Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Economics and Management
Department of Industrial Economics and Technology Management

Robin Andersen
Vegard Hassel

In-game Betting and the FA English Premier League: The Contribution of Prediction Models

Master's thesis in Industrial Economics and Technology Management
Supervisor: Magnus Stålhane, Lars Magnus Hvattum
June 2019

NTNU
Kunnskap for ei betre verd

Robin Andersen
Vegard Hassel

In-game Betting and the FA English Premier League: The Contribution of Prediction Models

Master's thesis in Industrial Economics and Technology Management
Supervisor: Magnus Stålhane, Lars Magnus Hvattum
June 2019

Norwegian University of Science and Technology
Faculty of Economics and Management
Department of Industrial Economics and Technology Management

NTNU
Norwegian University of
Science and Technology

Abstract

The in-game betting market for FA English Premier League matches is rapidly increasing in value. As in all financial markets, the ability to generate positive returns on investments in such a market is to a large extent dependent upon the quality of information about future events and a proper wealth allocation strategy. This thesis is written in collaboration with Sportradar AG, a provider of prediction services to suppliers of odds in the sports betting market. With the aim of improving these predictions, the performance of a set of generated prediction models is compared.

Models for the scoreline distribution and the 1X2 distribution are generated for two different architectures. The first architecture is based on a long short-term memory network, while the other relies on the Weibull count distribution (McShane et al., 2008) where the Frank copula is used to model dependence between the goal processes of the opposing teams. The models are trained by minimization of the cross entropy, which coincides with a maximum likelihood approach. The comparisons are conducted in an attempt to determine the relative performance of a parametric count distribution and a complex black box algorithm motivated by the hypothesis that the former is overly restrictive for the purpose at hand. Furthermore, comparisons are made between the scoreline models and their 1X2 equivalents to test the hypothesis that knowledge of the scoreline distribution is of the essence when modelling the 1X2 distribution.

The results suggest that all models perform similarly on the 1X2 distribution according to both the accuracy score and cross entropy, with the best scores obtained by any model on these metrics being 0.5783 and 0.9423 respectively. The models based on the Weibull count distribution perform slightly better than the long short-term memory networks with respect to both the accuracy score and the cross entropy when considering the overall performance during an entire match. The ranked probability score strongly indicates the opposite and the long short-term memory models have significantly better predictive performance when taking the ordinal structure of scorelines into account.

The betting performance of the generated models is also evaluated subject to theoretically sound wealth allocation strategies. One of these is a dynamic Kelly betting strategy proposed by the authors, while the other is used as a means to test the static predictive performance of the models and to serve as a benchmark for the dynamic strategy. The results from the static strategy indicate that all models are able to generate positive returns for certain partial Kelly parameters in some stages of a football match. The best model combined with the most risk-averse partial Kelly strategy frequently generate returns of up to 15%, indicating good potential in the estimated probabilities. The dynamic strategy provides higher and more volatile results than comparable results from the static strategy, where a Kelly fraction of 0.05 combined with one of the Weibull count distribution models provides a return of 30%. However, neither combination of strategy and predictive model is able to consistently generate positive returns.

Sammendrag

Den samlede verdien på markedet for live-betting på kamper i FA English Premier League øker raskt. For å kunne tjene penger i et slikt marked trenger man, som i alle finansielle markeder, pålitelig informasjon on fremtidige hendelser og gode investeringsstrategier. Denne oppgaven er skrevet i samarbeid med Sportradar AG, som tilbyr prediksjonstjenester til tilbydere av odds i sportsmarkeder. Med et mål om å forbedre disse prediksjonene blir et sett av prediksjonmodeller generert og evaluert.

Modeller av den bivariate scoringsdistribusjonen og HUB-distribusjonen er generert fra to ulike arkitekturer. Den første arkitekturen er basert på et long short-term memory-nettverk og den andre bygger på Weibull count-distribusjonen (McShane et al., 2008). I denne arkitekturen brukes Frank copula for å modellere avhengigheten mellom scoringsprosessene til to motstandere. Alle modellene er trent ved å minimere cross entropy, som er sammenfallende med maximum likelihood-estimering. Sammenligninger mellom modellene er gjort for å avgjøre om det er best å benytte en parametrisk distribusjon eller en kompleks black-box-metode. Dette er motivert av en hypotese om at det første alternativet medfører unødvendige restriktjoner på læringsprosessen. I tillegg sammenlignes ytelsen til scoringsmodellene og HUB-modellene for å teste om informasjon om scoringsdistribusjonen er essensiell for å modellere HUB-distribusjonen nøyaktig.

Resultatene tyder på at alle modellene har nesten lik prediksjonsevne basert på både cross entropy og accuracy score når de måles på HUB-distribusjonen. De beste resultatene som observeres er henholdsvis 0.9423 og 0.5783. Modellene som er basert på Weibull count-distribusjonen har gjennom hele kampen generelt litt bedre ytelse enn de andre modellene, når ytelsen måles ut fra cross entropy og accuracy score. Ranked probability score sier at long short-term memory-modellene er signifikant best hvis man tar hensyn til den ordinale strukturen i scoringsdistribusjonen.

Betting-resultatene for de ulike modellene er basert på gode, teoretiske investeringsstrategier. En av disse er en dynamisk Kelly-strategi foreslått av forfatterene. Den andre er en strategi som brukes for å teste den statiske ytelsen til modellene og for å være en referanse for den dynamiske strategien. De statiske resultatene indikerer at alle prediksjonsmodellene genererer positiv avkastning i enkelte tidspunkt av kampene ved bruk av partial-Kelly strategier. Den beste modellen kombinert med en risiko-avers partial-Kelly strategi genererer ofte avkastning opp mot 15% og indikerer dermed at potensialet i prediksjonene er godt. Den dynamiske strategien gir høyere og mer volatil avkastning enn den statiske. Ved å bruke en partiell Kelly-parameter på 0.05 og sannsynligheter fra en av Weibull distribusjons-modellene ga den 30% avkastning med en akseptabelt lav volatilitet. Det er likevel ingen kombinasjon av prediksjonmodell og strategi som konsekvent gir positiv avkastning.

Preface

This thesis was written for the Department of Industrial Economics and Technology Management at the Norwegian University of Science and Technology as a finalization of the authors' Master of Science in Industrial Economics and Technology Management. The thesis was written during the spring of 2019 and is to some extent a continuation of the authors' project thesis from the autumn of 2018. Both authors specialize in computer science, artificial intelligence and finance.

The thesis is the result of a collaboration between the Department of Industrial Economics and Technology Management and Sportradar AG, a provider of sports data services and statistics.

We would like to thank our supervisors, Associate Professor Magnus Stålhane and Professor Lars Magnus Hvattum for their excellent feedback and guidance. We would also like to thank Sportradar AG for providing data and support.

Table of Contents

List of Figures

List of Tables

Terminology

Term	Description
Class, label	Dependent variable, response variable, possible outcomes
Features	Covariates, independent variables, explanatory variables
Training	Estimation of parameters in a distribution - learning, fitting
Validation	Estimation of the test performance of candidate models
Evaluation	Estimation of the general inductive power of the final models
Training set/sample	Data set/sample used to train candidate models
Validation set/sample	Data set/sample used for model selection
Test set/sample	Data set/sample used to evaluate the final models
Hold-out data	Data independent from that in the training set; *hold-out*= $valid \bigcup test$
Match	A single football game between two opposing teams
Scoreline	The joint number of goals scored in a match
1X2	The winner of a match (1 = home, X = draw, 2 = away)
Kick-off	The moment at which a match begins
Pre-game	The moment right before kick-off
Full-time	The moment at which the match ends
In-game	The open time interval between kick-off and full-time
Live-odds	Odds available in-game
Lineup	The 11 players starting for a team in a football match
ANN	Artificial neural networks
RNN	Recurrent artificial neural networks
LSTM	Long short-term memory

First set of terms frequently used in this thesis.

Term	Description
i.i.d.	Independent and identically distributed
Univariate	One-dimensional random variable
Multivariate	Multi-dimensional random variable
pdf	Probability density function
cdf	Cumulative probability density function
Count distribution	Probability distribution for the number of arrivals in some time interval
Maximum likelihood	Paradigm for statistical parameter estimation
Likelihood function	Loss function in maximum likelihood paradigm with respect to some pdf
Log-likelihood	Logarithm of the likelihood-function
Architecture	Framework, idea used for generating models
Model	Estimator based on architecture, data and scientific process
Binary classification	Classification with only two possible outcomes
Multiclass classification	Classification with more than two possible outcome
Ordinal	Inherent structure in the classes - not pairwise equidistant
Nominal	Complement of ordinal - all classes are equally similar
Multinomial classification	Multiclass nominal classification
Prediction model, classifier	Regression model for some probability distribution
Football prediction model	Prediction model for some type of outcome of football matches

Second set of terms frequently used in this thesis.

Chapter 1

Introduction

> *"The gambling known as business looks with austere disfavor upon the business known as gambling." - Ambrose Bierce*

This statement (Bierce and Ford, 2010) refers to the statistical advantage held by suppliers of money games in the case of perfect information about probabilities, as well as the lure of these games to the human mind. In this scenario, gamblers know of their negative expected return at the time of wagering. Nevertheless, they happily place their money in such games. Ever since Bernoulli introduced the *St.Petersburg Paradox* in 1738, investors are often assumed to be risk averse (Hayden and Platt, 2009). The course of action mentioned above breaks with this assumption if one ignores the utility of the act of gambling itself. Thus, despite the *no fun in gambling axiom* (von Neumann and Morgenstern, 1944), there must be at least some fun in gambling (Kusyszyn, 1984; Griffiths, 1990; Wood and Griffiths, 2008).

When no party holds perfect information about the probabilities of the game, the expected value of a wager is not defined. As sports betting contracts fall into this category, it introduces the requirement of accurate probability estimates. Additionally, the value of the global betting market is expected to reach USD 253 bn by 2020 (BusinessWire, 2016). Due to the considerable technological advances in recent years, the accessibility and thus the revenue of online betting platforms is rapidly increasing. In the European betting market, online sports betting accounts for 37% of the revenue, a figure increasing by the minute (Killick and Griffiths, 2018). When combining this with the altered broadcasting technology and mobile internet access, the proportion of in-game betting is also increasing. As an example, 45% of sports bets are estimated to be placed through mobile phone applications by the end of 2019 (SportsBettingDime.com, 2018).

The FA English Premier League (*EPL*) is estimated to entertain 4.7 bn people by live television coverage. This is unmatched by any other sports competition when ignoring single events such as the Super Bowl, UEFA Champions League final and the FIFA World Cup final (British Council, 2015). Thus, the in-game betting market for the outcome of EPL matches can validly be assumed to be of significant interest to a vast number of parties. The focus of the research presented in this thesis is the task of generating accurate estimates of the unbiased probability distributions in association football (*football*), as well as choosing proper wealth allocation strategies for generating positive returns in the in-game football betting markets.

1.1 Motivation and Research Questions

Due to its low scoring nature, the inherent randomness in the game of football is substantial. This is perfectly reflected by Diego Simeone, the manager of Atletico Madrid - *"Today was not meant to be for us. There is no such thing as justice in football. ..."* (Corrigan, 2019). Suppliers of financial products in the football betting market are also subject to this randomness. With an estimated market value of USD 2.4 bn, Sportradar AG (*Sportradar*) has claimed a position as an important provider of prediction services to these suppliers (Ozanian, 2018). The research presented in this thesis is a product of a collaboration between Sportradar and the Department of Industrial Economics and Technology Management at NTNU. The topic of research is thus partly motivated by the possibility of aiding the former in their quest to improve their prediction services for in-game football betting.

Predicting the optimal odds at different times during a football match requires accurate estimates of the true probability distributions in the game, as well as reliable market information. However, for a bettor in this market, the former requirement may be sufficient to generate positive financial returns. This research project rests on the hypothesis that accurate estimates of the unbiased probability distributions in football are sufficient for generating positive financial returns in this market when combined with a proper wealth allocation strategy.

The literature review conducted during their project thesis suggested to the authors that prediction models for the outcome of football matches often rely on a parametric count distribution. Despite the Poisson distribution being a popular choice in this regard, the work of Boshnakov et al. (2017) and the author's project thesis indicate that a generalization of this distribution proposed by (McShane et al., 2008) is the preferred choice. This distribution is hereby denoted the Weibull count distribution.

In recent years, extensive research on probabilistic artificial intelligence has resulted in a wide array of advanced black box statistical algorithms. An interesting topic is the comparison of these algorithms to models restricted by an assumed probability distribution. This is based on a hypothesis that there exist complex relationships in football that cannot be sufficiently captured by most of these distributions, and that parameter inference is thus redundant to some extent. In addition, black box approaches are hypothesized to decrease the risk of introducing human bias in the probability estimates. The presence of such bias in sports betting markets is widely supported in existing research, as presented by Daunhawer et al. (2017). Artificial neural networks constitute a general framework for statistical prediction models able to approximate any continuous function (Hornik, 1991). This motivates the use of these networks as a representation of black box algorithms in the comparison.

The winner of a football match is completely determined by the number of goals scored by the opposing teams. Hence, information about the goal distribution is hypothesized to be important for predicting the match winner. Since 1X2 markets constitute a large portion of the football betting market, an interesting aspect is the relative prediction power of scoreline predictors and explicit 1X2 predictors in these markets.

Furthermore, wealth allocation strategies are assumed to be of the essence for generating positive returns in the in-game betting market, as well as for evaluating the relative ability of prediction models to contribute to these returns. The research conducted in this project is to a large degree motivated by these assumptions, and an important topic is therefore to develop a theoretically founded betting strategy.

Based on this discussion, the topics of interest in this thesis can explicitly be represented by the following research questions:

RQ1 *How does the performance of an artificial neural network compare to that of a Weibull count distribution model, when considered with respect to both pre-game and in-game prediction of the outcome of EPL matches?*

RQ2 *How do models of the scoreline distribution in EPL matches compare to otherwise equivalent models of the 1X2 distribution on the same set of matches, when their performance is measured on the latter distribution, and the prediction task is the one stated in the previous question?*

RQ3 *How do the generated prediction models perform in the in-game betting market when subject to a theoretically sound betting strategy, and when the live-odds estimated by Sportradar are taken as the supply in the market?*

1.2 Outline

This report presents the conducted research and its basis in the following manner: The theoretical foundation of the research is presented in Chapter 2, and Chapter 3 presents the existing academic literature deemed relevant to the research. Then, Chapter 4 is dedicated to a presentation of the data, as well as a discussion of its reliability and relevant descriptive statistics. Chapter 5 describes the methodology utilized to generate prediction models for the outcome of football matches and for developing betting strategies, while Chapter 6 and Chapter 7 presents the results and corresponding discussions regarding the research questions. Based on the aforementioned discussion, a conclusion is stated in Chapter 8, before Chapter 9 concludes the report with some recommendations for further research.

As an introduction, the entire overview of the entire scientific procedure for generating prediction models is presented in Figure 1.1. An important note is that this is a very brief and abstract representation of the procedure and that the implementation of it varies somewhat throughout the report due to the different model architectures chosen for the research.

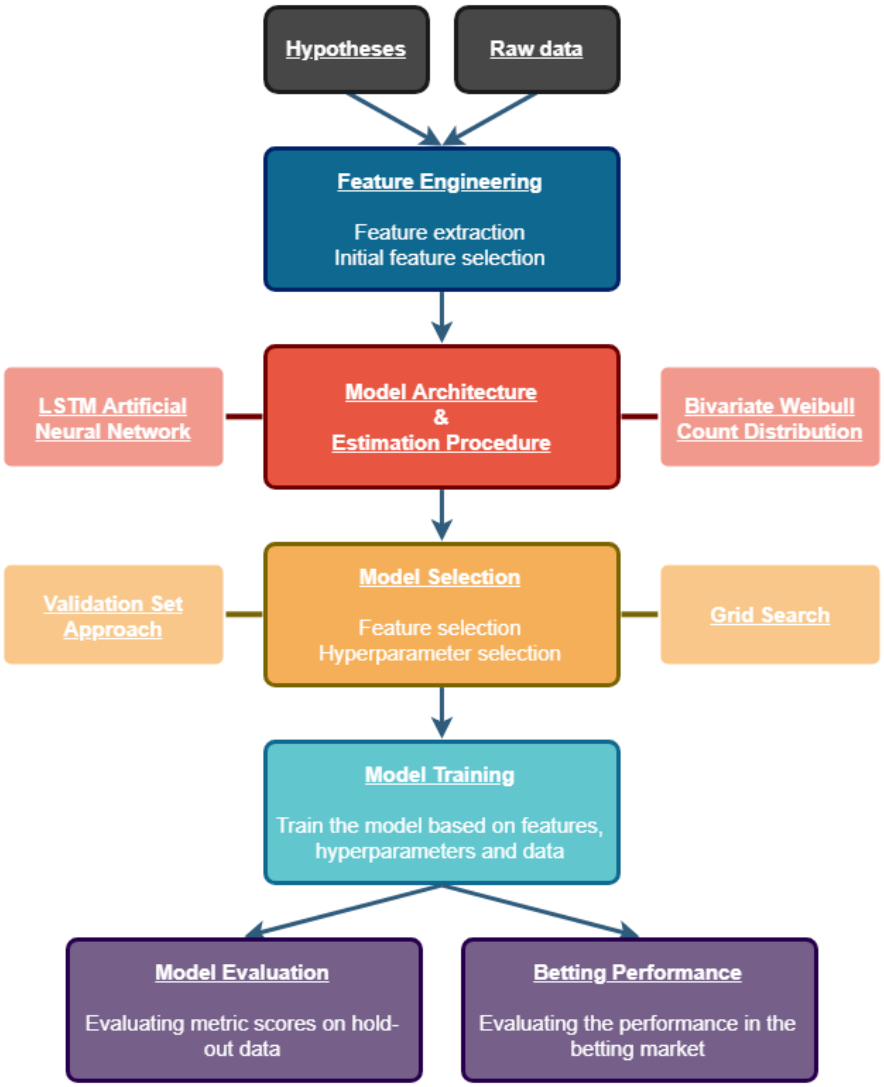


Figure 1.1: Abstract view of the scientific process.

Chapter 2

Theoretical Foundation

The theory deemed necessary for understanding the remainder of the thesis is presented in this chapter. Due to the mathematical nature of the research, a conventional notation is first presented in Section 2.1. Among the theory to be presented, the emphasis is first placed on metrics for evaluating probabilistic classifiers in Section 2.2, before Section 2.3 presents theory concerning the model selection process. Then, some parametric probability distributions and copula functions are presented in Section 2.4 and Section 2.5 respectively, before the focus turns to artificial neural networks in Section 2.6. The chapter ends with a presentation of theory relevant for the application of football prediction models in the in-game betting market. As a final note, this chapter presents the general strengths and weaknesses of the different methods and tools, along with their most relevant applications, while their explicit relevance to the research tasks is discussed more thoroughly in Chapter 5.

2.1 Mathematical Notation

Firstly, some mathematical notation is defined to avoid extensive use of definitions. This notation, which is presented in Table 2.1, constitute the convention throughout the thesis and variations of the convention are explained explicitly when deemed necessary. In general, the time index t is omitted for ease of notation when explaining general concepts that do not require the use of this index.

Notation	Description
A	Random variable or upper bound on index
$\boldsymbol{A}$	Matrix ($\boldsymbol{A} \in \mathcal{M}_{(n,m)}(\mathbb{R})$)
$\boldsymbol{a}$	Vector ($\boldsymbol{a} \in \mathcal{M}_{(n,1)}(\mathbb{R}) = \mathbb{R}^n$)
a	Scalar ($a \in \mathbb{R}$)
i	index - frequently used for football matches; $i \in \{1, .., N\}$
j	index - frequently used for features; $j \in \{1, .., J\}$
t	index - frequently used for time; $t \in \{1, ..., T\}$
x_{ijt}	Observation for feature j at time t in match i
$\boldsymbol{x_{it}}$	Observation of multiple features at time t in match i: $\boldsymbol{x_{it}} = (x_{ijt})_{j=1}^{J}$
$\boldsymbol{X_i}$	Matrix of observations for game i: $X_i = (\boldsymbol{x_{it}})_{t=1}^{T}$
$\boldsymbol{X}$	Matrix or 3D array of observations representing an entire data set
Y_i	Random variable for the outcome of game i
y_i	True classification for game i
$\boldsymbol{y}$	Vector of true classifications: $\boldsymbol{y} = (y_i)_{i=1}^{N}$
$\boldsymbol{\theta}$	Parameters in a probabilistic function to be estimated by a model
$\hat{\boldsymbol{\theta}}$	Estimate of $\boldsymbol{\theta}$
Ω_Z	Range of outcomes for the random variable Z
$f_Z(z)$	Probability of outcome z for the random variable Z
$F_Z(z)$	Cumulative probability of outcome z for the random variable Z
$f_{(W,Z)}(w, z)$	Joint probability of $W = w$ and $Z = z$
$f_Y(k\|\boldsymbol{x}, \hat{\boldsymbol{\theta}})$	Estimated probability of class k: $f_k(k\|\boldsymbol{x}, \hat{\boldsymbol{\theta}}) = Pr(Y = k\|\boldsymbol{x}, \hat{\boldsymbol{\theta}})$
$f_Y(\Omega_Y\|\boldsymbol{x}, \hat{\boldsymbol{\theta}})$	Estimated pdf over all classes Ω_Y: $f_Y(\Omega_Y\|\boldsymbol{x}, \hat{\boldsymbol{\theta}}) = (f_Y(k\|\boldsymbol{x}, \hat{\boldsymbol{\theta}}))_{k\in\Omega_Y}$
$I(P)$	Indicator variable for the proposition P
$R_{f_Y}(\boldsymbol{\theta}\|\boldsymbol{X}, \boldsymbol{y})$	Loss function - Function to be minimized during training

Table 2.1: Convention for mathematical notation used in this thesis.

2.2 Metrics - Evaluation of Probabilistic Classifiers

All candidate hypotheses in a scientific experiment must be tested, and some criteria must be defined to choose the proper candidate. Similarly, some criteria must be defined to test the chosen hypothesis on independent data. This section presents some of the most recognized criteria, or *metrics*, in this regard, as well as two important properties of statistical models.

2.2.1 Calibration and Discrimination

A necessity for a proper selection of metrics is a thorough understanding of the objective of the application of a model. A natural starting point in this regard are the terms *calibration* and *discrimination*. The calibration ability of a prediction model is defined as its ability to accurately estimate the entire probability distribution on independent data given some predefined classes, and when observed frequencies are taken as an estimate of the true distribution. The discriminatory ability of such a model refers to its ability to assign higher probabilities to the true classes than all the other classes (Steyerberg et al., 2010).

Good performance on one of these tasks does not imply good performance on the other. To see this, consider a hypothetical model that assigns the sample proportion of each class in the training set as the estimated probabilities for all future observations. Then, the calibration ability of this model should be decent, but its discriminatory ability is likely to be horrible. The opposite scenario is nicely illustrated by Daniel Kahneman, which states that by assigning a probability $p > 0.5$ to all outcomes that do happen, you would have a perfect discriminatory ability, but miserable calibration ability (Tetlock, 2015). Thus, a useful model should have a sufficient ability to both calibrate and discriminate (Alba et al., 2017).

2.2.2 Accuracy Score

A natural metric to consider for classification problems is the accuracy score, which is given by the proportion of correctly classified observations, as seen in the following equation:

$$Acc(\hat{\boldsymbol{\theta}}|f_Y, \boldsymbol{X}, \boldsymbol{y}) = \frac{1}{N}\sum_{i=1}^{N} I(y_i = argmax_k\{f_Y(k|\boldsymbol{x_i}, \hat{\boldsymbol{\theta}})\}) \tag{2.1}$$

Note that this metric does not give any indication of the calibration ability of the classifier. It merely yields an estimate of the classifier's performance on the task of predicting the most probable class, thus giving an indication of its discriminatory ability. Note also that this metric may be misleading as it assigns equal weight to each observation regardless of the distribution of classes in the sample (Tibshirani et al., 2017, p. 37-38).

2.2.3 Information Theory and Cross Entropy

Another widely used metric for classification problems is the *cross entropy*. This metric is proposed by Shannon (1948), who defines a mathematical theory of communication based on the notion of *entropy* in thermodynamics. This work is considered the basis for *information theory*, a field from which several quantities used in this thesis originate. Thus, some concepts are presented here based on Commenges (2015).

The entropy $H(Z)$ of a discrete random variable Z with true pdf $f = f_Z(z)$ is the expected value of $ln\big(\frac{1}{f_Z(\Omega_Z)}\big)$, that is

$$H(Z) = \sum_{z\in\Omega_Z} f_Z(z) ln\Big(\frac{1}{f_Z(z)}\Big) = -\sum_{z\in\Omega_Z} f_Z(z) ln(f_Z(z)) \tag{2.2}$$

where $\frac{1}{f_Z(z)}$ measures the degree of surprise in the observation that $Z = z$. Thus $H(Z)$ can be understood as a measure of the amount of information contained in Z. Now, given another distribution $g = g_Z(z)$, the cross entropy of g with respect to f is defined as the expected surprise $\mathbb{E}\big[ln\big(\frac{1}{g_Z(\Omega_Z)}\big)\big]$ given that f is true. In other words, it is a measure of the amount of information about Z contained in g given that f is true. Formally, the cross entropy is given by

$$CE_Z(g|f) = \sum_{z\in\Omega_Z} f_Z(z) ln\Big(\frac{1}{g_Z(z)}\Big) = -\sum_{z\in\Omega_Z} f_Z(z) ln(g_Z(z)) \tag{2.3}$$

Now, in a supervised learning scenario, one usually holds imperfect information about both of the distributions f and g. Thus, the sample distribution $I(\boldsymbol{y}) = I(y_{ik})_{i\in\{1,..,N\};k\in\Omega_Y}$ and the

estimator $f_Y(k|\boldsymbol{x_i}, \boldsymbol{\theta})$ of the distribution takes the roles of f, g respectively, as seen in Equation (2.4).

$$CE(\boldsymbol{\theta}|f_Y, \boldsymbol{X}, \boldsymbol{y}) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k\in\Omega_Y} I(y_i = k) ln\big(f_Y(k|\boldsymbol{x_i}, \boldsymbol{\theta})\big) \tag{2.4}$$

Here, the notation $CE(\boldsymbol{\theta}|f_Y, \boldsymbol{X}, \boldsymbol{y})$ is used to represent $CE_Y\big(f_Y(k|\boldsymbol{x_i}, \boldsymbol{\theta})|I(\boldsymbol{y}), \boldsymbol{X}\big)$ for simplicity, as well as for indicating that $\boldsymbol{\theta}$ is the only variable given an assumed probability distribution f_Y and data $\boldsymbol{X}$. This notation is hereafter used for the cross entropy. In the maximum likelihood paradigm, the loss function is the negative of the *log-likelihood function*. This function coincides with the cross entropy with respect to the sample distribution (Tibshirani et al., 2009, p.32). The maximum likelihood approach is assumed known and thus not discussed in detail in this thesis.

Note from Equation (2.4) that given a sample distribution $I(\boldsymbol{y})$, the performance of models based on different parametric distributions can be validly compared with the cross entropy, as it evaluates the combined choice of probability distribution and trained parameters with respect to $I(\boldsymbol{y})$. Furthermore, the cross entropy is a strictly proper metric, which means that its expected value is minimized by the true odds (Merkle and Steyvers, 2013). In other terms, it is a proper metric for evaluating the combined calibration and discrimination ability of a statistical model. Finally, the cross entropy can accurately capture uncertainty for nominal discrete random variables where the variance has no inherent meaning. However, this does not hold for ordinal random variables, as variance may be important in these scenarios (Commenges, 2015).

2.2.4 Ranked Probability Score

In order to measure the predictive performance of a model where the true classes follow an ordinal structure, one should also use a metric able to determine the quality of a prediction based on the similarity between classes. In this regard, Epstein (1969) proposed the *ranked probability score (RPS)*, which is also a strictly proper metric (Murphy, 1970). Given a discrete random variable Y, the RPS is given by

$$RPS(\hat{\boldsymbol{\theta}}|f_Y, \boldsymbol{X}, \boldsymbol{y}) = \frac{1}{N(|\Omega_Y| - 1)}\sum_{i=1}^{N}\sum_{k\in\Omega_Y}\Big(F_Y(k|\boldsymbol{x_i}, \hat{\boldsymbol{\theta}}) - I(y_i \geq k)\Big)^2 \tag{2.5}$$

where $I(y_i \geq k) \in \{0, 1\}$ is an indicator variable for the fact that y_i is considered larger than k with respect to an ordinal structure of the classes. Note that the RPS is a mean squared distance between the estimated cumulative distribution and a cumulative indicator variable. Thus, for two classes k_1, k_2 satisfying $k_1 < k_2 < y_i$, the penalty given to a prediction of k_1 is larger than if k_2 is predicted. This is due to the fact that the ordinal structure implies $F_Y(k_1|\boldsymbol{x_i}, \hat{\boldsymbol{\theta}}) < F_Y(k_2|\boldsymbol{x_i}, \hat{\boldsymbol{\theta}})$.

2.3 Model Selection

Based on the chosen evaluation metrics, one must specify a proper procedure for choosing among all the candidate models. Furthermore, all models of reality depend on both *hyperparameters* and *features*, where the former is a set of model-specific parameters, and the latter is used to represent the true distribution of the population that the model is meant to estimate. Thus,

the selection procedure should rely on a reasonable validation process of the model performance conditional on all proposed hyperparameters and features.

2.3.1 Validation Methods

Validation is the process of estimating $L = \mathbb{E}[R_{f_Y}(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{y})]$ on hold-out data, that is, the expected loss on an independent data sample when the estimator $f_Y(\Omega_Y|\boldsymbol{X}, \boldsymbol{\theta})$ is applied. The motivation for doing this is that L is underestimated if the estimates are solely based on the training set. A popular set of approaches for this task is the set of *cross validation (CV)* methods. These methods are often used when the sample at hand is too small to hold out a separate validation set, an approach conventionally denoted the *validation set approach*. Referring to the *bias-variance trade-off*, the validation set approach yields a low variance in the estimates of L due to a large validation set, but a high bias since a large proportion of the training data is removed for validation. On the other hand, *leave-one-out CV*, which implies training $N - 1$ models and holding out one observation for validation in each model, yields low bias and high variance by the same arguments. *K-Fold CV*, which is a generalization of leave-one-out CV due to the possibility of choosing $K \neq 1$, is a trade-off between the two former approaches and aims to obtain both low bias and low variance in the estimates of L (Tibshirani et al., 2017, p.176-184).

2.3.2 Feature and Hyperparameter Selection

Based on the chosen validation approach, a set of hyperparameters and features must be chosen. Regarding the hyperparameters, a widely used approach for choosing these parameters is to perform a *grid search*, which simply entails training and validating the model for every combination of some suggested hyperparameter values. Then, the chosen combination is the one with the best validation score, as measured by a chosen metric. Considering the feature selection process, the best approach usually differs depending on the chosen model framework and the amount of available data. The most relevant approaches for this research project are presented here.

Subset selection algorithms

One of these approaches is to utilize a *subset selection algorithm*, which is a grid search for selecting the best subset of data features. Due to the $O(2^d)$ complexity of an exhaustive search given d candidate features, greedy variations of this approach such as *stepwise selection algorithms* are most commonly used. These algorithms work by either greedily adding or removing features based on a criterion. This approach is usually performed by using a metric that penalizes complex models when the amount of available data is very limited, but may also be performed within a validation loop (Tibshirani et al., 2017, p.206-209).

Regularizers

Another approach for selecting or partly ignoring features is to use a regularization technique. A regularizer is a function incorporating a penalty term in the loss function to automatically shrink or nullify irrelevant features. Thus, a penalty is given during the training procedure by introducing some bias rather than being imposed during evaluation. The main argument for this approach is that it may prevent overfitting. It is therefore often chosen in combination

with complex model architectures, as the risk of overfitting imposed by such architectures is substantial.

The penalty terms are usually chosen to be L_q *norms* $\|\boldsymbol{\theta}\|_q = (\sum_{j=1}^{J} \theta_j^q)^{\frac{1}{q}}$ of the vector of parameters to be estimated. The most widely used penalty terms are the L_1 and squared L_2 norms, usually referred to as *lasso* and *ridge* penalties respectively. A common approach in recent years is to combine the lasso and ridge penalties to obtain all the benefits of their respective properties. This penalty term is usually called an *elastic net* and is given by

$$ElasticNet(\theta) = \lambda_1 \|\theta\|_1 + \lambda_2 \|\theta\|_2^2 \tag{2.6}$$

where λ_1 and λ_2 are parameters to be chosen and θ are other model parameters. A common approach is to determine λ_1 and λ_2 in a grid search. By incorporating the lasso penalty in the loss function, the learning algorithm is able to automatically perform model selection during training by fixing parameters to zero given that λ_1 is large enough, which is due to the properties of the L_1 norm. This property is not held by the ridge penalty unless $\lambda_2 \to \infty$, since the squared penalty terms imply a penalty proportional to the size of θ_j for all features j (Tibshirani et al., 2009, p.61-73).

Mutual Information

A completely different approach for feature selection is to consider a measure of dependency between the true classes and the features of the training set. This is a typical approach in scenarios where the entire data set is required for training due to limited data access. The most commonly used measure in this regard is the correlation coefficient. A problem with this coefficient is that it can only capture linear dependencies, and may thus be misleading if the true dependencies are nonlinear. The *mutual information* is based on a different idea than correlation and accounts for nonlinear dependence. This measure is closely related to the notion of entropy presented in Section 2.2.3. Specifically, it is a measure of the amount of information obtained about a random variable by observing another random variable. Another important property of the mutual information is that it can capture dependencies between discrete random variables, in contrast to the correlation coefficient (Commenges, 2015).

Formally, let (Z, W) be a pair of random variables, $MI(Z, W)$ be their mutual information and $H(Z)$, $H(W)$ denote the entropy of Z, W respectively. Then

$$MI(Z, W) \in [0, max\{H(Z), H(W)\}]$$

determines the similarity between the joint distribution $f_{(Z,W)}(z, w)$ and the same distribution under the assumption of statistical independence, that is $f_{(Z,W)}(z, w) = f_Z(z) f_W(w)$. Furthermore, $MI(Z, W) = 0 \iff Z, W$ are independent, and $MI(Z, W) = H(Z) = H(W) \iff X, Y$ are deterministic functions of each other (Church and Hanks, 1990). The mutual information is given in Equation (2.7) for the case where Z and W are discrete.

$$MI(Z, W) = \sum_{z \in \Omega_Z} \sum_{w \in \Omega_W} f_{(Z,W)}(z, w) ln \left(\frac{f_{(Z,W)}(z, w)}{f_Z(z) f_W(w)} \right) \tag{2.7}$$

2.4 Probability Distributions for Count Processes

The Poisson distribution is widely used to model the number of events occurring during a given time interval. It is the count distribution equivalent of the exponential distribution, meaning that the times between two pairs of events in disjoint time intervals in a Poisson process are i.i.d. exponentially distributed random variables. These distributions are assumed known and no further elaboration is therefore made about them.

The Weibull distribution is closely related to the exponential distribution. Its pdf is given by

$$P(T = t) = f_T(t) = \lambda c t^{c-1} \exp(-\lambda t^c), \text{ for t} \geq 0, \tag{2.8}$$

where c and λ are the shape and scale parameters respectively. Consider its *hazard function*, $h(t) = \lambda c t^{c-1}$, which represent the development of the occurrence rate over time. This function is monotonically increasing for $c > 1$ and monotonically decreasing for $c < 1$. When fixing $c = 1$, the Weibull distribution coincides with the exponential distribution and the hazard function reduces to $h(t) = \lambda$. Thus, it becomes clear that the Weibull distribution is a generalization of the exponential distribution.

The corresponding count distribution of the Weibull distribution, hereby referred to as the *Weibull count distribution*, is thus a generalization of the Poisson distribution (McShane et al., 2008). Specifically, the Weibull count distribution relaxes the assumption of equidispersion implicit in the Poisson distribution, meaning that it allows the mean and the variance of the distribution to differ. According to (McShane et al., 2008), the pdf and the cdf of the Weibull count distribution are given by

$$f_{W_t}(w|c, \lambda) = \sum_{i=w}^{\infty} \frac{(-1)^{w+i} (\lambda t^c)^i \alpha_i^w}{\Gamma(ci + 1)}; w \in \mathbb{N}. \tag{2.9}$$

$$\alpha_i^w = \begin{cases} \frac{\Gamma(ci+1)}{\Gamma(i+1)}; & w = 0; i \in \{w, w+1, ..\} \\ \sum_{j=w-1}^{i-1} \alpha_j^{w-1} \frac{\Gamma(ci-cj+1)}{\Gamma(i-j+1)}, & w \in \mathbb{N}^+; i \in \{w, w+1, ..\} \end{cases} \tag{2.10}$$

$$F_{W_t}(w|c, \lambda) = P(W_t \leq w|c, \lambda) = \sum_{i=1}^{w} f_{W_t}(i|c, \lambda) \tag{2.11}$$

2.5 Copula Functions

Copula functions are multivariate probability distributions used to describe the dependence between uniformly distributed random variables (Haugh, 2016). Formally, the joint cdf of two random variables can be expressed by a copula and their marginal cdfs $F_Z(z), F_W(w) \sim Uniform(0, 1)$ in the following manner (Sklar, 1959):

$$F_{(Z,W)}(z, w) = C(F_Z(z), F_W(w)). \tag{2.12}$$

A special family of copulas are the Archimedean copulas, which can be defined by $C(u, v) = \phi^{-1}(\phi(u)+\phi(v))$ in the bivariate case, where $\phi : [0, 1] \to [0, \infty]$ is a continuous, strictly increasing and convex generating function (Fischer and Köck, 2012). A subset of this family is particularly

popular due to their ability to model dependence in arbitrarily high dimensions by estimating a single parameter. One such copula is the *Frank copula*, defined by

$$C(u,v) = -\frac{1}{\kappa}\ln\left(1 + \frac{(e^{-\kappa u}-1)(e^{-\kappa v}-1)}{e^{-\kappa}-1}\right) \tag{2.13}$$

where κ is the parameter representing the dependence between the marginal cumulative distributions. It is worth noting that the copula is a cumulative distribution, but the pdf for discrete random variables can easily be obtained by the relation (Nelsen, 2006)

$$f(z,w) = F(z,w) - F(z-1,w) - F(z,w-1) + F(z-1,w-1) \tag{2.14}$$

where the subscripts are omitted for simplicity.

2.6 Artificial Neural Networks

Artificial neural networks (ANNs) constitute a computational framework inspired by biological neural networks and mimics the computational mechanisms of the animal brain. As for their biological equivalents, the core elements of ANNs are neurons connected in a structure for the purpose of propagating information (van Gerven and Bohte, 2017). From a mathematical perspective, ANNs are, in simple terms, nonlinear statistical models (Tibshirani et al., 2009, p.392). To explain the concept, a simple ANN architecture is presented here. This architecture represents a two-stage regression procedure and is usually referred to as a *feed-forward network* with a single hidden layer. An example of such a network and its functionality is presented by a network diagram in Figure 2.1, where $\Theta^0_{(5,3)}$,$\Theta^1_{(4,1)}$ are parameter matrices to be estimated. The extra dimension accounts for the *bias* in the linear transformation, which is not shown in the figure.

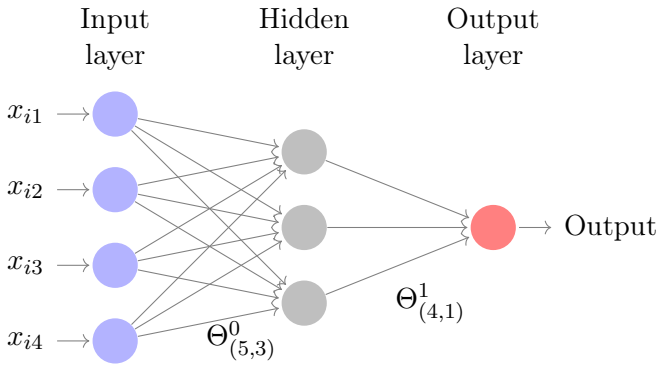


Figure 2.1: Architecture of a feed-forward network with one hidden layer.

Now, some terminology is required. Figure 2.1 contains a *directed acyclic graph* consisting of *edges* and *nodes*. These represent the *information flow* from input to output and neurons respectively. All the neurons are structured in *layers* with respect to this information flow. That is, a layer may be defined as a set of neurons L that all receive information from the same set of neurons, $S = parents(L)$. All layers which are neither input layers nor output layers are called *hidden layers*, as the features derived from these layers are not directly observable.

The special case of one hidden layer and a general classification problem with $K > 2$ classes is considered here, in accordance with Figure 2.1. In this case, the output layer of the network

consists of one node calculating the entire distribution $f_Y(\Omega_Y|\boldsymbol{x_i}, \hat{\boldsymbol{\theta}})$. The dynamics of this network is presented below (Tibshirani et al., 2009, p.392-397).

Each neuron $m \in \{1, .., M\}$ in the hidden layer performs a linear transformation $l_m(\cdot)$ of the input observation $\boldsymbol{x}$, before an *activation function* $h_m = \sigma(l_m(\cdot))$ is applied and taken as the output from m. A typical activation function in the hidden layers is the *hyperbolic tangent* $\sigma(\cdot) = tanh(\cdot)$, while the output activation function is usually the *softmax function* for multinomial models. This function is a generalization of the *sigmoid function* $\sigma(\cdot) = 1/(1 + e^{-(\cdot)})$ utilized in binary logistic regression. Hence, an ANN trained with the softmax as the output activation function and cross entropy as the loss function is simply a multinomial logistic regression of the features derived in its hidden layer. The softmax is given by

$$softmax_Y(\Omega_Y|\boldsymbol{z}) = \left(\frac{e^{z_k}}{\sum_{l=\Omega_Y} e^{z_l}} \right)_{k \in \Omega_Y} \tag{2.15}$$

The output layer takes the combined output from the hidden layer as an input vector, but a linear transformation is not conducted on this vector. Formally, let $\boldsymbol{h} = (h_1, ..., h_M)$, and consider the case where we have one node in the output layer. Then, the procedure can be stated as follows:

$$l_m = \theta_{0,m} + \boldsymbol{\theta_m^T} \boldsymbol{x}, m \in \{1, ..., M\}; \tag{2.16}$$

$$h_m = \sigma(l_m), m \in \{1, ..., M\}; \tag{2.17}$$

$$f_Y(\Omega_Y|\boldsymbol{X}, \hat{\boldsymbol{\theta}}) = softmax_Y(\Omega_Y|\boldsymbol{h}) \tag{2.18}$$

Here, $\boldsymbol{\theta_0} = (\theta_{0m})_{m=1}^M$ represent the bias introduced in the hidden layers. Note that in this special case we require that $M = |\Omega_Y|$. A presentation of the two-stage transformation conducted by each hidden neuron m can also be seen in Figure 2.2.

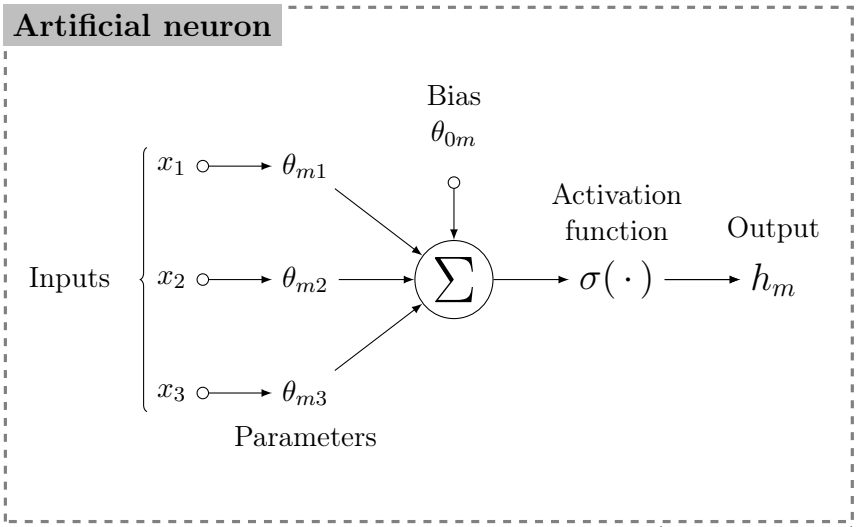


Figure 2.2: Dynamics of a hidden artificial neuron (Taskjelle, 2017).

The procedure presented here can be generalized to include several hidden layers. In this case, the ANNs are called deep neural networks and are usually used in complex learning tasks such as

logical games (Mnih et al., 2013; Silver et al., 2016). By the *Universal Approximation Theorem*, an ANN with at least one hidden layer and a finite number of nodes has the ability to approximate any continuous function (Hornik, 1991).

Artificial neural networks are usually trained using a variant of the *backpropagation algorithm*, which is a gradient descent approach based on the chain rule of mathematical differentiation. Training is therefore conducted by loops of forward propagation of information through the network and parameter adjustments are made by propagating information about the loss function back through the network (Tibshirani et al., 2009, p.397). One such information loop is called an *epoch*, and the number of epochs is usually chosen as a hyperparameter. An advanced variation of the backpropagation algorithm is the adaptive moment estimation (ADAM) algorithm, a first-order gradient-based stochastic optimization algorithm, which generates and utilizes adaptive estimates of the lower-order moments of the stochastic loss function (Kingma and Ba, 2014).

2.6.1 Recurrent Neural Networks

A significant limitation of ordinary feed-forward neural networks is that they are *amnesiacs*, meaning that they cannot store previously learned information. This implies that they are not capable of learning sequences or processes, such as time series. To address this issue, Jordan (1986) proposed the *recurrent neural network (RNN)*. This network contains loops of information such that previously learned information can be utilized when a new set of data is encountered. Due to this ability, RNNs have been widely used in complex learning tasks, like sequence learning (Sutskever et al., 2014) and natural language processing (Bahdanau et al., 2014; Cho et al., 2014). The information loops can be understood as a copy of the entire network and its state at a given step in time or space. An illustration of this concept can be seen in Figure 2.3, where the observation index i is omitted.

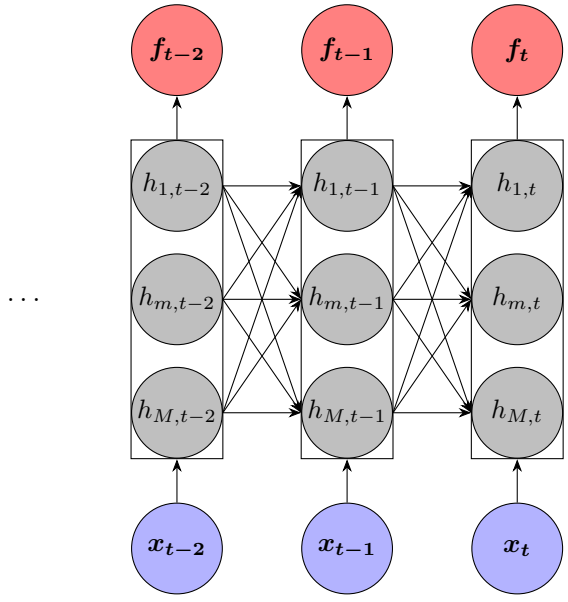


Figure 2.3: Illustration of a many-to-many RNN architecture.

Note that the network in Figure 2.3 only has recurrent connections in the hidden layer and that it outputs a prediction at every step of the sequence. The latter aspect is conventionally called a *many-to-many* approach, in contrast to a *many-to-one* approach, in which the network only outputs a prediction at the last step. In the former approach, the loss function is the average loss

over all steps, a function that may be very misleading if the output at all nodes is not equally important. On the contrary, a many-to-one RNN is trained using an ordinary loss function on the output at the last step.

Now, consider the dynamics of RNNs. Let $\boldsymbol{h_t} = (h_{1t}, ..., h_{Mt})$ be the output of the hidden layer at step $t \in \{1, .., T\}$. The equivalent of Equation (2.17) for an RNN is then given by

$$h_{m,t} = \begin{cases} \sigma(l_{m,t}) & t = 0 \\ \sigma(l_{m,t} + \boldsymbol{\theta}_{\boldsymbol{m,t}}^{\boldsymbol{T}} \boldsymbol{h_{m,t-1}}) & t \in \{1, .., T\} \end{cases} \tag{2.19}$$

where $m \in \{1, \ldots, M\}$, $l_{m,t}$ is defined equivalently to Equation (2.16) and $\boldsymbol{\theta_{m,t}}$ are the parameters in the recurrent connections from step $t-1$ to t in node m. A well-known issue for RNNs is that they are usually unable to learn long sequences. This is due to the problem of *vanishing* or *exploding* gradients in the loss function of the backpropagation procedure. The reason for this problem is that stored information is subject to the multiplication implied by the chain rule, which entails a high risk of numerical instability. Thus, the loss signal may increase or decrease with an exponential rate, inhibiting a proper propagation of the signal and thus also the learning process (Hochreiter et al., 2001).

2.6.2 Long Short-Term Memory Networks

One widely recognized approach for avoiding the problem of vanishing or exploding gradients is the use of *long short-term memory (LSTM) networks* (Hochreiter and Schmidhuber, 1997). These networks contain designated *LSTM cells* that themselves learn how long to store information. The significant difference between LSTM networks and RNNs is that stored information is not subject to the chain rule due to these designated cells. LSTM networks are therefore often used on complex sequential learning problems such as machine translation (Graves, 2013), music composition (Eck and Schmidhuber, 2019; Agrawal et al., 2018), image generation (Vinyals et al., 2015) and general question answering (Wang and Nyberg, 2015). Also note that each layer in an LSTM network is represented by a single LSTM cell. The architecture of such a cell can be seen in Figure 2.4.

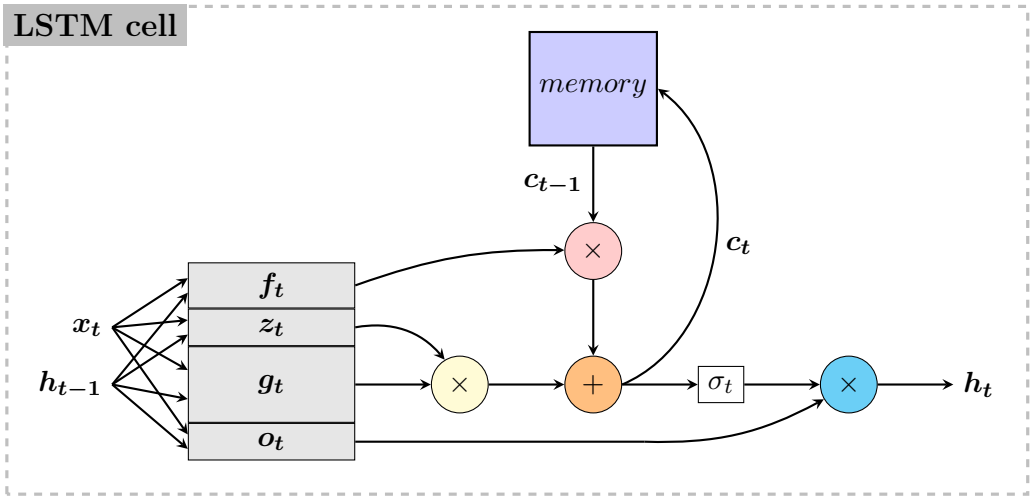


Figure 2.4: The internal architecture of an LSTM cell (Veličković, 2017).

Note that the observation index i is omitted in Figure 2.4. Now, consider the process from input

$(\boldsymbol{h_{l,t-1}}, \boldsymbol{x_{it}})$ to output $\boldsymbol{h_{l,t}}$ in a such a cell l. The presentation of this process follows that of Hochreiter and Schmidhuber (1997). Let $\sigma(\cdot)$ denote the sigmoid function, and let $\boldsymbol{W_q}, \boldsymbol{U_q}$ and $\boldsymbol{b_q}$ denote parameter matrices and a bias vector for gate $q \in \{f, g, o, z\}$.

The first step of the process is to choose the proportion of information to be ignored among that stored in the previous *cell state* $\boldsymbol{c_{l,t-1}}$. This task is performed by the *forget gate* $\boldsymbol{f_{l,t}}$ in the following manner:

$$\boldsymbol{f_{l,t}} = \sigma(\boldsymbol{W_{l,f}}\boldsymbol{x_{l,it}} + \boldsymbol{U_{l,f}}\boldsymbol{h_{l,t-1}} + \boldsymbol{b_{l,f}}) \tag{2.20}$$

Here, $\sigma(\cdot)$ operations are to be understood element-wise. Also, $f_{l,jt} = 0$ indicate that no information should be stored about feature j, while $f_{l,jt} = 1$ indicate the opposite. The amount of new information to be carried forward and stored in the new cell state must now be chosen. This is done in the *input gate* $\boldsymbol{z_{l,t}}$ in a equivalent manner as to that for $\boldsymbol{f_{l,t}}$, while a separate procedure in the *candidate gate* $\boldsymbol{g_{l,t}}$ construct a vector of candidate values to be stored

$$\boldsymbol{z_{l,t}} = \sigma(\boldsymbol{W_{l,z}}\boldsymbol{x_{l,it}} + \boldsymbol{U_{l,z}}\boldsymbol{h_{l,t-1}} + \boldsymbol{b_{l,z}}) \tag{2.21}$$

$$\boldsymbol{g_{l,t}} = tanh(\boldsymbol{W_{l,g}}\boldsymbol{x_{l,it}} + \boldsymbol{U_{l,g}}\boldsymbol{h_{l,t-1}} + \boldsymbol{b_{l,g}}) \tag{2.22}$$

Here, the interpretation of $\boldsymbol{z_{l,t}}$ is equivalent to that of $\boldsymbol{f_{l,t}}$. The next step is to use $\boldsymbol{f_{l,t}}$ and $\boldsymbol{z_{l,t}}$ to construct the new cell state by weighting the old and new information $\boldsymbol{c_{l,t-1}}$ and $\boldsymbol{g_{l,t}}$ respectively

$$\boldsymbol{c_{l,t}} = \boldsymbol{f_{l,t}} \odot \boldsymbol{c_{l,t-1}} + \boldsymbol{z_{l,t}} \odot \boldsymbol{g_{l,t}} \tag{2.23}$$

Here, $\odot$ is the *Hadamard product*, which represents element-wise multiplication. Finally, the new output must be chosen. First, the *output gate* $\boldsymbol{o_{l,t}}$ performs an equivalent calculation to that in the forget gate and the input gates. Then, this value is multiplied by a transformed version of the new cell state $\boldsymbol{c_{l,t}}$ to determine the weights assigned to the transformed features.

$$\boldsymbol{o_{l,t}} = \sigma(\boldsymbol{W_{l,o}}\boldsymbol{x_{l,t}} + \boldsymbol{U_{l,o}}\boldsymbol{h_{l,t-1}} + \boldsymbol{b_{l,o}}) \tag{2.24}$$

$$\boldsymbol{h_{l,t}} = \boldsymbol{o_{l,t}} \odot tanh(\boldsymbol{c_{l,t}}) \tag{2.25}$$

As stated, LSTM cells are used to build the entire network architecture of an LSTM network. An example of such a network consisting of three cells is shown in Figure 2.5, where the notation follows that from Figure 2.4.

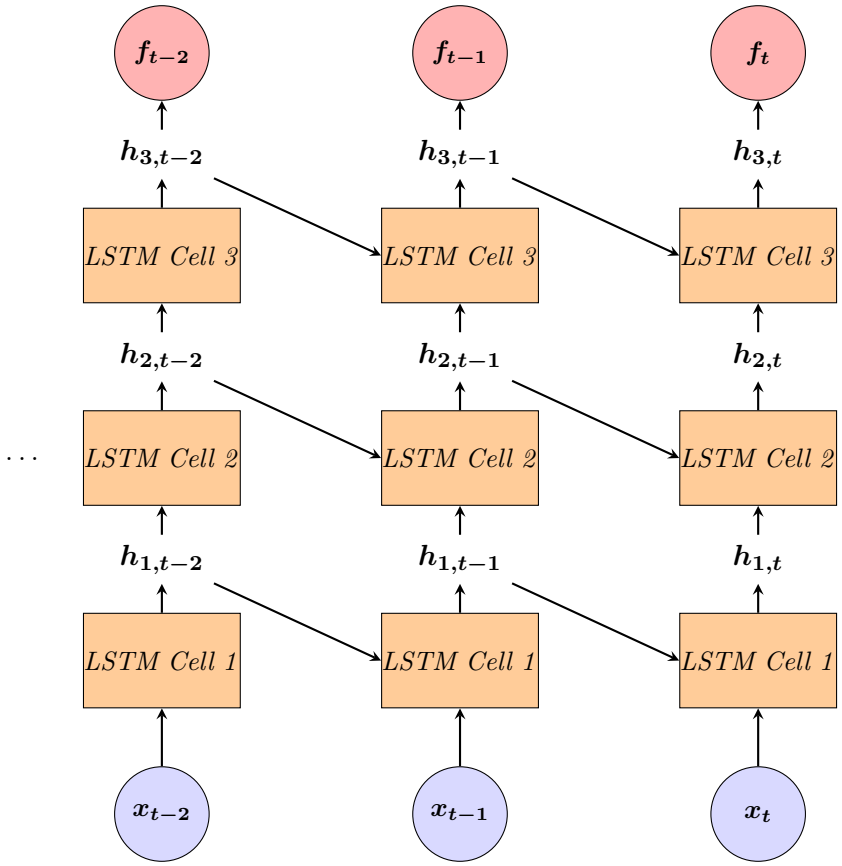


Figure 2.5: Illustation of an LSTM network with three layers.

2.6.3 Estimation Procedures

In addition to the problems discussed regarding ANNs thus far, the complexity of such model architectures entails some typical problems in the estimation procedure. These problems and some recognized methods for avoiding them are presented here.

Standardization

Unlike most statistical methods, ANNs are not *scale invariant*, meaning that the numerical scale of the data may considerably alter the learning ability. Thus, processing of the data prior to training is often a necessary part of the learning process. The most widely used approach in this regard is to *standardize* the data by performing the transformations given in Equation (2.26)-Equation (2.28).

$$\boldsymbol{X}_{train} = \frac{\boldsymbol{X}_{train} - \hat{\boldsymbol{\mu}}_{train}}{\hat{\boldsymbol{\sigma}}_{train}} \tag{2.26}$$

$$\boldsymbol{X}_{valid} = \frac{\boldsymbol{X}_{valid} - \hat{\boldsymbol{\mu}}_{train}}{\hat{\boldsymbol{\sigma}}_{train}} \tag{2.27}$$

$$\boldsymbol{X}_{test} = \frac{\boldsymbol{X}_{test} - \hat{\boldsymbol{\mu}}_{train}}{\hat{\boldsymbol{\sigma}}_{train}} \tag{2.28}$$

Where $\hat{\boldsymbol{\mu}}_{train}$ and $\hat{\boldsymbol{\sigma}}_{train}$ are unbiased estimates of the mean and the standard deviation of the training data, and the operations are to be understood element-wise. Note that these estimates are not based on information from the validation and test sets, as no information about this data should be used for training.

Dropout

Methods for avoiding overfitting is of the essence for neural networks. The most frequently used approach to reduce this risk is to use a *dropout* technique. This approach is based on the idea that by randomly ignoring signals, one reduces the risk of learning relationships only present in the training data. A dropout layer has a simple function: For every propagation of information through the network, drop any node in the previous layer by a user-specified probability p (Srivastava et al., 2014). An illustration of this approach can be seen in Figure 2.6

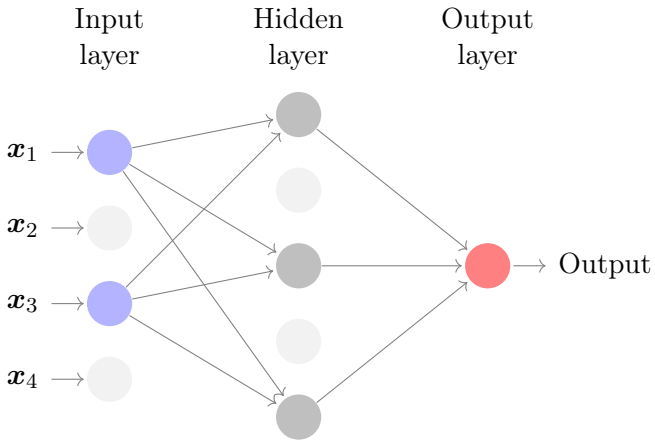


Figure 2.6: Illustration of the dropout technique.

Regularization revisited

Another aid against overfitting is to use regularization, which may be used as a substitute or a complement to dropout. The strongest argument for the use of regularizers when training ANNs is to avoid the problem of choosing the size of the network. This is done by choosing a network assumed to be at least *large enough*, then determining proper hyperparameters for the regularizer and thus the amount of information to ignore (Zaremba et al., 2014). The chosen regularizer in this context is an elastic net, as seen in Equation (2.6).

Batch normalization

Recall that every layer l^* in an ANN is responsible for performing a two-stage transformation of its input. Thus, for each round of backpropagation and every layer l^*, small changes in the output of the preceding layer may shift the distribution of inputs to the layer to a range subject to vanishing gradients, halting the learning process in layers $\{1, .., l^*\}$. This is called *internal covariate shift*. It has been shown in several studies that normalizing the input to every layer by introducing normalization layers can alleviate this problem (Ioffe and Szegedy, 2015; Cooijmans et al., 2016; Laurent et al., 2016). ANNs are usually trained in *batches*, or subsets, of training data, meaning that each loop of information propagation is performed on a single subset. Thus,

the approach of normalizing the input is called *batch normalization*. According to Cooijmans et al. (2016), ordinary batch normalization can be conducted in the following manner in a deep ANN

$$BN(\boldsymbol{h}|\gamma,\beta) = \boldsymbol{\beta} + \boldsymbol{\gamma}\frac{\boldsymbol{h} - \hat{\mathbb{E}}[\boldsymbol{h}]}{\sqrt{\hat{Var}[\boldsymbol{h}] + \epsilon}} \tag{2.29}$$

where $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ are vectors of size $\|\boldsymbol{h}\|$ with each element being equal to γ and β respectively, $\epsilon \in \mathbb{R}$ is a regularization parameter, and the transformation is to be understood element-wise. They also propose a methodology for performing batch normalization for LSTM networks. The dynamics of a LSTM cell subject to this approach is shown in Equation (2.30) to Equation (2.32).

$$\begin{bmatrix} \tilde{\boldsymbol{f}}_{\boldsymbol{t}}^{\boldsymbol{T}} \\ \tilde{\boldsymbol{i}}_{\boldsymbol{t}}^{\boldsymbol{T}} \\ \tilde{\boldsymbol{o}}_{\boldsymbol{t}}^{\boldsymbol{T}} \\ \tilde{\boldsymbol{g}}_{\boldsymbol{t}}^{\boldsymbol{T}} \end{bmatrix} = BN(\boldsymbol{W}\boldsymbol{h}_{\boldsymbol{t-1}}|\gamma_h,\beta_h) + BN(\boldsymbol{U}\boldsymbol{x}_{\boldsymbol{t}}|\gamma_x,\beta_x) + \boldsymbol{b} \tag{2.30}$$

$$\boldsymbol{c}_{\boldsymbol{t}} = \sigma(\tilde{\boldsymbol{f}}_{\boldsymbol{t}}) \odot \boldsymbol{c}_{\boldsymbol{t-1}} + \sigma(\tilde{\boldsymbol{i}}_{\boldsymbol{t}}) \odot tanh(\tilde{\boldsymbol{g}}_{\boldsymbol{t}}) \tag{2.31}$$

$$\boldsymbol{h}_{\boldsymbol{t}} = \sigma(\tilde{\boldsymbol{o}}_{\boldsymbol{t}}) \odot tanh(\boldsymbol{c}_{\boldsymbol{t}}) \tag{2.32}$$

Note that the cell index l and the gate subscript is omitted in these equations for simplicity. In textual terms, Cooijmans et al. (2016) propose to perform the transform $BN(\cdot)$ on both the recurrent term $\boldsymbol{W}\boldsymbol{h}_{\boldsymbol{t-1}}$ and the new input $\boldsymbol{U}\boldsymbol{x}_{\boldsymbol{t}}$ separately, aiding the model in the task of controlling the relative contribution of these terms. They also propose to set $\beta_h = \beta_x = 0$ to avoid redundancy, as well as not performing the $BN(\cdot)$ transform in the cell update in order to preserve the gradient flow through $\boldsymbol{c}_{\boldsymbol{t}}$. Otherwise, they found $\gamma_h = \gamma_x = 0.1$ to be a good choice over a large set of problems.

2.7 Betting Strategies

A core part of the research presented in this thesis is the application of prediction models in the in-game football betting market. In this regard, some prerequisites are presented in this section.

2.7.1 Odds

First, with the aim of explaining the financial products supplied in the football betting market, the fundamental quantity representing the price of these products is defined. This quantity is the *odds*, which represents the payment in the case of a successful bet from the perspective of the bettor. There exist several conventions for quoting this price, but the only convention considered in detail here is that explicitly following the mathematical definition of odds.

Formally, let p be the true probability of a given outcome of an event. Then, the odds of this outcome is given by

$$s(p) = \frac{p}{1-p} \tag{2.33}$$

Next, let p be the probability of a favourable outcome for the bettor. Then $s(p)^{-1}$ is denoted the *fractional odds* and is the supplied odds in the betting market (Nakharutai et al., 2019). Thus, in a betting contract where both the supplier and bettor have the same belief or estimate $\hat{p}_s = \hat{p}_b = \hat{p}$ of p, the fair supplied odds is given by

$$s(\hat{p}, \alpha) = \frac{1 - \hat{p}}{\alpha \hat{p}} = \frac{s(\hat{p})^{-1}}{\alpha} \tag{2.34}$$

where the risk premium $\frac{1}{\alpha} - 1 = 0$. This reflects the fact that the expected return $\mathbb{E}\big[R(\hat{p})\big] = s(\hat{p}, 1)\hat{p} - (1 - \hat{p}) = 0$ for both parties. Due to both the underlying risk in p and the uncertainty in the estimate $\hat{p}$, the odds is supplied incorporating a risk premium $\frac{1}{\alpha} - 1 > 0 \iff \alpha > 1$. Note that a rational bettor would never enter a betting contract with such odds given the same belief as the supplier. This is because the expected return of the bettor is given by

$$\mathbb{E}\big[R(\hat{p}_b)\big] = s(\hat{p}_s, \alpha)\hat{p}_b - (1 - \hat{p}_b) \tag{2.35}$$

which is negative if $\hat{p}_b = \hat{p}_s$ and $\alpha > 1$. In fact, a rational bettor would only agree on the bet if and only if $\hat{p}_g > (s(\hat{p}_s, \alpha) + 1)^{-1}$. Note that $s(\hat{p}_s, \alpha)$ is the predefined return in the case of a successful bet.

2.7.2 Convexity Theory

This section presents aspects of convexity theory deemed useful for understanding the theoretical foundation of betting strategies considered in this thesis.

First, the definitions of a concave function and a convex set are given by (Lundgren et al., 2012, p. 30-31)

Definition 1. *The function $g(\boldsymbol{x})$ is a concave function on the feasible region X if for all choices of points $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)} \in X$ and $0 \leq \lambda \leq 1$ we have that $f(\lambda \boldsymbol{x}^{(1)} + (1-\lambda)\boldsymbol{x}^{(2)}) \geq \lambda f(\boldsymbol{x}^{(1)}) + (1-\lambda) f(\boldsymbol{x}^{(2)})$.*

Definition 2. *A set $X \subseteq \mathbb{R}^n$ is a convex set if for any pair of points $\boldsymbol{x^{(1)}}, \boldsymbol{x^{(2)}} \in X$ and $0 \leq \lambda \leq 1$ we have*

$$\boldsymbol{x} = \lambda \boldsymbol{x}^{(1)} + (1 - \lambda)\boldsymbol{x}^{(2)} \in X$$

If $g(\boldsymbol{x})$ satisfies a strict inequality in Definition 1, it is a strictly concave function. Next, if a function $g(\boldsymbol{x})$ is concave, then the function $f(\boldsymbol{x}) = -g(\boldsymbol{x})$ is convex (Lundgren et al., 2012, p. 30). Furthermore, the following theorem states that a linear combination of convex functions is a convex function (Lundgren et al., 2012, p. 248).

Theorem 1. *If $f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_p(\boldsymbol{x})$ are convex functions and we have $\lambda_k \geq 0, k = 1, \ldots, p$, then the function $f(\boldsymbol{x}) = \sum_{k=1}^{p} \lambda_k f_k(\boldsymbol{x})$ is convex.*

Based on the two latter results, if $f_k(\boldsymbol{x}) = -g_k(\boldsymbol{x}); k \in \{1, .., K\}$, $g_1(\boldsymbol{x}), \ldots, g_K(\boldsymbol{x})$ are concave and $f(\boldsymbol{x})$ is defined as in Theorem 1, the following holds:

$$g(\boldsymbol{x}) = \sum_{k=1}^{p} \lambda_k \big(g_k(\boldsymbol{x})\big) = \sum_{k=1}^{p} \lambda_k \big(-f_k(\boldsymbol{x})\big) = -\sum_{k=1}^{p} \lambda_k f_k(\boldsymbol{x}) = -f(\boldsymbol{x}) \text{ concave} \tag{2.36}$$

Now, the definition of a convex maximization problem is given by (Lundgren et al., 2012, p. 29)

Definition 3. *A maximization problem P*

$$\boldsymbol{max}\ g(\boldsymbol{x}) \ \ \textit{subject to} \ \ \boldsymbol{x} \in X$$

is convex if X is a convex set and $g(\boldsymbol{x})$ in concave on X.

If a problem P is convex, than each local maximum is also a global maximum. If $g(\boldsymbol{x})$ is strictly concave, then the a maximum is also unique (Lundgren et al., 2012, p. 245). Lastly, note that the following holds (Conway, 1985, p. 8)

Theorem 2. *An intersection of convex sets is itself a convex set.*

2.7.3 Portfolio Optimization and The Kelly Criterion

The task of choosing the optimal wealth allocation amongst a set of candidate securities in financial markets is usually referred to as *portfolio optimization*. This term was first discussed by (Markowitz, 1952), a paper considered to be the foundation of *modern portfolio theory (MPT)*. The core of MPT is the *Mean-variance model*, which rests on the assumption that an optimal portfolio can be constructed in such a way that the financial return is maximized for a given risk level or vice-versa. The portfolio return is here defined as a linear combination of the returns on the individual investments and the risk coincides with the covariance matrix of these returns. However, it can be proved that the optimal expected return of a portfolio of investments is not a linear combination of the proportion of wealth placed in the individual investments (Peterson, 2018).

Another approach to portfolio optimization is proposed by Kelly (1956) based on a completely different problem in information theory. This approach is, in simple terms, to allocate fractions of wealth such that these fractions maximize the expected logarithmic return on investments. In fact, this allocation also maximizes the expected utility for investors with a logarithmic utility function with respect to their wealth (Mossin, 1968; Bellman and Kalaba, 1957). An important note is that this allocation does not necessarily lie on the efficient frontier of the mean-variance model (Thorp, 1975).

Following Moffitt (2017), let $\boldsymbol{Y} = (Y_i \in \{0,1\})_{i=1}^{N}$ denote a sequence of i.i.d. random variables for event $i \in \{1,...,N\}$. Let also p be the known probability of the reference outcome $Y_i = 1$. Now, assume that a bettor successively bets a fixed proportion $f \in [0,1]$ of its initial wealth W_0 on the events, with the supplied odds $s > 0$ fixed. Note that this implies that leveraging is not allowed. Then the expected compounded wealth after wagering the same fraction on all N events is given by

$$W_N(f, \boldsymbol{Y}) = W_0(1 + fS)^{\sum_{i=1}^{N} Y_i}(1 - f)^{N - \sum_{i=1}^{N} Y_i} \tag{2.37}$$

The *asymptotic logarithmic return* of the fixed fractional betting scheme can then be defined as

$$G_f = \lim_{N \to \infty} \frac{1}{N} ln\Big(\frac{W_N(f, \boldsymbol{Y})}{W_0}\Big) \tag{2.38}$$

which, by substitution from Equation (2.37) becomes

$$G_f = \lim_{N\to\infty} \frac{1}{N} ln\Big((1+fS)^{\sum_{i=1}^{N} Y_i}(1-f)^{N-\sum_{i=1}^{N} Y_i}\Big) \tag{2.39}$$

Now, the derivation of the Kelly criterion rests on the following result (Loève, 1977):

Theorem 3 (Strong Law of Large Numbers)**.** *Suppose $X_1, \ldots, X_N$ are i.i.d random variables and that $\mathbb{E}[|X|]$ is finite. Define $\bar{X}_N = \frac{1}{N}\sum_{i=1}^{N} X_i$. Then $\bar{X}_N$ converges almost surely to $\mathbb{E}[X]$, that is*

$$P(\lim_{N\to\infty} \bar{X}_N = \mathbb{E}[X]) = 1 \tag{2.40}$$

Since the random variables in Equation (2.37) are given by $X_i = (1+fS)^{Y_i}(1-f)^{1-Y_i}$ with corresponding expected value $\mathbb{E}[|X|] = p(1+fs)+(1-p)(1-f) \in [1-f, 1+fs]$, Equation (2.39) converges almost surely to

$$G_{f_k} = ln\Big((1+fS)^{p}(1-f)^{1-p}\Big) \tag{2.41}$$

by Theorem 3, as shown by Kelly (1956). Maximization of this function with respect to f provides the following rule:

$$f^* = \begin{cases} \frac{p(S+1)-1}{S} & ; \frac{p(S+1)-1}{S} > 0 \\ 0 & ; \frac{p(S+1)-1}{S} \leq 0 \end{cases}$$

This is the Kelly criterion for the stated scenario. Note that if short positions $f < 0$ are allowed, then one would bet $\frac{1-p(S+1)}{S}$ in the case where this quantity is positive (Moffitt, 2017).

Chapter 3

Literature Review

This chapter presents the existing academic literature deemed relevant to this thesis. First, Section 3.1 considers research relevant to the prediction of the outcome of football matches, before Section 3.2 presents research considering investment strategies based on the Kelly criterion. The emphasis in both sections is placed on the relevance to in-game prediction models and the corresponding live-betting market. To conclude the chapter, Section 3.3 presents the academic contribution of this thesis with respect to the relevant research stated in the first two sections.

3.1 Relevant Prediction Models

This section presents existing research that considers the prediction of the outcome of football matches or other relevant sports events. Since existing research on in-game prediction models is limited, most of the presented work consider pre-game prediction models, which means that they only utilize the information available before the event starts. The research papers are presented categorically according to their most interesting aspect for this thesis, although multiple papers fall into several of these categories.

3.1.1 The Poisson Assumption and Generalizations

Maher (1982) makes the assumption that goals scored by each team in a football match follow independent time-homogeneous Poisson processes. He then employs a parametric regression model to estimate scoring rates as a function of offensive and defensive team strengths in the marginal Poisson distributions using a maximum likelihood approach. Since then, an abundance of research has been conducted based on this work. Dixon and Coles (1997) extend the model proposed by Maher (1982) by using a copula function to allow for dependence between the scoring processes of opposing teams. In addition, they modify the likelihood function by using a time decay factor. This is done in order to assign most relevance to observations from recent matches and is motivated by the assumption that team strengths vary over time. Another approach to account for this is proposed by Rue and Salvesen (2000). They develop a Bayesian linear dynamic model and employ a Markov Chain Monte Carlo approach for inference. Koopman and Lit (2012) also account for time-varying team strengths by developing a stochastic process for modelling the strengths, while otherwise following the same approach as Dixon and Coles (1997). Regarding the dependence between goals scored by opposing teams, Dixon and Robinson (1998) account for this by modelling a two-dimensional goal process rather than fitting a copula

function around two independent distributions. McHale and Scarf (2007) examine the hypothesis that such dependence exists by developing a model for FIFA World Cup matches based on FIFA ratings. They find that there is a statistically significant negative dependence and that the magnitude of this dependence grows as a function of the difference in the FIFA ratings. They also suggest that there exist a small positive dependence between goals scored by opposing teams in domestic football competitions such as the EPL.

In addition to Rue and Salvesen (2000), several researchers have utilized a Bayesian network to account for conditional probability. Joseph et al. (2006) develop a Bayesian network for prediction of the matches of Tottenham Hotspur in the $2005 - 2007$ seasons of the EPL. Based on this, Constantinou et al. (2013) subsequently modify the Bayesian network for prediction of all matches in the EPL season 2011-2012, reporting superior performance in the betting market. Owramipur et al. (2013) developed a model for predicting the matches of Barcelona FC during one season, using features such as weather conditions, psychological state of players and injuries. Yet another approach is taken by Schauberger and Groll (2018), who consider the use of random forests for predicting the outcome of international football matches based on data from the four FIFA World Cups between 2002 and 2014. They develop one random forest for the entire scoreline distribution and one for the 1X2 outcome, reporting decent results for both models.

With respect to choosing a model architecture, research comparing different approaches to do so are particularly interesting. A good example in this regard is the work of Goddard (2005). He develops a bivariate Poisson model for the scoreline distribution as well as an ordered probit model for the 1X2 distribution to compare their performance on the task of predicting outcomes from the latter. He found that the two approaches provide fairly similar results and that a hybrid model performs slightly better. He also suggests that information about the importance of a match for the opposing teams hold some predictive power on the outcome of football matches. Another interesting paper is that of Hvattum (2017), who compares an ordinal and a multinomial logit model. This is motivated by the fact that many researchers, including most of those presented above, choose an ordinal regression for modelling 1X2 outcomes in football matches due to the inherent ordinal structure of goals. He finds, despite this ordinal nature, that the multinomial model performs slightly better than its ordinal equivalent. He suggests that this is due to the challenge of correctly predicting draws implied by the *proportional odds assumption* inherent in ordinal models. The proportional odds assumption is that independent variables influence the logarithm of the odds of each outcome is the same manner.

The assumption that scoring processes follow time-homogeneous Poisson distributions, which is made in all the models mentioned thus far, may be overly restrictive. The extensive use of this assumption may be motivated by the fact that few other realistic options were available at their time of publishing. Based on the observation that Poisson models tend to underestimate the probability of a draw, Karlis and Ntzoufras (2003) incorporate an inflation factor to their model. Albeit not specifically intended for goal processes in football, McShane et al. (2008) derive the Weibull count distribution, as presented in Section 2.4. Boshnakov et al. (2017) subsequently develop a model for the scoreline distribution of EPL matches based on the assumption that the goal scoring processes follow this distribution and use the Frank copula to model dependence. They otherwise follow a similar approach to that of Dixon and Coles (1997). In addition, they conduct a *Chi-squared* hypothesis test for comparing the fit of the Poisson and Weibull count distributions to the goal distribution observed during the 2011/2012-2014/2015 seasons of the EPL. They find that a null hypothesis stating that the goals follow a Weibull Count distribution cannot be rejected at significance level $\alpha = 0.1$ for either average home or away goals, while an equivalent null hypothesis for the homogeneous Poisson distribution is rejected even at $\alpha = 0.005$ in both cases.

Note that all the models mentioned above construct some form of team ratings with respect to a parametric probability distribution. Another approach considered in several papers is to create a rating independent from the choice of a parametric probability distribution. Some also consider the predictive power of existing rating systems. Elo (1978) develop a rating system for assessing chess players based on exchanges of rating points between the winner and loser. Hvattum and Arntzen (2010) examine the predictive power of this rating on football matches by using it for feature selection. They then employ these features in an ordered logit regression model for prediction. Constantinou et al. (2012) propose the Performance index (Pi) rating system and suggest that their model outperforms that of Hvattum and Arntzen (2010) in the betting market, where it yields statistically significant positive returns. This rating system is constructed by utilizing a Bayesian network to incorporate information about the conditions under which a match is played, thus capturing some information not available from historical data. Silver (2009) develops the Soccer Power. In 2014, they updated the system to account for international football matches. Silver and Boice (2018) subsequently use the SPI rating in their model of expected goals in football matches. The expected goals estimates are then used as the scoring rate in two independent Poisson distributions. Their model also utilizes information about the importance of matches from the perspective of the opposing teams and market values of players from Transfermarkt.com (2018) as estimates of their ability at the beginning of each season. These market prices are determined by votes from a large number of people, thus representing common knowledge about the ability of a player. In a similar fashion to Karlis and Ntzoufras (2003), they use an inflation factor to increase the predicted proportion of draws.

The predictive power of common knowledge on the outcome of football matches is also examined by several others. Peeters (2018) utilize market prices from Transfermarkt.com (2018) to predict the outcome of football matches. He suggests that the predictions are more accurate than those based on the FIFA rating and the Elo rating. Godin et al. (2014) develop a prediction model for the 1X2 outcome of football matches by extracting and aggregating information in Twitter posts. They suggest that common knowledge can beat predictions made by bookmakers and betting experts based on the performance of their model. Schumaker et al. (2016) test a somewhat similar approach based on sentiment analysis from Twitter posts and find that it performs better than wagering on the match favourite.

Lastly, Nevo and Ritov (2012) consider the explanation power of the time of the first goal in a football match on the number of goals scored in the remainder of the match. They find that this time has a statistically significant effect in this regard, but that this effect might be both impeding and expediting. Furthermore, they suggest that scoring rates increase given that a goal is scored and that this effect is the same regardless of whether it was scored or conceded.

3.1.2 In-game Prediction Methods

Asif and McHale (2016) develop a dynamic logistic regression model for in-game prediction of one-day cricket matches. They do this by training independent logistic regression models for every inning in the match and then applying a smoothing scheme on the parameters of each of these models so that they vary continuously throughout the match and reflect time decay. Volf (2009) develop a prediction model for football matches based on two dependent random point processes, each of which is a product of a non-parametric baseline scoring rate and dynamic regression model represented by a stochastic differential equation. Feng et al. (2016) employ a Skellam process to represent real-time betting odds for EPL matches. They estimate the expected scoring rates for each team based on a matrix of market odds on all possible scorelines. They then use these rates as an estimate of the implied volatility of the match. As the match evolves they

re-estimate the expected scoring rates and thus the implied volatility measure. This provides a dynamic representation of the expected outcome of the match. Nyquist and Pettersson (2017) investigate the use of artificial neural networks for predicting the outcome of football matches. In particular, they develop a long short-term memory network for learning sequences of states present in matches by utilizing the information only available during the match. An interesting aspect of this research is that they use features to model the quality of a team on the time of prediction rather than inferring it directly from historical team performance.

3.1.3 Artificial Neural Networks and Machine Learning

Although research on applications of artificial neural networks for predictions of football matches is rather sparse, there exists relevant research where ANNs are applied in other areas.

Bunker and Thabtah (2019) present a critical analysis of the existing literature on the use of ANNs for predicting the outcome of different sports events. Based on this analysis, they and suggest a scientific framework for creating such models. Baboota and Kaur (2019) consider the prediction of football matches and place emphasis on feature engineering based on an extensive review of football prediction models. They suggest that features such as EA SPORTS^{TM} FIFA ratings and indicators of the quality of recent performances of a team have decent predictive power. They then use a gradient boosting algorithm to obtain their predictions. Ulmer and Fernandez (2014) also focus their work on feature engineering and develop several models based on machine learning methods and use a feature set consisting of match day data and recent team performances. They are not able to beat the accuracy of betting experts with any of their models. Arabzad et al. (2014) construct a rather simple ANN for prediction of football matches in the 2013-2014 season of the Iranian Pro League and suggest that the decent predictive power of their simplistic network implies that ANNs should be able to accurately predict the outcome of football matches. McCabe and Trevathan (2008) construct an ANN for general team sports predictions in a similar manner to Arabzad et al. (2014), but by using a much more complex network. They otherwise choose a team independent representation similar to Nyquist and Pettersson (2017) in order to enable modelling of team qualities across different sports and environments. They report that the network compares favourably with predictions made by betting experts in several of these environments.

There is extensive coverage of racing events among the research on sports prediction, especially of horse races and greyhound races. There are some relevant aspects of the research conducted on the former, despite its completely different nature to football as a sport. Davoodi and Khanteymoori (2010) develop a model for horse race predictions based on data collected from Aqueduct Race Track in New York across January 2010. They use an RNN to be able to hold information about the most recent runs in a given race, thus trying to learn sequences of performances by a given horse on a given day of racing. Pudaruth et al. (2015) develop an ordinary feed-forward ANN for the same task, but report low prediction performance from their model. Except for the difference in architecture, the two approaches are very similar. Thus, there seems to be some extra information to obtain from learning sequences of events. This proposition is also supported to some extent by the results mentioned in the previous paragraph.

3.2 Investment Strategies Under Fixed Conditional Returns

This section presents existing research related to strategies for generating profits in betting markets for the outcome of football matches. Although the degree of information efficiency in

these markets is in itself a heavily researched topic (Paton et al., 2006; Franck et al., 2009; Stekler et al., 2010; Croxson and Reade, 2013; Hvattum, 2013), the markets in question are assumed to offer no arbitrage opportunities during this project. Thus, the degree of information efficiency and strategies for exploiting such opportunities are not discussed further.

The emphasis is placed on the Kelly criterion based on the assumptions that bettors act according to a logarithmic utility function. Furthermore, the application of this criterion in betting markets is the topic of interest. Note that conditional returns are fixed by the odds in these markets, a property not shared by most financial markets. Due to this considerable difference in nature, research considering applications of the Kelly criterion in markets with uncertain conditional returns is omitted.

The Kelly's criterion is widely used among bettors and especially in high-frequency games since the optimal logarithmic growth property of this strategy only holds in the limit $N \to \infty$ for the number of placed bets N (Peterson, 2018). Recall that the formulas presented in Section 2.7.3 rests on the assumptions that the events corresponding to successive bets are i.i.d. binary random variables and that the bettor has perfect information about the probabilities in each game. The former assumption is very restrictive and does not guarantee for optimal betting on several mutually exclusive events such as $1X2$ outcomes of football matches. Furthermore, each side of a betting contract only hold estimates of the true probabilities of the outcomes. Thus, some adjustments to Kelly's criterion should be used for portfolio optimization in the football betting market, some of which are presented here.

Maclean et al. (2011) present a thorough examination of the existing literature on the properties of the Kelly criterion. An important property is that the Kelly criterion is an optimal myopic strategy, meaning that the strategy is constant regardless of prior and subsequent bets under the presented assumptions. Hakansson (1971) proves that this property extends to investments on dependent events given the logarithmic utility function, while Algoet and Cover (1988) show that past outcomes can be accounted for by maximizing the expected logarithmic return conditional on these outcomes. Note, however, that none of these papers considers the task of allowing multiple bets at different times on a single outcome of a single event.

Based on their review, Maclean et al. (2011) also state that the main disadvantage with the Kelly criterion is that its suggested wagers are consistently larger than implied by rational behaviour according to a logarithmic utility function for short investments horizons. The reason is that the mean is much more important for determining the optimal fractions than the variance (Kallberg and Ziemba, 1984). This is also indicated by Hsieh et al. (2018), who consider the optimal frequency for updating the Kelly fractions in the case where the sequence of games corresponds to i.i.d. random variables. They suggest that, in the absence of transaction costs, the highest possible frequency is optimal. Another issue is presented by Griffin (1984) who suggests that, since the Kelly fractions are fixed as a function of total wealth, the Kelly criterion may yield a lower return than expected. This is because the unweighted geometric return rate converges to half the arithmetic return rate.

As stated, the Kelly criterion only holds when there exists perfect knowledge of the winning probability p. This has motivated the considerable amount of research conducted on the topic of *partial Kelly* strategies, which imply shrinking the Kelly fractions f^* to γf^* where $\gamma \in (0, 1)$. MacLean et al. (1992) consider the use of these strategies for analysis of dynamic portfolio optimization in discrete time, while Thorp (2008) tests the strategies for sports betting as well as for Blackjack and in the stock market. Furthermore, Kadane (2011) shows that *half-Kelly* strategies, which simply mean choosing $\gamma = \frac{1}{2}$, do not optimize any utility function exactly, but that partial Kelly strategies approximately maximizes the constant relative risk aversion utility

function $U(f) = \frac{1-f^{1-w}}{w-1}$ by choosing $f = f^*$ and $w = \gamma$. Baker and McHale (2013) find that, although the Kelly fraction f^* maximize the expected logarithmic utility when p is known, there always exist an optimal solution for $\gamma \in (0, 1)$, while $\gamma > 1$ implies terminal ruin for the bettor. They also show that the optimal γ is a monotonically decreasing function of the variance in the estimates of p. Yet another approach to account for uncertainty in p is presented by Wu et al. (2016), who suggest to use the historical winning rate $\hat{p}_N = \frac{number\ of\ wins}{N}$ as the estimate of p after N bets in the betting sequences. They show that this approach yields similar returns to the perfect information scenario for sufficiently large N.

The extension of the Kelly criterion to multivariate portfolios is also a topic often considered in existing research. Nekrasov (2014) presents a Kelly strategy for multivariate portfolios in the stock market based on estimates of the first and second order moments of excess returns. Cao et al. (2017) follow a similar approach, but also propose a partial Kelly strategy based on volatility regulation to account for uncertainty. The performance of these approaches relies to a large extent on an accurate estimate of the correlation matrix of the assets in question. Several others consider the use of the Kelly criterion for betting on sports events with more than two mutually exclusive outcomes. O'Shaughnessy (2012) derives an adjusted Kelly criterion for betting on 1X2 outcomes of football matches on a betting exchange. This criterion accounts for taxes to be paid in the case of a winning bet and the possibility of engaging in short positions. In a similar manner, Noon (2014) propose a strategy for sports betting markets where outcomes are mutually exclusive and where both short and long positions are allowed, showing that this strategy maximizes the logarithmic utility. Chapman (2006) evaluates the use of the Kelly criterion for *spread betting*, that is, distributing bets over the range of outcomes for a continuous random variable. This topic is also considered by Fitt (2008) for the time of arrival of events in football matches, albeit using the approach suggested by Markowitz (1952) rather than the Kelly criterion. Smoczynski and Tomkins (2010) develop an algorithm based on the Kelly criterion for placing bets on the winner of horse races and proves that it asymptotically maximizes the logarithmic return rate. However, all the mentioned approaches only consider pre-game betting.

3.3 Contribution to the Existing Literature

The continuation of this report is devoted to a presentation of the data and methodology utilized and results generated to answer the research questions.

The data utilized in this research project data is to a large extent motivated by ideas and results from existing literature. Both Goddard (2005) and Silver and Boice (2018) indicate that information about the importance of football matches from the perspective of the opposing teams has some ability to explain the outcome. Furthermore, Constantinou et al. (2012) and Hvattum and Arntzen (2010) suggest that team rating systems have a large prediction power on the outcome of football matches, and the same is suggested by Baboota and Kaur (2019) for information about recent performances and the EA SPORTS^{TM} FIFA player ratings.

Regarding the generation of prediction models, the choice of ANNs as a model architecture is fixed by the initial hypotheses. Nevertheless, this choice is supported by the poor results of Ulmer and Fernandez (2014) for other machine learning architectures. The work of Hvattum (2017) inspires the use of a multinomial rather ordinal logistic regression approach in the ANNs. Among the ANN architectures encountered, the results of McCabe and Trevathan (2008) suggest that the quality of a team should be represented implicitly rather than a time series connected to its name. Furthermore, a comparison between Davoodi and Khanteymoori (2010) and Pudaruth et al. (2015) indicate that a recurrent ANN architecture should be utilized to generate in-game

predictions of the outcome of sports events. The most novel approach encountered regarding the use of ANNs for prediction of football matches is that of Nyquist and Pettersson (2017) and their use of an LSTM network. Although the former model does not perform particularly well, the contrast in methodology to that of Asif and McHale (2016) indicate that this poor performance is due to the scientific process rather than the LSTM architecture itself. This is due to the similarity in how dynamics are represented in these models. The scientific process suggested by Bunker and Thabtah (2019) is also interesting in this regard.

The contribution of this research project to the existing literature lies first and foremost in combining the presented ideas, as well as extending the application of these to the in-game prediction of football matches. Explicitly, a considerably altered LSTM architecture to that of Nyquist and Pettersson (2017) is chosen, and aspects of the scientific procedures of Asif and McHale (2016) and Bunker and Thabtah (2019) are combined to propose a proper scientific process for prediction of football matches. The LSTM architecture is compared to an architecture based on the Weibull count distribution and the Frank Copula, inspired by Boshnakov et al. (2017), with respect to the equivalent scientific procedure in order to answer RQ1. The proposition of Goddard (2005) that the prediction power of otherwise equivalent models of the scoreline and 1X2 distributions is rather similar is also evaluated. This is done by constructing a model for each of the mentioned distributions for both model architectures.

Regarding betting strategies, the approach of Smoczynski and Tomkins (2010) is chosen to evaluate the static betting performance of the generated models at distinct points in time during football matches. Based on this approach, a dynamic betting strategy is proposed as a theoretical contribution to the literature, along with a proof of its optimality under a set of stated assumptions. Both of these strategies assume perfect information about probabilities, and different scaling strategies are therefore tested based on the results of MacLean et al. (1992).

Chapter 4

Data

This chapter gives a presentation and an analysis of the data utilized in this research project. First, Section 4.1 presents the origin of the data, before Section 4.2 explains the chosen representation of matches as time intervals. A presentation of the Elo and EA SPORTS^{TM} FIFA rating systems as well as the available in-game event data can be found in Section 4.3. The chapter ends with a discussion of the descriptive statistics of the mentioned features in Section 4.4. To evaluate the choice of the Weibull count distribution over the Poisson distribution as a model for goal distributions in the EPL, a chi-square hypothesis test is conducted. The results of this test strong support the choice, as can be seen in Appendix A, along with plots indicating the same.

4.1 Origin of the Data

Most of the data used in this research is provided by Sportradar. This data includes pre-game, in-game and full-time event information from all matches during the seasons 2008/2009 - 2017/2018 of the EPL, as well as odds data for the 2016/2017 and 2017/2018 seasons of the same competition. The event data is very thorough and includes all directly observable in-game events. The odds data includes both pre-game and in-game odds for a variety of markets. These odds are generated by Sportradar's own prediction model, which relies on market information as well as a model of the true scoreline distribution in EPL. The reason for the use of market information to generate these odds is motivated by the objective of providing optimal odds to their customers.

Although Sportradar is a large corporation with a significant role in the betting markets, its data is collected mostly manually and is therefore subject to human error. For this reason, the provided data has undergone simple error processing as well as deletion of misplaced events. The total number of events deleted or changed is less than 1% and is assumed not to be of major significance for the analyses and model results. Due to Sportradar's size and reputation, no general fact-checking of the live events has been conducted apart from the mentioned error checking. The full-time scorelines have however been verified through comparison with data from Football-Data.co.uk (2019), which is an online site providing free historical odds and event data.

In addition to the data from Sportradar, the Elo and EA SPORTS^{TM} FIFA ratings are used as potential model features. Historical Elo ratings are downloaded from clubelo.com (Schiefler, 2019) and contains ratings for all the mentioned seasons of the EPL. These ratings are a modification of those proposed by Elo (1978) suitable for football due to the inclusion of a home field advantage and a goal difference dependence in the rating exchange formula. EA SPORTS^{TM}

FIFA ratings are obtained from fifaindex.com (2019) by modifying an information scraper provided by Grantham (2018). The same scraper is also utilized to acquire team line-ups from BetStudy.com (2019) required to generate combined player ratings for a given match.

4.2 Time Intervals

The analyses, figures and tables of the data throughout this thesis have a specific time format. The overall aim is to accurately predict the outcome at all times $t \in [\textit{pre-game}, \textit{full-time}]$, which implies accurate predictions of the goals scored in the time intervals $I_t = [t, \textit{full-time}]$. Thus, the relevant event occurrences are first and foremost those in I_t and the notion $time = t$, therefore, refers to these occurrences throughout the thesis. Furthermore, I_t is divided into a varying number of subsets based on the granularity required to give a good presentation of the relevant statistics. Regarding the first half stoppage time, the chosen format is, due to practical purposes, that all events occurring during this stoppage time are registered at the next time index. Thus, estimates of the performance of a prediction model at $t = 45$ are pessimistic estimates of its performance at half-time.

4.3 Description of the Data

This section presents the mentioned data in detail. Some information about a match is available prior to kick-off, while other data is only made available as the game progresses. The following discussion is structured similarly - The mentioned rating systems are presented first, followed by the in-game event data.

4.3.1 Pre-game Rating Systems

As mentioned in Section 3.1.1, Constantinou et al. (2012) suggested that the Pi rating outperforms the Elo rating. However, the former is no longer made available, and the latter is therefore chosen here. The EA SPORTS^{TM} FIFA ratings are produced by Electronic Arts Inc. for their annually released FIFA games. The formulas and algorithms behind the ratings are not publicly known, but the ratings are assumed to be sufficiently reliable, both due to the reputation of the producer and the decent predictive power suggested by Baboota and Kaur (2019). Since there are considerable difficulties in obtaining all player ratings due to transfers and name formatting, the mean of all available player ratings in a lineup is used as an estimate of the missing ratings.

For the 2008/2009 - 2010/2011 seasons of the EPL, only one EA SPORTS^{TM} FIFA rating was published along with the video game, approximately one month after the start of the season. Although far more frequent rating updates are available for more recent seasons, the time required to collect these rating is considerable. Thus, the chosen methodology is to obtain ratings for each season at three different time points - right before the first game week, after the January transfer window and in the last game week, all of which are assumed to be fixed for intermediate matches. For the 2008/2009 - 2010/2011 seasons, the ratings are dated back to the first game week based on the assumption that the expected ratings in this game week coincide with these ratings.

4.3.2 In-game Events

The data provided by Sportradar contains varying levels of detail. The 2008/2009-2010/2011 and parts of the 2011/2012 seasons of the EPL does not contain live events for goal kicks, throw-ins and free kicks, while these are included for more recent seasons. This means that there is a trade-off between a larger sample and a larger set of potential in-game features. The sample size is already somewhat limited for the purpose at hand, as there are only 380 matches in one season of the EPL. Consequently, the latter is chosen to ensure a robust scientific process. Furthermore, the mentioned features are hypothesized to have very little predictive power, implying that little information is lost by excluding them from the set of candidate features. Table 4.1 lists the available live events and the seasons for which they are included.

Event type	Available seasons	Chosen
Corner kick	08/09 - 17/18	Yes
Goal	08/09 - 17/18	Yes
Offside	08/09 - 17/18	Yes
Red card	08/09 - 17/18	Yes
Shot off goal	08/09 - 17/18	Yes
Shot on goal	08/09 - 17/18	Yes
Yellow card	08/09 - 17/18	Yes
Free kick	11/12 - 17/18	No
Goal kick	11/12 - 17/18	No
Throw-in	11/12 - 17/18	No

Table 4.1: In-game events and the seasons for which they are available.

4.4 Descriptive Statics

The descriptive statistics of the candidate features should be evaluated to allow for the generation of hypotheses about the processes to be modelled, as well as to enable a proper posterior analysis of the model performance. The following sections consider estimates of the mean, standard deviation and correlation coefficient of events from Table 4.1 in different intervals throughout the matches. Note that the data sets used for model selection and evaluation are excluded to avoid drawing biased inferences.

4.4.1 Mean and Standard Deviation

The sample mean and sample standard deviation, hereafter denoted by $\hat{\mu}$ and $\hat{\sigma}$, are unbiased estimators of the arguably most important properties of a probability distribution. Table 4.2 presents $\hat{\mu}$ and $\hat{\sigma}$ for all the chosen event types, where the interpretation of time follows that explained in Section 4.2.

	Time = 0		Time = 30		Time = 60		Time = 90	
	$\hat{\mu}$	$\hat{\sigma}$	$\hat{\mu}$	$\hat{\sigma}$	$\hat{\mu}$	$\hat{\sigma}$	$\hat{\mu}$	$\hat{\sigma}$
Home corner kick	5.692	3.205	4.009	2.583	2.080	1.761	0.270	0.562
Away corner kick	4.510	2.749	3.211	2.257	1.694	1.555	0.232	0.529
Home goal	1.550	1.314	1.139	1.119	0.613	0.796	0.085	0.287
Away goal	1.166	1.159	0.857	0.977	0.475	0.719	0.066	0.252
Home offside	1.846	1.644	1.252	1.283	0.627	0.855	0.081	0.289
Away offside	1.673	1.566	1.132	1.217	0.577	0.812	0.082	0.284
Home red card	0.060	0.245	0.053	0.229	0.032	0.183	0.008	0.089
Away red card	0.087	0.295	0.077	0.277	0.054	0.233	0.012	0.110
Home shot off goal	4.998	2.683	3.536	2.153	1.866	1.460	0.233	0.484
Away shot off goal	3.983	2.304	2.845	1.893	1.510	1.314	0.203	0.459
Home shot on goal	3.502	2.293	2.499	1.904	1.333	1.319	0.178	0.441
Away shot on goal	2.848	1.991	2.054	1.650	1.083	1.144	0.150	0.405
Home yellow card	1.445	1.189	1.227	1.109	0.733	0.869	0.123	0.358
Away yellow card	1.782	1.290	1.499	1.191	0.874	0.930	0.151	0.389

Table 4.2: Mean and standard deviation for events at different time intervals in the match.

Two interesting aspects from these results is that $\hat{\sigma}$ is lower than the corresponding $\hat{\mu}$ at $time = 0$ for all events except red cards and that the value of $\hat{\sigma}$ increases relative to $\hat{\mu}$ over time. One can also observe that both $\hat{\mu}$ and $\hat{\sigma}$ decreases as the matches progress, which is as expected. It is also worth noting that $\hat{\mu}_{RC}$ is very small relative to $\hat{\sigma}_{RC}$, indicating that there may be a large uncertainty in the importance assigned to information about red cards. As red cards are assumed to affect the outcome of football matches, this further supports the choice of a large samples size in contrast to more potential in-game features.

A considerable difference in $\hat{\mu}$ between home and away teams can be seen, indicating that a home-field advantage exists. Table 4.3 give an explicit presentation of this advantage as $\hat{Advantage} = \frac{\hat{\mu}_{Home\ goal}}{\hat{\mu}_{Away\ goal}}$, where its estimated presence throughout the match indicate that a model should be able to capture it.

Time	$\hat{Advantage}$
0	1.330
15	1.320
30	1.329
45	1.317
60	1.290
75	1.282
90	1.281

Table 4.3: Home field advantage throughout the matches as measured by goals scored.

4.4.2 Correlation Coefficient

The correlation coefficient ϕ between two random variables is a measure of their linear dependence. Despite its limitations, ϕ is easily interpreted, and can be used as a simple means of

generating initial hypotheses about the data. The estimated correlation coefficients $\hat{\phi}$ can be seen in Table 4.4 for different times t, where $\hat{\phi}$ indicate the linear dependence between the value or number of occurrences of a given event on the interval *[kick-off,t]* and the number of goals scored in *[t,full-time]*. Note that since the Elo and EA SPORTS^{TM} FIFA ratings are constant throughout the match, these take the role of the former random variable in the calculation.

	Time = 0		Time = 30		Time = 60		Time = 90	
	Home	Away	Home	Away	Home	Away	Home	Away
Home FIFA	0.270	-0.183	0.243	-0.153	0.161	-0.122	0.084	-0.059
Away FIFA	-0.194	0.256	-0.169	0.210	-0.134	0.179	-0.039	0.072
Home Elo	0.291	-0.187	0.258	-0.157	0.174	-0.117	0.077	-0.049
Away Elo	-0.207	0.260	-0.188	0.215	-0.146	0.181	-0.052	0.079
Home corner kick			0.125	-0.071	0.112	-0.052	0.047	-0.035
Away corner kick			-0.078	0.087	-0.074	0.090	-0.035	0.027
Home goal			0.042	-0.032	0.068	-0.040	0.034	-0.030
Away goal			-0.017	0.073	0.008	0.079	0.014	0.072
Home offside			0.059	-0.001	0.040	-0.032	0.012	-0.015
Away offside			-0.007	0.050	0.006	0.049	0.005	0.014
Home red card			-0.045	0.009	-0.049	0.066	-0.006	0.047
Away red card			0.023	-0.044	0.098	-0.062	0.068	-0.007
Home shot off goal			0.133	-0.042	0.124	-0.047	0.073	-0.041
Away shot off goal			-0.073	0.105	-0.076	0.123	-0.072	0.036
Home shot on goal			0.074	-0.083	0.083	-0.071	0.045	-0.037
Away shot on goal			-0.090	0.091	-0.074	0.085	-0.034	0.044
Home yellow card			-0.044	0.019	-0.004	0.032	-0.001	0.032
Away yellow card			0.009	0.002	0.015	-0.020	0.038	0.000

Table 4.4: Correlation between goals scored after a given time and the other events up to that time.

The absolute values of $\hat{\phi}$ seen here indicate a small linear dependence with the expected number of goals. However, both of the rating systems seem to have decent predictive power. Another interesting aspect from this table is that $\hat{\phi}$ decreases as the match progresses, either indicating that the goal processes become subject to a larger amount of uncertainty, or that there are considerable nonlinear dependencies between the goal processes and the presented features. Note also that if the events were sorted by their absolute value of $\hat{\phi}$, the order would change throughout the match. This implies that a predictive model should be able to utilize different sets of features during a match, as well as assign varying importance to them. Although Table 4.4 indicate a small ϕ between most in-game event types and the expected goals scored, it also indicates that the linear dependence varies throughout the match. In addition, it may be the case that some subsets of these features have a strong combined predictive power in regards to the goal processes. Thus, it is worth considering them as candidates for model selection.

A final aspect to consider is the dependence between goals scored by opposing teams. Although such dependence is often modelled using nonlinear functions, for instance copulas, ϕ may yield some information about the importance of modelling it. Table 4.5, which presents $\hat{\phi}$ between goals scored by the home and away teams in EPL, indicate that the linear dependence is small, negative and varying throughout the match. Although these values do not indicate the presence

of dependence, it should not be deemed completely absent due to the possible existence of non-linear dependencies, as indicated by Boshnakov et al. (2017), Dixon and Coles (1997) and McHale and Scarf (2007) among others. As a final note, allowing for a varying dependence during football matches seems like a necessity based on these results.

Time	Correlation
0	-0.071
15	-0.076
30	-0.063
45	-0.038
60	-0.039
75	-0.047
90	-0.023

Table 4.5: Correlation between goals scored by the home team and the away team for different time intervals.

Chapter 5

Methodology

This chapter presents the methodology used to answer the research questions. Section 5.1 and Section 5.2 together give a detailed presentation of the scientific procedure that was briefly presented in Section 1.2. This procedure is designed to ensure that the models are generated using a sound statistical procedure, as well as to enable valid comparison of their performance. Then, Section 5.4 and Section 5.5 present model architectures based on a Weibull count distribution and an LSTM network respectively, where both of these architectures are used to generate models of the scoreline and 1X2 distributions. These sections also present the corresponding estimation procedures used to obtain the final models. Section 5.6 concludes this chapter with a presentation of the investment strategies chosen for application of prediction models in the in-game football betting market, as well as the mathematical foundation behind these strategies.

5.1 Feature Engineering

The raw data available in a scientific experiment is often inappropriate for generating statistical models. Thus, the process of structuring, generating and removing information from the raw data is often regarded as an integral part of statistical model generation processes. This process is usually referred to as *feature engineering*, a term also used here. An abstract view of this phase is presented in Figure 5.1, where *feature extraction* is the process of generating new features hypothesized to be of relevance, and *feature selection* is the subsequent removal of redundant or misleading features. The methodology used for these tasks is presented in this section.

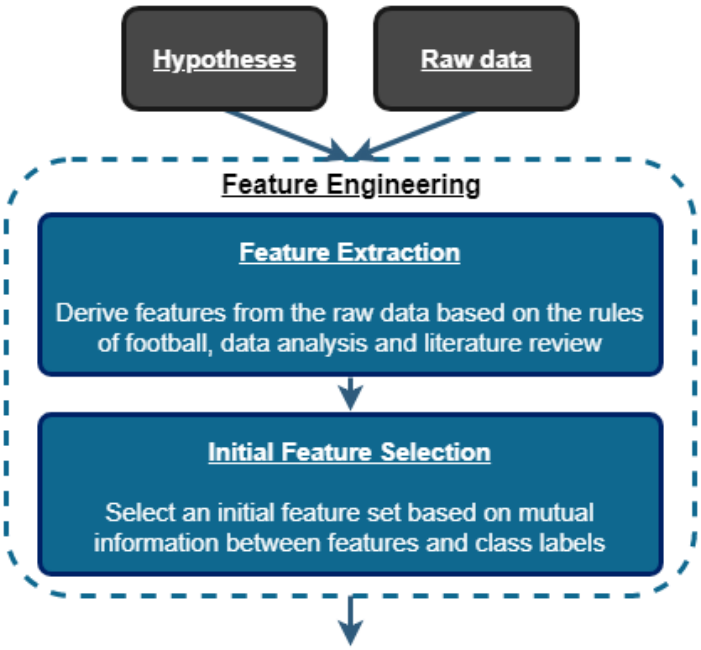


Figure 5.1: Abstract view of the feature engineering step in the scientific process.

5.1.1 Feature Extraction

Advanced model architectures such as ANNs are able to generate features during the training procedure. Nevertheless, the training task is considerably harder if no feature extraction is conducted as a preliminary step (Guyon et al., 2006). Furthermore, architectures based on the Weibull count distribution do not share this ability. Feature extraction is therefore deemed necessary in this project as a means of simplifying the training process, as well as enabling valid comparison of model performance across the mentioned architectures. The chosen approach is to generate new features based on ideas and results encountered during the literature review and data analysis, as well as from intuition about the game of football.

Features Motivated by Data Analysis and the Rules of Football

A subset of the extracted features are based on knowledge about the game of football or motivated by hypotheses generated from the results in Chapter 4. Based on the rules of football, features are extracted from a replication of the EPL table generated from the full-time results present in the available data. This feature set includes, amongst others, the number of wins, points accumulated and goals scored during the current season. It also includes the average of these quantities taken over all matches in a season and the difference in feature value between two opposing teams. Based on intuition, a home-field advantage is likely to exist in football, a hypothesis supported by the results in Section 4.4.1. Hence, indicator variables are included in the set of candidate features to separate the home team from the away team in a given match. An overview of the features discussed here is given in Table 5.1.

Feature	Description
Away indicator	Equals 1 for the away team and 0 for the home team
Home indicator	Equals 1 for the home team and 0 for the away team
Draws	Number of draws
Losses	Number of losses
Wins	Number of wins
Goals conceded	Number of goals conceded
Goals scored	Number of goals scored
Goal difference	Difference between goals scored and goals conceded
Goals conceded/match	Average number of goals conceded per match
Goals scored/match	Average number of goals scored per match
Goal difference/match	Average goal difference per match
Matches	Number of matches played
Matches left	Number of matches not yet played
Points	Number of points accumulated
Points/match	Average number of points gained per match
Position	Position in the EPL table, ranked from most to least points

Table 5.1: Features motivated by intuition, the rules of football and the data analysis.

Features Inspired by the Existing Literature

Another set of features are motivated by the work of Silver and Boice (2018), Scarf and Shi (2008) and Goddard (2005) on modelling of the importance of matches from the perspective of the opposing teams, as well as the research conducted on feature importance by Baboota and Kaur (2019).

Inspired by Baboota and Kaur (2019), a feature denoted *form* is constructed. This feature is a measure of the strength of a team relative to its most recent opponents. It is constructed by successive exchanges between teams based on the outcome of matches, much in the same manner as the Elo rating. The exchanges are conducted based on a weighting scheme, and where every team is assigned the value $form = 1$ at the beginning of each season. Explicitly, the form feature is calculated as follows:

$$Form_{winner}(w) = Form_{winner} + w * Form_{loser}$$

$$Form_{loser}(w) = (1 - w) * Form_{loser}$$

The weight w determines the amount of form exchanged in a given match. A form feature is created for all values $w \in \{0.05, 0.1, ..., 0.5\}$, where these values are chosen based on Baboota and Kaur (2019). It is worth noting that lower values of w imply a large degree of similarity between the form feature and the Elo rating, which also utilizes a low exchange rate, indicating slow updates in the performance measure.

Another feature inspired by Baboota and Kaur (2019) is denoted *streak* and is a measure of the recent performance of a team. In its most basic form, the streak is defined as the average number of points obtained during a given number l of preceding matches. Based on this idea, a set of features are created by introducing different weighting schemes in order to assign more

importance to recent performances (tw), to account for the Elo rating of the opposition (ew), or both ($ew\ tw$). As the intention is that these features should represent short-term performance, and thus complement the information inherent in the Elo rating, the maximum lag length $\bar{l} = 10$ is chosen. In addition, all the extracted streak features are initialized to 0 at the beginning of each season. Consequently, for a given $l \leq \bar{l}$, the lag is chosen to be $m = max\{l, m_s\}$, where m_s is the number of matches played by a given team in a given season. Formally, the mentioned extensions of the streak feature are defined as

$$Streak(l) = \frac{1}{3m} \sum_{i=1}^{i=m} p_i$$

$$Streak\ tw(l) = \frac{1}{3m} \sum_{i=1}^{i=m} (1 - \frac{i}{20}) p_i$$

$$Streak\ ew(l) = \frac{1}{3m} \sum_{i=1}^{i=m} \frac{e_i}{2000} p_i$$

$$Streak\ ew\ tw(l) = \frac{1}{3m} \sum_{i=1}^{i=m} \frac{e_i}{2000} (1 - \frac{i}{20}) p_i$$

where i represents the $i'th$ most recent match, and p_i and e_i denote the number of points obtained by the team in question and the Elo rating of its opponent in match i. As the maximum number of points in a given match is 3 and the Elo ratings for the best teams are close to 2000, the features are scaled by these quantities to obtain values approximately $\in [0, 1]$.

Motivated by the work of Silver and Boice (2018) and Goddard (2005), a set of features representing the importance of a match for the opposing teams is generated. Although these features may be derived from simulations and a prediction model, as proposed by Scarf and Shi (2008), the chosen approach is to construct simplifying deterministic equations based on discussions in Goddard (2005).

The rules of the EPL imply that several table positions have bearing on the next season. In addition to the fact that the three lowest positioned teams are relegated and that the best positioned team wins the league, teams finishing in the four highest positions are directly qualified for the UEFA Champions League. Furthermore, the fifth position implies qualification for the UEFA Europa League. Based on this, the given positions are used as a reference to construct four different importance features. Formally, for the fifth position, the importance feature is defined as

$$Importance\ 5(w) = |100 - |Points - Points_5||(1 - \frac{m}{38})^w I(|Points - Points_5| \leq 3m),$$

where $Points$ denotes the number of points already obtained in a given season for the team in question, $Points_5$ is the number of points obtained by the team in the fifth position, m is the number of matches left in the given season, and $w \in \{0.25, 0.5, 1, 2, 3\}$. If the team in question is the fifth highest positioned team, $Points_5$ is replaced by the number of points obtained by the closest team. In a similar manner to the form and streak features, the importance is defined as a non-negative quantity and where a high value indicates high importance. Since 100 points are the maximum ever obtained by a single team in an EPL season, this is taken as a measure of the

maximum point difference. The time weighting scheme is introduced to give higher importance to matches late in the season. Equations for *Importance 1*, *Importance 4* and *Importance 18* are defined in a similar manner, where 4 and 18 represents the limit for Champions League qualification and relegation respectively. It is also worth noting that by the given definition, the importance is 0 if the point difference exceeds the limit for obtainable points in the remaining matches.

The last set of generated features is also motivated by match importance and is based on the hypothesis that matches late in the season are considered more important from a psychological perspective. It is also generated based on another hypothesis that the outcome of matches at the beginning of a season is more random than at later stages. These features are simply constructed as indicator variables for differentiating between different phases of a season. Explicitly, given a lower and upper bound l and u, these indicators are defined as

$$Match\ num[l,u] = \begin{cases} 1;\ matches\ played \in [l,u] \\ 0;\ otherwise \end{cases}$$

Table of Features

All the subsets of features discussed here are presented in Table B.1 along with some of their properties. The interpretation of these properties is as follows: The *difference* property indicates that there exists an additional feature which represents the difference of the feature in question for opposing teams. Features marked *in-game* are only available during a match and those with the *seasonal* property are reset for each new season and depend only on the current season. The *variations* property is stated along with a set of parameters for which there exist variations of the feature. Note that all features, except for the indicators for home and away teams, are available for both teams in a match. This implies a large set of 335 candidate features.

5.1.2 Feature Selection

After completion of the feature extraction, the next step in the process is the removal of redundant or misleading features. In addition to complicating the training task, the inclusion of such features also increases the risk of overfitting. It is important to note that the term *feature selection* here refers to the task of selecting features solely based on information contained in the training sample.

The first step of this selection procedure entails dividing some of the features into categories based on their similarities. Then, for each category, only the feature with the highest mutual information score with the univariate goal distribution is retained in the feature set. This step is only conducted on the extracted features which are inspired by existing literature, as these are already grouped in subsets of very similar features. After this step is conducted, the initial feature set (IFS) is obtained. This set constitutes the candidate features considered in the model selection phase, and also includes the event types and rating systems discussed in Table 4.1 and Section 4.3.1 respectively.

The second step of the procedure is not conducted in the feature engineering phase of the scientific process, but rather in the *model selection* phase, where the number of features d is considered as a hyperparameter of the candidate models. However, this step is explained here due to its connection to the feature engineering process.

Formally, given a fixed $d \geq 1$, it can be seen as a univariate selection algorithm $FS(IFS)$, where features are chosen greedily based on their mutual information with the class labels. For every d considered in the model selection procedure, FS chooses d features that are subsequently used for training. The optimal combination of parameters is thus the combination chosen by FS given the optimal number of features d^* found in the model selection procedure. For a given time t during the matches, the feature values taken as input to FS are those representing the state of a match at t. Regarding the class labels, these are taken as the number of goals scored in the interval $[t, \textit{full-time}]$ for the scoreline models, and the full-time 1X2 outcome for the 1X2 models. Note that a univariate selection procedure may propose suboptimal feature sets, since the chosen features may hold a large amount of mutual information.

5.2 Evaluation

A sound procedure for validating, choosing and evaluating models heavily depends on the quality and relevance of the chosen metrics, as stated in Section 2.2. Thus, the process of defining metrics is a core part of any model generation procedure. The presentation of the chosen metrics and the reasoning behind these choices is structured in two sections, one for the metrics used on the scoreline distribution and one for the 1X2 distribution.

5.2.1 Metrics - Scoreline

An important reason for creating models of the full scoreline distribution in football matches is that it may be used to derive a large set of odds and thus financial portfolios in the in-game betting market. To be used for the mentioned purpose, the entire distribution should be estimated with decent precision, while the model should also understand the ordinal structure of scorelines. This is especially the case if one allows bets on multiple outcomes of a given match. However, metrics that only indicate the discrimination power of a model is also of importance, as they indicate the ability of the model to predict the actual outcome.

For the purpose at hand, RPS is considered important due to its theoretical properties as a strictly proper metric, as well as its ability to accurately measure performance on ordinal probability distributions. A requirement for the RPS to accurately measure performance on the scoreline distribution, is that its ordinal structure must be represented in a reasonable manner. This is done by using Equation (2.14) to derive the cumulative distribution. Due to the mentioned properties, the RPS is assigned the most importance in the evaluation of the scoreline models. The cross entropy has equivalent theoretical properties to the RPS except for its inability to accurately measure performance on ordinal distributions. As the cross entropy considers the pdf and RPS the cdf, the former is a stronger indicator of the discrimination ability on small samples. Hence, it is also deemed of relevance for a proper performance evaluation. Although the accuracy score is not a strictly proper metric, its presence in existing literature suggests that it should at least be given some attention during evaluation.

5.2.2 Metrics - 1X2

The quality of the estimates over the entire distribution is of most interest in general, so the same arguments regarding the cross entropy and RPS apply here. Note that although there is an inherent ordinal structure in the 1X2 distribution, defining a cumulative distribution over these classes seems unnatural. Thus, RPS is not used to evaluate model performance on this

distribution. The accuracy score of a model is arguably more important in the 1X2 scenario since it is a stronger indicator of the total model performance when there are fewer classes. As a final remark, note that the 1X2 distribution can easily be derived from the scoreline distribution. Thus, the metrics presented here are also means of measuring the performance of a scoreline model on the task of predicting the latter distribution, the topic of RQ2.

5.3 Validation and Model Selection

As mentioned in Chapter 2, the model selection procedure has considerable implications on the ability of the entire scientific process to generate good statistical models. Since validation and model selection are tightly interlinked, the two processes and the dependence between them are presented together in this section under the term *selection procedure*. Two selection procedures are presented in this section, where the first is chosen for this project due to time limitations and the second is a recommended approach for similar research.

5.3.1 Chosen Selection Procedure

The validation set approach simply implies holding out a separate validation set for model selection purposes, in addition to the designated training and test sets. The validation set, which is independent of the training sample, is used for model selection by evaluating the cross entropy for the candidate models on this sample. As discussed regarding the bias-variance trade-off in Section 2.3.1, a significant disadvantage by this approach is a large bias in the estimate of the generalization power in the case of a limited amount of available data. A more robust option would be to use a cross-validation approach, but this is deemed infeasible due to time limitations and the considerable time required to train LSTM networks. The chosen approach is therefore to use the validation set approach, but in a manner that aims to lower the bias by accepting a somewhat larger variance. Explicitly, the approach chosen here is to first designate 20% of the observation for the test set. Then, only 20% of the remaining observations are designated for the validation set, leaving most of the original sample for training.

Based on a consideration regarding reproducibility and comparison with existing literature, these data sets are constructed based on the order of observations in a time series and are not chosen at random. The explicit date ranges corresponding to these sets are presented in Table 5.2. However, such an assignment of observations does not alleviate the problem of biased estimates due to the possibility of considerable variations in the distribution of outcomes of football matches between seasons. As an example, the designated validation set used in this thesis includes the 2015/2016 season of the EPL, in which Leicester City was crowned champions at a record low 81 points despite a pre-season odds of $5000 : 1$ (Rayner and Brown, 2019) and usual big hitters Chelsea and Liverpool finished 10. and 8. respectively. This may cause a biased estimate of the predictive performance of the candidate models.

Data set	Start date	End date
Training set	16/08/2008	20/12/2014
Validation set	21/12/2014	15/05/2016
Test set	14/08/2016	13/05/2018

Table 5.2: Date ranges for the training, validation and test sets.

The constructed validation set is utilized for model selection by performing a grid search for selecting the hyperparameters of a model. Recall from Section 2.3.2 that the time complexity of an exhaustive grid search is $O(2^P)$, and that greedy variations are often chosen due to feasibility. A greedy approach is also taken here.

In explicit terms, the chosen approach consists of two steps, in which the first is to perform a grid search on model-specific hyperparameters and every fifth value of d, the number of features. For every proposed value, the d features are chosen as described in Section 5.1.2. The chosen model specific hyperparameters from this search are then taken as fixed and a second search is conducted for $d \in \{d^* - 5, d^* - 4, \ldots, d^* + 5\}$, where d^* is the chosen value from the first search. Note that this procedure is only guaranteed to find the optimal number of features in the case where the loss function is convex as a function d. This is not necessarily the case and this approach is therefore not recommended in general. Also note that this procedure is very time-consuming, even though it was chosen due to time limitations. As a consequence, one grid search is performed for pre-game predictions and another is performed at $time = 45$ for the in-game models based on the assumption that, although a different set of features may be optimal at different times during a match, the number of features required to obtain sufficient estimates is constant.

5.3.2 Recommended Selection Procedure

An entirely different selection procedure is planned and implemented for the project. However, the estimated running time based on initial testing suggested that the approach was infeasible within the time frame. Nevertheless, it is presented here as a suggestion for further research as well as an integral part of the suggested scientific framework.

Explicitly, the suggested approach is to perform a single grid search based on k-fold cross validation, where every proposed combination of hyperparameters is tested in the search. Inside this loop, an algorithm should perform feature selection for every proposed d based on multivariate mutual information rather than the chosen univariate selection procedure.

By choosing this approach one alleviates the problem of potentially heavily biased estimates of the generalization ability, as well as reduces the risk of choosing suboptimal combinations of hyperparameters and features. This is especially the case for in-game predictions of football matches, as reliable in-game data is hard to obtain. In short, the authors deem this to be a more robust approach, and it should be considered when the research does not rely on the generation of several prediction models, as is implied by the research questions in this project.

5.4 Architecture 1: Weibull Count Distribution

This section presents an architecture based on the Weibull count distribution (WCD) and the Frank copula. This model architecture is based on previous research conducted by the authors. The emphasis in the presentation is first placed on the architecture, before the chosen estimation procedure for obtaining the two models based on this architecture is discussed in detail. To conclude this section, a summary of the entire model generation process is given.

5.4.1 Architecture

The pdf of the WCD can incorporate a regression model for its hazard function. Explicitly, this can be done by assuming that a linear combination of some features is the logarithm of the scale parameter λ. As proposed by McShane et al. (2008), the hazard function of the WCD can then be represented as

$$h(t) = \lambda c t^{c-1} = e^{\boldsymbol{x_i}^T \boldsymbol{\theta}} c t^{c-1} \tag{5.1}$$

where $\boldsymbol{x_i}$ and $\boldsymbol{\theta}$ follow the conventional notation and $\boldsymbol{x_i}$ includes a bias feature $b_i = 1$ representing the intercept in the regression.

The procedure used for generating estimates of the scoreline distribution is therefore to calculate a univariate WCD for each team in a match and then combine these using the Frank copula. A copula is chosen to allow for non-linear dependence between the goal distributions based to the discussion in Section 4.4.2. The use of the Frank copula is motivated by the work of Boshnakov et al. (2017) and the previous work of the authors, as well as the properties of Archimedean copulas. Parameter estimation is performed by minimizing the cross entropy, which coincides with the maximum likelihood approach. This procedure is conducted for all times $t \in \{0, 5, \ldots, 90\}$, where the parameter estimates obtained at t are valid for all times $t^* \leq t$. The described approach is used to create two distinct models, where the only difference between them is the distribution for which the cross entropy is calculated. The first model is obtained by minimizing the cross entropy of the scoreline distribution, and the other of the 1X2-distribution. These models are hereafter referred to as WCD_{score} and WCD_{1X2} respectively.

Formally, let $Y_{it} = Y_i - g_{it}$ be the random variable representing the difference between the goals scored by a given team at full-time, Y_i, and at time t, g_{it}, in a given match i. Also, let h denote the home team and a the away team. The scoreline outcomes for the remainder of the match and the full-time 1X2 outcomes are then represented by

$$S_{it} = (Y_{it}^h, Y_{it}^a)$$

and

$$W_{it} = \big[I(Y_i^h > Y_i^a), I(Y_i^h = Y_i^a), I(Y_i^h < Y_i^a)\big].$$

respectively. Furthermore, let $F_{Y_{it}}(k|\boldsymbol{x_{it}}, \boldsymbol{\theta}, c)$ represent the cumulative probability of class k for Y_{it} according to the WCD. Let also $C(\,\cdot\,,\,\cdot\,)$ denote the Frank copula, and $\boldsymbol{\theta}$ and $\boldsymbol{c}$ be parameters subject to estimation. The bivariate cumulative scoreline distribution for the remainder of the match is then given by

$$F_{S_{it}}(\Omega_{S_{it}}|\boldsymbol{x_{it}}, \boldsymbol{\theta}, \boldsymbol{c}) = \Big(C\big(F_{Y_{it}^h}(m|\boldsymbol{x_{it}}, \boldsymbol{\theta}, c^h), F_{Y_{it}^a}(n|\boldsymbol{x_{it}}, \boldsymbol{\theta}, c^a)\big)\Big)_{(m,n)\in\Omega_{S_{it}}} \tag{5.2}$$

where $\Omega_{S_{it}} = \Omega_{Y_{it}^h} \times \Omega_{Y_{it}^a}$ and the copula parameter κ is included in $\boldsymbol{\theta}$. The corresponding bivariate pdf, defined as $f_{S_{it}}(\Omega_{S_{it}}|\boldsymbol{x_{it}}, \boldsymbol{\theta}, \boldsymbol{c})$, is constructed by utilizing Equation (2.14). Obtaining the full-time 1X2 distribution given information up until time t entails summing over all scoreline

probabilities that result in a given 1X2 outcome. This yields the following equation

$$f_{W_{it}}(\Omega_{W_{it}}|\Omega_{S_{it}}, \boldsymbol{g_{it}}, \boldsymbol{x_{it}}, \boldsymbol{\theta}, \boldsymbol{c}) = \left(\sum_{(m,n)\in\Omega_{S_{it}}} I\big(k = WI(m,n,\boldsymbol{g_{it}})\big) f_{S_{it}}(\Omega_{S_{it}}|\boldsymbol{x_{it}}, \boldsymbol{\theta}, \boldsymbol{c}) \right)_{k\in\Omega_{W_{it}}} \tag{5.3}$$

where the 1X2 indicator WI is given by

$$WI(m,n,\boldsymbol{g}) = [I(m+g^h > n+g^a), I(m+g^h = n+g^a), I(m+g^h < n+g^a)].$$

5.4.2 Estimation Procedure

Following the above definitions, the estimation procedure for WCD_{score} and WCD_{1X2} is presented here. These models both represent estimates of the bivariate scoreline distributions at all times $t \in \{0, 5, \ldots, 90\}$, but the procedures for obtaining estimates of $\boldsymbol{\theta}$ and $\boldsymbol{c}$ differ, as stated above. The estimates of these parameters in WCD_{score}, denoted $\hat{\boldsymbol{\theta}}_s$ and $\hat{\boldsymbol{c}}_s$, are obtained by minimizing the cross entropy given by

$$CE(\boldsymbol{\theta_s}, \boldsymbol{c_s}|f_{S_t}, \boldsymbol{X_t}, \boldsymbol{s_t}) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k\in\Omega_{S_{it}}} I(s_{it}=k) ln\big(f_{S_{it}}(k|\boldsymbol{x_{it}}, \boldsymbol{\theta_s}, \boldsymbol{c_s})\big) \tag{5.4}$$

where $\boldsymbol{s_t} = (y^h_{it}, y^a_{it})_{i\in\{0,1,\ldots,N\}}$ and y_{it} is the true value of the corresponding random variable. Similarly, the parameters for WCD_{1X2} are obtained by minimizing

$$CE(\boldsymbol{\theta_w}, \boldsymbol{c_w}|f_{W_t}, \boldsymbol{X_t}, \boldsymbol{w_t}, \Omega_{S_{it}}, \boldsymbol{g_{it}}) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k\in\Omega_{W_{it}}} I(w_{it}=k) ln\big(f_{W_{it}}(k|\Omega_{S_{it}}, \boldsymbol{g_{it}}, \boldsymbol{x_{it}}, \boldsymbol{\theta_w}, \boldsymbol{c_w})\big) \tag{5.5}$$

where $\boldsymbol{w_t} = ([I(y^h_i > y^a_i), I(y^h_i = y^a_i), I(y^h_i < y^a_i)])_{i\in\{0,1,\ldots,N\}}$. The optimal solution for the minimiziation problems corresponding to Equation (5.4) and Equation (5.5) are obtained using the *Constrained optimization by linear approximation (COBYLA)* algorithm (Jones et al., 2001), which is not discussed further.

A simple graphic representation of the entire estimation procedure can be seen in Figure 5.2 for WCD_{score}, and the necessary extension for obtaining WCD_{1X2} is presented in Figure 5.3.

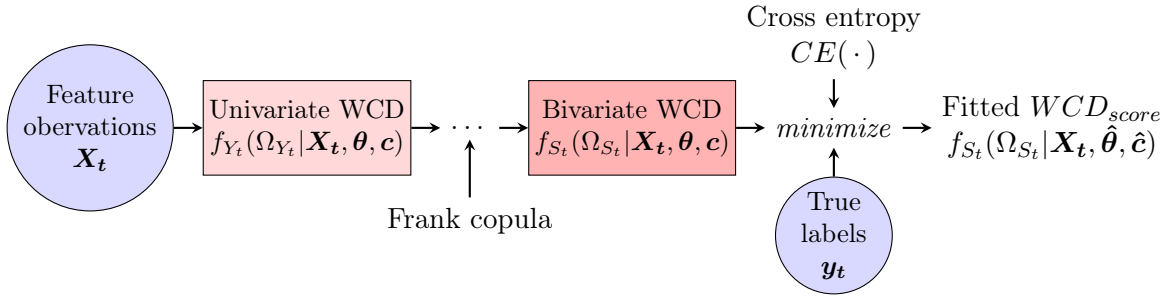


Figure 5.2: Illustration of the estimation procedure for WCD_{score}.

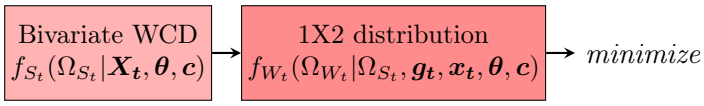


Figure 5.3: Illustration of the extension required by WCD_{1X2}.

5.4.3 Summary - Model Generation Procedure

Now that the scientific framework and the individual components of the model generation process have been extensively discussed, the time is due to present the entire process. A natural way to do this is by presenting it in a step by step manner, as can be seen in Table 5.3. Note that the term WCD_{model} is to be replaced by either WCD_{score} or WCD_{1X2}, and that $\hat{f}_{S_t}$ represents the trained models. Furthermore, the scaling of the input data is conducted to ease the computational burden placed on the optimization algorithm by ensuring that all feature values lie in the interval $[-1, 1]$.

Model Generation Process WCD

1: Initial Selection and Scaling

All features $\xrightarrow{feature\ selection}$ Initial feature set *(IFS)*

$\boldsymbol{X}[IFS] \rightarrow \boldsymbol{X}$

$\boldsymbol{X} \xrightarrow{split} \boldsymbol{X}_{train}, \boldsymbol{X}_{valid}, \boldsymbol{X}_{test}$

$\boldsymbol{y} \xrightarrow{split} \boldsymbol{y}_{train}, \boldsymbol{y}_{valid}, \boldsymbol{y}_{test}$

$\boldsymbol{\rho} = (\max\{|\boldsymbol{X}_{train,f}|\})_{f \in features}$

$\boldsymbol{X}_{train}, \boldsymbol{X}_{valid}, \boldsymbol{X}_{test} \xrightarrow{Scaling: \frac{\boldsymbol{X}}{\boldsymbol{\rho}}} \boldsymbol{X}_{train}, \boldsymbol{X}_{valid}, \boldsymbol{X}_{test}$

2: Model Selection

hyperparameters$=[d_{pre}, d_{in}]$

$d^*_{pre} \xleftarrow{Grid\ search} WCD_{model}, \boldsymbol{X}_{train}, \boldsymbol{X}_{valid}, \boldsymbol{y}_{train}, \boldsymbol{y}_{valid}, \{d_{pre}\}$

$d^*_{in} \xleftarrow{Grid\ search} WCD_{model}, \boldsymbol{X}_{train}, \boldsymbol{X}_{valid}, \boldsymbol{y}_{train}, \boldsymbol{y}_{valid}, \{d_{in}\}, d^*_{pre}$

3: Estimation

Select d^*_{pre} best pre-game features $\rightarrow \boldsymbol{X}_0$

$\hat{f}_{S_0} \xleftarrow{t=0}$ Estimation procedure from Figure 5.2 and Figure 5.3

for all time steps $t \in \{5, 10, \ldots, 90\}$ **do**:

  Select d^*_{in} best features at time $t \rightarrow \boldsymbol{X}_t$

  $\hat{f}_{S_t} \leftarrow$ Estimation procedure from Figure 5.2 and Figure 5.3

end for

Table 5.3: Model generation process for the WCD models.

5.5 Architecture 2: Long Short-Term Memory Network

The emphasis is now moved to the LSTM architecture. As in the previous sections, the focus is first placed on the architecture itself along with the corresponding hyperparameters. Then, the chosen estimation procedure and three different models are presented. The section ends with a

summary of the entire model generation process with respect to the special requirements of this architecture.

5.5.1 Architecture and Hyperparameters

This section introduces the chosen LSTM architecture and the corresponding hyperparameters. The core component of this architecture is a simple form of LSTM networks, as seen in Section 2.6.2. However, a series of alterations are made to this architecture to enhance the model generation procedure. These alterations impose a new set of hyperparameters on the models, in addition to the ones inherent in ANNs. The entire set of hyperparameters is presented below, before the focus shifts to the model architecture.

A subset of the hyperparameters determines the size of the network. As discussed in Section 2.6.3, the convention for choosing this size is to ensure that the network is *large enough* and then use a regularizer to avoid overfitting. This convention is also chosen here, where the regularizer is the elastic net and the parameter values for the norm penalties in Equation (2.6) are taken as hyperparameters. Regarding the term *large enough*, initial testing suggested that three hidden layers are sufficient to not restrict the training procedure. This is supported by the Universal approximation theorem. The dimensions of the LSTM cells M_l in each layer l must also be sufficiently large. These are chosen as $M_l > \max\{\textit{Number of features}, \textit{Number of classes}\}$; $l \in \{1, 2, 3\}$ to ensure that the *degrees of freedom* is not restricted.

The output layer and all hidden layers in the network require activation functions. Prior to choosing these for the task of modelling football matches, one must decide whether to create an ordinal or a nominal model. Motivated by the work of Hvattum (2017), the softmax function is used as the output activation function. The network also requires an optimization algorithm that utilizes the output from the network to ensure proper training. Initial testing suggested that the ADAM optimizer is the one indicating the best ability to converge to an optimal solution among the options available in *Keras* (Chollet et al., 2015), the environment used for training the LSTM networks. Consequently, it is the chosen optimization algorithm used for this architecture.

Furthermore, two additional types of layers are introduced to provide specific benefits. Recall from Section 2.6.3 that dropout layers can prevent overfitting by randomly ignoring signals during training. These layers are therefore included after each LSTM layer in the network, where the drop probability p is chosen as a hyperparameter. In addition, the batch normalization approach and its layers are presented as an aid against internal covariate shift. These layers are included according to the approach suggested by Cooijmans et al. (2016). The choice of the hyperbolic tangent and sigmoid functions as activation functions in all hidden layers follows from this approach.

Another aspect to consider is the problem of defining the optimal number of epochs e. This parameter must be chosen such that convergence is ensured without a large risk of overfitting. A simple and time-consuming approach is to let e be chosen as a hyperparameter in a grid search. The chosen approach here as a means of decreasing the risk of overfitting is to define a set of *early stopping* criteria based on the cross entropy of the model on the validation set. This halts the training process when the metric does not improve by a sufficient amount within a given number of epochs. Thus, the only requirement of e is that it is large enough for convergence.

In addition to the presented alterations, there is a set of other hyperparameters to be chosen, such as the initial values of the bias vectors $\boldsymbol{b}$. However, these are chosen to be their default values in *Keras* and no further elaboration is made regarding these parameters, except for the fact that the default $\boldsymbol{b} = \mathbf{1}$ follows the convention for regression tasks. Recall that the optimal number of

features, d_{pre} and d_{in}, are also taken as hyperparameters, and are chosen by the general selection procedure presented in Section 5.3.1. A summary of all the mentioned hyperparameters and their values can be seen in Table 5.4.

Hyperparameter	Value
Hidden activation function	Hyperbolic tangent
Output activation function	Softmax
Number of epochs	1000
Number of hidden layers	3
Number of dimension in a cell	$M_l > \max\{\textit{Number of features}, \textit{Number of classes}\}$; $\textit{layer } l \in \{1, 2, 3\}$
Elastic net weights; λ_1, λ_2	Found by grid search
Dropout rate; p	Found by grid search
Number of in-game features; d_{in}	Found by grid search
Number of pre-game features; d_{pre}	Found by grid search
Learning rate	Keras default value
Initial bias vector	Keras default value
Other ANN parameters	Keras default values

Table 5.4: Hyperparameters for the LSTM models.

A visualization of the described architecture can be seen in Figure 5.4, where $\boldsymbol{W}$ and $\boldsymbol{U}$ are parameter matrices, $\boldsymbol{x}$ is the input data and $\boldsymbol{f}$ is the output. In accordance with the conventional notation, $\boldsymbol{W}$ and $\boldsymbol{U}$ are subsets of $\boldsymbol{\theta}$. Furthermore, $D(\cdot)$ and $BN(\cdot)$ represent dropout and batch normalization layers respectively. Note that the information propagation differs somewhat from that represented in the figure. The cell outputs $\boldsymbol{h}$ are propagated through the batch normalization layer, before one copy is forwarded through the recurrent connections and another through a dropout layer and taken as input in the LSTM layer above. The batch normalization parameters, dropout rate and the regularization operations are omitted for simplicity. Also note that this is a many-to-one approach, in contrast to that presented in Figure 2.5. The reasoning for this choice is based on the results presented by Nyquist and Pettersson (2017), which indicates that many-to-many networks do not perform well over the entire course of football matches.

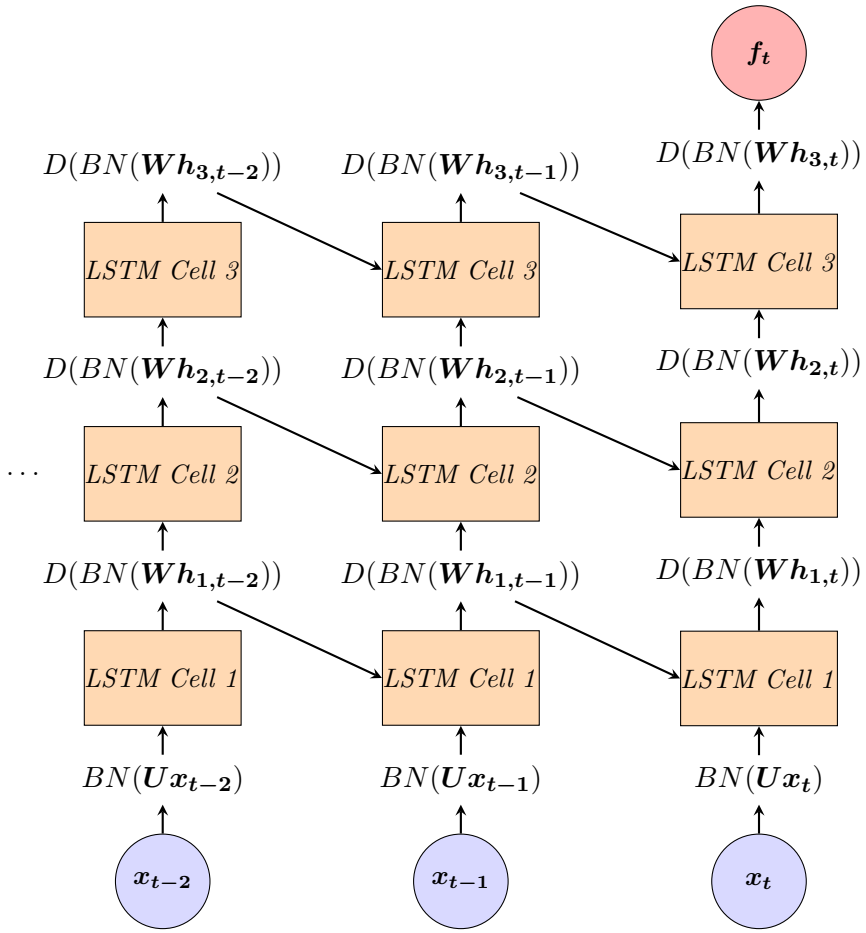


Figure 5.4: The architecture of the LSTM networks.

5.5.2 Estimation Procedure

The presented architecture is used as a foundation for three different models, all of which are generated by minimization of the cross entropy. In a similar manner to the WCD models, versions of all three models are estimated at all times $t \in \{0, 5, \ldots, 90\}$ during the match. This is due to the chosen many-to-one approach, which only outputs predictions at one time t in the sequence. Based on this, all three models and their variations are presented below.

Two of the models are entirely based on the LSTM architecture and only differ in the probability distribution they are meant to estimate. The first model, hereafter referred to as $LSTM_{score}$, is an estimate of the bivariate scoreline distribution, for which the set of possible outcomes are $\Omega_S = \{0, \ldots, 9\} \times \{0, \ldots, 9\}$. The second model, referred to as $LSTM_{1X2}$, is an estimate of the 1X2 distribution. This model is consequently obtained by minimizing the cross entropy over this distribution rather than over Ω_S.

The estimation procedure for obtaining both of these models is visualized in Figure 5.5, where $\boldsymbol{\theta}$ represents all parameters to be trained and Z_t is the distribution to be estimated. The class labels for both distributions are defined equivalently to those used for the WCD architecture.

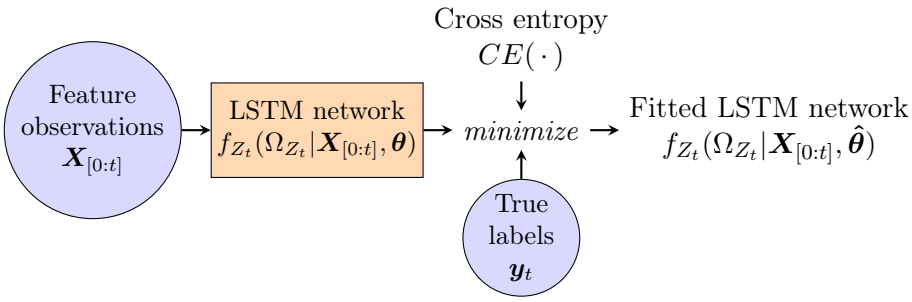


Figure 5.5: Visualization of the estimation procedure for LSTM models.

The third model, $LSTM_{copula}$, is also an estimate the scoreline distribution, but this estimate is obtained by a two-stage process. First, an LSTM network is trained on the univariate goal distribution, before the estimated probability distributions for opposing teams are combined and a Frank copula is fitted to these distributions. The former stage is equivalent to that used to train $LSTM_{score}$ except for the difference in distribution, while the latter stage is similar to that used to fit Frank copula for WCD_{score}. As for the previous two models, the estimation procedure is visualized in Figure 5.6, where all notation follows the previous definitions.

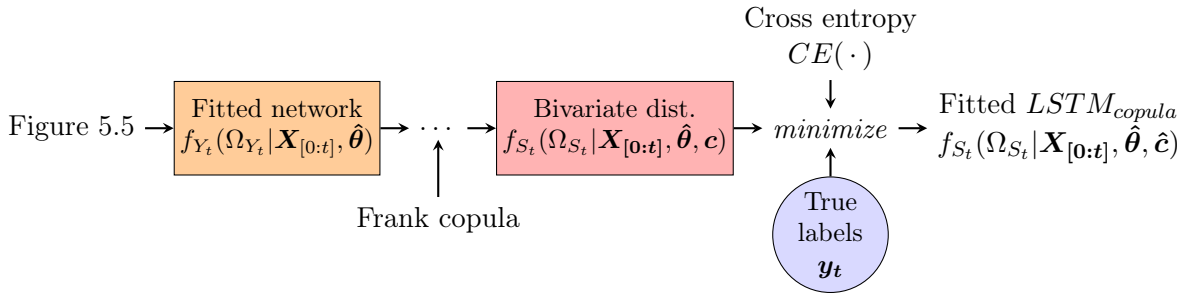


Figure 5.6: Visualization of the estimation procedure for $LSTM_{copula}$.

5.5.3 Summary - Model Generation Procedure

In a similar fashion to the Weibull count architecture, the entire model generation process is presented in Table 5.5, where Z_t is replaced by the respective probability distributions and $\hat{f}_{Z_t}$ and $\hat{f}_{S_t}$ represents the trained models. Note that the input data is standardized for numerical stability, as discussed in Section 2.6.3. Another important note is that the selection procedure used to construct the $LSTM_{copula}$ is entirely independent of Frank Copula. This approach is chosen due to practical purposes, but should ideally be conducted based on the loss function over the entire scoreline distribution.

Model Generation Process LSTM

1: Initial Selection and Scaling

All features $\xrightarrow{\textit{feature selection}}$ Initial feature set *(IFS)*

$\boldsymbol{X}[IFS] \rightarrow \boldsymbol{X}$

$\boldsymbol{X} \xrightarrow{split} \boldsymbol{X}_{train}, \boldsymbol{X}_{valid}, \boldsymbol{X}_{test}$

$\boldsymbol{y} \xrightarrow{split} \boldsymbol{y}_{train}, \boldsymbol{y}_{valid}, \boldsymbol{y}_{test}$

$\boldsymbol{\mu} = \big(\text{mean}(\boldsymbol{X}_{train,f})\big)_{f \in features}$

$\boldsymbol{\sigma} = \big(\text{standard deviation}(\boldsymbol{X}_{train,f})\big)_{f \in features}$

$\boldsymbol{X}_{train}, \boldsymbol{X}_{valid}, \boldsymbol{X}_{test} \xrightarrow{Standardization: \frac{\boldsymbol{X}-\boldsymbol{\mu}}{\boldsymbol{\sigma}}} \boldsymbol{X}_{train}, \boldsymbol{X}_{valid}, \boldsymbol{X}_{test}$

2: Model Selection

hyperparameters := $\boldsymbol{hp} = [\lambda_1, \lambda_2, p, d_{pre}, d_{in}]$

$\lambda_1^*, \lambda_2^*, p^*, d_{pre}^* \xleftarrow{Grid\ search} LSTM_{Z_t}, \boldsymbol{X}_{train}, \boldsymbol{X}_{valid}, \boldsymbol{y}_{train}, \boldsymbol{y}_{valid}, \boldsymbol{hp}$

$d_{in}^* \xleftarrow{Grid\ search} LSTM_{Z_t}, \boldsymbol{X}_{train}, \boldsymbol{X}_{valid}, \boldsymbol{y}_{train}, \boldsymbol{y}_{valid}, k_{in}, \lambda_1^*, \lambda_2^*, p^*, d_{pre}^*$

3a: Estimation $LSTM_{score}$ and $LSTM_{1X2}$

Select d_{pre}^* best pre-game features $\rightarrow \boldsymbol{X}_0$

$\hat{f}_{Z_0} \xleftarrow{t=0}$ Estimation procedure from Figure 5.5

for all time steps $t \in \{5, 10, \ldots, 90\}$ **do**:

  Select d_{in}^* best features at time $t \rightarrow \boldsymbol{X}_t$

  $\hat{f}_{Z_t} \leftarrow$ Estimation procedure from Figure 5.5

end for

3b: Estimation $LSTM_{copula}$

Select d_{pre}^* best pre-game features $\rightarrow \boldsymbol{X}_0$

$\hat{f}_{Y_0} \xleftarrow{t=0}$ Estimation procedure from Figure 5.5

$\hat{f}_{S_0} \xleftarrow{t=0}$ Estimation procedure from Figure 5.6

for all time steps $t \in \{5, 10, \ldots, 90\}$ **do**:

  Select d_{in}^* best features at time $t \rightarrow \boldsymbol{X}_t$

  $\hat{f}_{Y_{t^*}} \leftarrow$ Estimation procedure from Figure 5.5

  $\hat{f}_{S_{t^*}} \leftarrow$ Estimation procedure from Figure 5.6

end for

Table 5.5: Model generation process for the LSTM models.

5.6 Betting Strategies

This section presents two betting strategies based on the Kelly criterion for use in the football betting market. First, the strategy proposed by Smoczynski and Tomkins (2010) is discussed in detail in order to introduce some ideas and mathematical notation. Then, a strategy for in-game betting on mutually exclusive outcomes of fixed odds games is proposed along with its mathematical foundation. The former strategy is used as a benchmark to the latter on in-game betting, as well as a means of testing the betting performance of the predictive models in a static scenario. The term static here refers to the fact that bets are only allowed to be placed at a fixed point in time during the lifetime of the odds. The section ends with a presentation of the

methodology used to apply these strategies based on the probabilities from the WCD and LSTM models, as well as a means of accounting for imperfect information about the probabilities.

First, some assumptions are required. A fundamental requirement for the Kelly criterion to be optimal for a bettor is that the bettor acts rationally and according to a logarithmic utility function with respect to its wealth ($A1$). Furthermore, the Kelly criterion only maximizes the logarithmic growth rate asymptotically. Thus, assume in the following that for any given match m, a bettor encounters a sufficiently large amount of matches N_m with i.i.d. outcomes to that of m ($A2$). In addition, assume that this holds for every interval $I_t = [t, full - time]$ for all t during any match ($A3$). Assume further that the bettor can only bet on mutually exclusive outcomes in every match ($A4$), that there are no transaction costs ($A5$), and that there exists no lower limit on bet sizes ($A6$). For now, also suppose that the bettor has perfect information regarding the probabilities ($A7$).

5.6.1 Mutually Exclusive Static Kelly

Smoczynski and Tomkins (2010) proposed an algorithm for optimal wealth allocation in a static betting scenario when the outcomes are mutually exclusive and no events occur simultaneously. This approach is presented in detail here.

Formally, take the perspective of a bettor considering the optimal wealth allocation on a given match m. The m notation is omitted in the following for simplicity. Then, by otherwise following the notation in Section 2.7.3, let Ω_Y denote the set of possible outcomes for i.i.d. $Y_i \,; i \in \{1, .., N\}$, and let $p_k \in (0, 1)$ be the known probability of outcome $k \in \Omega_Y$. Now, assume that a bettor successively makes a wager on a subset of the supplied odds $\boldsymbol{s} = \{s_k \in (0, \infty)\}_{k \in \Omega_Y}$, which is fixed over all these events, by placing a fixed proportion $\boldsymbol{f} = (f_k)_{k\Omega_Y}$ of their initial wealth W_0. Furthermore, assume that no leveraging ($A8$) nor short positions ($A9$) are allowed. The two latter assumptions can be formally stated as $\sum_{k \in \Omega_Y} f_k < 1$ and $f_k \in [0, 1) \,; k \in \Omega_Y$ respectively. Then, the combined expected compounded wealth of the bettor after wagering on N matches is given by

$$W_N(\boldsymbol{f}, \boldsymbol{Y}) = W_0 \sum_{k \in \Omega_Y} \big(1 + f_k(s_k + 1) - \sum_{j \in \Omega_Y} f_j\big)^{\sum_{i=1}^{N} I(Y_i = k)} \tag{5.6}$$

This follows from the fact that the bettor pays the fraction $\sum_{j \in \Omega_Y} f_j$ regardless of the outcome of Y_i, and receives the payment $f_k(s_k + 1)$ if $y_i = k$. Now, this yields an asymptotic logarithmic growth rate

$$G_{\boldsymbol{f}} = \lim_{N \to \infty} \frac{1}{N} ln\Big(\sum_{k \in \Omega_Y} \big(1 + f_k(s_k + 1) - \sum_{j \in \Omega_Y} f_j\big)^{\sum_{i=1}^{N} I(Y_i = k)} \Big) \tag{5.7}$$

In a similar manner to the binary equivalent in Equation (2.39), this converges almost surely to

$$G_{\boldsymbol{f}} = \sum_{k \in \Omega_Y} ln\Big((1 + f_k(s_k + 1) - \sum_{j \in \Omega_Y} f_j)^{p_k} \Big) = \sum_{k \in \Omega_Y} p_k ln\Big(1 + f_k(s_k + 1) - \sum_{j \in \Omega_Y} f_j\Big) \tag{5.8}$$

by the strong law of large numbers. The corresponding optimization problem can be formally stated as

$$
\begin{aligned}
(P)\ \underset{\boldsymbol{f}}{\text{maximize}} \quad & \mathrm{G}(\boldsymbol{f}) \\
\text{subject to} \quad & f_k \geq 0;\ k \in \Omega_y && (1) \\
& \sum_{k \in \Omega_Y} f_k \leq 1, && (2) \\
& 1 + f_k(s_k + 1) - \sum_{j \in \Omega_Y} f_j > 0\ ; k \in \Omega_Y && (3)
\end{aligned}
$$

Smoczynski and Tomkins (2010) show that P is a convex optimization problem. Then, they prove that Algorithm 1 maximizes Equation (5.8).

Algorithm 1 - $MutexKelly(\boldsymbol{s}, \hat{\boldsymbol{p}})$

for all outcomes k $\in \Omega_Y$ **do**
Calculate expected revenue: $\hat{\mu}_k = \hat{p}_k(1 + s_k)$
end for
Sort the outcomes k based on $\hat{\mu}_k$ *in non-increasing order*

Let $S = \{\}, R(S) = 1, k^* = argmax_k\{\hat{\mu}_k\}$
while $\hat{\mu}_{k^*} > R(S)$ **do**
$S = S \bigcup \{k^*\}$
$\hat{p}_{sum} = \sum_{k \in S} \hat{p}_k; s_{sum} = \frac{1}{\sum_{k \in S} S_k}$
$R(S) = \frac{1 - \hat{p}_{sum}}{1 - s_{sum}}$
$k^* = argmax_{\Omega_Y \setminus S}\{\hat{\mu}_k\}$
end while

$S_{optimal} = S$
$f_k^* = \frac{\hat{\mu}_k - R(S_{optimal})}{S_k}, k \in S_{optimal}$
$f_k^* = 0; k \notin S_{optimal}$
$\boldsymbol{f}^* = (f_k^*)_{k \in \Omega_Y}$
return $\boldsymbol{f}^*$

5.6.2 Extending the Kelly Criterion

This section presents a strategy based on the ideas presented above. The strategy is applicable for the case where subsequent bets can be placed on mutually exclusive outcomes and the return is fixed conditional upon the outcome. The Kelly criterion is an optimal myopic betting strategy given A1 when at most one bet is allowed for a single outcome of a given event. However, for in-game betting, a bettor is not subject to this restriction. Thus, one should account for the expected return of the previously placed bets in the investment decisions at any given time during the lifetime of the odds. Although the investment decision should ideally incorporate information about the expected path of probabilities until maturity, it is outside the scope of this thesis. Thus, the bettor is only allowed to take into account the amount already wagered when making this decision ($A10$).

Now, assume that assumptions A1-A10 hold, and consider a given time t in a hypothetical sequence of N identical matches. By A2 and A3, the optimization problem is equal for all these N matches. Thus, the sequence of optimal bets made before time t in all matches are also equal

and, by applying A2 once more, the optimization problem at t is similar in all these matches. In other words, the optimal strategy is myopic across all N matches at a given time t.

Before considering the mathematics of this problem, some further notation and restrictions are required. First, let

$$r_{kt} = \int_{l=0}^{t-\delta} \Big(f_{kl}(s_{kl}+1) - \sum_{j \in \Omega_Y} f_{jl} \Big)\, dl \tag{5.9}$$

denote the fixed return of bets placed at all times $0 \leq l \leq t-\delta$ for small $\delta > 0$ in a match conditional on the outcome $y = k$. Note that l can also be defined over discrete points in time. Furthermore, let

$$\phi_t = \int_{l=0}^{t-\delta} \Big(\sum_{j \in \Omega_Y} f_{jl} \Big)\, dl \tag{5.10}$$

be the total fraction of the bettor's initial wealth W_0 wagered up until time $t-\delta$ during the match. Then, the no leveraging assumption A8 implies that

$$\sum_{j \in \Omega_Y} f_{jt} < 1 - \phi_t \tag{5.11}$$

must hold. Now, consider the optimization problem in the stated scenario. After wagering on N identical matches, the wealth of the bettor is

$$W_{N,t}(\boldsymbol{f}, \boldsymbol{Y}) = W_0 \sum_{k \in \Omega_Y} \Big(1 + r_{kt} + f_{kt}(s_{kt}+1) - \sum_{j \in \Omega_Y} f_{jt} \Big)^{\sum_{i=1}^{N} I(Y_i = k)} \tag{5.12}$$

Then, as in the static scenario, the asymptotic logarithmic growth $G(\boldsymbol{f}_t)$ is

$$\lim_{N \to \infty} \frac{1}{N} ln \Big(\sum_{k \in \Omega_Y} \big(1 + r_{kt} + f_{kt}(s_{kt}+1) - \sum_{j \in \Omega_Y} f_{jt} \big)^{\sum_{i=1}^{N} I(Y_i = k)} \Big) := \lim_{N \to \infty} \frac{1}{N} G_N(\boldsymbol{f}_t) \tag{5.13}$$

To proceed in the same manner as earlier, it must be shown that Theorem 3 applies for Equation (5.13). But first, by the assumption of no allowed short positions A9, note that

$$1 + r_{kt} + f_{kt}(s_{kt}+1) - \sum_{j \in \Omega_Y} f_{jt} > 0 \quad ; k \in \Omega_Y \tag{5.14}$$

must hold. In fact, this is necessary to validly state the following:

Proposition 1. *$G(\boldsymbol{f})$ converges almost surely to $\sum_{k \in \Omega_Y} p_{kt} ln\Big(1 + r_{kt} + f_{kt}(s_{kt}+1) - \sum_{j \in \Omega_Y} f_{jt}\Big)$ by Theorem 3.*

Proof. Let $X_i(\boldsymbol{f}_t) = ln\Big(\sum_{k\in\Omega_Y}\big(1 + r_{kt} + f_{kt}(s_{kt}+1) - \sum_{j\in\Omega_Y} f_{jt}\big)^{I(Y_i=k)}\Big)$.
Then, since $ln(a^{\sum_{i=1}^{N} Y_i}) = \sum_{i=1}^{N} Y_i ln(a)$ for $a \in \mathbb{R}\backslash\{0\}$, we have $G_N(\boldsymbol{f}_t) = \sum_{i=1}^{N} X_i(\boldsymbol{f}_t)$.

Now, $\mathbb{E}[|X_i(\boldsymbol{f}_t)|] = p_{kt}|ln\Big(\sum_{k\in\Omega_Y}\big(1+r_{kt}+f_{kt}(s_{kt}+1)-\sum_{j\in\Omega_Y} f_{jt}\big)\Big)|$, since $p_{kt} \in (0,1)\,; k \in \Omega_Y$.

By Equation (5.14) and the fact that the odds are finite and positive,
$\mathbb{E}[|X_i(\boldsymbol{f}_t)|] < \infty$ since $|ln(a)| < \infty$ for $a > 0 \in \mathbb{R}$. Thus, $\frac{1}{N}G_N(\boldsymbol{f}_t)$ converges almost surely to $\sum_{k\in\Omega_Y} p_{kt} ln\Big(1 + r_{kt} + f_{kt}(s_{kt}+1) - \sum_{j\in\Omega_Y} f_{jt}\Big)$ by Theorem 3. ■

From this result, the asymptotic logarithmic growth rate is given by

$$G(\boldsymbol{f}_t) = \sum_{k\in\Omega_Y} p_{kt} ln\Big(1 + r_{kt} + f_{kt}(s_{kt}+1) - \sum_{j\in\Omega_Y} f_{jt}\Big) \tag{5.15}$$

Given the stated assumptions and Equation (5.15), the optimization problem in question can be defined formally as

$$\begin{aligned}
(*)\quad \underset{\boldsymbol{f}_t}{\text{maximize}} \quad & \mathrm{G}(\boldsymbol{f}_t) \\
\text{subject to} \quad & f_{kt} \geq 0;\ k \in \Omega_y && (4)\\
& \sum_{k\in\Omega_Y} f_{kt} < 1 - \phi_t, && (5)\\
& 1 + r_{kt} + f_{kt}(s_{kt}+1) - \sum_{j\in\Omega_Y} f_{jt} > 0;\ k \in \Omega_Y && (6)
\end{aligned}$$

Note that the stated constraints are renumbered to $(4) - (6)$ for simplicity. Now, if $(*)$ is a convex problem, then a unique solution exists on the feasible region $\mathcal{R}^*(\boldsymbol{f}_t)$, and an optimization algorithm should be able to solve the problem reasonably fast if Ω_Y is not too large. To show that $(*)$ is convex, it suffices to show that $\mathcal{R}^*(\boldsymbol{f}_t)$ is convex and that $G(\boldsymbol{f}_t)$ is concave on $\mathcal{R}^*(\boldsymbol{f}_t)$ by Definition 3. This is shown below.

Proposition 2. *The feasible region $\mathcal{R}^*(\boldsymbol{f}_t)$ defined by $(4) - (6)$ is a convex set.*

Proof. First, let $\mathcal{R}_6(\boldsymbol{f}_t) = \{\boldsymbol{f} \in \mathbb{R}^{|\Omega_Y|} : \boldsymbol{f}_t \ \ satisfies \ \ (6)\}$. Since $\mathcal{R}_6(\boldsymbol{f}_t)$ is the intersection of the sets

$R_6(f_{kt}) = \{f_{kt} \in \mathbb{R} : 1 + r_{kt} + f_{kt}(s_{kt}+1) - \sum_{j\in\Omega_Y} f_{jt} > 0\}$, then it suffices to show that $R_6(f_{kt})$ is convex for an arbitrary $k \in \Omega_Y$ to validly claim that $\mathcal{R}_6(\boldsymbol{f}_t)$ is convex by Theorem 2.

Thus, let $f_{kt}^1, f_{kt}^2 \in R_6(f_{kt})$ for some arbitrary $k \in \Omega_Y$, and let $\lambda \in [0,1]$, where $f_{kt}^1, f_{kt}^2, \lambda$ is chosen arbitrarily. Then

$1 + r_{kt} + (\lambda f_{kt}^1 + (1-\lambda) f_{kt}^2)(s_{kt}+1) - \lambda \sum_{j\in\Omega_Y} f_{jt}^1 - (1-\lambda)\sum_{j\in\Omega_Y} f_{jt}^2 =$
$\lambda(1+r_{kt}+f_{kt}^1(s_{kt}+1)-\sum_{j\in\Omega_Y} f_{jt}^1)+(1-\lambda)(1+r_{kt}+f_{kt}^2(s_{kt}+1)-\sum_{j\in\Omega_Y} f_{jt}^2) > \lambda 0+(1-\lambda)0 = 0$,
so $\lambda f_{kt}^1 + (1-\lambda) f_{kt}^2 \in R_6(f_{kt})$.

Thus, by Definition 2, $R_6(f_{kt})$ is convex, and $\mathcal{R}_6(\boldsymbol{f}_t) = \bigcap_{k\in\Omega_Y} R_6(f_{kt})$ is convex, since k was chosen arbitrarily. Now,

$\lambda \sum_{k\in\Omega_Y} f_{kt}^1 + (1-\lambda)\sum_{k\in\Omega_Y} f_{kt}^2 < \lambda(1-\phi_t) + (1-\lambda)(1-\phi_t) = 1-\phi_t$, so $\lambda \boldsymbol{f}_t^1 + (1-\lambda)\boldsymbol{f}_t^2 \in \mathcal{R}_5(\boldsymbol{f}_t)$ for any $f_{kt}^1, f_{kt}^2 \in R_5(f_{kt})$ and $\lambda \in [0,1]$. Thus $\mathcal{R}_5(\boldsymbol{f}_t)$ is also convex by Definition 2.

By the same argument, the set of fractions $\mathcal{R}_4(\boldsymbol{f}_t)$ of $\boldsymbol{f}_t$ satisfying (4) is also convex, since $\lambda \sum_{k\in\Omega_Y} f^1_{kt} + (1-\lambda)\sum_{k\in\Omega_Y} f^2_{kt} \geq \lambda 0 + (1-\lambda)0 = 0$ for any $f^1_{kt}, f^2_{kt} \in \mathcal{R}_4(\boldsymbol{f}_t)$ and $\lambda \in [0,1]$.

Now, since $\mathcal{R}^*(\boldsymbol{f}_t)$ is the intersection of the convex sets $\mathcal{R}_4(\boldsymbol{f}_t), \mathcal{R}_5(\boldsymbol{f}_t), \mathcal{R}_6(\boldsymbol{f}_t)$, then it is also convex by Theorem 2. ■

Proposition 3. *The function $G(\boldsymbol{f}_t)$ is convex on $\mathcal{R}^*(\boldsymbol{f}_t)$.*

Proof. To show that this proposition holds, define $g_k(\boldsymbol{f}_t) = ln\big(1 + r_{kt} + f_{kt}(s_{kt}+1) - \sum_{j\in\Omega_Y} f_{jt}\big)$.

Consider $\mathcal{R}_6(\boldsymbol{f}_t)$ defined above. A basis of the proof is that $g_k(\boldsymbol{f}_t)$ is concave on $\mathcal{R}_6(\boldsymbol{f}_t)$ for any $k \in \Omega_Y$. This follows directly from the fact that $ln(a)$ is strictly concave on its entire domain $\mathbb{R}\backslash\{0\}$, and that $1 + r_{kt} + f^1_{kt}(s_{kt}+1) - \sum_{j\in\Omega_Y} f^1_{jt} > 0$ for any $k \in \Omega_Y$ and any $\boldsymbol{f}_t \in \mathcal{R}_6(\boldsymbol{f}_t)$.

Thus, by Theorem 1, $G(\boldsymbol{f}_t) = \sum_{k\in\Omega_Y} p_{kt} g_k(\boldsymbol{f}_t)$ is also concave on $\mathcal{R}_6(\boldsymbol{f}_t)$, as $p_{kt} > 0 \;\; ; k \in \Omega_Y$.

Now, it suffices to show that this implies that $G(\boldsymbol{f}_t)$ is concave on $\mathcal{R}^*(\boldsymbol{f}_t)$. Note that since $\mathcal{R}^*(\boldsymbol{f}_t)$ is the intersection of $\mathcal{R}_4(\boldsymbol{f}_t), \mathcal{R}_5(\boldsymbol{f}_t), \mathcal{R}_6(\boldsymbol{f}_t)$, then $\mathcal{R}^*(\boldsymbol{f}_t) \subseteq \mathcal{R}_6(\boldsymbol{f}_t)$. Hence, since $\mathcal{R}^*(\boldsymbol{f}_t)$ is itself convex, we have that

$\tilde{\boldsymbol{f}}_t = \lambda \boldsymbol{f}^1_t + (1-\lambda)\boldsymbol{f}^2_t \in \mathcal{R}^*(\boldsymbol{f}_t)$ for any $\boldsymbol{f}^1_t, \boldsymbol{f}^2_t \in \mathcal{R}^*(\boldsymbol{f}_t) \subseteq \mathcal{R}_6(\boldsymbol{f}_t)$ and any $\lambda \in [0,1]$ by Definition 2.

But then, since $G(\boldsymbol{f}_t)$ is strictly concave on $\mathcal{R}_6(\boldsymbol{f}_t)$, and $\tilde{\boldsymbol{f}}_t = \lambda \boldsymbol{f}^1_t + (1-\lambda)\boldsymbol{f}^2_t \in \mathcal{R}^*(\boldsymbol{f}_t) \subseteq \mathcal{R}_6(\boldsymbol{f}_t)$ for any $\boldsymbol{f}^1_t, \boldsymbol{f}^2_t \in \mathcal{R}^*(\boldsymbol{f}_t)$ and any $\lambda \in [0,1]$, it is also strictly concave on $\mathcal{R}^*(\boldsymbol{f}_t)$. ■

As stated, since $(*)$ is a convex problem with $G(\boldsymbol{f}_t)$ strictly convex, then a unique maximum of $G(\boldsymbol{f}_t)$ exists on $\mathcal{R}^*(\boldsymbol{f}_t)$. Thus, the problem can be solved in a reasonable time for in-game betting purposes by an optimization algorithm, or by solving $\nabla G(\boldsymbol{f}) = 0$ by some variation of the Newton-Raphson algorithm. As a final note, no matches are assumed to be played simultaneously in this approach. The proposed framework can be further extended to account for bets on E simultaneous events given that A2 and A3 hold for all such sets of simultaneous events. This can be done by altering Equation (5.6) to

$$W^E_{N,t}((\boldsymbol{f}_e, \boldsymbol{Y}_e)_{e\in\Omega_e}) = W_0 \sum_{e\in\Omega_e} \sum_{k\in\Omega_Y} \Big(1 + r_{kte} + f_{kte}(s_{kte}+1) - \sum_{j\in\Omega_Y} f_{jte}\Big)^{\sum_{i=1}^N I(Y_{ie}=k)} \tag{5.16}$$

where $\Omega_e = \{1, 2, \ldots, E\}$. However, the approach is deemed out of scope for this thesis.

5.6.3 Application of the Betting Strategies

Now that both the static benchmark strategy and the proposed dynamic in-game strategy are discussed, the methodology used for applying the generated prediction models in these strategies is presented here.

First, note that the optimality of the two approaches rests on the assumption of perfect information in the probabilities. As discussed in Section 3.2, Baker and McHale (2013) show that a shrinkage $f^* \to \gamma f^*$ for $\gamma \in (0,1)$ of the Kelly fractions f^* is always optimal in the case of imperfect information about the probabilities. Since the historical live odds available for this research project only covers the matches in the chosen test set, and due to the time complexity of generating accurate sample variances of the estimated probabilities, optimizing the value of γ is not considered worthwhile for the purpose of this research. Instead, the chosen approach is to

test the proposed strategies for several values of γ. Ideally, the optimal γ should be a decreasing function of the uncertainty in the probabilities, as suggested by Baker and McHale (2013).

Static Kelly betting

The static $MutexKelly$ algorithm is used to consider the problem of betting at a single time t during the lifetime of the odds. This is conducted to evaluate the betting performance of the predictive models at every time step, in a similar fashion to the evaluation of their predictive performance. This approach is presented in Table 5.6. Note that k is the correct outcome for a given match, $\boldsymbol{S}_t$ are odds provided by Sportradar and $\boldsymbol{P}_t$ constitutes the estimated probabilities from a prediction model.

Static Kelly Betting
1: Initialization
Sort all matches according to start time so they occur sequentially
$t \leftarrow$Time in match at which bets are placed
$\gamma \leftarrow$Maximum fraction of wealth to bet on a single match
$\boldsymbol{P_t} \leftarrow$1X2 predictions for all matches
$\boldsymbol{S_t} \leftarrow$1X2 fractional odds all matches
$wealth \leftarrow 1.0$
2: Find Optimal Fractions and Update Wealth
for $i \in \{1, \ldots, number\ of\ matches\}$
$\quad \boldsymbol{f}^* \leftarrow MutexKelly(\boldsymbol{p}_{i,t}, \boldsymbol{s}_{i,t})$
$\quad profit = \left(f_k^*(s_{i,kt}+1) - \sum_{j\in\Omega_Y} f_j^*\right) \times \gamma \times wealth$
$\quad wealth = wealth + profit$
end for

Table 5.6: Overview of the static Kelly betting procedure.

When evaluated over multiple points in time, the static strategy represents a hypothetical scenario where a bettor splits its wealth among several agents a_t ; $t \in \{1, \ldots, T\}$, each of which is responsible for placing bets according to the static betting scenario at a unique time t during all matches. At the end of the investment horizon, the return of the bettor is a linear combination of the returns generated by the agents. Thus, since $\sum_{t=1}^{T} ln(r_t) \neq ln(\sum_{t=1}^{T} r_t)$ in general, the bettor does not satisfy A1, although all these agents do. Consequently, assume that this hypothetical bettor (*irrational log bettor*) represents an investor acting irrationally according to its assumed logarithmic utility function. As stated, this bettor is used to evaluate the performance of the WCD and LSTM models for different time intervals as well as serving the role of a benchmark for the dynamic strategy.

Dynamic Kelly Betting

Now, consider the proposed dynamic strategy, which is shown to act as a rational investor according to a logarithmic utility function. The explicit approach entails solving $(*)$ for every time step $t \in \{0, 10, \ldots, 90\}$ in every match i by updating the coefficients r_{kt} and ϕ_t in Equation (5.9) and in Equation (5.10) based on bets placed prior to t in the same match. The intervals of t are chosen to allow for relatively frequent bets and at the same time ensure that the bets are

spread throughout the match. Furthermore, the odds $\boldsymbol{S}$ are provided by Sportradar and the probabilities $\boldsymbol{P}$ are estimated by the WCD and LSTM models. An overview of this functionality can be seen in Table 5.7, where k is the correct outcome of a given match.

Dynamic Kelly Betting

1: Initialization
Sort all matches according to start time so they occur sequentially
$\gamma \leftarrow$Maximum fraction of wealth to bet on a single match
$\boldsymbol{P} \leftarrow$1X2 predictions for all matches and time intervals
$\boldsymbol{S} \leftarrow$1X2 fractional odds all matches and time intervals
$wealth \leftarrow 1.0$
$\Omega_t = \{0, 10, \ldots, 90\}$

2: Find Optimal Fractions and Update Wealth
for $i \in \{1, \ldots, number\ of\ matches\}$
  for $t \in \Omega_t$
    $\boldsymbol{f}_t^* \xleftarrow{maximize} (*)$
  end for
  $profit = \sum_{t \in \Omega_t} \left(f_{kt}^*(s_{i,kt} + 1) - \sum_{j \in \Omega_Y} f_{jt}^* \right) \times \gamma \times wealth$
  $wealth = wealth + profit$
end for

Table 5.7: Overview of the dynamic Kelly betting procedure.

Chapter 6

Results - Predictive Performance

This chapter is dedicated to a presentation of the results from the model selection and model evaluation processes, as well as a corresponding discussion with a special emphasis on the research questions RQ1 and RQ2. The chosen hyperparameters and features for each model are put forward in Section 6.1, before a comparison of the generated models based on the chosen metrics follows in Section 6.2. The chapter is concluded with a discussion regarding RQ1 and RQ2 in Section 6.3.

6.1 Model Selection

The results from the model selection procedure is presented in this section. The presentation is structured based on the scientific procedure, where hyperparameters are chosen before the explicit subset of features.

6.1.1 Chosen Hyperparameters

The only hyperparameter of the WCD architecture is the number of features d, while the LSTM architecture also depends on several others. Among these, the dropout rate and the parameters of the elastic net are chosen by grid search. Their optimal values are presented and discussed here.

Regarding the dropout rate p and the parameters λ_1 and λ_2 in the elastic net, they are chosen separately for each LSTM model. However, the same values are chosen in the grid search for all these models, indicating that this combination is optimal for the task of predicting the outcome of football matches. Explicitly, the chosen values for these models are $p = 0.3$ and $(\lambda_1, \lambda_2) = (0, 0.1)$. These results imply that 30% of the input signals to all three layers in the LSTM network are ignored. Furthermore, no automatic model selection is conducted by the regularizer due to the absence of the lasso penalty in the loss function. However, the ridge penalty does to some extent penalize high parameter values and ensures that no single parameter is assigned too much predictive power. Based on this, the loss function to be minimized to obtain the LSTM models is given by

$$CE(\boldsymbol{\theta}|f_Y, \boldsymbol{X}, \boldsymbol{y}) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k\in\Omega_Y}\Big[I(y_i = k)ln\big(f_Y(k|\boldsymbol{x_i}, \boldsymbol{\theta})\big)\Big] + 0.1\|\boldsymbol{\theta}\|_2^2 \tag{6.1}$$

where the cross entropy is taken over the relevant probability distribution.

Next, consider the number of features d. The optimal values of d for each model according to the grid search are presented in Table 6.1. Although these values are fairly similar across models and lie within the range $[10, 17]$ in the pre-game scenario, this is not the case in-game. Furthermore, the top five suggestions from the grid search, as can be seen in Appendix C, show that the optimal number of features seem rather random. This can to some extent be attributed to the chosen univariate selection procedure, since its ignorance of the combined explaining power of features entails a high risk of choosing suboptimal feature sets for each proposed d. This again implies a large degree of randomness in the manner of which d itself is chosen. Additionally, the two-stage grid search procedure implies that d is not chosen simultaneously with the other hyperparameters in the models based on the LSTM architecture, imposing more randomness in the choice. This randomness may also be partly due to biased estimates of the generalization ability or the assumption that the cross entropy is a convex function of d in the grid search. A combination of all these moments largely supports the use of the recommended selection procedure.

Model	Pre-game d	In-game d
WCD_{score}	15	37
WCD_{1X2}	10	10
$LSTM_{score}$	14	24
$LSTM_{copula}$	14	24
$LSTM_{1X2}$	17	45

Table 6.1: The chosen number of features for all models.

6.1.2 Chosen Features

Now that d is chosen, consider the importance assigned to the features based on mutual information with the class labels. Note that the corresponding parameter estimates for the models are not presented nor discussed, as the focus of this research is valid model comparison based on a fixed scientific process rather than conducting parameter inference.

Although the importance assigned to most features varies considerably between the different models, some features are strongly indicated to have predictive power on the outcomes of football matches. The latter can be observed in Table 6.2, which presents the eight highest ranking features for some of the models at different times in the matches. The ranking represents the ordinality of mutual information scores with the class labels.

Feature	Time 0			Time 45			Time 90		
	M1	M2	M3	M1	M2	M3	M1	M2	M3
Diff Importance 1 pw0.25	8	7	7	3	3	3	1	1	1
Diff Importance 18 pw0.25	3	3	3	2	2	2	2	3	2
Diff Points		9		8	6	9	3	2	3
Diff Elo	1	2	2	1	1	1			
Diff Avg player strength	2	1	1	7		7			
Diff Goals scored	9	8	9	6		5			
Diff Goals scored/Match	10	10		4	4	4			
Diff Points/Match	5	5	5	10	7				

Table 6.2: The features that are most often in the top 10 based on mutual information. M1 = WCD_{score}, M2 = WCD_{1X2}, M3 = $LSTM_{copula}$.

Among the top ranking features, the importance of a match from the perspective of the opposing teams is heavily represented, indicating that they are useful for predicting the outcome of football matches. This aligns with the results of Silver and Boice (2018) and Goddard (2005). The same can also be seen for both the Elo and FIFA ratings, except at $Time = 90$. This coincides with the observed correlation presented in Section 4.4.2. The presence of the number of points and goals scored during a season among the top ranking features is supported by their inherent meaning in the game of football. An important note considering all these features, however, is that they all represent very similar information.

A more detailed overview of feature rankings based on mutual information is presented in Appendix D. Here, all features deemed to be among the ten top ranking features for at least one model is included along with their ranking. From this, one can see that very few in-game features are included, and a presentation of these in-game features are also found in Appendix D. They are for the most part poorly ranked, except for the score features in $LSTM_{1X2}$ and $LSTM_{score}$. This can be attributed to the selection procedure, in which the mutual information is calculated directly with the 1X2 and scoreline distributions respectively, while the class labels for the other models is the univariate goal distribution. Although the suggested limited explanatory power of the number of goals scored in the rest of the match may seem counterintuitive, the estimated correlation coefficients in Table 4.5 support this claim. The opposite is suggested by Nevo and Ritov (2012) considering the total amount of goals scored in a match, where the first goal is proposed to have an expediting effect of the scoring rate of both teams. This is in line with the higher importance assigned to the goal features for the explicit 1X2 and scoreline distributions, as represented in $LSTM_{1X2}$ and $LSTM_{score}$.

The low presence of in-game features in the models indicates that pre-game information and the current scoreline can be used to generate reasonably sound in-game predictions. As this claim is also supported by the estimated correlation coefficients, there seems to be a small dependence between these in-game events and the goal processes in the EPL.

6.2 Model Evaluation

As a means of answering RQ1 and RQ2, a comparison of the performance of all the generated models is presented here. First, some reference models are introduced to give perspective to the discussion, before all models are compared based on each of the chosen metrics and the reference models.

6.2.1 Benchmark Models

To put the predictive ability of the generated models in perspective, a set of benchmarks is presented. This set includes a pre-game prediction model for the 1X2 distribution, a naïve in-game reporter, and implied probabilities from the estimated 1X2 odds supplied by Sportradar as well as from average odds in the pre-game 1X2 betting market.

In order for the probabilities from the mentioned sources to serve as valid benchmarks, the quantities in question should be generated on the defined test set. These benchmark values are here generated from the prediction model of Silver and Boice (2018) and the implied probabilities from the Sportradar odds model. The choice is based on Sportradars position as an important supplier of sports predictions and the reputation of Nate Silver, both suggesting that beating the prediction power of these models requires very good estimates of the probability distribution. In addition, the implied probabilities from the average odds in the market, as gathered from Football-Data.co.uk (2019), is considered a reasonable estimate of the predictive power inherent in the pre-game betting market. The pre-game metrics for Silver and Boice (2018) and Football-Data.co.uk (2019) is presented in Table 6.3, while the performance of the implied probabilities from Sportradar are presented along with the metrics from the WCD and LSTM models.

Name	Acc. 1X2	CE 1X2
Silver and Boice	0.5663	0.9327
Football-Data.co.uk	0.5776	0.9163

Table 6.3: Metrics calculated based on the predictions from Silver and Boice (2018) and the odds from Football-Data.co.uk.

Furthermore, a predictive model should have at least some inductive ability. Thus, an algorithm only reporting the most frequent outcome according to the sample average in the test set is constructed to serve as an expected lower bound on the 1X2 and scoreline accuracy. This algorithm is hereafter referred to as the *naïve reporter*, and states results in the following manner: The class reported at time t for the 1X2 and scoreline distributions are given by

$$k^*_{1X2}(t) = argmax_{k\in\{1,X,2,current\}}\{\bar{p}_k\} \tag{6.2}$$

$$k^*_{score}(t) = argmax_{score\in\{0,1,\ldots,9\}\times\{0,1,\ldots,9\}}\{\bar{p}_{score}\} \tag{6.3}$$

respectively, where $\bar{p}_1, \bar{p}_X, \bar{p}_2$ refers to the average proportion of 1X2 outcomes, $\bar{p}_{current}$ is the average proportion of matches with unchanged 1X2 outcome on the interval $[t, full-time]$, and $\bar{p}_{(i,j)}$ has the same interpretation for a scoreline (i, j) in the remainder of a match. The ex-post accuracy scores generated by the naïve reporter can be seen in Table 6.4 and are considerably lower than those stated above for pre-game predictions. One should expect to see a convergence to these ex-post scores in late stages of the matches for all models, as the current state becomes an increasingly likely outcome as full-time approaches.

Time	1X2	Scoreline
0	0.4735	0.1074
10	0.4735	0.1233
20	0.4735	0.1419
30	0.4973	0.1552
40	0.5637	0.1844
50	0.6406	0.2427
60	0.7095	0.3302
70	0.7666	0.4443
80	0.8302	0.6207
90	0.9324	0.8488

Table 6.4: Accuracy scores for the naïve reporter.

Now, the emphasis is moved to the generated models and a comparison between them, as well as with the presented benchmarks. The comparisons are conducted for each metric, where the predictive power of the models on the 1X2 distribution is evaluated first.

6.2.2 Cross entropy - 1X2

A comparison of the models based on cross entropy on the 1X2 distribution is presented here. An overview of the values of this metric is presented in Table 6.5. For perspective on the quantities presented, the difference $CE_2 - CE_1$ in cross entropy between two models m_1 and m_2 can be interpreted as $e^{CE_2 - CE_1} = \frac{p_1}{p_2}$, where p_1 and p_2 are the probabilities of the true class of an average match for $m1$ and $m2$ respectively. See Appendix E for the derivation behind this statement. Thus, this difference indicates the average ratio of the probabilities of the true class for model m_1 as a fraction of those for model m_2. As an example, $CE_2 - CE_1 = 0.05 \implies \frac{p_1}{p_2} = e^{0.05} \approx 1.05$.

Time	WCD_{score}	WCD_{1X2}	$LSTM_{copula}$	$LSTM_{score}$	$LSTM_{1X2}$	Sportradar
0	0.9755	0.9437	0.9524	0.9486	0.9423	0.9234
10	0.9237	0.9226	0.9327	0.9339	0.9359	0.9022
20	0.8913	0.8913	0.8939	0.9036	0.9051	0.8654
30	0.8352	0.8462	0.8788	0.8518	0.8766	0.8280
40	0.7852	0.7853	0.7991	0.7956	0.8096	0.7777
50	0.7202	0.7181	0.7239	0.7317	0.7281	0.7116
60	0.6413	0.6434	0.6589	0.6558	0.6700	0.6346
70	0.5700	0.5688	0.5744	0.5831	0.6213	0.5714
80	0.4359	0.4402	0.4454	0.4467	0.4844	0.4385
90	0.2338	0.2379	0.2319	0.2411	0.2626	0.2725

Table 6.5: Cross entropy of all models with respect to the 1X2 sample distribution.

First, consider the pre-game predictions. While four of the models are indicated to have similar predictive power, WCD_{score} is outperformed by all the other models. Furthermore, the cross entropy of all models are higher than those of the implied probabilities from Sportradar, Silver and Boice (2018) and Football-Data.co.uk (2019). This also holds throughout most of the match when using Sportradar as a benchmark, a proposition that becomes even more apparent in

Figure 6.1, where the difference between the generated models and the Sportradar probabilities are plotted. An interesting result is that all the models are indicated to have a considerably better fit to the distribution at $t = 90$ than the Sportradar probabilities, except for $LSTM_{1X2}$ which performs approximately on par with these probabilities. This seems reasonable, since Sportradar takes market information into account for generating these probabilities. Due to the large uncertainty relative to the mean as matches approach full-time, as indicated in Section 4.4.2, it seems reasonable to assume that the probabilities from Sportradar are heavily based on market information in these stages. It may also be the case that the Sportradar odds data have a delay compared to their event data, causing a change in scoreline close to the 90th minute to be registered only in the latter data set.

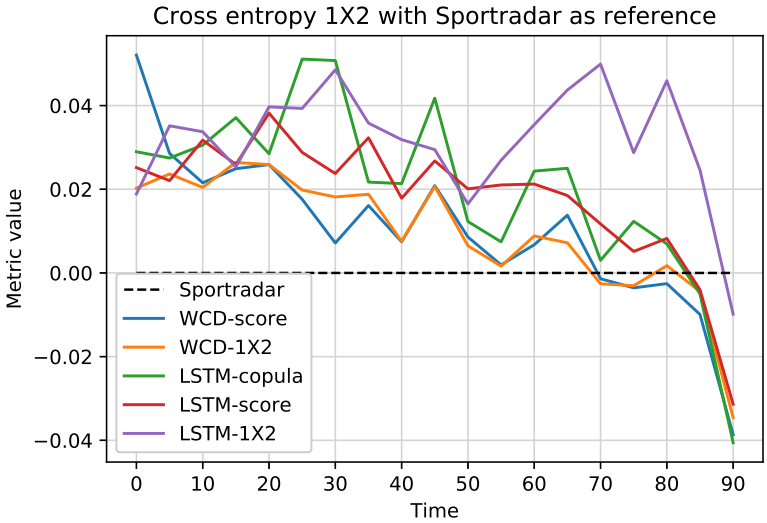
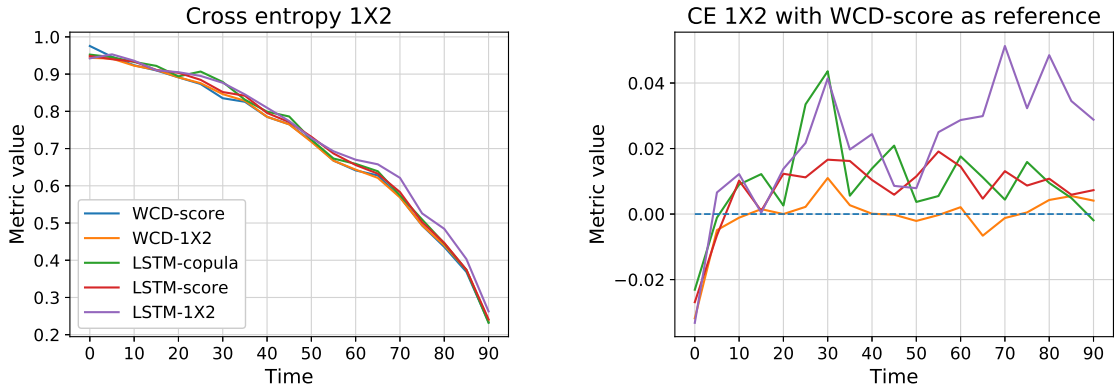


Figure 6.1: Plot of difference in cross entropy between each model and implied probabilities from Sportradar.

Now, consider only the generated models. To ease the comparison between the models, the cross entropy is plotted as a function of elapsed time during matches in Figure 6.2. From this, one can see that the cross entropy is approximately monotonically decreasing as the matches progress. This is as expected due to less uncertainty in the outcome, but still indicates that the predictions improve as time decreases. Furthermore, when taking WCD_{score} as a reference, the differences in performance become more apparent. WCD_{score} is indicated to perform worse than the other models pre-game, and the $LSTM_{1X2}$ model seems to struggle at later stages in comparison to the other models. This is an interesting aspect, as this is the only model generated for predicting the 1X2 distribution directly. Another interesting aspect is that the general performance of models based on the WCD architecture seems to be slightly better than the LSTM models.



(a) Cross entropy value.

(b) Difference between WCD_{score} and the other models.

Figure 6.2: Plot of cross entropy of all models with respect to the 1X2 sample distribution.

6.2.3 Accuracy - 1X2

The accuracy score is the second metric chosen for evaluating the performance with respect to the 1X2 distribution. The values of this metric is presented in Table 6.6.

Time	WCD_{score}	WCD_{1X2}	$LSTM_{copula}$	$LSTM_{score}$	$LSTM_{1X2}$	Sportradar
0	0.5442	0.5751	0.5670	0.5770	0.5783	0.5790
10	0.5818	0.5764	0.5751	0.5770	0.5676	0.5790
20	0.5952	0.5925	0.5912	0.5850	0.5770	0.6042
30	0.6206	0.6139	0.5831	0.6118	0.5957	0.6242
40	0.6314	0.6394	0.6220	0.6064	0.6238	0.6494
50	0.6716	0.6783	0.6743	0.6653	0.6586	0.6826
60	0.7198	0.7145	0.7105	0.7135	0.7229	0.7251
70	0.7668	0.7641	0.7668	0.7671	0.7470	0.7530
80	0.8311	0.8311	0.8324	0.8286	0.8166	0.8287
90	0.9316	0.9316	0.9316	0.9317	0.9331	0.9150

Table 6.6: Accuracy score of all models with respect to the 1X2 sample distribution.

The first thing to note is that the accuracy score suggests that all models perform better than the naïve reporter until late stages of the matches, where the performance is indicated to converge to that of the latter. Furthermore, these results support the claims made based on the cross entropy, where the pre-game performance of WCD_{score} also here is indicated to be worse than for the other models, and the Sportradar probabilities seem to have the better prediction power up until 90 minutes. However, the accuracy scores indicate that all the generated models, except for $LSTM_{1X2}$, performs slightly better than the Sportradar model at 70 and 80 minutes. These aspects should become clear when considering Figure 6.3. It is also worth noting that, in addition to WCD_{score}, the pre-game accuracy score of $LSTM_{copula}$ and Silver and Boice (2018) are slightly lower than the rest of the models, including that of Football-Data.co.uk (2019).

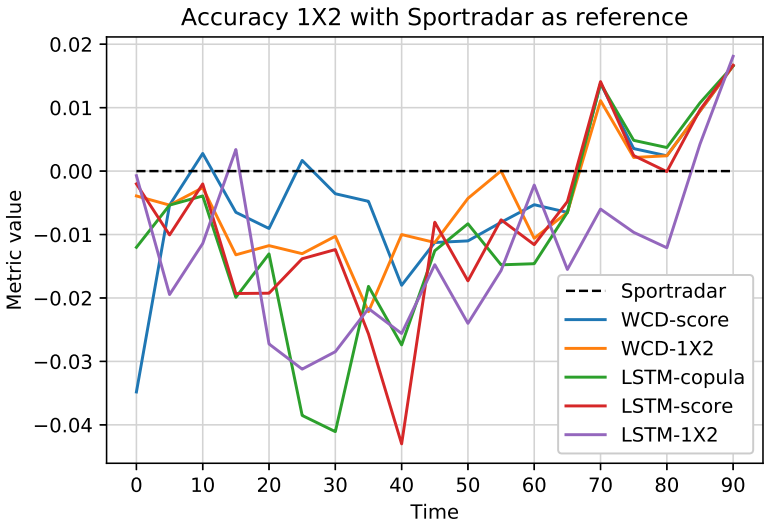
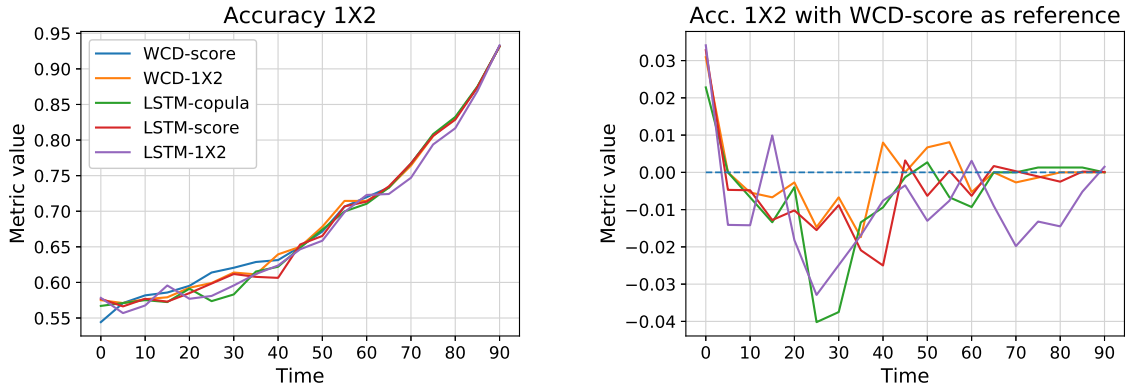


Figure 6.3: Plot of difference in accuracy between each model and implied probabilities from Sportradar.

Figure 6.4 shows the same as for the cross entropy, where the performance improves over time in general and the models based on the WCD architecture seems to slightly outperform the LSTM models. Sudden decreases in the accuracy, as can be seen for $LSTM_{copula}$ at $time = 30$ can most likely be attributed to the varying feature sets. Furthermore, $LSTM_{1X2}$ again seems to have the worst performance based on the 1X2 accuracy score. Next, note that WCD_{score} improves greatly from pregame to $time = 5$ with respect to both cross entropy and accuracy, possibly indicating that it either overfits the scoreline distribution or that it relies on an unfavourable set of pregame features.



(a) Accuracy score.

(b) Difference between WCD_{score} and the other models.

Figure 6.4: Plot of accuracy of all models with respect to the 1X2 sample distribution.

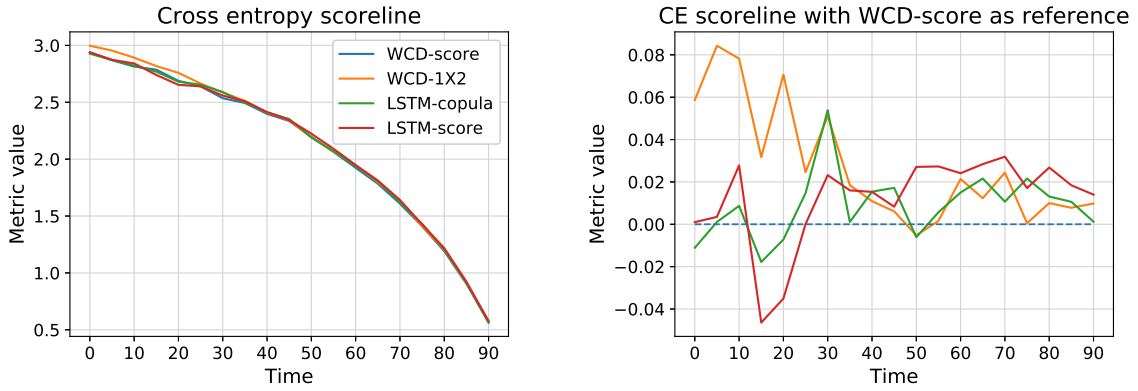
6.2.4 Cross Entropy - Scoreline

The emphasis is now shifted to the scoreline distribution. Since the odds data supplied by Sportradar does not include odds for the scoreline, and $LSTM_{1X2}$ directly estimates the 1X2 distribution, their performance cannot be considered with respect to the former distribution. A presentation of the cross entropy of the other models is given Table 6.7.

Time	WCD_{score}	WCD_{1X2}	$LSTM_{copula}$	$LSTM_{score}$
0	2.9383	2.9970	2.9272	2.9393
10	2.8139	2.8921	2.8226	2.8417
20	2.6881	2.7587	2.6809	2.6530
30	2.5358	2.5869	2.5896	2.5590
40	2.3979	2.4088	2.4133	2.4132
50	2.1969	2.1917	2.1908	2.2240
60	1.9245	1.9459	1.9395	1.9486
70	1.6091	1.6335	1.6198	1.6410
80	1.1911	1.2011	1.2042	1.2179
90	0.5637	0.5735	0.5649	0.5777

Table 6.7: Cross entropy of all models with respect to the scoreline sample distribution.

Based on these results, the performance of all four models seems quite similar, although with a few exceptions. First, the in-game performance of WCD_{score} is, as for the previous two metrics, slightly higher than for the other models. However, it has improved considerably with respect to its pre-game metric scores, with the cross entropy being on par with the two LSTM models. Secondly, WCD_{1X2} has a considerably higher cross entropy than the other models during the first 30 minutes, which seems intuitive as the loss function minimized to obtain WCD_{1X2} is the cross entropy over the 1X2 distribution. Figure 6.5 presents the same type of plot as seen for the 1X2 metrics, and a rather strange result can also be seen at $time \in [15, 25]$. The cross entropy scores of both LSTM models are considerably lower than that of WCD_{score} on this interval. This may be a sign that the optimal sequence length for the networks lie between 3 and 5, implying that one should rely on an nth order Markov assumption with $n \in [3, 5]$.



(a) Cross entropy value.

(b) Difference between WCD_{score} and the other models.

Figure 6.5: Plot of cross entropy of all models with respect to the scoreline sample distribution.

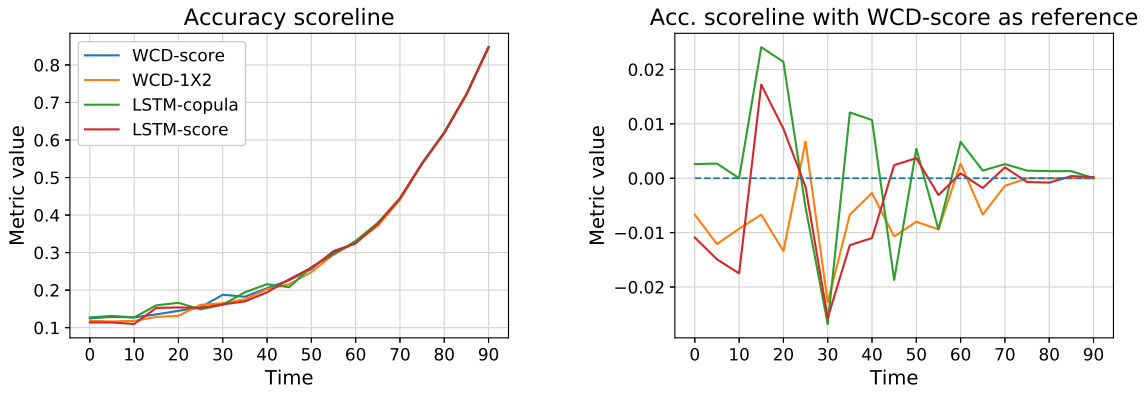
6.2.5 Accuracy - Scoreline

Although the accuracy score may be a rather weak indicator of the overall prediction performance for the scoreline distribution, it is briefly discussed here based on Table 6.8.

Time	WCD_{score}	WCD_{1X2}	$LSTM_{copula}$	$LSTM_{score}$
0	0.1247	0.1180	0.1273	0.1138
10	0.1273	0.1180	0.1273	0.1098
20	0.1448	0.1314	0.1662	0.1539
30	0.1877	0.1649	0.1609	0.1620
40	0.2051	0.2024	0.2158	0.1941
50	0.2560	0.2480	0.2614	0.2597
60	0.3244	0.3271	0.3311	0.3253
70	0.4424	0.4410	0.4450	0.4444
80	0.6193	0.6193	0.6206	0.6185
90	0.8472	0.8472	0.8472	0.8474

Table 6.8: Accuracy score of all models with respect to the scoreline sample distribution.

These values indicate that all models perform considerably better than the naïve reporter throughout the first half, while converging to its performance after 50 minutes. Due to a large number of classes, and thus a large uncertainty in the true class, it seems intuitive that this convergence occurs earlier than in the 1X2 scenario. Furthermore, the convergence indicates that all four models are likely to assign the highest probability to the current scoreline as the matches approach full-time. Regarding the comparison between the generated models, the performance ranking is rather random as a function of time elapsed, which is made even more apparent in Figure 6.6. No inferences should thus be drawn from these results regarding the relative performance of the models.



(a) Accuracy score.

(b) Difference between WCD_{score} and the other models.

Figure 6.6: Plot of accuracy of all models with respect to the scoreline sample distribution.

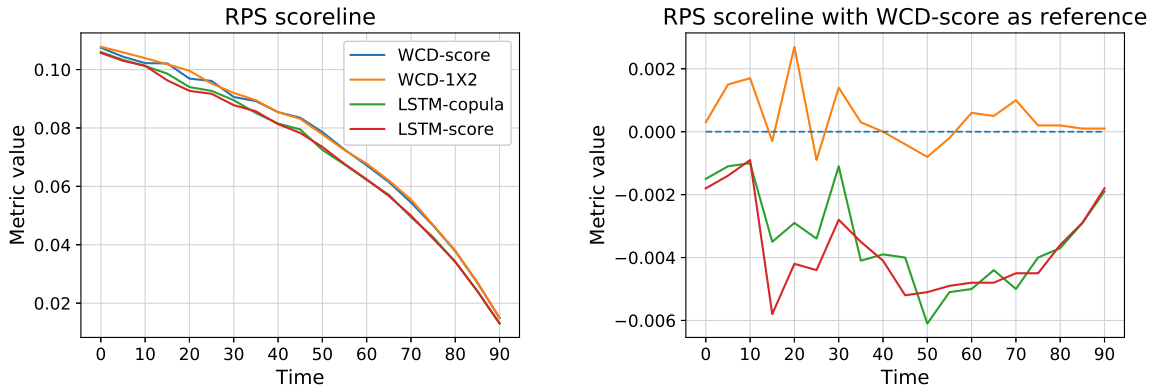
6.2.6 Ranked Probability Score - Scoreline

An important topic of the discussion surrounding the scoreline distribution, in addition to RQ1 and RQ2, is the ability of the models to capture information about the ordinal structure of scorelines. Consequently, the discussion ends with an emphasis on the ranked probability score, where the values of this metric are presented in Table 6.9.

Time	WCD_{score}	WCD_{1X2}	$LSTM_{copula}$	$LSTM_{score}$
0	0.1075	0.1078	0.1060	0.1057
10	0.1022	0.1039	0.1012	0.1013
20	0.0969	0.0996	0.0940	0.0927
30	0.0906	0.0920	0.0895	0.0878
40	0.0854	0.0854	0.0815	0.0813
50	0.0786	0.0778	0.0725	0.0735
60	0.0672	0.0678	0.0622	0.0624
70	0.0545	0.0555	0.0495	0.0500
80	0.0378	0.0380	0.0341	0.0342
90	0.0149	0.0150	0.0130	0.0131

Table 6.9: Ranked probability score of all models with respect to the scoreline sample distribution.

Here, an interesting pattern occurs. The models based on the LSTM architecture have a lower ranked probability score than the models based on the WCD architecture. Note that although the total difference is small, the proportional difference is quite large. This is an interesting result, as the WCD architecture implies an inherent ordinal structure due to WCD being a count distribution. This implies that, although the WCD is indicated to be a better model of the univariate scorelines in EPL than the Poisson distribution, the combination of the WCD and Frank copula does not seem to sufficiently model the ordinality in the bivariate scoreline. Furthermore, these results do to some extent coincide with the findings of Hvattum (2017). A visualization of the stated results is given in Figure 6.7 below, showing a consistently better RPS score for the LSTM models. The difference is made even more apparent by pairwise Welch's t-tests of the RPS values for the WCD and LSTM models, as can be seen in Appendix F.



(a) Ranked probability score.

(b) Difference between WCD_{score} and the other models.

Figure 6.7: Plot of ranked probability score of all models with respect to the scoreline sample distribution.

6.3 Discussion

A summary and an extension of the discussion in the previous section is presented here, with a special emphasis on RQ1 and RQ2. First, some general aspects from this discussion are pointed

out, before an explicit summary surrounding each of the research questions is given. As a reminder, RQ1 and RQ2 are restated here.

RQ1 *How does the performance of an artificial neural network compare to that of a Weibull count distribution model, when considered with respect to both pre-game and in-game prediction of the outcome of EPL matches?*

RQ2 *How do models of the scoreline distribution in EPL matches compare to otherwise equivalent models of the 1X2 distribution on the same set of matches, when their performance is measured on the latter distribution, and the prediction task is the one stated in the previous question?*

As shown, the implied probabilities of the average odds collected from Football-Data.co.uk (2019) is indicated to have the best pre-game predictive ability on the 1X2 distribution. In addition, the implied probabilities from the Sportradar odds seems to have the best overall performance throughout the match. A common property of both models is that they utilize market information, a property not held by any of the generated models. This indicates that the wisdom of crowds should be taken into account when creating football prediction models. This is supported by the claims of Peeters (2018) regarding market prices from Transfermarkt.com (2018), the findings of Godin et al. (2014) and Schumaker et al. (2016) regarding information in Twitter posts and the performance of Forza Football user votes (Nyquist and Pettersson, 2017).

The aspect of market information is also likely to explain some of the degraded performance seen by the Sportradar probabilities at $time = 90$. Sportradar has stated that their main interest is to estimate the optimal odds, which at $time = 90$ is likely to deviate from the fair probabilities due to the supply and demand in the betting market. This approach may generate optimal odds, but it seems to decrease the predictive performance of the implied probabilities compared to the generated models. However, this decrease in performance may also be attributed to a potential difference in the time stamps for the provided odds and event data, as previously stated.

Next, the copula functions are chosen to allow for a potential dependency between the goal processes. The estimated dependence parameter $\hat{\kappa}$ can be seen in Table 6.10, where the value is approximately zero for $LSTM_{copula}$ and also low for WCD_{score}. However, the variance in $\hat{\kappa}$ for WCD_{1X2} is larger, indicating a stronger dependency in some stages of the matches. This indicates that a dependency is likely to exist, supporting the findings from Section 4.4.2. Furthermore, the change in the magnitude of $\hat{\kappa}$ is likely a result of the rather small feature set used by WCD_{1X2}, implying that certain features indirectly model the dependence. This also supports the low values in the two other models, and it can be validly stated that a bivariate scoreline model should allow for dependence or utilize features that capture such a dependence.

Time	WB_{score}	WB_{1X2}	$LSTM_{copula}$
0	0.0223	0.4142	-0.0008
10	0.0766	0.3241	-0.0024
20	0.0444	0.0545	0.0029
30	-0.0079	-0.0082	0.0001
40	0.0166	0.2956	0.0020
50	-0.0183	-0.0385	0.0001
60	-0.0525	0.1157	-0.0001
70	-0.0127	0.0495	-0.0005
80	-0.0931	0.0811	0.0000
90	-0.0779	-0.0496	0.0000

Table 6.10: Chosen values of the copula dependency parameter κ.

6.3.1 Research Question 1

Now, based on the relative performance of the models based on two model architectures, a discussion surrounding RQ1 is presented here. As the general predictive ability of the models is the topic of interest, all metrics are taken into consideration. As a note, the performance of all models can be deemed acceptable with respect to the benchmarks, but their properties vary slightly.

The accuracy score on the 1X2 distribution indicates that both WCD models, and especially WCD_{score}, is able to more frequently assign the highest probability to the correct outcome, although WCD_{score} seems to be the worst performing model pre-game. This superiority is to some degree supported by the cross entropy with respect to both distributions. However, due to the inability of the cross entropy to accurately measure performance for ordinal random variables, as well as the somewhat varying ranking of the models based on this metric, few inferences can be drawn based on this alone. Now, consider the RPS on the scoreline distribution. This metric strongly suggests that the LSTM models have the best performance, which implies that the probability distribution generated by the LSTM networks better approximates the ordinality of the scoreline distribution than the Weibull count distribution.

Based on this discussion, the following general proposition can be stated regarding RQ1: The WCD architecture seems to be the superior architecture if the objective is solely to predict the true class, while the LSTM networks generate probabilities that better fit the scoreline distribution when accounting for its inherent ordinal structure. Therefore, the choice of architecture is dependent upon the purpose of the model.

6.3.2 Research Question 2

The focus of the discussion turns to the ability of the models to accurately estimate the 1X2 distribution. Note that, although WCD_{1X2} provides an estimate of the scoreline distribution, it is evaluated as a 1X2 model due to the loss function on which it is trained.

First, consider the models created based on the WCD architecture. Apart from the strong indication that the pre-game performance of WCD_{score} is inferior to WCD_{1X2}, few clear inferences can be drawn regarding the comparison of their performance. There is a slight indication that the former model is superior in the first half based on both accuracy score and cross entropy,

but the difference is not substantial. The almost identical performance of these models is in line with the result of Goddard (2005) for logistic regression models. In addition, as the 1X2 distribution is derived directly from the scoreline distribution based on the WCD distribution, it seems reasonable that the difference in loss function does not imply considerable differences between the trained models.

The performance of the LSTM models can be considered for two separate time intervals, the first being $time \in [0, 60)$ minutes. On this interval, $LSTM_{1X2}$ has comparable performance to the two other LSTM models and all three models score similarly for both the cross entropy and the accuracy score. The same is not indicated for $time \in [60, 90]$ minutes, where the 1X2 model seems to perform poorly. Except for a few points in time, it is indicated to be the most inferior among all models, including the WCD models. This may be attributed to $LSTM_{1X2}$ being an explicit 1X2 model without knowledge of the fact that the scoreline determines the 1X2 outcome. Thus, it is hypothesised that this lack of knowledge complicates the classification task. Furthermore, this effect is likely to be amplified by the fact that features are chosen based on mutual information with the 1X2 outcome instead of the scorelines.

Explicitly, the following statement summarizes the discussion regarding RQ2: The WCD models perform similarly, with the scoreline model slightly outperforming the 1X2 model on in-game prediction tasks, while the opposite is indicated pre-game. Thus, no statement can be made regarding an overall superior model, but a 1X2 model may be preferred to ensure acceptable pre-game performance. For LSTM models, the metrics indicate that the scoreline models are superior as the matches approach full-time, thus making them the preferable choices for estimating the 1X2 distribution.

Chapter 7

Results - Betting Performance

This chapter is devoted to a presentation of the results from applying the betting strategies presented in Section 5.6.1 and Section 5.6.2 in a hypothetical in-game betting market. First, this market is presented in Section 7.1, before Section 7.2 presents an evaluation of the ability of the prediction models to contribute to financial returns in a static scenario. Section 7.3 follows with a comparison of the performance of the two betting strategies, before section 7.4 concludes the chapter with a summary of the presented results and a corresponding discussion surrounding RQ3.

7.1 Fundamentals

The betting market is in this thesis represented by the odds provided by Sportradar. As these odds do not have an inherent risk premium, such a premium must be chosen. An analysis of pre-game odds obtained from Football-Data.co.uk (2019) shows that most standalone bookmakers supply odds with an incorporated premium of about 5%, while the premium is approximately zero for combinations of maximum odds present in the market. Based on the assumption that such odds combinations can be obtained reasonably fast, a risk premium of 0.001% is used when evaluating the betting performance. Although this choice may imply overly optimistic results, it is supported by the vast amount of odds comparison sites making these combinations accessible if they exist (BetStudy.com, 2019; BetBrain.com, 2019; Oddschecker.com, 2019; Easyodds.com, 2019).

Furthermore, bookmakers are likely to place an upper bound on the odds as well as significantly increase their premium as matches progress and certain outcomes become highly unlikely. All odds $s > 35$ are therefore shifted to 1.0 in order to remove the risk of allowing for bets corresponding to odds not supplied in the real market. This is also based on the hypothesis that the uncertainty in the probability estimates increases as outcomes become more unlikely.

Although Sportradar provided odds for several markets, the only odds considered here are those present in the 1X2 market. Two main arguments support this decision. First, the estimated probabilities of the 1X2 outcome can either be derived or used directly from all the generated models. Secondly, and most importantly, the performance of prediction models in this market is extensively covered in existing research. Hence, the decision is made based on a consideration regarding further research on the topic, as it eases the process of comparing the methods chosen here with those presented in the academic literature. As a final note, the matches used for

evaluating the betting performance correspond to the original test set, and the performance can therefore also be compared to the presented metric scores on the 1X2 distribution.

7.2 Static Betting Performance of Prediction Models

This section presents an evaluation of the betting performance of all the generated prediction models at distinct times $t \in \{0, 5, \ldots, 90\}$ during matches. As a means of evaluating this performance, the static Kelly strategy is used to determine the bet sizes. But first, the irrational log bettor presented in Section 5.6.3 is considered in order to measure results over different intervals during the matches.

The performance of the irrational log bettor is evaluated with respect to partial Kelly strategies corresponding to $\gamma \in \{0.02, 0.05, 0.1, 0.25, 0.5\}$ as well as the full Kelly strategy. The proportion of total wealth obtained by this bettor over the investment horizon is presented in Figure 7.1. These results are based on probabilities from $LSTM_{score}$. From this plot, it becomes clear that high values of γ are not appropriate, suggesting that the uncertainty in the generated probabilities is substantial. Specifically, a high γ implies higher volatility in returns and seems to imply certain terminal ruin for the bettor in the case where its probability estimates are inferior to those of the odds suppliers. Especially is this the case for the full Kelly strategy $\gamma = 1.0$. This tendency supports the claim of Baker and McHale (2013) that the optimal $\gamma \in (0, 1)$ under uncertainty in probabilities. The same aspect can be seen for the other models in Section G.2. These results suggest that a high γ is beneficial for models generating superior estimates to those of the odds supplier, but again generally suggests certain terminal ruin for $\gamma = 1$.

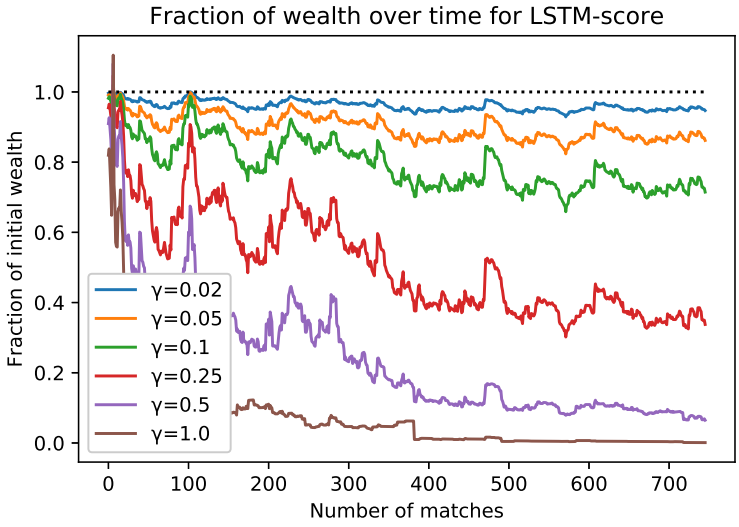


Figure 7.1: Proportion of wealth for $LSTM_{score}$ and different values of γ over the investment horizon. The wealth is the average over all time intervals.

It can be seen that the overall best and most stable performance is obtained with $\gamma = 0.02$. Based on all these results, $\gamma = 0.02$ is chosen when further evaluating the performance of the models.

Now, consider the relative performance of all predictive models. The terminal wealth obtained by the static Kelly strategy for all five predictive models and all $t \in \{0, 5, \ldots, 90\}$ over the entire

test set can be seen in Table 7.1. These results indicate that this strategy is unable to generate a positive return based on probabilities from any of the models for most t. Also note that the terminal wealth lies in the range $90 - 110\%$ of the initial wealth in most cases, indicating that the volatility in returns is rather low. Similar tables for each model and different values of γ and t can be seen in Section G.1. The behaviour seen in these tables is generally as expected, with higher values of γ increasing the volatility in returns. Note that certain combinations of model, γ and t are able to generate very high returns in excess of 100%. Apart from these observations, some interesting findings are worth mentioned based on Table 7.1.

Time	WCD_{score}	WCD_{1X2}	$LSTM_{copula}$	$LSTM_{score}$	$LSTM_{1X2}$
0	0.6923	0.8811	0.9054	0.8536	0.9412
5	0.8279	0.8945	0.8958	0.8824	1.0798
10	0.9253	0.9410	0.8642	0.9691	0.9339
15	0.9223	0.8748	0.8415	0.8888	1.1547
20	0.8530	0.8729	0.9022	0.8648	0.9077
25	0.9982	0.8682	0.8788	0.7740	0.9471
30	1.1505	1.0811	1.0263	0.9172	0.9042
35	1.0551	0.8933	0.8929	1.0108	0.9903
40	1.1460	1.0861	1.0630	1.0678	1.0040
45	0.9438	0.9184	0.8778	0.8732	1.1696
50	0.9737	1.0209	1.0135	1.0152	1.2605
55	1.0453	1.0366	0.8473	1.0425	1.1960
60	1.0731	1.1222	0.9667	0.9902	1.0849
65	0.9657	1.0520	1.0482	0.9573	0.8989
70	1.0814	1.1065	0.9829	1.0846	0.9934
75	1.0709	1.0466	1.0404	0.9375	1.0164
80	1.0171	0.9846	0.9447	0.9014	0.8890
85	0.9923	0.9886	0.9955	0.9651	0.9073
90	1.0223	1.1192	1.0052	1.0363	0.9870

Table 7.1: Wealth at the end of the investment horizon for different models and in-game time points.

The poor pre-game betting performance of WCD_{score}, with a loss of over 30%, is far below that of the other models at the same point in time. This aligns with the poor pre-game predictive performance of this model relative to the other models as well as the Sportradar odds. The substantial improvement on the interval $time \in [0, 10]$ is also in line with the predictive performance of this model. This indicates that the betting performance relies to a high degree on the quality of the probability estimates of the bettor relative to those of the supplier. This proposition is supported by an evaluation of the performance of $LSTM_{1X2}$, which indicates slight improvements in prediction metrics as well as a great betting performance at times $t \in \{15, 45, 50, 55\}$.

Certain trends seem to be present regarding the time intervals during which the irrational log bettor generates positive returns based on probabilities from the different models. These trends become more apparent in Figure 7.2, which shows the progression of wealth over a set of intervals. First, neither model is able to contribute to a positive return for the irrational log bettor at $time = 0$, although $LSTM_{1X2}$ achieves this at approximately half the interval corresponding to the investment horizon. Similar behaviour can be seen for the next time interval, except for WCD_{score} which improves greatly. Observations for both of these intervals coincide with the value of the prediction metrics, during which Sportradar has a considerably better score.

The results during the last two time intervals show considerable improvements for all models, as presented in Figure 7.2. Especially is this the case for the WCD models, as they contribute to a positive return during both intervals. Another important finding is the results based on $LSTM_{1X2}$ on the interval $time \in [35, 60]$ with a return exceeding 10%. However, the returns generated based on the latter model are poor as the matches approach full-time, which again coincides with the analysis of its prediction ability. As a general trend, all models perform better for later stages of the game, except for $LSTM_{1X2}$, again supporting the proposition that the predictive ability of all models increases relative to the Sportradar odds as matches approach full-time.

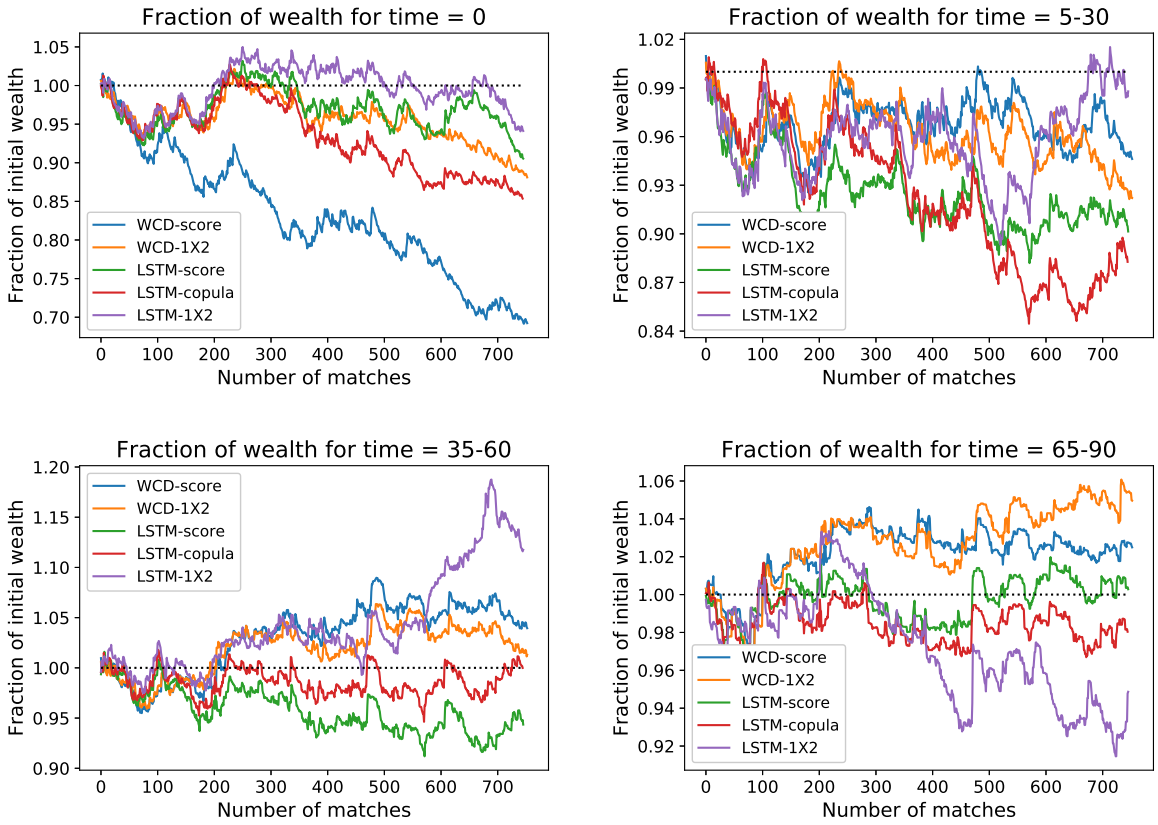


Figure 7.2: Proportion of wealth for different models and time intervals over the investment horizon. The wealth is the average over the specified time intervals.

7.3 Dynamic Strategy

The proposed dynamic betting strategy is meant as a theoretical contribution to existing literature. However, its performance in the in-game betting market is considered with respect to the generated predictive models as a means of evaluating its practical potential. In the continuation, denote this strategy the dynamic bettor.

The terminal wealth of the dynamic bettor conditional on probabilities from all models and proposed values of γ can be seen in Table 7.2. Note that WCD_{score} contributes to considerable positive returns for $\gamma \leq 0.10$, and that returns conditional on probabilities from all other models are negative for all γ.

Gamma	WCD_{score}	WCD_{1X2}	$LSTM_{copula}$	$LSTM_{score}$	$LSTM_{1X2}$
0.02	1.1321	0.9734	0.9533	0.9276	0.9805
0.05	1.3053	0.8973	0.8322	0.7773	0.8906
0.10	1.4805	0.7049	0.5663	0.4926	0.6406
0.25	1.0010	0.1631	0.0624	0.0418	0.0747
0.50	0.0522	0.0015	0.0001	0.0000	0.0001
1.00	0.0000	0.0000	0.0000	0.0000	0.0000

Table 7.2: Wealth at the end of the investment horizon for different models and values of γ.

The observations also coincide with the returns of the static strategy, indicating that a low γ is generally most suitable, while higher values have the potential to generate higher returns due to increased volatility. This is further supported by Figure 7.3, which shows the development of wealth for the dynamic bettor given different γ based on probabilities from WCD_{score}. Since the strategy seems to perform well, some of the higher values of γ are able to increase the return per bet and thus obtain a higher overall return. As previously, note that high γ causes rapid change and sudden jumps in wealth. Similar plots for the four other models can be seen in Section G.3, all of which indicate behaviour similar to what has been discussed.

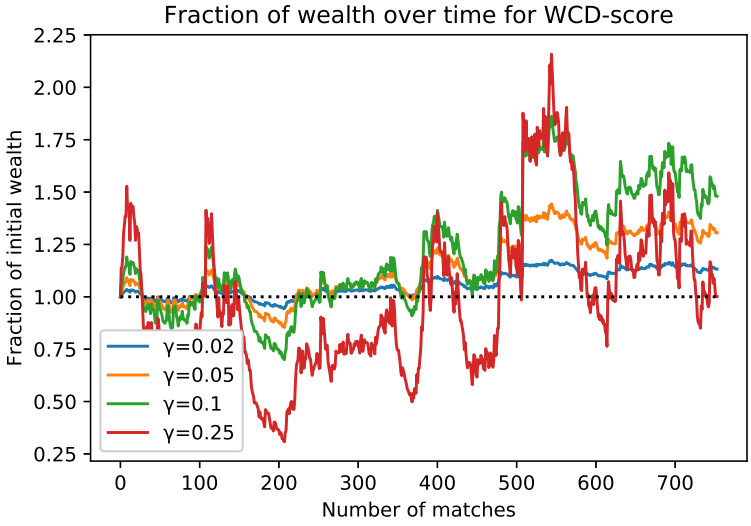


Figure 7.3: Proportion of wealth for WCD_{score} and different values of γ over the investment horizon..

A comparison between the dynamic and irrational log bettor is also appropriate. The wealth development of both can be seen Figure 7.4. Among the two, the dynamic bettor appears to accept investments with more inherent risk. In line with the arguments based on Figure 7.3, it is able to generate higher return based on WCD_{score}. However, the terminal wealth based on the other models indicates approximately equal returns for both bettors. The increased volatility can be attributed to the ability of the dynamic bettor to place multiple bets on the same outcome of each match, implying that it is more likely to allocate a higher total fraction of wealth during each match. Overall, the dynamic strategy indicates higher uncertainty in returns and higher expected return given properly estimated probabilities.

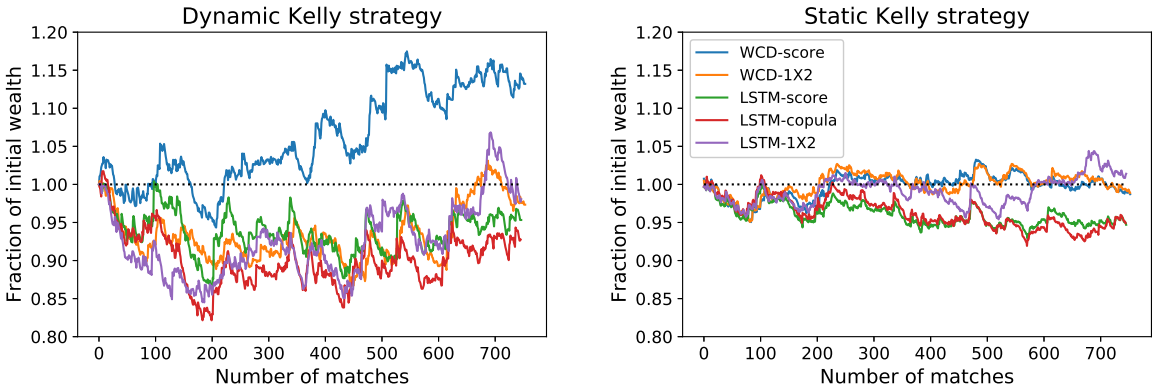


Figure 7.4: Proportion of wealth for the two betting strategies over the investment horizon. The wealth of the static strategy is the average over all time intervals.

Now, consider the timing of the investments made by the dynamic bettor. This can be observed in Table 7.3, which shows the proportion of placed fractions per time t conditional on each model. Note that for all models, the majority of fractions are placed before the half-time break. As discussed regarding both the irrational log bettor and the prediction metrics, the second half is indicated to be the preferable interval for placing bets. The observed wealth allocation pattern is likely attributed to the fact that no restrictions are placed on the chosen fractions except for the constraints than no leveraging nor short positions are allowed during a single match. Thus, the dynamic bettor is likely to invest all the assigned wealth γW_0 during the early stages of the matches.

Based on this result, a scaling γ_t of the fractions placed at each time t seems appropriate, ensuring that bets can be placed throughout the match. Furthermore, comparing the performance of the dynamic and irrational log bettor during the first half of the match indicates that the dynamic strategy outperforms the static one. This supports the argument that the dynamic strategy has the highest potential and is likely to converge to higher growth than the static strategy.

Time	WCD_{score}	WCD_{1X2}	$LSTM_{copula}$	$LSTM_{score}$	$LSTM_{1X2}$
0	0.1964	0.1453	0.1731	0.1682	0.1681
10	0.2576	0.1849	0.1448	0.1151	0.1876
20	0.1546	0.1947	0.2267	0.1839	0.1917
30	0.1309	0.1379	0.1523	0.1434	0.1440
40	0.1020	0.1446	0.1033	0.1818	0.1560
50	0.0894	0.0771	0.0784	0.0844	0.0933
60	0.0315	0.0624	0.0581	0.0551	0.0272
70	0.0238	0.0275	0.0284	0.0330	0.0184
80	0.0094	0.0189	0.0273	0.0262	0.0104
90	0.0043	0.0066	0.0076	0.0089	0.0034

Table 7.3: Proportion of the total amount of fractions placed at different time steps for each model.

As a final note, the high volatility in terminal returns observed in these results is likely to decrease over longer investment horizons. This is because the Kelly criterion consistently chooses more

risky positions than a logarithmic utility function entails given that the number of bets is not sufficiently large for Theorem 3 to apply (Maclean et al., 2011).

7.4 Discussion

A brief summary of the findings from the two previous sections is presented here. The emphasis is placed on the aspects deemed relevant for a discussion surrounding RQ3, which is restated below.

RQ3 *How do the generated prediction models perform in the in-game betting market when subject to a theoretically sound betting strategy, and when the live-odds estimated by Sportradar are taken as the supply in the market?*

For the purpose of this discussion, the assumed great standard of Sportradar's estimates suggests that exiting the market on par at the end of the investment horizon is deemed acceptable. The general trend for both the static and dynamic betting strategy is a small negative return, although predictions by all models can be used to generate positive returns for at least some combinations of t and γ. The times at which these positive returns occur seem to coincide well with those at which the predictive ability of the generated probability estimates compare favourably to that of Sportradar's implied probabilities. Furthermore, the proposed dynamic strategy yields more volatile returns than its static counterpart due to the allowance of multiple bets on a single outcome. This aspect may also explain the superior returns generated by the dynamic strategy when the probability estimates are sufficiently accurate. Among the generated models, WCD_{score} is strongly suggested to be most applicable for use in the in-game betting market when subject to the dynamic strategy.

The hypothesis that the generated models are sufficient for generating a positive return in the in-game betting market when subject to a proper wealth allocation strategy should not be rejected. As the returns are indicated to be highly dependent on the predictive ability of the models, it seems reasonable to train the models without knowledge of the betting market.

Chapter 8

Conclusion

An important topic of this thesis was the relative performance of statistical models on the task of predicting the outcome of football matches in the FA English Premier League. A comparison between these models was conducted to evaluate a set of initial hypotheses represented by three posed research questions. The hypothesis that parametric count distributions impose some unwanted limitations on the task of modelling relationships in football was represented by RQ1. The motivation for posing RQ2 was the hypothesis that information about the entire scoreline distribution is essential for obtaining accurate estimates of the 1X2 distribution. To answer these research questions, five prediction models were generated based on an LSTM network and a Weibull count distribution.

The third hypothesis, represented by RQ3, was motivated by the assumption that predictive models can generate positive returns in the in-game betting market without using market information in the estimation procedure. To accurately assess the hypothesis, two theoretically founded wealth allocation strategies were utilized. One of these strategies is proposed by the authors as a theoretical contribution to the existing literature. It was shown to be an optimal dynamic strategy for an investor acting rationally according to its logarithmic utility function under a large set of simplifying assumptions, and where the investment problem concerns wagering on mutually exclusive outcomes of a single event.

Regarding RQ1, the architecture based on the Weibull count distribution was indicated to generate the best predictors of the true class. However, the results strongly suggest that the LSTM architecture can generate models that better capture the inherent ordinal structure of the scorelines in football matches in the EPL. The latter result indicates that the hypothesis represented by RQ1 hold, although the effect of this is not reflected in the relative predictive ability. As for RQ2, the scoreline models were indicated as slightly better models of the 1X2 distribution than their 1X2 equivalents, supporting the hypothesis that information about the scoreline distribution is of the essence for predicting the winner of a football match.

The probabilities generated by the prediction models indicated an ability to occasionally generate positive returns in the market when subject to a static betting strategy. The dynamic strategy proposed by the authors was able to achieve higher returns in comparable situations but was subject to higher volatility. The results also indicated that positive returns mostly coincided with good predictive ability of the generated models compared to the implied probabilities of Sportradar. However, neither combination of strategy and predictive model was able to consistently generate positive returns.

Chapter 9

Recommendations for Further Research

A foundation for a valid comparison of the predictive ability of the generated models is that they originate from the same scientific setup. A significant drawback of this setup is the chosen selection and validation approach. A recommended approach was presented in Section 5.3.2, and it is highly recommended that this approach is chosen for future work on the topic given that a sufficient amount of time and computational resources are available to the researcher. Another potential weakness of the generated models is that they do not utilize the information available in the betting market nor common knowledge, such as twitter post and prediction polls. Since the wisdom of crowds is suggested to have a great prediction power in existing research, such information should ideally be included in the feature set. Furthermore, the results indicated that the choice of sequence length can considerably alter the performance of LSTM networks. Based on this, it is suggested to include the sequence length as a hyperparameter in the model selection procedure.

Although most of the generated models estimate the entire scoreline distribution, the only market used to test the betting performance in this thesis was the 1X2 market. An interesting topic for further research is to extend the analysis presented here to other markets for mutually exclusive outcomes. Regarding the proposed betting strategy, relaxation of the requirement of mutually exclusive outcomes is considered an important extension, as one should allow for bets on multiple markets in the same match. Allowing for bets in simultaneous matches is also of interest. An approach in this regard is presented in Section 5.6.2. Another restriction inherent in the betting results is the fixed set of values for the partial Kelly parameter γ. Ideally, γ should be optimized for different intervals during the course of a match. This proposal is based on the observation that the dynamic strategy invested all or most of its wealth in the early stages of the match, even though the estimated probabilities were shown to be superior for later stages.

As a concluding remark, both of the evaluated model architectures were indicated to generate models with a similar predictive performance to that observed for implied probabilities from Sportradar. This indicates that these architectures should be viable options for further research on the task of predicting the outcome of football matches.

References

Agrawal, P., Banga, S., Pathak, N., Goel, S., and Kaushik, S. (2018). Automated music composition using LSTM. In *INDIACom 2018, International Conference on "Computing for Sustainable Global Development"*, pages 1395–1399.

Alba, A. C., Agoritsas, T., Walsh, M., Hanna, S., Iorio, A., Devereaux, P. J., McGinn, T., and Guyatt, G. (2017). Discrimination and calibration of clinical prediction models: Users' guides to the medical literature. *JAMA*, 318:1377–1384.

Algoet, P. H. and Cover, T. M. (1988). Asymptotic optimality and asymptotic equipartition properties of log-optimum investment. *The Annals of Probability*, 16:876–898.

Arabzad, S. M., Araghi, M., Soheil, S.-N., and Ghofrani, N. (2014). Football match results prediction using artificial neural networks; the case of Iran Pro League. *International Journal of Applied Research on Industrial Engineering*, 1:159–179.

Asif, M. and McHale, I. (2016). In-play forecasting of win probability in one-day international cricket: A dynamic logistic regression model. *International Journal of Forecasting*, 32:34–43.

Baboota, R. and Kaur, H. (2019). Predictive analysis and modelling football results using machine learning approach for English Premier League. *International Journal of Forecasting*, 35:741–755.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. Available at arXiv: `https://arxiv.org/abs/1409.0473`. [Online; accessed <17.2.2019>].

Baker, R. and McHale, I. (2013). Optimal betting under parameter uncertainty: Improving the Kelly criterion. *Decision Analysis*, 10:189–199.

Bellman, R. and Kalaba, R. (1957). On the role of dynamic programming in statistical communication theory. *IRE Transactions on Information Theory*, 3:197–203.

BetBrain.com (2019). `https://no.betbrain.com/`. [Online; accessed <7.6.2019>].

BetStudy.com (2019). `https://www.betstudy.com`. [Online; accessed <20.3.2019>].

Bierce, A. and Ford, J. (2010). *The Devil's Dictionary of Ambrose Bierce - Complete and Unabridged - Special Edition*. El Paso Norte Press.

Boshnakov, G., Kharrat, T., and McHale, I. G. (2017). A bivariate Weibull count model for forecasting association football scores. *International Journal of Forecasting*, 33:458–466.

British Council (2015). Playing the game: The soft power of sport. `https://www.britishcouncil.org/organisation/policy-insight-research/insight/playing-game-soft-power-sport"`. [Online; accessed <14.12.2018>].

Bunker, R. P. and Thabtah, F. (2019). A machine learning framework for sport result prediction. *Applied Computing and Informatics*, 15:27–33.

BusinessWire (2016). Top 4 emerging trends impacting the sports betting market: Technavio. https://www.businesswire.com/news/home/20161019005427/en/Top-4-Emerging-Trends-Impacting-Sports-Betting?fbclid=IwAR2p3JWfHqygpc3stuMaccNxT_rbFWCYYsJ2dav4A9iH-dZZ5LBaLogAbmw. [Online; accessed <14.12.2018>].

Cao, R., Liu, Z., Wang, S., and Zhou, W. (2017). Multivariate volatility regulated Kelly strategy: A superior choice in low correlated portfolios. *Theoretical Economics Letters*, 07:1453–1472.

Chapman, S. (2006). The Kelly criterion for spread bets. *IMA Journal of Applied Mathematics*, 72:43–51.

Cho, K., van Merriënboer, B., Gulcehre, C., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734.

Chollet, F. et al. (2015). Keras. https://keras.io.

Church, K. W. and Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16:22–29.

Commenges, D. (2015). Information theory and statistics: An overview. Available at arXiv: https://arxiv.org/abs/1511.00860. [Online; accessed <11.3.2019>].

Constantinou, A., Fenton, N., and Neil, M. (2012). pi-football: A bayesian network model for forecasting association football match outcomes. *Knowledge-Based Systems*, 36:332–339.

Constantinou, A., Fenton, N., and Neil, M. (2013). Profiting from an inefficient association football gambling market: Prediction, risk and uncertainty using bayesian networks. *Knowledge-Based Systems*, 50:60–86.

Conway, J. B. (1985). *A Course in Functional Analysis*, page 8. Springer-Verlag New York.

Cooijmans, T., Ballas, N., Laurent, C., and Courville, A. C. (2016). Recurrent batch normalization. Available at arXiv: https://arxiv.org/abs/1603.09025. [Online; accessed <21.2.2019>].

Corrigan, D. (2019). Atletico Madrid boss Diego Simeone: Real Madrid 'were better than us again'. https://africa.espn.com/football/atletico-madrid/story/2882273/atletico-boss-diego-simeone-says-real-madrid-were-better-than-us-again. [Online; accessed <7.6.2019>].

Croxson, K. and Reade, J. J. (2013). Information and efficiency: Goal arrival in soccer betting. *The Economic Journal*, 124:62–91.

Daunhawer, I., Schoch, D., and Kosub, S. (2017). Biases in the football betting market. Available at SSRN: https://ssrn.com/abstract=2977118. [Online; accessed <7.6.2019>].

Davoodi, E. and Khanteymoori, A. (2010). Horse racing prediction using artificial neural networks. In *Recent Advances in Neural Networks, Fuzzy Systems & Evolutionary Computing*, pages 155–160.

Dixon, M. and Robinson, M. (1998). A birth process model for association football matches. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 47:523–538.

Dixon, M. J. and Coles, S. G. (1997). Modelling association football scores and inefficiencies in the football betting market. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 46:265–280.

Easyodds.com (2019). `https://easyodds.com/`. [Online; accessed <7.6.2019>].

Eck, D. and Schmidhuber, J. (2019). A first look at music composition using LSTM recurrent neural networks. Technical report, Instituto Dalle Molle di studi sull' intelligenza artificiale.

Elo, A. E. (1978). *The rating of chess players, past and present*. Arco Publishing, New York.

Epstein, E. S. (1969). A scoring system for probability forecasts of ranked categories. *Journal of Applied Meteorology*, 8:985–987.

Feng, G., G. Polson, N., and Xu, J. (2016). The market for English Premier League (EPL) odds. *Journal of Quantitative Analysis in Sports*, 12:167–178.

fifaindex.com (2019). `https://www.fifaindex.com`. [Online; accessed <20.3.2019>].

Fischer, M. and Köck, C. (2012). Constructing and generalizing given multivariate copulas: A unifying approach. *Statistics: A Journal of Theoretical and Applied Statistics*, 46:1–12.

Fitt, A. D. (2008). Markowitz portfolio theory for soccer spread betting. *IMA Journal of Management Mathematics*, 20:167–184.

Football-Data.co.uk (2019). `http://www.football-data.co.uk`. [Online; accessed <13.3.2019>].

Franck, E. P., Verbeek, E., and Nüesch, S. (2009). Prediction accuracy of different market structures - bookmakers versus a betting exchange. *International Journal of Forecasting*, 26:448–459.

Goddard, J. (2005). Regression models for forecasting goals and match results in association football. *International Journal of Forecasting*, 21:331–340.

Godin, F., Zuallaert, J., Vandersmissen, B., De Neve, W., and Van de Walle, R. (2014). Beating the bookmakers: Leveraging statistics and Twitter microposts for predicting soccer results. `https://www.researchgate.net/publication/264977879_Beating_the_Bookmakers_Leveraging_Statistics_and_Twitter_Microposts_for_Predicting_Soccer_Results`. [Online; accessed <05.12.2018>].

Grantham, B. (2018). `https://github.com/BradleyGrantham/pl-predictions-using-fifa`. [Online; accessed <18.3.2019>].

Graves, A. (2013). Generating sequences with recurrent neural networks. Available at arXiv: `https://arxiv.org/abs/1308.0850`. [Online; accessed <11.3.2019>].

Griffin, P. A. (1984). Different measures of win rate for optimal proportional betting. *Management Science*, 30:1540–1547.

Griffiths, M. (1990). The cognitive psychology of gambling. *Journal of gambling studies / co-sponsored by the National Council on Problem Gambling and Institute for the Study of Gambling and Commercial Gaming*, 6:31–42.

Guyon, I., Gunn, S., Nikravesh, M., and Zadeh, L. (2006). *Feature extraction: foundations and applications*. Springer, Berlin, Heidelberg.

Hakansson, N. H. (1971). On optimal myopic portfolio policies, with and without serial correlation of yields. *The Journal of Business*, 44:324–334.

Haugh, M. (2016). An introduction to copulas. http://www.columbia.edu/~mh2078/QRM/Copulas.pdf. [Online; accessed <02.12.2018>].

Hayden, B. and Platt, M. (2009). The mean, the median, and the St. Petersburg Paradox. *Judgment and Decision Making*, 4:256–272.

Hochreiter, S., Bengio, Y., Frasconi, P., and Schmidhuber, J. (2001). Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. In Kolen, J. and Kremer, S., editors, *Field Guide to Dynamical Recurrent Networks*, pages 237–243. IEEE Press.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9:1735–80.

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4:251–257.

Hsieh, C., Barmish, B. R., and Gubner, J. A. (2018). At what frequency should the Kelly bettor bet? In *2018 Annual American Control Conference (ACC)*, pages 5485–5490.

Hvattum, L. M. (2013). Analyzing information efficiency in the betting market for association football league winners. *The Journal of Prediction Markets*, 7:55–70.

Hvattum, L. M. (2017). Ordinal versus nominal regression models and the problem of correctly predicting draws in soccer. *International Journal of Computer Science in Sport*, 16:50–64.

Hvattum, L. M. and Arntzen, H. (2010). Using ELO ratings for match result prediction in association football. *International Journal of Forecasting*, 26:460–470.

Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. Available at arXiv: https://arxiv.org/abs/1502.03167. [Online; accessed <21.2.2019>].

Jones, E., Oliphant, T., Peterson, P., et al. (2001). SciPy: Open source scientific tools for Python. https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html. [Online; accessed <06.12.2018>].

Jordan, M. I. (1986). Attractor dynamics and parallelism in a connectionist sequential machine. In *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, pages 531–546. Hillsdale, NJ: Erlbaum.

Joseph, A., Fenton, N., and Neil, M. (2006). Predicting football results using bayesian nets and other machine learning techniques. *Knowledge-Based Systems*, 19:544–553.

Kadane, J. B. (2011). Partial-Kelly strategies and expected utility: Small-edge asymptotics. *Decision Analysis*, 8:4–9.

Kallberg, J. G. and Ziemba, W. T. (1984). Mis-specifications in portfolio selection problems. In *Risk and Capital*, pages 74–87. Springer Berlin Heidelberg.

Karlis, D. and Ntzoufras, I. (2003). Analysis of sports data using bivariate Poisson models. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 52:381–393.

Kelly, J. (1956). A new interpretation of information rate. *IRE Transactions on Information Theory*, 2:185–189.

Killick, E. and Griffiths, M. (2018). In-play sports betting: A scoping study. *International Journal of Mental Health and Addiction*, pages 1–40. Available at: https://doi.org/10.1007/s11469-018-9896-6. [Online; accessed <23.10.2018>].

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations*. Available at: `http://arxiv.org/abs/1412.6980`. [Online; accessed <21.2.2019>].

Koopman, S. J. and Lit, R. (2012). A dynamic bivariate Poisson model for analysing and forecasting match results in the English Premier League. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 178:167–196.

Kusyszyn, I. (1984). The psychology of gambling. *The ANNALS of the American Academy of Political and Social Science*, 474:133–145.

Laurent, C., Pereyra, G., Brakel, P., Zhang, Y., and Bengio, Y. (2016). Batch normalized recurrent neural networks. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2657–2661.

Loève, M. (1977). *Elementary Probability Theory*, page 14. Springer New York, New York, NY.

Lundgren, J., Rönnqvist, M., and Värbrand, P. (2012). *Optimization*, pages 29–31, 245, 248. Studentlitteratur AB.

Maclean, L. C., Thorp, E., and Ziemba, W. (2011). Long-term capital growth: The good and bad properties of the Kelly and fractional Kelly capital growth criteria. *Quantitative Finance*, 10:681–687.

MacLean, L. C., Ziemba, W., and Blazenko, G. (1992). Growth versus security in dynamic investment analysis. *Management Science*, 38:1562–1585.

Maher, M. (1982). Modelling association football scores. *Statistica Neerlandica*, 36:109–118.

Markowitz, H. (1952). Portfolio selection. *Journal of Finance*, 7:77–91.

McCabe, A. and Trevathan, J. (2008). Artificial intelligence in sports prediction. In *Fifth International Conference on Information Technology: New Generations (itng 2008)*, pages 1194–1197.

McHale, I. and Scarf, P. (2007). Modelling soccer matches using bivariate discrete distributions with general dependence structure. *Statistica Neerlandica*, 61:432–445.

McShane, B., Adrian, M., Bradlow, E., and Fader, P. (2008). Count models based on Weibull interarrival times. *Journal of Business and Economic Statistics*, 26:369–378.

Merkle, E. and Steyvers, M. (2013). Choosing a strictly proper scoring rule. *Decision Analysis*, 10:292–304.

Mnih, V., Kavukcuoglu, K., et al. (2013). Playing Atari with deep reinforcement learning. Available at arXiv: `https://arxiv.org/abs/1312.5602`. [Online; accessed <15.2.2019>].

Moffitt, S. (2017). Gambling for quants, part 1: A simple fractional betting system. Available at SSRN: `https://ssrn.com/abstract=2914620`. [Online; accessed <14.5.2019>].

Mossin, J. (1968). Optimal multiperiod portfolio policies. *The Journal of Business*, 41:215–229.

Murphy, A. (1970). The ranked probability score and the probability score: A comparison. *Monthly Weather Review*, 98:917–924.

Nakharutai, N., Caiado, C. C. S., and Troffaes, M. C. M. (2019). Evaluating betting odds and free coupons using desirability. *International Journal of Approximate Reasoning*, 106:128–145.

Nekrasov, V. (2014). Kelly criterion for multivariate portfolios: A model-free approach. Available at SSRN: https://ssrn.com/abstract=2259133. [Online; accessed <14.5.2019>].

Nelsen, R. (2006). *An Introduction to Copulas, second edition*. Springer-Verlag New York.

Nevo, D. and Ritov, Y. (2012). Around the goal: Examining the effect of the first goal on the second goal in soccer using survival analysis methods. *Journal of Quantitative Analysis in Sports*, 9:165–177.

Noon, E. (2014). *Extending Kelly Staking Strategies to Peer-to-Peer Betting Exchanges*. PhD thesis, Imperial College London. Available at: https://www.doc.ic.ac.uk/~wjk/publications/noon-2014.pdf. [Online; accessed <16.5.2019>].

Nyquist, R. and Pettersson, D. (2017). Football match prediction using deep learning: Recurrent neural network applications. Master's thesis, Chalmers University of Technology, Gothenburg, Sweden. Available at: http://publications.lib.chalmers.se/records/fulltext/250411/250411.pdf. [Online; accessed <21.2.2019>].

Oddschecker.com (2019). https://www.oddschecker.com/. [Online; accessed <7.6.2019>].

O'Shaughnessy, D. (2012). Optimal exchange betting strategy for win-draw-loss markets. In *Proceedings of the Eleventh Australasian Conference on Mathematics and Computers in Sport*, pages 62–66.

Owramipur, F., Eskandarian, P., and Sadat Mozneb, F. (2013). Football result prediction with bayesian network in Spanish league-Barcelona team. *International Journal of Computer Theory and Engineering*, pages 812–815.

Ozanian, M. (2018). Sportradar is the pick and shovel maker of sports betting. https://www.forbes.com/sites/mikeozanian/2018/08/02/sportradar-is-the-pick-and-shovel-maker-of-sports-betting. [Online; accessed <7.6.2019>].

Paton, D., Williams, L. V., and Smith, M. A. (2006). Market efficiency in person-to-person betting. *Economica*, 73:673–689.

Peeters, T. (2018). Testing the wisdom of crowds in the field: Transfermarkt valuations and international soccer results. *International Journal of Forecasting*, 34:17–29.

Peterson, Z. (2018). Kelly's criterion in portfolio optimization: A decoupled problem. *Journal of Investment Strategies*, 7:53–76.

Pudaruth, S., Jogeeah, M., and Kumar Chandoo, A. (2015). Using artificial neural networks to predict winners in horseraces: A case study at the champs de mars. In *IST-Africa 2015 Conference Proceedings*, pages 1–8. Available at: https://www.researchgate.net/publication/228847950_Horse_racing_prediction_using_artificial_neural_networks. [Online; accessed <11.3.2019>].

Rayner, G. and Brown, O. (2019). Leicester City win Premier League and cost bookies biggest ever payout. https://www.telegraph.co.uk/news/2016/05/02/leicester-city-win-premier-league-and-cost-bookies-biggest-ever/. [Online; accessed <11.5.2019>].

Rue, H. and Salvesen, O. (2000). Prediction and retrospective analysis of soccer matches in a league. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 49:399–418.

Scarf, P. and Shi, X. (2008). The importance of a match in a tournament. *Computers & Operations Research*, 35:2406–2418.

Schauberger, G. and Groll, A. (2018). Predicting matches in international football tournaments with random forests. *Statistical Modelling*, 18:460–482.

Schiefler, L. (2019). Football club Elo ratings. `https://www.clubelo.com`. [Online; accessed <13.3.2019>].

Schumaker, R., Jarmoszko, T., and Labedz, C. (2016). Predicting wins and spread in the Premier League using a sentiment analysis of Twitter. *Decision Support Systems*, 88:76–84.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423.

Silver, D., Huang, A., et al. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, 529:484–489.

Silver, N. (2009). A guide to ESPN's SPI ratings. `http://www.espn.com/world-cup/story/_/id/4447078/ce/us/guide-espn-spi-ratings`. [Online; accessed <22.2.2019>].

Silver, N. and Boice, J. (2018). How our club soccer projections work. `https://fivethirtyeight.com/features/how-our-club-soccer-projections-work/`. [Online; accessed <22.2.2019>].

Sklar, A. (1959). Fonctions de répartition à n dimensions et leurs marges. *Publications de l'Institut Statistique de l'Université de Paris*, 8:229–231.

Smoczynski, P. and Tomkins, D. (2010). An explicit solution to the problem of optimizing the allocations of a bettor's wealth when wagering on horse races. *The Mathematical Scientist*, 35:10–17.

SportsBettingDime.com (2018). The size and increase of the global sports betting market. `https://www.sportsbettingdime.com/guides/finance/global-sports-betting-market/`. [Online; accessed <14.12.2018>].

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

Stekler, H. O., Sendor, D., and Verlander, R. (2010). Issues in sports forecasting. *International Journal of Forecasting*, 26:606–621.

Steyerberg, E., Vickers, J., et al. (2010). Assessing the performance of prediction models: A framework for traditional and novel measures. *Epidemiology (Cambridge, Mass.)*, 21:128–138.

Sutskever, I., Vinyals, O., and V. Le, Q. (2014). Sequence to sequence learning with neural networks. *Advances in Neural Information Processing Systems*, 27:3104–3112.

Taskjelle, T. (2017). Diagram of an artificial neural network. `https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network`. [Online; accessed <14.3.2019>].

Tetlock, P. (2015). Edge master class 2015: A short course in superforecasting, class ii tournaments: Prying open closed minds in unnecessarily polarized debates. `https://www.edge.org/conversation/philip_tetlock-edge-master-class-2015-a-short-course-in-superforecasting-class-ii`. [Online; accessed <15.05.2019>].

Thorp, E. (1975). Portfolio choice and the Kelly criterion. In *Stochastic Optimization Models in Finance*, pages 599–619. Academic Press.

Thorp, E. O. (2008). Chaper 9 - the Kelly criterion in blackjack sports betting, and the stock market. In *Handbook of Asset and Liability Management*, pages 385 – 428. North-Holland, San Diego.

Tibshirani, R., Hastie, T., and Friedman, J. (2009). *The Elements of Statistical Learning, 2.Ed.*, pages 32,61–73,249–254, 119–127,392–397. Springer.

Tibshirani, R., Hastie, T., James, G., and Witten, D. (2017). *Introduction to Statistical Learning, 7.Printing*, pages 37–38,176–184,206–209. Springer.

Transfermarkt.com (2018). `https://www.transfermarkt.com`. [Online; accessed <07.12.2018>].

Ulmer, B. and Fernandez, M. (2014). Predicting soccer match results in the English Premier League. `http://cs229.stanford.edu/proj2014/Ben%20Ulmer,%20Matt%20Fernandez,%20Predicting%20Soccer%20Results%20in%20the%20English%20Premier%20League.pdf`. [Online; accessed <06.12.2018>].

van Gerven, M. and Bohte, S. (2017). Editorial: Artificial neural networks as models of neural information processing. *Frontiers in Computational Neuroscience*, 11:114.

Veličković, P. (2017). Complete collection of my PGF/TikZ figures. `https://github.com/PetarV-/TikZ/tree/master/Long%20short-term%20memory`. [Online; accessed <14.3.2019>].

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015). Show and tell: A neural image caption generator. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3156–3164.

Volf, P. (2009). A random point process model for the score in sport matches. *IMA Journal of Management Mathematics*, 20:121–131.

von Neumann, J. and Morgenstern, O. (1944). *Theory of Games and Economic Behavior*. Princeton University Press.

Wang, D. and Nyberg, E. (2015). A long short-term memory model for answer sentence selection in question answering. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers)*, pages 707–712.

Wood, R. and Griffiths, M. (2008). The psychology of lottery gambling. *International Gambling Studies*, 1:27–45.

Wu, M.-E., Tsai, H.-H., Tso, R., and Weng, C.-Y. (2016). An adaptive Kelly betting strategy for finite repeated games. In *Genetic and Evolutionary Computing*, pages 39–46. Springer International Publishing.

Zaremba, W., Sutskever, I., and Vinyals, O. (2014). Recurrent neural network regularization. Available at arXiv: `https://arxiv.org/abs/1409.2329`. [Online; accessed <17.2.2019>].

Appendices

Appendix A

Evaluating the Choice of Count Distribution

A.1 Chi-squared Hypothesis Test - Weibull Count vs. Poisson

	Home Team		Away Team	
Time	Poisson	Weibull	Poisson	Weibull
0.0	0.003	0.336	0.000	0.726
5.0	0.017	0.489	0.000	0.613
10.0	0.012	0.268	0.000	0.579
15.0	0.000	0.075	0.000	0.677
20.0	0.004	0.222	0.000	0.640
25.0	0.001	0.163	0.000	0.926
30.0	0.004	0.453	0.000	0.910
35.0	0.015	0.487	0.000	0.745
40.0	0.002	0.017	0.000	0.698
45.0	0.072	0.219	0.000	0.828
50.0	0.244	0.365	0.000	0.969
55.0	0.524	0.826	0.002	0.907
60.0	0.662	0.903	0.001	0.636
65.0	0.691	0.800	0.013	0.743
70.0	0.430	0.487	0.269	0.632
75.0	0.398	0.205	0.681	0.411

Table A.1: p-values from chi-squared hypothesis tests with H0: Fitted distribution = data.

A.2 Distribution Plots

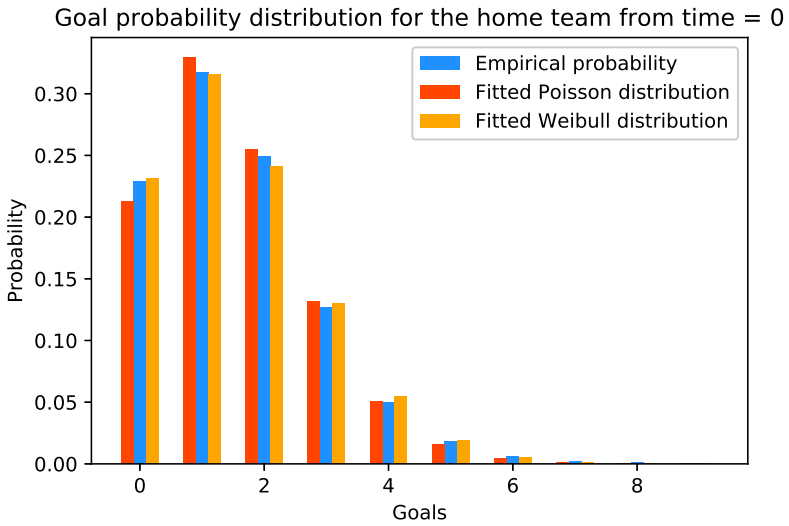


Figure A.1: Probability for the number of goals scored by the home team based on empirical data, a fitted Poisson distribution and a fitted Weibull distribution.

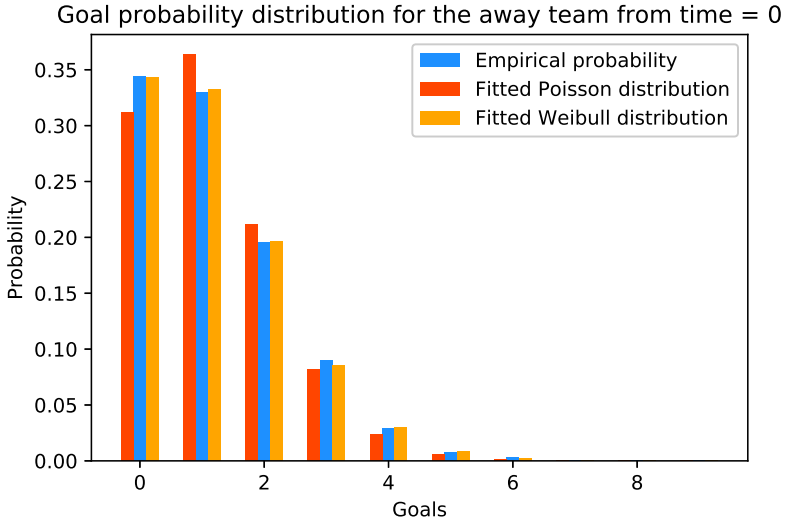


Figure A.2: Probability for the number of goals scored by the away team based on empirical data, a fitted Poisson distribution and a fitted Weibull distribution.

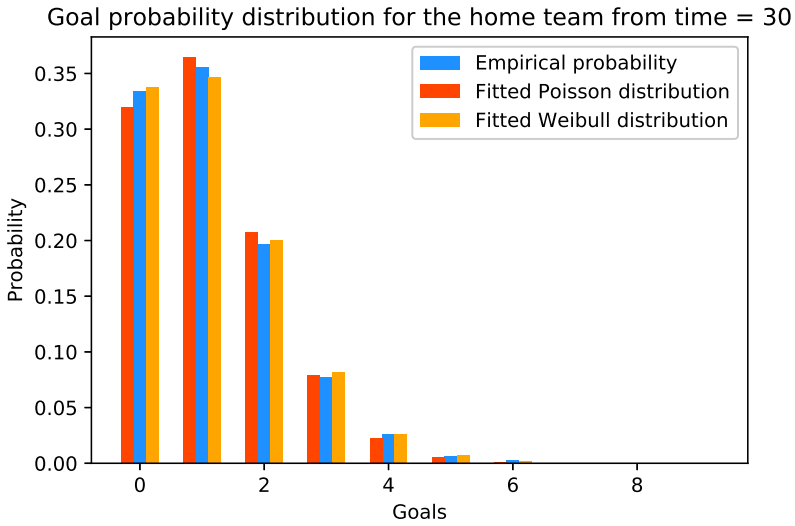


Figure A.3: Probability for the number of goals scored by the home team based on empirical data, a fitted Poisson distribution and a fitted Weibull distribution.

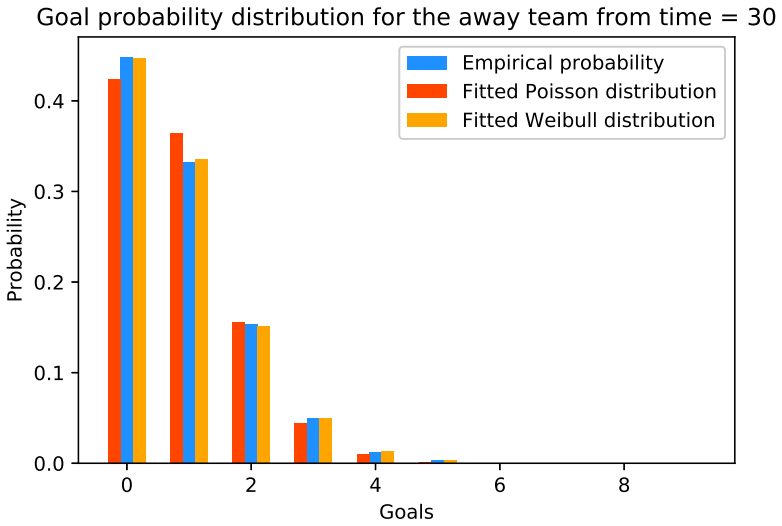


Figure A.4: Probability for the number of goals scored by the away team based on empirical data, a fitted Poisson distribution and a fitted Weibull distribution.

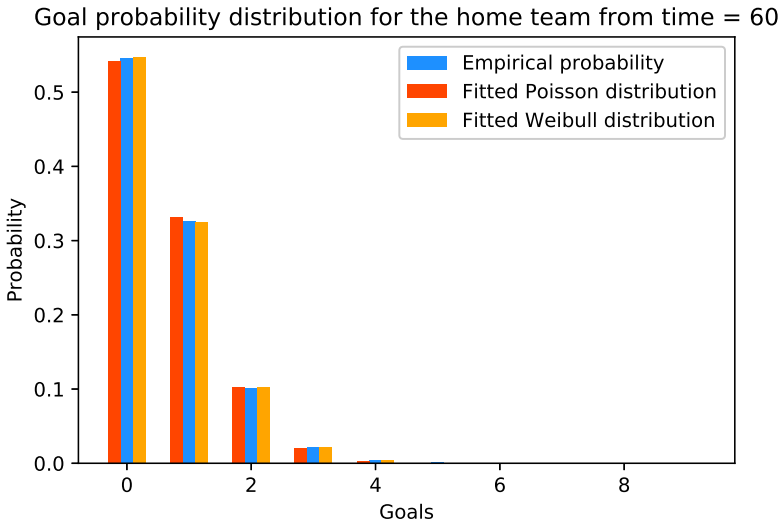


Figure A.5: Probability for the number of goals scored by the home team based on empirical data, a fitted Poisson distribution and a fitted Weibull distribution.

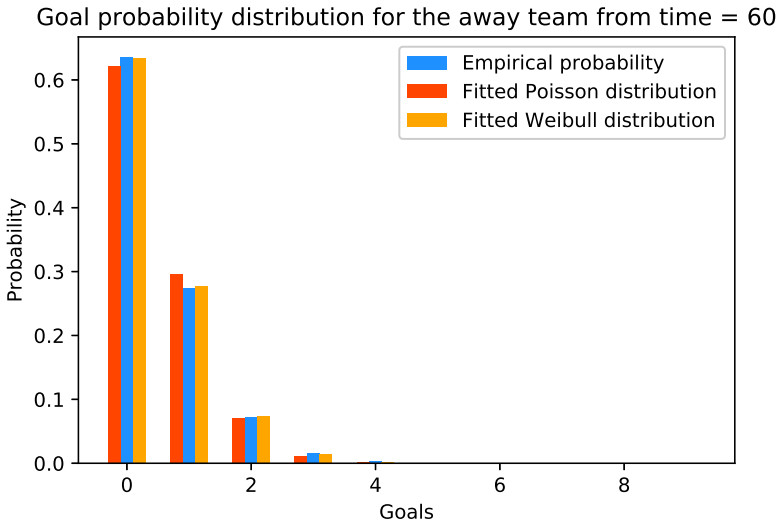


Figure A.6: Probability for the number of goals scored by the away team based on empirical data, a fitted Poisson distribution and a fitted Weibull distribution.

Appendix B

Table with all Features

Feature	Difference	Seasonal	In-game	Variations	Num. features
Away indicator					1
Corner kicks	X		X		3
Draws	X	X			3
Elo rating	X				3
FIFA rating	X				3
Form(w)	X	X		$w \in \{0.05, 0.1, \ldots, 0.5\}$	30
Goal difference	X	X			3
Goal difference/match	X	X			3
Goals conceded	X	X			3
Goals conceded/match	X	X			3
Goals scored	X	X			3
Goals scored/match	X	X			3
Home indicator					1
Importance 1(w)	X	X		$w \in \{0, 0.25, 0.5, 1, 2, 3\}$	18
Importance 18(w)	X	X		$w \in \{0, 0.25, 0.5, 1, 2, 3\}$	18
Importance 4(w)	X	X		$w \in \{0, 0.25, 0.5, 1, 2, 3\}$	18
Importance 5(w)	X	X		$w \in \{0, 0.25, 0.5, 1, 2, 3\}$	18
Losses	X	X			3
Match num 0-19	X	X			3
Match num 0-4	X	X			3
Match num 0-9	X	X			3
Match num 10-14	X	X			3
Match num 10-19	X	X			3
Match num 15-19	X	X			3
Match num 20-24	X	X			3
Match num 20-29	X	X			3
Match num 20-38	X	X			3
Match num 25-29	X	X			3
Match num 30-34	X	X			3
Match num 30-38	X	X			3
Match num 35-38	X	X			3
Match num 5-9	X	X			3
Matches	X	X			3
Matches left	X	X			3
Offsides	X		X		3
Points	X	X			3
Points/match	X	X			3
Position	X	X			3
Red cards	X		X		3
Score	X		X		3
Shots off goal	X		X		3
Shots on goal	X		X		3
Streak ew tw(l)	X	X		$l \in \{1, 2, \ldots, 10\}$	30
Streak ew(l)	X	X		$l \in \{1, 2, \ldots, 10\}$	30
Streak tw(l)	X	X		$l \in \{1, 2, \ldots, 10\}$	30
Streak(l)	X	X		$l \in \{1, 2, \ldots, 10\}$	30
Wins	X	X			3
Yellow cards	X		X		3
Total					335

Table B.1: All available features and their properties.

Appendix C

Grid Search

Scoreline				1X2			
Pre-game		In-game		Pre-game		In-game	
d	Loss	d	Loss	d	Loss	d	Loss
15	2.8947	37	2.2615	10	1.0172	10	0.8478
16	2.8952	48	2.2618	11	1.0176	11	0.8504
32	2.8953	17	2.2621	12	1.0214	12	0.8505
42	2.8959	43	2.2625	8	1.0217	13	0.8539
50	2.8960	10	2.2631	20	1.0228	16	0.8540

Table C.1: Optimal values of d from grid search for the WCD models.

Scoreline				Copula				1X2			
Pre-game		In-game		Pre-game		In-game		Pre-game		In-game	
d	Loss	d	Loss	d	Loss	d	Loss	d	Loss	d	Loss
14	2.9164	24	2.4210	14	1.4483	24	1.1734	17	1.0160	17	0.8443
8	2.9200	20	2.4213	11	1.4486	18	1.1762	18	1.0179	45	0.8452
13	2.9203	22	2.4241	20	1.4492	26	1.1779	16	1.0187	41	0.8481
9	2.9212	18	2.4273	17	1.4493	40	1.1803	19	1.0204	19	0.8505
15	2.9213	38	2.4292	18	1.4495	30	1.1804	20	1.0213	39	0.8523

Table C.2: Optimal values of d from grid search for the LSTM models.

Appendix D

Feature Selection

Feature
Diff Form 0.45
Diff Importance 1 pw0.25
Diff Importance 4 pw0.25
Diff Importance 5 pw0.25
Diff Importance 18 pw0.25
Diff Match num 30-34
Diff Match num 35-39
Diff Streak Tw Ew 4

Table D.1: Features chosen during the initial feature selection process.

Feature	Time 0			Time 45			Time 90		
	M1	M2	M3	M1	M2	M3	M1	M2	M3
Diff Importance 1 pw0.25	8	7	7	3	3	3	1	1	1
Diff Importance 18 pw0.25	3	3	3	2	2	2	2	3	2
Diff Points		9		8	6	9	3	2	3
Diff Elo	1	2	2	1	1	1			
Diff Avg player strength	2	1	1	7		7			
Diff Goals scored	9	8	9	6		5			
Diff Goals scored/match	10	10		4	4	4			
Diff Points/Match	5	5	5	10	7				
Elo	4	4	4				8		
Diff Goal difference	7		8		10				
Diff Goal difference/match	6	6	6						
Diff Importance 4 pw0.25				5	5	6			
Diff Wins			10		8	10			
Opp Avg player strength							4	4	5
Avg player strength				9		8			
Diff Corner kick							7	5	
Diff Goals conceded/match							10		8
Diff Shot off goal							6		7
Home							5		4
Opp Goals conceded/match							9		9
Position					9			8	
Diff Losses								6	
Diff Matches left								7	
Draws								9	
Opp Matches								10	
Opp Goal									6
Shot on goal									10

Table D.2: The top 10 features based on mutual information. M1 = WCD_{score}, M2 = WCD_{1X2}, M3 = $LSTM_{copula}$.

Feature	Time 30			Time 60			Time 90		
	M1	M2	M3	M1	M2	M3	M1	M2	M3
Corner kick	29			34		20	25		
Diff Corner kick				24		16	7	5	
Diff Shot off goal	28						6		7
Diff Shot on goal	33						13		12
Diff Red card							24		21
Offside	35				10				
Opp Corner kick				30	8				
Opp Offside							19		18
Opp Goal			24						6
Shot on goal				33					10
Diff Goal	34								
Diff Yellow card							26		
Opp Red card				35					
Opp Yellow card							33		
Opp Shot off goal							14		
Opp Shot on goal	31								
Goal							15		
Yellow card							35		
Shot off goal						18			

Table D.3: Ranking of the in-game features based on mutual information. M1 = WCD_{score}, M2 = WCD_{1X2}, M3 = $LSTM_{copula}$.

Feature	Time 0		Time 45		Time 90	
	M4	M5	M4	M5	M4	M5
Diff Goal difference	5	3	9	6	3	6
Diff Avg player strength	2	2	2	5		5
Diff Elo	1	1	1	2		4
Diff Wins		7	7	10	10	10
Diff Goals scored		4	6	8		8
Diff Goals scored/match	8	9	10		8	
Diff Points	10	5		7		7
Diff Points/match	3	6		9		9
Away Goal				4	2	3
Diff Goal difference/match	4	10	5			
Diff Importance 1 pw0.25		8	8		4	
Home Goal				3	1	2
Away Avg player strength	6				9	
Diff Score				1		1
Away Elo			3			
Diff Importance 4 pw0.25					6	
Diff Position			4			
Home Avg player strength	9					
Home Elo					7	
Home Points/match	7					

Table D.4: The top 10 features based on mutual information. M4 = $LSTM_{score}$, M5 = $LSTM_{1X2}$.

Feature	Time 30		Time 60		Time 90	
	M4	M5	M4	M5	M4	M5
Away Goal		12		3	2	3
Diff Corner kick			17	40	15	43
Home Goal		11		2	1	2
Diff Goal		1		1		1
Away Shot on goal		43		41		
Diff Yellow card		44				40
Diff Shot off goal		41			21	
Diff Shot on goal				39		41
Home Shot on goal			21		17	
Away Corner kick	21					
Away Offside	24					
Away Red card						42
Away Yellow card						45
Away Shot off goal	19					
Diff Offside			24			
Diff Red card					19	
Home Corner kick		42				
Home Offside			20			
Home Red card					20	
Home Yellow card		45				
Home Shot off goal					22	

Table D.5: Ranking of the in-game features based on mutual information. M4 = $LSTM_{score}$, M5 = $LSTM_{1X2}$.

Appendix E

Difference in Cross Entropy

$$CE = CE(\boldsymbol{\theta}|f_Y, \boldsymbol{X}, \boldsymbol{y}) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k\in\Omega_Y} I(y_i = k) ln\big(f_Y(k|\boldsymbol{x_i}, \boldsymbol{\theta})\big)$$

$$CE = -\frac{1}{N}\sum_{i=1}^{N} ln(p_{k,i}),$$

where $p_{k,i} = f_Y(k|\boldsymbol{x_i}, \boldsymbol{\theta})$ and k is the correct class. Looking at the average probabilities gives $p_{k,1} = p_{k,2} = \cdots = p_{k,N} = \bar{p}_k$.

$$\begin{aligned}
CE_1 - CE_2 &= -\frac{1}{M}\sum_{i=1}^{M} ln(p_{k,i}) - (-1)\frac{1}{N}\sum_{j=1}^{N} ln(p_{l,j}) \\
&= \frac{1}{N}\sum_{j=1}^{N} ln(p_{l,j}) - \frac{1}{M}\sum_{i=1}^{M} ln(p_{k,i}) \\
&= \frac{1}{N} ln(\bar{p}_l^N) - \frac{1}{M} ln(\bar{p}_k^M) \\
&= ln(\bar{p}_l) - ln(\bar{p}_k) \\
&= ln(\frac{\bar{p}_l}{\bar{p}_k})
\end{aligned}$$

$$CE_2 - CE_1 = ln(\frac{\bar{p}_k}{\bar{p}_l}) \iff e^{CE_2 - CE_1} = \frac{\bar{p}_k}{\bar{p}_l}$$

Appendix F

Hypothesis Tests for RPS - WCD vs LSTM

Time	WCD_{score}-$LSTM_{score}$ Statistic	p-value	WCD_{score}-$LSTM_{copula}$ Statistic	p-value	WCD_{1X2}-$LSTM_{score}$ Statistic	p-value	WCD_{1X2}-$LSTM_{copula}$ Statistic	p-value
0	-3.71	0.0002	-3.43	0.0006	-3.55	0.0004	-3.28	0.0011
5	-5.25	0.0000	-4.81	0.0000	-4.59	0.0000	-4.17	0.0000
10	-5.67	0.0000	-5.04	0.0000	-5.03	0.0000	-4.42	0.0000
15	-7.13	0.0000	-5.34	0.0000	-7.09	0.0000	-5.34	0.0000
20	-8.26	0.0000	-7.20	0.0000	-7.37	0.0000	-6.30	0.0000
25	-7.58	0.0000	-7.23	0.0000	-8.29	0.0000	-7.92	0.0000
30	-9.92	0.0000	-8.98	0.0000	-9.40	0.0000	-8.48	0.0000
35	-11.16	0.0000	-10.25	0.0000	-11.08	0.0000	-10.18	0.0000
40	-12.97	0.0000	-12.73	0.0000	-12.71	0.0000	-12.47	0.0000
45	-13.85	0.0000	-13.26	0.0000	-14.53	0.0000	-13.90	0.0000
50	-15.99	0.0000	-16.90	0.0000	-16.78	0.0000	-17.70	0.0000
55	-19.47	0.0000	-19.80	0.0000	-19.27	0.0000	-19.61	0.0000
60	-21.89	0.0000	-22.48	0.0000	-21.90	0.0000	-22.50	0.0000
65	-23.69	0.0000	-24.17	0.0000	-24.00	0.0000	-24.48	0.0000
70	-26.14	0.0000	-27.89	0.0000	-26.29	0.0000	-28.07	0.0000
75	-29.18	0.0000	-30.27	0.0000	-29.34	0.0000	-30.44	0.0000
80	-33.16	0.0000	-32.98	0.0000	-32.98	0.0000	-32.80	0.0000
85	-37.26	0.0000	-37.17	0.0000	-37.12	0.0000	-37.02	0.0000
90	-41.91	0.0000	-41.60	0.0000	-41.89	0.0000	-41.58	0.0000

Table F.1: Test statistics and p-values for pairwise Welch's t-test for the difference in RPS.

Appendix G

Betting Performance - Tables and Figures

G.1 Tables Based on the Static Strategy

Time	0.02	0.05	0.1	0.25	0.5	1.0
0	0.6923	0.3845	0.1312	0.0026	0.0000	0.0000
5	0.8279	0.6101	0.3464	0.0418	0.0003	0.0000
10	0.9253	0.8025	0.5919	0.1470	0.0033	0.0000
15	0.9223	0.7932	0.5719	0.1253	0.0019	0.0000
20	0.8530	0.6565	0.3989	0.0573	0.0005	0.0000
25	0.9982	0.9714	0.8705	0.3892	0.0213	0.0000
30	1.1505	1.3868	1.7804	2.4262	1.0243	0.0013
35	1.0551	1.1159	1.1495	0.7936	0.1004	0.0000
40	1.1460	1.3766	1.7688	2.5343	1.3509	0.0045
45	0.9438	0.8460	0.6640	0.2064	0.0068	0.0000
50	0.9737	0.9132	0.7700	0.2889	0.0120	0.0000
55	1.0453	1.0986	1.1429	0.9421	0.2559	0.0005
60	1.0731	1.1741	1.3076	1.3235	0.4775	0.0005
65	0.9657	0.8987	0.7571	0.3114	0.0210	0.0000
70	1.0814	1.1961	1.3567	1.4751	0.6903	0.0057
75	1.0709	1.1736	1.3276	1.5480	0.9837	0.0226
80	1.0171	1.0328	1.0324	0.8544	0.3407	0.0049
85	0.9923	0.9745	0.9292	0.7053	0.2777	0.0041
90	1.0223	1.0523	1.0919	1.1270	0.9216	0.2077

Table G.1: Wealth at the end of the investment horizon with WCD_{score} and for different values of γ and in-game time points.

Time	0.02	0.05	0.1	0.25	0.5	1.0
0	0.8811	0.7160	0.4839	0.1069	0.0030	0.0000
5	0.8945	0.7402	0.5094	0.1089	0.0022	0.0000
10	0.9410	0.8388	0.6511	0.1956	0.0066	0.0000
15	0.8748	0.7000	0.4557	0.0826	0.0013	0.0000
20	0.8729	0.6969	0.4525	0.0820	0.0012	0.0000
25	0.8682	0.6907	0.4513	0.0904	0.0021	0.0000
30	1.0811	1.1808	1.2693	0.9262	0.1046	0.0000
35	0.8933	0.7412	0.5184	0.1256	0.0037	0.0000
40	1.0861	1.2076	1.3750	1.4430	0.5211	0.0010
45	0.9184	0.7911	0.5824	0.1522	0.0040	0.0000
50	1.0209	1.0294	0.9832	0.5578	0.0553	0.0000
55	1.0366	1.0755	1.0931	0.8266	0.1799	0.0001
60	1.1222	1.3088	1.6091	2.0828	1.0029	0.0025
65	1.0520	1.1160	1.1783	1.0091	0.2820	0.0004
70	1.1065	1.2587	1.4716	1.5746	0.5332	0.0009
75	1.0466	1.1095	1.1915	1.2241	0.7080	0.0263
80	0.9846	0.9495	0.8630	0.4993	0.0810	0.0000
85	0.9886	0.9623	0.8960	0.5901	0.1394	0.0001
90	1.1192	1.3073	1.6387	2.6396	3.5223	1.1944

Table G.2: Wealth at the end of the investment horizon with WCD_{1X2} and for different values of γ and in-game time points.

Time	0.02	0.05	0.1	0.25	0.5	1.0
0	0.9054	0.7600	0.5304	0.1100	0.0017	0.0000
5	0.8958	0.7430	0.5134	0.1105	0.0021	0.0000
10	0.8642	0.6722	0.4064	0.0492	0.0002	0.0000
15	0.8415	0.6355	0.3756	0.0505	0.0004	0.0000
20	0.9022	0.7481	0.5024	0.0817	0.0005	0.0000
25	0.8788	0.7050	0.4556	0.0742	0.0007	0.0000
30	1.0263	1.0371	0.9801	0.4871	0.0289	0.0000
35	0.8929	0.7314	0.4856	0.0815	0.0007	0.0000
40	1.0630	1.1317	1.1651	0.7403	0.0628	0.0000
45	0.8778	0.7041	0.4565	0.0769	0.0008	0.0000
50	1.0135	1.0027	0.9085	0.3767	0.0132	0.0000
55	0.8473	0.6453	0.3854	0.0524	0.0004	0.0000
60	0.9667	0.8993	0.7538	0.2930	0.0152	0.0000
65	1.0482	1.0994	1.1216	0.7787	0.1100	0.0000
70	0.9829	0.9330	0.8003	0.3184	0.0167	0.0000
75	1.0404	1.0889	1.1330	0.9815	0.3292	0.0011
80	0.9447	0.8526	0.6866	0.2552	0.0150	0.0000
85	0.9955	0.9786	0.9256	0.6416	0.1778	0.0005
90	1.0052	1.0071	0.9946	0.8556	0.4562	0.0177

Table G.3: Wealth at the end of the investment horizon with $LSTM_{score}$ and for different values of γ and in-game time points.

Time	0.02	0.05	0.1	0.25	0.5	1.0
0	0.8536	0.6565	0.3970	0.0550	0.0005	0.0000
5	0.8824	0.7144	0.4723	0.0875	0.0013	0.0000
10	0.9691	0.8909	0.7046	0.1833	0.0030	0.0000
15	0.8888	0.7222	0.4714	0.0740	0.0006	0.0000
20	0.8648	0.6809	0.4323	0.0738	0.0011	0.0000
25	0.7740	0.5066	0.2257	0.0097	0.0000	0.0000
30	0.9172	0.7728	0.5219	0.0759	0.0003	0.0000
35	1.0108	1.0028	0.9290	0.4681	0.0352	0.0000
40	1.0678	1.1380	1.1578	0.6549	0.0386	0.0000
45	0.8732	0.6876	0.4209	0.0498	0.0002	0.0000
50	1.0152	1.0159	0.9605	0.5380	0.0559	0.0000
55	1.0425	1.0899	1.1192	0.8556	0.1761	0.0001
60	0.9902	0.9522	0.8370	0.3582	0.0195	0.0000
65	0.9573	0.8749	0.7053	0.2275	0.0067	0.0000
70	1.0846	1.1916	1.3022	1.0859	0.2219	0.0001
75	0.9375	0.8351	0.6547	0.2147	0.0085	0.0000
80	0.9014	0.7601	0.5497	0.1526	0.0058	0.0000
85	0.9651	0.9081	0.8038	0.4738	0.1062	0.0002
90	1.0363	1.0908	1.1813	1.4390	1.7504	1.5607

Table G.4: Wealth at the end of the investment horizon with $LSTM_{copula}$ and for different values of γ and in-game time points.

Time	0.02	0.05	0.1	0.25	0.5	1.0
0	0.9412	0.8389	0.6500	0.1900	0.0054	0.0000
5	1.0798	1.1623	1.1799	0.5756	0.0164	0.0000
10	0.9339	0.8117	0.5830	0.1084	0.0008	0.0000
15	1.1547	1.3838	1.7110	1.7276	0.2633	0.0000
20	0.9077	0.7576	0.5115	0.0822	0.0005	0.0000
25	0.9471	0.8406	0.6247	0.1282	0.0011	0.0000
30	0.9042	0.7476	0.4929	0.0707	0.0003	0.0000
35	0.9903	0.9444	0.8014	0.2674	0.0066	0.0000
40	1.0040	0.9814	0.8761	0.3596	0.0138	0.0000
45	1.1696	1.4300	1.8314	2.0608	0.3599	0.0000
50	1.2605	1.7228	2.6494	5.0417	1.9133	0.0001
55	1.1960	1.5132	2.0545	2.7832	0.6900	0.0000
60	1.0849	1.1830	1.2467	0.7655	0.0475	0.0000
65	0.8989	0.7396	0.4883	0.0749	0.0005	0.0000
70	0.9934	0.9288	0.7234	0.1420	0.0008	0.0000
75	1.0164	1.0087	0.9175	0.3895	0.0163	0.0000
80	0.8890	0.7195	0.4625	0.0655	0.0003	0.0000
85	0.9073	0.7647	0.5403	0.1255	0.0032	0.0000
90	0.9870	0.9502	0.8538	0.4755	0.0941	0.0007

Table G.5: Wealth at the end of the investment horizon with $LSTM_{copula}$ and for different values of γ and in-game time points.

G.2 Plots Based on the Static Strategy

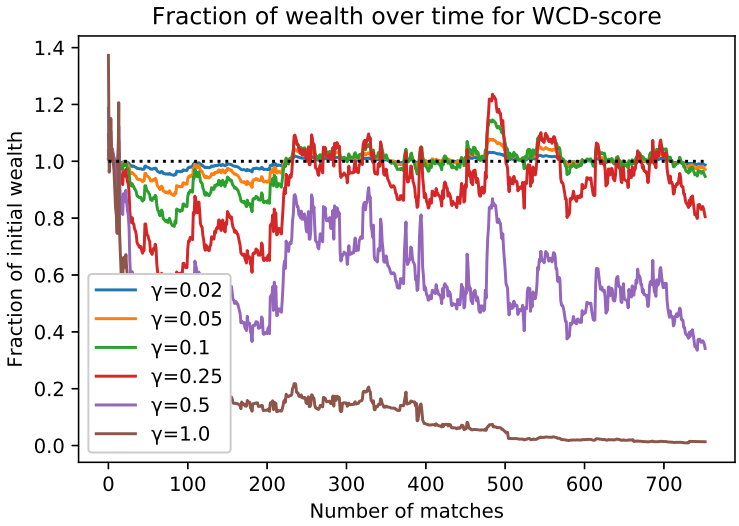


Figure G.1: Proportion of wealth for WCD_{score} and different values of γ over the investment horizon. The wealth is the average over all time intervals.

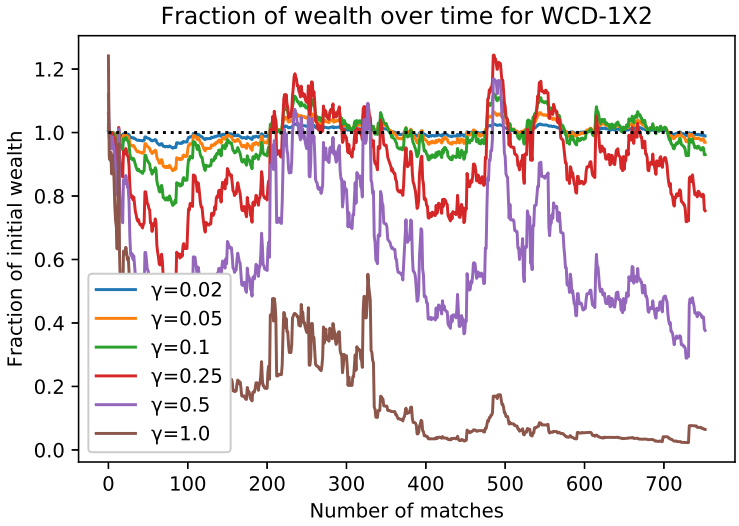


Figure G.2: Proportion of wealth for WCD_{1X2} and different values of γ over the investment horizon. The wealth is the average over all time intervals.

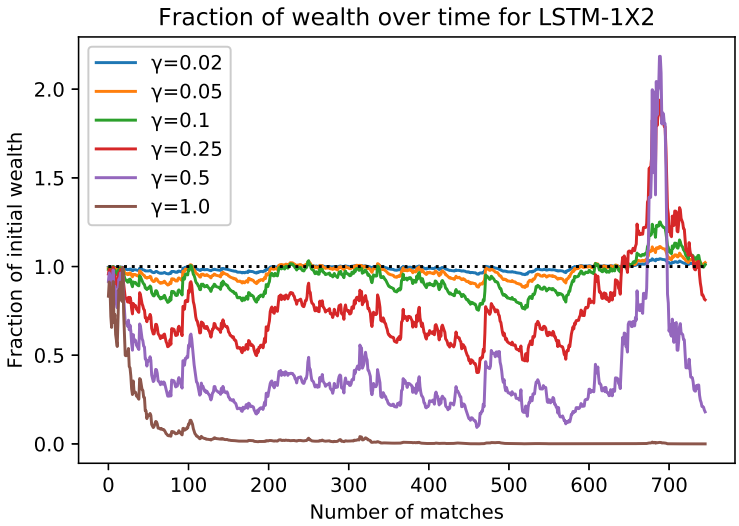


Figure G.3: Proportion of wealth for $LSTM_{1X2}$ and different values of γ over the investment horizon. The wealth is the average over all time intervals.

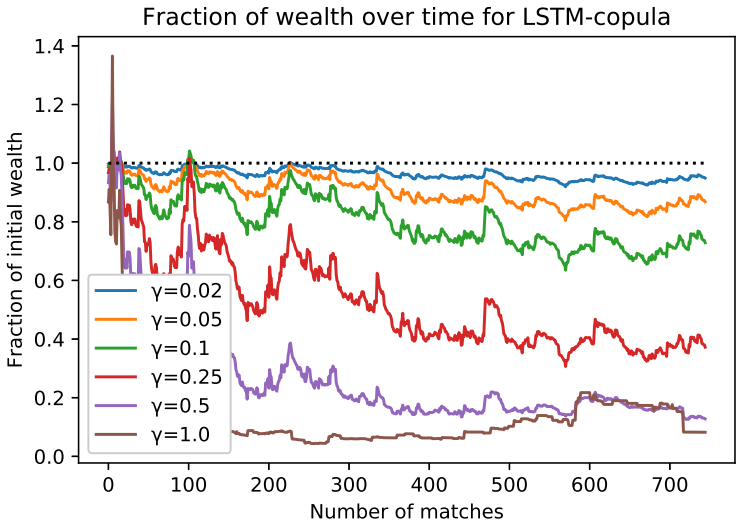


Figure G.4: Proportion of wealth for $LSTM_{copula}$ and different values of γ over the investment horizon. The wealth is the average over all time intervals.

G.3 Plots Based on the Dynamic Strategy

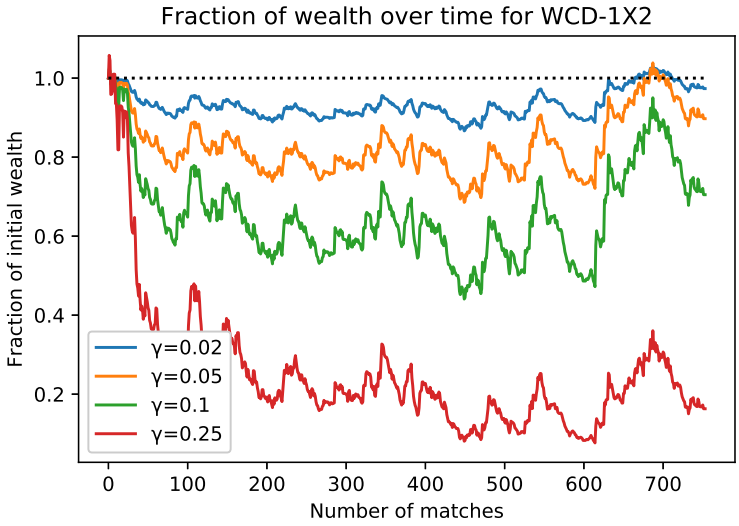


Figure G.5: Proportion of wealth for WCD_{1X2} and different values of γ over the investment horizon.

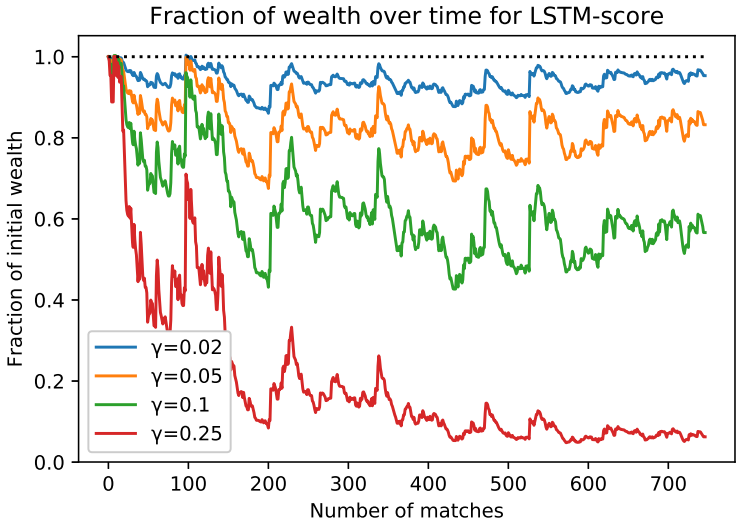


Figure G.6: Proportion of wealth for $LSTM_{score}$ and different values of γ over the investment horizon.

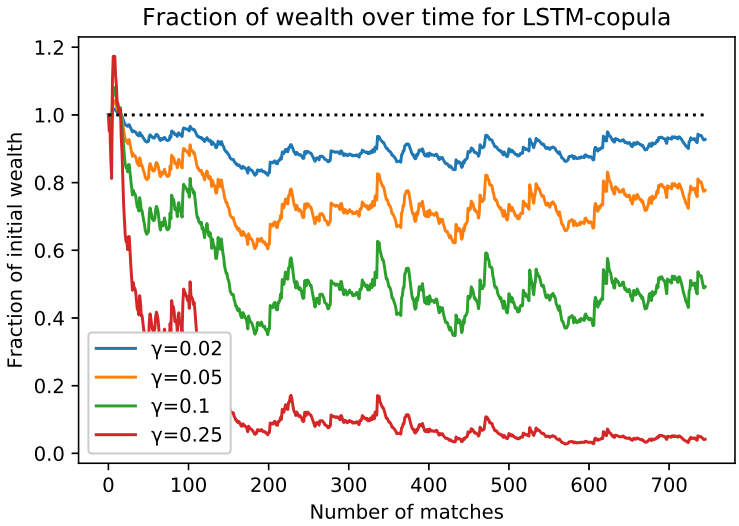


Figure G.7: Proportion of wealth for $LSTM_{copula}$ and different values of γ over the investment horizon.

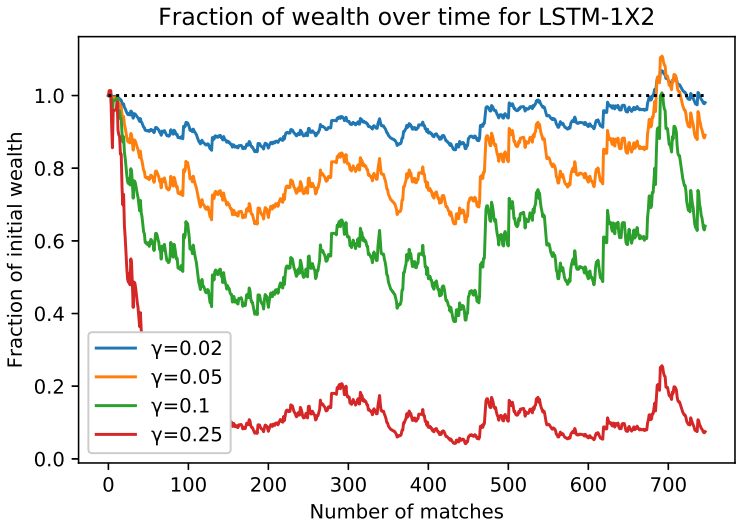


Figure G.8: Proportion of wealth for $LSTM_{1X2}$ and different values of γ over the investment horizon.

Robin Andersen, Vegard Hassel

In-game Betting and the FA English Premier League: The Contribution of Prediction Models

NTNU
Kunnskap for ei betre verd