatment models additionally use betting odds from other sources. For this reason, the data needed for this project is divided into two groups: the SPITCH data and the betting odds. Furthermore, it is stated in the previous section that the final tool is divided into two phases: a batch processing offline phase and a stream processing online phase. These classifications end in four different sources needed to implement the tool: SPITCH data and betting odds for each of the online and offline stages.

The tool's architecture is designed after the microservice principles. The microservices are described throughout the *Implementation* chapters, and the final architecture is again summarised in the last section of this chapter. The tool and with it each associated microservice is launched on a virtual private server (VPS) hosted on *Hetzner*. (see Hetzner, 2021) The core of the architecture is the PostgreSQL *Database* microservice, which stores all of the data mentioned in the following paragraphs.

SPITCH Data

The SPITCH data derives from the same source regardless of the ongoing tool's phase. The data is crawled directly from their API-endpoint https://api.spitch.live from the *Crawler* microservice. The *Crawler* therefore executes a script in a manually definable schedule. Additionally, the *Proxy* microservice is running. The requests sent from the *Crawler* are

piped over the *Proxy*. The *Proxy* then sends the request via a scaling number of different IP addresses (nodes) within the *Tor* network. This procedure bypasses the rate limits of the recipient's firewall, as the requests can no longer be clearly assigned to a single IP address, allowing the *Crawler* to send multiple requests per second. However, with programming ethics in mind, the requests per second still were set to a relatively small number to avoid over floating the endpoint.

SPITCH data refers to player-specific and event-based data. Player-specific data includes tables such as the *Player* table with information about their position, transfer market value, names, and team membership. The player data was crawled from the endpoint https://api.spitch.live/**contestants**. This endpoint sends all players and teams from the SPITCH database in the *JavaScript Object Notation* (JSON) format. The JSON is received, converted, and stored in two separate tables *Player* and *Team* in the own *Database*. This procedure took place once in the offline phase and takes place before each matchday in the online phase. The latter ensures that the player and team data is up to date for the model's predictions. During the conversion of the request, the transfer market value of each player is stored in a different table *Market Value* with an additional period of validity. In data management or warehousing terms, the table *Market Value* is implemented as *Slowly Changing Dimension Type 2*. This modification was made because the transfer market value is a feature of the players that changes frequently but whose history is helpful for later investigations. For this reason, information would be lost if the transfer market value were merely overwritten on update.

However, the core data is event-specific, stored in the *Event* table. Each event on the pitch during a match is stored in this table, represented by one row. Therefore, one row in the table interprets like: '*Player A has performed event B in minute C on matchday D.*,' i.e. '*Manuel Neuer played a pass in the 35th minute of the 6th matchday.*' This data can only be gathered matchday- and player-specific, using the corresponding endpoints following the structure:

https://api.spitch.live/matchdays/**matchday_id**/players/**player_id**/events

Consequently, the *Crawler* first has to get all the player and matchday identifiers (id) to gather all the data provided by these endpoints. The tool is able to get all the player identifiers using the method described above in the player paragraph. Simultaneously, the *Crawler* has to get all the matchday identifiers by requesting the endpoint

https://api.spitch.live/matchdays and storing the received information in the table *Matchday*. If player and matchday identifiers are provided, the event data can be queried. The service then takes two extra checks before starting the crawling. First, it checks for the latest matchday in the *Event* table to prevent duplicate queries. This step is particularly essential in the online phase, as the tool is executed regularly on different occasions. Second, it proves which of the matchdays in the table *Matchday* have already occurred by taking the current timestamp and comparing it with the timestamps of the matchdays. This checkup ensures that no endpoints from matchdays that lay in the future are getting requested, leading to empty responses. After these two checks are made, the *Crawler* first gets and then stores all the available event data in the *Database*.

It is important to notice that SPITCH most likely uses an official API as well to gather this information, like https://www.api-football.com/ for instance. For further studies, it would be beneficial to investigate to what extend the data from SPITCH matches with the data from such APIs since they are easier to request and provide much more data, which reaches far into the past. At the same time, however, it must be noted that more data could also lead to less accurate results, as the development of individual players over the years represents an additional and very complex influence. Only the connection between player data and betting odds could be examined more closely. In both cases, the amount of data now available is likely to be sufficient.

The following table serves as an overview of the various SPITCH-based tables in the *Database*, explaining how many records are included and how much memory they occupy.

Betting Odds

The betting odds data in this thesis are obtained in two different ways. Here, a distinction is made between the online and offline phases. In the offline phase, a comma-separated values (CSV) file is taken from the following website:
https://www.football-data.co.uk/germanym.php. In this CSV file are all betting odds for each game of a season for different betting odds providers. Furthermore, columns like the average betting odds from all providers are added. In the online phase, the data is taken from this API: https://the-odds-api.com/. In this phase, care is taken to request the odds as late as possible, as they often change before the start of the matchday given

to the most recent information. Here, no average betting odds column is provided and therefore needs to be calculated itself by the *Crawler* microservice. This data gets stored in the *Odds* Table of the *Database*. Each row thereby represents the winning, draw, and losing odds for a team for a matchday.

4.2.2 Processment

After the data has been obtained in the previous section, it is processed and merged in this section. In the offline phase, these are the first steps in the Jupyter Notebook, while in the online phase, these steps take place in the *Prediction* microservice.

In the database query, the player-specific tables are joined with the *Event* table. This join converts the identifier columns into human-readable and understandable columns. For example, the name, club, and position are obtained from the player identifier. After all the event data in the database has been fetched and made comprehensible, this data must first be supplemented because the API only returns events that have indeed happened on the pitch. However, it could also be interesting to know that a player **did not** play, for example, a single pass on a matchday. This information is currently not yet available in the data set and will be added in the following step. Therefore, the points catalog from SPITCH (see SPITCH, 2021a) has to be loaded, which contains all events that can happen in SPITCH. After that, the events can be grouped by player and matchday. For each group, the points catalog gets iterated over. If there exists no corresponding entry in the group for an event type, an entry is added with the occurrence of zero. After this step, the event data is ready to use.

In the Jupyter Notebook, the betting odds data gets read from the CSV file mentioned in the previous section. From the data obtained, only the relevant columns get extracted, which include the date, name of the home and away team, and the average of the betting odds. Each row in the dataset is equal to a match between two teams. In order to assign the correct match day to these matches, the date column must first be converted into a *datetime* object. Now the *Matchday* table in the *Database* can be used to check to which matchday the date belongs belongs. After this step, the rows can be split, resulting in two rows, with each row being equal to betting odds for one team for one matchday. Therefore, the odds for the home team are renamed in odds to win for the home team and vice versa

for the away team. Additionally, the column *is_home* is added to investigate the **home advantage** mentioned often in the literature review.
In the online phase, the data was already written to the database in this format by the *Crawler*. This is why the data only needs to be read from the database by giving the upcoming matchday.

After the event data and the betting odds have been converted into the desired form, they can be merged. The key with which the two data sets must be linked contains the matchday and the team. Thereby the following problem occurs: while the team data obtained by SPITCH is in German format, the betting odds data is in International format. Since the names of the teams in the respective data sets do not match, each International formatted name must first be assigned its German equivalent. Furthermore, as it is best practice to always join via identifiers, the German identifier from the database is assigned to each international name. The Python library *smart-match* is used to achieve this. Smart-match calculates the similarity between two entered strings using various methods such as the *Levenshtein distance* or the *Smith-Waterman* algorithm. All methods were evaluated and compared with each other, and finally, the *Smith-Waterman* algorithm was chosen, as it produced by far the highest similarity values. The algorithm iterates over each international name in the betting odds dataset, checking the similarity with each German name using the *Smith-Waterman* formula and permanently storing the name and the identifier for the highest similarity. In this way, each row in the betting odds dataset has a team identifier assigned, which makes it possible to merge the two datasets.

With the help of the SPITCH points catalog and the individual events for each player, the player score can now be calculated for each match day. It is important to mention that the transfer market value of a player is not deducted from the score, as the models should only concentrate on the points actually achieved. In this case, the transfer market values would only blur the scores. To calculate each player's individual score, according to equation (2) on page 6, each number of appearances is multiplied by the corresponding point value per event, and these products are then summed. In the resulting dataset, each row represents the player's score for a matchday together with his position, team, and betting odds.

Through this, an additional feature can be implemented utilizing the individual player score: the player *performance trend*. This feature takes the recent scores of a player and

calculates a trend score based on them. How differently the historical values had to be weighted was investigated in the course of the thesis. Several methods were compared: the Simple Moving Average (SMA), the Cumulative Moving Average (CMA), and the Exponential Moving Average (EMA). In addition, different parameters were used for the SMA and EMA: the SMA was calculated with three, five, and ten historical values and the EMA with an α of 0.1, 0.3, and 0.5. The SMA sums up the n past values and divides the total by n, resulting in each past value having the same weight. The EMA, however, weights more recent values by the exponential factor α. From these investigations, the EMA with an α of 0.5, also known as the half-life, performed best in predicting future performances merely based on recent values. Therefore, the player performance trend column is created by calculating the EMA with an α of 0.5 based on the individual player score column.

According to Géron's methodology and the intention of this work, several machine learning models are compared with each other. For this reason, the data now available must be made equally feasible for each model. Because of this, all numerical features must be normalized and all categorical features encoded. The features *player performance trend* and all betting odds-specific columns are normalized using a technique called *Min-Max-Scaling*. With this technique, each feature X is scaled in the range between zero and one using the following formula:

$$X_{\mathrm{SC}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{7}$$

The categorical features *team* and *position* are encoded using the technique *One-Hot-Encoding*. This technique creates a boolean column for each instance of a categorical feature. Thus, for example, the position column becomes four columns according to the pattern *is_goalkeeper, is_defender*, and so on. By converting the data in this way, the final data set put into the models has 31 columns.

After the data has been merged and made human-readable in the previous paragraphs, a distinction must be made between two data sets. The first data set contains all event types with the number of occurrences for each player and each matchday. The second data set, in contrast, only includes the calculated final score for each player for each matchday. Thus, there are two ways in which machine learning models could be used

to predict player performance. The first option would be to predict for each player the number of occurrences for each event type. Then the final score can be calculated based on the predicted values. The second option is to predict the final score directly. Since a proportional relationship between the events and the final score can be established through the points catalog, only the influence of the features is decisive for which variant is chosen.

4.2.3 Exploration

As already mentioned in the previous section, two data sets exist which can be interpreted. First, this section examines all features that are present in both datasets. Finally, the different features are examined to finally decide on a dataset to be used for the models.

The following table describes all the features that appear throughout the datasets, their description, and datatype. In *Python*, *Strings* are stored as objects.

Table 3: Overview of Features and their Dataype

Feature	**Datatype**	**Description**
name	object	name of a player
position	object	position of a player, i.e. goalkepper, attacker
matchday	int64	matchday where the results happened, i.e. 1, 2, 34
team_name	object	name of the team in which the player competes
is_home	bool	boolean indicating whether the game was played at home or not
odds_win	float64	odds for the team to win
odds_draw	float64	odds for the team to draw
odds_lose	float64	odds for the team to lose
event_type	object	event type, i.e. goal, pass, unsuccessfullTackle
occurrences	int64	number of occurences of event_type for this matchday
performance_trend	int64	recent performance trend for a player
score	int64	individual player score S_{P_i}

Each feature is first analyzed individually and then in combination with each other. This analysis is mainly done in the order of the table, starting with the dataset containing event types and their occurrences.

Name

The name of a player is synonymous with the player itself in this context. As can be seen from the following figure, there aren't identical numbers of entries for each player, which should not be the case as in the previous section an entry was assigned to each player for each matchday and event type. The reason for this difference is that there are entries for the event types **player_on** and **player_off** in the data set, which are not found in the points catalog from SPITCH and have no influence on the score. For this reason, these events can be ignored.

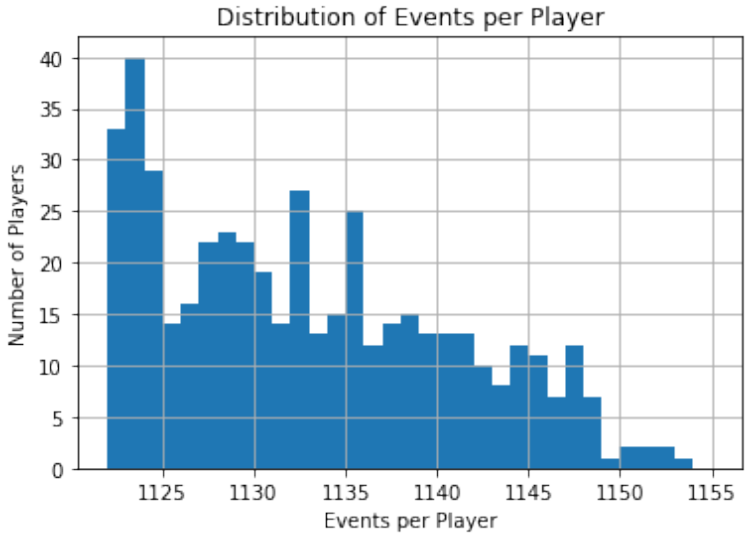

Figure 4.1: Distribution of Events per Player

After deleting the two event types mentioned above from the data set, there are 1,122 rows for each of the 467 players. This is equivalent to 33 different event types on 34 matchdays. Therefore, the data set has a total of 1,122 rows times 467 players, i.e. 523,974 entries.

Matchday, Event Type, and Home Advantage

Accordingly to the calculation from the previous section, for each matchday, there are 33 event types times 467 players, i.e., 15,411 rows, and for each event type, there are 34 matchdays times 467 players, 15,878 entries. Since in the German *Bundesliga* every team plays every team twice, once at home and once away, there are precisely the same number of rows for matches played at home and matches played away.

Position

From the figure 4.2 below, it can be seen that there are four different positions in the dataset, with the midfielder being the most common position with 193 entries out of 467 (41.3%) and goalkeeper being the rarest position with 35 entries (7.5%). This distribution is reasonable given that most formations have more midfielders than any other position, and there is always only one goalkeeper.

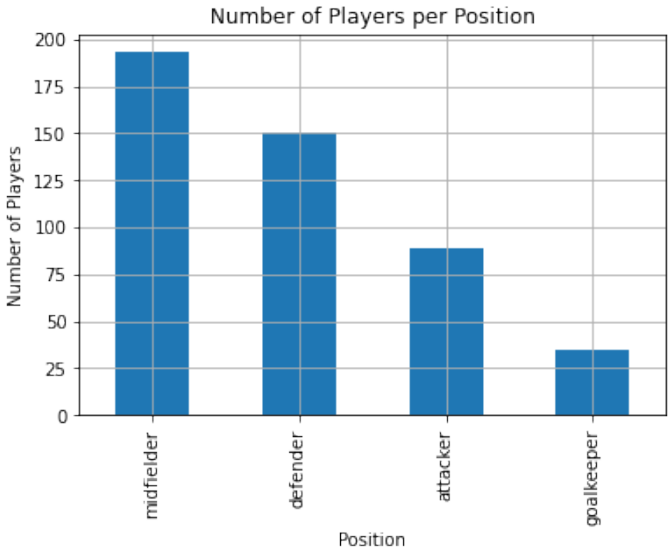

Figure 4.2: Number of Players per Position

Team Name

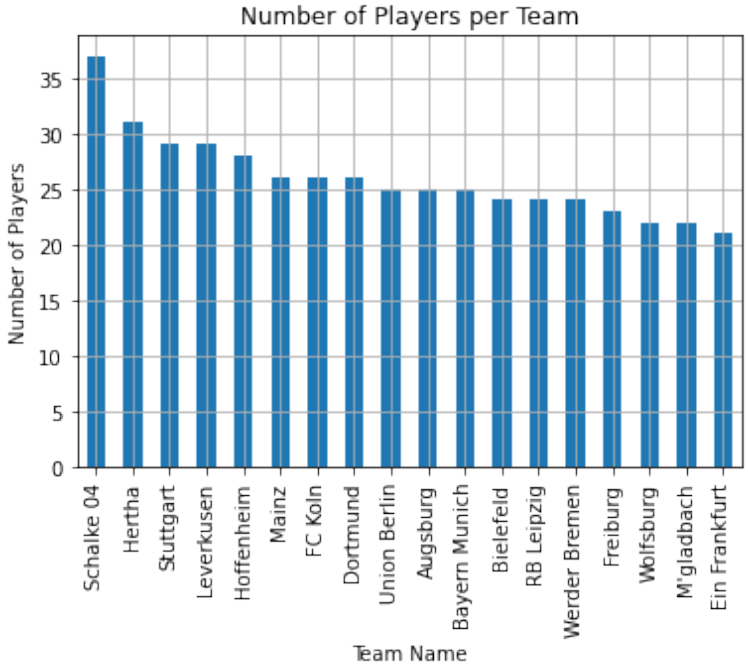

Figure 4.3: Number of Players per Team

The figure shows that Eintracht Frankfurt has the fewest players with 21. Schalke 04 has the most players with 37. The average number of players per team is 25.

Betting Odds

Many different betting odds systems exist. The betting odds used in this thesis follow the most popular model. In this model, the betting odds represent the factor by which the stake is multiplied if the event on which the bet was placed occurs. Through this system, bets can be placed on any game outcome with a chance of winning. The more probable the outcome, the lower the factor. This factor is influenced by various aspects such as the assumed playing strength of the teams, but also how many bettors have already bet on this outcome. Since every bookmaker has its own methods for this calculation, an average of 13 different platforms is chosen. This betting odds model has the advantage that it is lucrative for the bookmaker in every case, as he chooses the factors in such a

way that no matter what the outcome of the match, the stakes paid in exceed the profit to be cashed out. For this reason, these betting odds should be assessed with caution. However, they represent a relatively general and, above all, up-to-date picture of the teams' probabilities of winning. The figure below shows that the betting odds are mostly in the range between two and four for winning or losing. Anything above or below that range indicates a one-sided game. Here, 24.69 has been the maximum value, and the corresponding counterpart 1.09 was the minimum value. While the range for a draw is smaller, the factor itself is higher in general.

Figure 4.4: Distribution of Betting Odds

Occurences

The bar chart below shows that most of the rows have a value of 0 (**87.3%**). The frequency monotonically decreases as one moves further away from zero. Therefore, the main task of machine learning models would be to predict situations where the result are not zero, which leads to a classification problem instead of the initial regression problem. As announced at the beginning of the section, a final decision has to be made in favor of one data set. In addition to the below figure, further investigations with this dataset, which cannot be presented in this thesis due to the large volume, showed that its use is unsuitable. For this reason, it was decided to use the data set that contains the calculated scores.

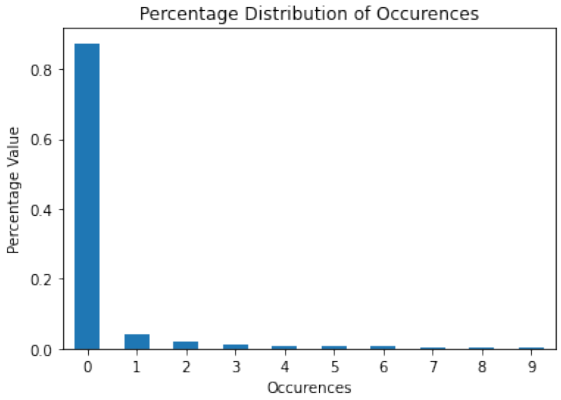

Figure 4.5: Percentage Distribution of Occurences

Score

An interesting statistic in relation to the score, especially about the significance of the performance trend, is the autocorrelation. The autocorrelation indicates the extent to which previous values are predictive of future values of the same feature. Figure 4.6 shows an autocorrelation between 0.65 and 0.7 for the lags between -10 and 10. Additionally, the correlation slowly decreases as one moves further away from the actual value. This small decrease indicates that more current values give better predictions than past values. This conclusion confirms the choice of exponential moving averages for the calculation of the performance trend. Overall, an autocorrelation of 0.7 indicates a medium to strong relationship between past and current values.

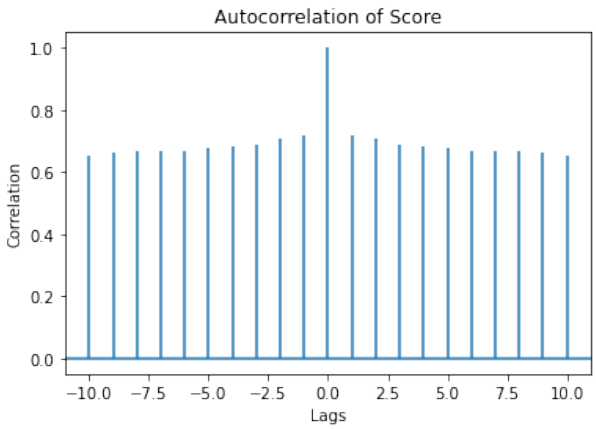

Figure 4.6: Autocorrelation of Score

Looking at the score, two groups of players are particularly interesting. The first group consists of players who rarely score many points and are consequently not drafted by many managers. The second group consists of players who score relatively confidently. The players in this group are, therefore, probably the most expensive because many managers draft them. Two different approaches are taken in the analysis of the data to separate the players in these two groups. To begin with, the players who belong to the first group will be examined by looking for the players with the highest average of points per game. Secondly, it is examined which players have scored the most points overall over the entire season. These are the players in the second group.

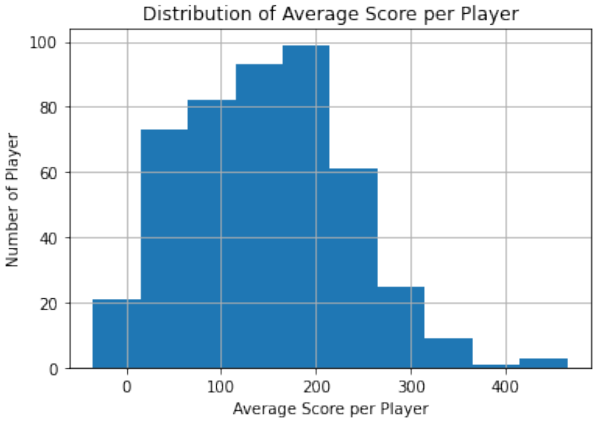

Figure 4.7: Distribution of Average Score per Player

Figure 4.7 shows the distribution of the average scores of all players. This figure indicates that most players receive an average score of around 200. There exists a small group of players who achieve an average score of over 300. Interestingly, more players achieve more than 400 points than players who achieve between 350 and 400 points.

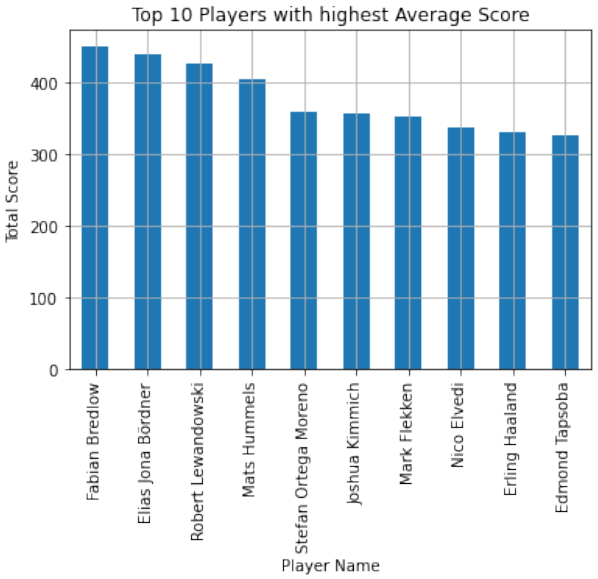

Figure 4.8: Top 10 Players with highest Average Score

The above figure displays the ten players with the best average score. Surprisingly, many players in this group are not well-known and therefore inexpensive. These are the players who need to be ideally lined up by the model because, on the one hand, they score a lot of points and, on the other, their low transfer market value means they don't have to earn a lot of points to be valuable. Nevertheless, there are also very well-known players, such as *Mats Hummels* or *Robert Lewandowski*. With these players, the models have to weigh up whether to put them in the line-up or not, as their high transfer market value means that they have to earn a lot of points first in order to recoup the manager points they have cost. These players are especially worth choosing as captain because the points scored are doubled, but their transfer market value deduction is not.

Regarding the second group of players, figure 4.9 shows that there are many players who have scored between zero and 1,000 points throughout the season. Given this and the prior observations, it can be concluded that many players in the data set are rarely fielded, but when they are fielded, they still achieve an average score.

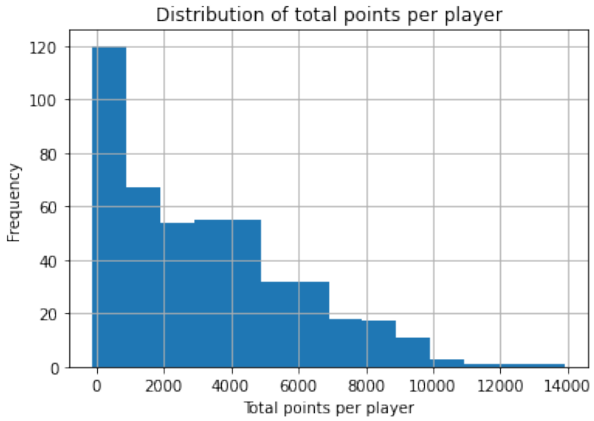

Figure 4.9: Distribution of Total Score per Player

On the other end of the axis, there is only a small number of extraordinary players who scored over 10,000 points throughout the season. These players have two characteristics: they are fielded often and score confidently. The following figure shows the Top 10 players regarding the total score over the entire season.

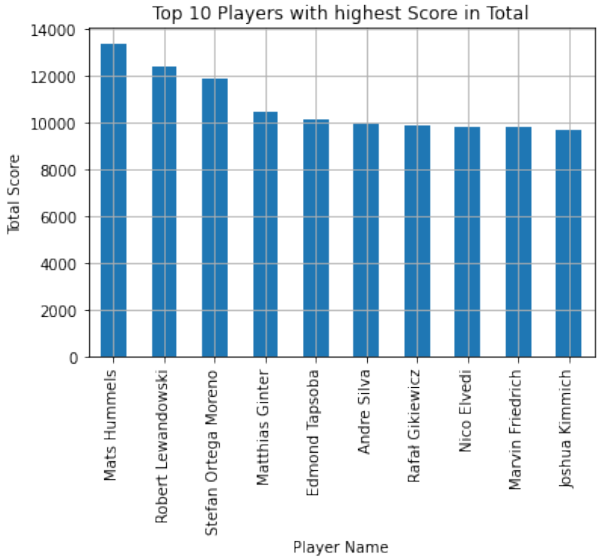

Figure 4.10: Top 10 Players with highest Total Score

In contrast to the previous group, this top 10 contains many well-known players, once again including *Mats Hummels* and *Robert Lewandowski* at the front positions.

Correlation between position and score

Interestingly, all positions are represented in the previous figure 4.10. For this reason, further insights into the correlation between the position and the scores are provided in this paragraph. The figure 4.11 shows that goalkeepers score the most points on average, followed by defenders and midfielders. Attackers score the fewest points. This finding is counterintuitive as goals are scored highest in SPITCH and most goals are scored by attackers. It can be deduced from this finding that it often depends on many small events, such as passing or taking the ball away, to achieve a high score.

Furthermore, conclusions for the formation can be drawn from this diagram. For example, if goalkeepers score the most points on average, it might be a good strategy to choose the goalkeeper as captain. In addition, a defensive formation should be chosen, as defenders and midfielders score more points on average than attackers.

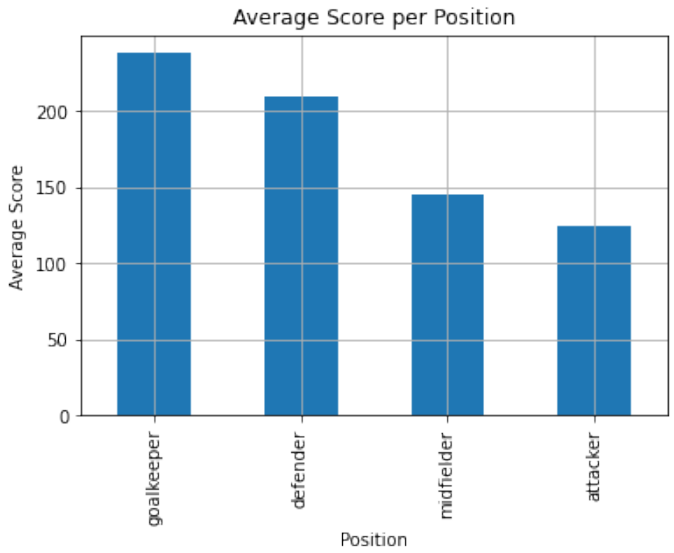

Figure 4.11: Average Score per Position

The following two data explorations are carried out exclusively with the Betting Odds CSV data set (Football-Data, 2021) and should therefore be considered separately from the previous data set.

Correlation between Betting Odds and Match Result

This section aims to examine the influence betting odds have in the prediction of future player performance. For this reason, the match result is taken from the CSV in addition to the betting odds. First, the percentage of cases in which betting odds predicted the correct final result is examined. Therefore, the delta between the odds for the home and away team is calculated. From the previous investigations of the betting odds, it is known that when the odds for a draw are highest, the odds for the home and away team usually differ by a maximum of 0.5. For this reason, a draw is expected for a delta of less than 0.5. Otherwise, a win is expected for the team with lower odds. When this procedure is applied to the entire data set, the betting odds predict the correct outcome **53.6%** of the time. This result is a remarkable 21% difference to the probability of one-third if one had to guess the result. Thus, this study concludes that betting odds are indicative of which team is likely to win.

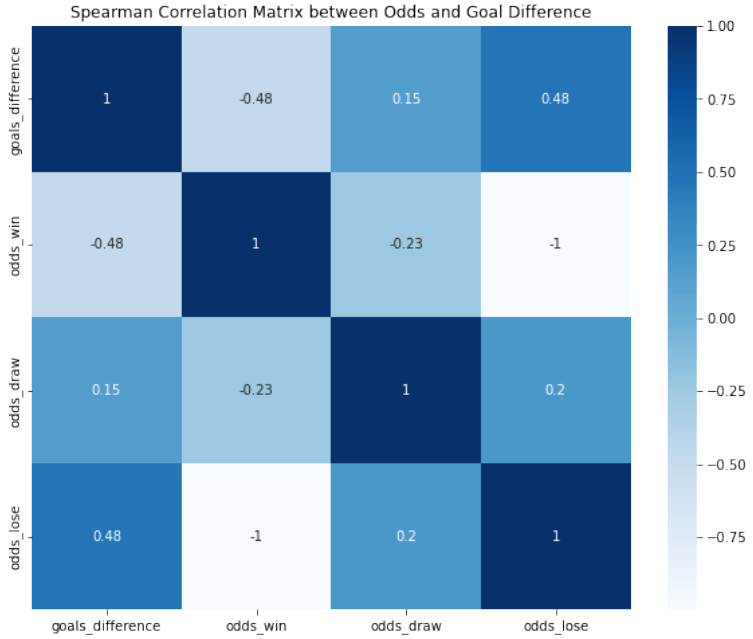

Figure 4.12: Spearman Correlation Matrix between Odds and Goal Difference

However, the game of SPITCH is more complex than just predicting the winning team. It would also be interesting to know whether betting odds can also predict the level of victory or defeat. Because teams that are expected to win higher than others consequently

score more goals, which leads to higher scores. In order to prove this influence, the final goal difference is computed. Afterward, the correlation between this variable and the betting odds is calculated using *Spearman's rank correlation coefficient*. The correlation matrix in figure 4.12 indicates a medium correlation of 0.48 or -0.48 respectively between the goal difference and the odds for winning or losing. Accordingly, the betting odds not only allow to predict which team is likely to win but also to estimate with a certain degree of accuracy how clear the result will be.

Home advantage

The literature review showed that home advantage had been used in many models written to predict sports outcomes. In the data set used for the models in this thesis, a column regarding the home advantage was also added. The following figure shows that home advantage also played a role in the 2020/2021 season. From the perspective of the home team, 42% of the matches were won and 31% lost.

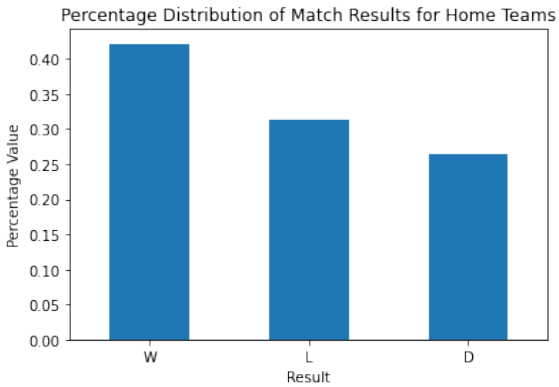

Figure 4.13: Percentage Distribution of Match Results for Home Teams

4.3 Models

4.4 Application

Chapter 5

Conclusion

List of Figures

4.1 Distribution of Events per Player 30
4.2 Number of Players per Position 31
4.3 Number of Players per Team 32
4.4 Distribution of Betting Odds 33
4.5 Percentage Distribution of Occurences 34
4.6 Autocorrelation of Score 34
4.7 Distribution of Average Score per Player 35
4.8 Top 10 Players with highest Average Score 36
4.9 Distribution of Total Score per Player 37
4.10 Top 10 Players with highest Total Score 37
4.11 Average Score per Position 38
4.12 Spearman Correlation Matrix between Odds and Goal Difference 39
4.13 Percentage Distribution of Match Results for Home Teams 40

List of Tables

1 SPITCH Glossary 5

2 Concept Matrix 12

3 Overview of Features and their Dataype 29

Bibliography

[1] *About ASA*. URL: https://www.amstat.org/ASA/about/home.aspx?hkey=6a706b5c-e60b-496b-b0c6-195c953ffdbc (visited on 07/20/2021).

[2] Aminul Islam Anik et al. "Player's Performance Prediction in ODI Cricket Using Machine Learning Algorithms". en. In: *2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT)*. Dhaka, Bangladesh: IEEE, Sept. 2018, pp. 500–505. ISBN: 978-1-5386-8279-1. DOI: 10.1109/CEEICT.2018.8628118. URL: https://ieeexplore.ieee.org/document/8628118/ (visited on 07/01/2021).

[3] Adrian Becker and Xu Andy Sun. "An analytical approach for fantasy football draft and lineup management". en. In: *Journal of Quantitative Analysis in Sports* 12.1 (Mar. 2016). Publisher: De Gruyter Section: Journal of Quantitative Analysis in Sports, pp. 17–30. ISSN: 1559-0410. DOI: 10.1515/jqas-2013-0009. URL: https://www.degruyter.com/document/doi/10.1515/jqas-2013-0009/html (visited on 06/02/2021).

[4] J Beliën, D Goossens, and D Van Reeth. "Optimization modeling for analyzing fantasy sport games". en. In: (2017), p. 16.

[5] F. Bonomo, G. Durán, and J. Marenco. "Mathematical programming as a tool for virtual soccer coaches: a case study of a fantasy sport game". en. In: *International Transactions in Operational Research* 21.3 (May 2014), pp. 399–414. ISSN: 09696016. DOI: 10.1111/itor.12068. URL: http://doi.wiley.com/10.1111/itor.12068 (visited on 07/01/2021).

[6] Jan vom Brocke et al. "Standing on the Shoulders of Giants: Challenges and Recommendations of Literature Search in Information Systems Research". en. In: *Communications of the Association for Information Systems* 37 (2015). ISSN: 15293181. DOI:

10.17705/1CAIS.03709. URL: https://aisel.aisnet.org/cais/vol37/iss1/9/ (visited on 07/16/2021).

[7] Simon Demediuk et al. "Performance Index: A New Way To Compare Players". en. In: (2021), p. 16.

[8] Wuhuan Deng and Eric Zhong. "Analysis and Prediction of Soccer Games: An Application to the Kaggle European Soccer Database". en. In: *Insight - Statistics* 3.1 (Nov. 2020), p. 1. ISSN: 2661-3115. DOI: 10.18282/i-s.v3i1.332. URL: http://insight.piscomed.com/index.php/I-S/article/view/332 (visited on 07/01/2021).

[9] Steven J. Edwards. "Analyzing Fantasy Sport Competitions with Mixed Integer Programming". en. In: *Data and Decision Sciences in Action 2*. Ed. by Andreas T. Ernst et al. Lecture Notes in Management and Industrial Engineering. Cham: Springer International Publishing, 2018, pp. 167–182. ISBN: 978-3-030-60135-5. DOI: 10.1007/978-3-030-60135-5_12.

[10] Leonardo Egidi and Jonah Gabry. "Bayesian hierarchical models for predicting individual performance in soccer". en. In: *Journal of Quantitative Analysis in Sports* 14.3 (Sept. 2018). Publisher: De Gruyter Section: Journal of Quantitative Analysis in Sports, pp. 143–157. ISSN: 1559-0410. DOI: 10.1515/jqas-2017-0066. URL: https://www.degruyter.com/document/doi/10.1515/jqas-2017-0066/html (visited on 06/02/2021).

[11] Fabiano. *Fantasy football 101*. en-US. 2007. URL: https://www.nfl.com/news/fantasy-football-101-09000d5d80021ece (visited on 10/19/2021).

[12] Football-Data. *Germany Football Results and Betting Odds*. URL: https://www.football-data.co.uk/germanym.php (visited on 11/02/2021).

[13] Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. Sebastopol, UNITED STATES: O'Reilly Media, Incorporated, 2019. ISBN: 978-1-4920-3261-8. URL: http://ebookcentral.proquest.com/lib/htw-berlin/detail.action?docID=5892320 (visited on 10/25/2021).

[14] Daniel G. Goldstein, Randolph Preston McAfee, and Siddharth Suri. "The wisdom of smaller, smarter crowds". en. In: *Proceedings of the fifteenth ACM conference on Economics and computation*. Palo Alto California USA: ACM, June 2014, pp. 471–488. ISBN: 978-1-4503-2565-3. DOI: 10.1145/2600057.2602886. URL: https://dl.acm.org/doi/10.1145/2600057.2602886 (visited on 07/01/2021).

[15] Green. *'Wink': Wilfred 'Bill' Winkenbach invented Fantasy Football way back in 1962 with GOPPPL in Oakland - newsnet5.com Cleveland*. Sept. 2014. URL: https://web.archive.org/web/20150929163914/http://www.newsnet5.com/sports/wink-wilfred-bill-winkenbach-invented-fantasy-football-way-back-in-1962-with-gopppl-in-oakland (visited on 10/19/2021).

[16] Michael Gusenbauer. "Google Scholar to overshadow them all? Comparing the sizes of 12 academic search engines and bibliographic databases". en. In: *Scientometrics* 118.1 (Jan. 2019), pp. 177–214. ISSN: 0138-9130, 1588-2861. DOI: 10.1007/s11192-018-2958-5. URL: http://link.springer.com/10.1007/s11192-018-2958-5 (visited on 07/16/2021).

[17] Hetzner. *About Us - Hetzner Online GmbH*. 2021. URL: https://www.hetzner.com/unternehmen/ueber-uns (visited on 10/28/2021).

[18] Hoffmann. *Millionen Daten pro Spiel: So werden Fußball-Statistiken erfasst*. de. Section: Sport. Nov. 2014. URL: https://www.hna.de/sport/fussball/millionen-daten-spiel-werden-fussball-statistiken-erfasst-4483893.html (visited on 10/19/2021).

[19] K. Karthik et al. "Analysis and Prediction of Fantasy Cricket Contest Winners Using Machine Learning Techniques". en. In: *Evolution in Computational Intelligence*. Ed. by Vikrant Bhateja et al. Vol. 1176. Series Title: Advances in Intelligent Systems and Computing. Singapore: Springer Singapore, 2021, pp. 443–453. ISBN: 9789811557873 9789811557880. DOI: 10.1007/978-981-15-5788-0_43. URL: http://link.springer.com/10.1007/978-981-15-5788-0_43 (visited on 07/28/2021).

[20] Jonathan Robert Landers and Brian Duperrouzel. "Machine Learning Approaches to Competing in Fantasy Leagues for the NFL". en. In: *IEEE Transactions on Games* 11.2 (2017), pp. 159–172. ISSN: 2475-1502, 2475-1510. DOI: 10.1109/TG.2018.2841057. URL: https://ieeexplore.ieee.org/document/8367900/ (visited on 07/01/2021).

[21] Roman Lutz. "Fantasy Football Prediction". en. In: *arXiv:1505.06918 [cs]* (May 2015). arXiv: 1505.06918. URL: http://arxiv.org/abs/1505.06918 (visited on 06/02/2021).

[22] Tim Matthews, Sarvapali D Ramchurn, and Georgios Chalkiadakis. "Competing with Humans at Fantasy Football: Team Formation in Large Partially-Observable Domains". en. In: (2012), p. 7.

[23] Alan M. Nevill and Roger L. Holder. "Home Advantage in Sport: An Overview of Studies on the Advantage of Playing at Home". en. In: *Sports Medicine* 28.4 (1999), pp. 221–236. ISSN: 0112-1642. DOI: 10.2165/00007256-199928040-00001. URL: http://link.springer.com/10.2165/00007256-199928040-00001 (visited on 07/09/2021).

[24] Robert Rein and Daniel Memmert. "Big data and tactical analysis in elite soccer: future challenges and opportunities for sports science". en. In: *SpringerPlus* 5.1 (Dec. 2016), p. 1410. ISSN: 2193-1801. DOI: 10.1186/s40064-016-3108-2. URL: http://springerplus.springeropen.com/articles/10.1186/s40064-016-3108-2 (visited on 07/01/2021).

[25] Kushal Shah, James Hyman, and Dominic Samangy. "A Poisson Betting Model with a Kelly Criterion Element for European Soccer". en. In: (2021), p. 16.

[26] Brian Skinner and Stephen J. Guy. "A Method for Using Player Tracking Data in Basketball to Learn Player Skills and Predict Team Performance". en. In: *PLOS ONE* 10.9 (Sept. 2015). Ed. by Frank Emmert-Streib, e0136393. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0136393. URL: https://dx.plos.org/10.1371/journal.pone.0136393 (visited on 07/01/2021).

[27] Martin Spann and Bernd Skiera. "Sports forecasting: a comparison of the forecast accuracy of prediction markets, betting odds and tipsters". en. In: *Journal of Forecasting* 28.1 (Jan. 2009), pp. 55–72. ISSN: 02776693, 1099131X. DOI: 10.1002/for.1091. URL: http://doi.wiley.com/10.1002/for.1091 (visited on 07/01/2021).

[28] SPITCH. *Points Catalogue.* URL: https://www.spitch.live/en/points-catalogue/ (visited on 10/21/2021).

[29] SPITCH. *Rules of Play and Participation:* URL: https://www.spitch.live/en/rules-of-play-and-participation/ (visited on 10/21/2021).

[30] SPITCH. *SPITCH | The Live Football Manager*. URL: https://www.spitch.live/en/ (visited on 10/21/2021).

[31] VHB e. V. *VHB-JOURQUAL3: Wirtschaftsinformatik*. 2015. URL: https://vhbonline.org/fileadmin/user_upload/JQ3_WI.pdf.

[32] Jane Webster and Richard T Watson. "Guest Editorial: Analyzing the Past to Prepare for the Future: Writing a literature Review". en. In: (2002), p. 11.

[33] Edward Wheatcroft. "Profiting from overreaction in soccer betting odds". en. In: *Journal of Quantitative Analysis in Sports* 16.3 (Sept. 2020). Publisher: De Gruyter Section: Journal of Quantitative Analysis in Sports, pp. 193–209. ISSN: 1559-0410. DOI: 10.1515/jqas-2019-0009. URL: https://www.degruyter.com/document/doi/10.1515/jqas-2019-0009/html (visited on 06/03/2021).

[34] Ronald Yurko, Samuel Ventura, and Maksim Horowitz. "nflWAR: a reproducible method for offensive player evaluation in football". en. In: *Journal of Quantitative Analysis in Sports* 15.3 (Sept. 2019). Publisher: De Gruyter Section: Journal of Quantitative Analysis in Sports, pp. 163–183. ISSN: 1559-0410. DOI: 10.1515/jqas-2018-0010. URL: https://www.degruyter.com/document/doi/10.1515/jqas-2018-0010/html (visited on 06/02/2021).

Appendix A

A.1 Diagrams

A.2 Tables

A.3 Screenshots

A.4 Graphs

Decleration of Authenticity

I declare that I wrote this thesis on my own and did not use any unnamed sources or aid. Thus, to the best of my knowledge and belief, this thesis contains no material previously published or written by another person except where due reference is made by correct citation. This includes any thoughts taken over directly or indirectly from printed books and articles as well as all kinds of online material. It also includes my own translations from sources in a different language. The work contained in this thesis has not been previously submitted for examination. I also agree that the thesis may be tested for plagiarized content with the help of plagiarism software. I am aware that failure to comply with the rules of good scientific practice has grave consequences and may result in expulsion from the program.

Leipzig, 08.11.2021

Jakob Heine