University of Economics in Prague

Faculty of Informatics and Statistics

Prediction models in e-sports: Generalized autoregressive score and common opponent models

Master Thesis

Department: Department of Econometrics

Study program: Econometrics and Operations Research

Author: Bc. Miroslav Pikhart

Supervisor: Mgr. Vladimír Holý, Ph.D.

Prague, April 2019

Declaration

I hereby declare that I am the sole author of the thesis entitled *Prediction models in e-sports: Generalized autoregressive score and common opponent models* and all cited literature and sources are stated in the attached list of references.

In Prague on 28th May 2019 ....

Bc. Miroslav Pikhart

Acknowledgement

I would like to thank my supervisor, Mira Holy, for supporting my choice of what is an unusual field of study and also co-writing papers on the subject. Also my work colleagues who understood that I did not have time to grab an after work beer for several weeks. And my girlfriend for not killing me for writing while she tried to sleep. Thank you all.

Abstrakt

Kromě populární sportovní statistiky se v posledních letech, společně se vzrůstající oblíbeností e-sports akcí zvyšuje zájem i o e-sport statistiku. V rámci této práce je představena adaptace dvou modelů převzatých ze sportovní statistiky. Oba tyto modely, známy jako general autoregressive score model a common opponent model disponují schopností predikovat výsledek zápasu pouze na základě veřejně dostupných informací jako je čas konání a výsledek. V rámci empirické studie je testována predikční schopnost zmíněných modelů na datech pocházející ze hry Counter-Strike. Kromě aplikace již vytvořených modelů je dále představen alternativní přístup k common opponent modelování, který předpokládá, že síla týmů v čase není konstantní. Tento přístup slabě zvyšuje predikční schopnosti modelu.

Klíčová slova

e-sport statistika, general autoregressive score model, predikce výsledků, common opponent modelování

Abstract

The field of e-sports statistics is getting more popular as e-sports become generally known. In this thesis, we discuss the adaptations of two models with proven usage in sports statistics and their possibilities in the realm of e-sports. The most significant advantage shared by both of these models, known as general autoregressive score model and common opponent model, is that they require only publicly available data in order to forecast future match results. In an empirical study, we test the predictive power of said models on the game Counter-Strike, abbreviated as CS:GO, and introduce another approach to the common opponent model, which accounts for the time-varying performance of modelled teams and slightly improves the predictive ability of the model

Keywords

e-sport statistics, general autoregressive score model, result forecasting, common opponent modelling

Contents

List of Figures

List of Tables

1. Introduction

1.1 Motivation

E-sports, short for electronic sports, is one of the phenomenons of the current century. Originally being perceived as just games for children, the way computer games became received by businesses, sponsors and wide audiences has changed significantly. From small, mostly unknown events, e-sports moved to selling out the biggest stadiums in the recent years.

In the year 2012, when e-sports started to become official, the worldwide e-sports market revenue was estimated at 120 million dollars. Figure 1.1 shows the rapid growth during recent years all the way up to a predicted 1.1 billion USD market revenue for the complete year 2019.[1]

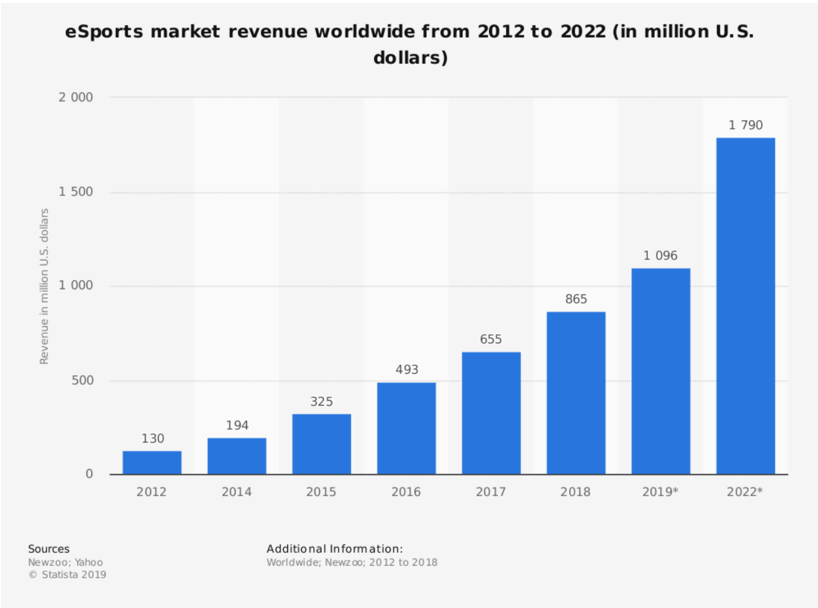

Figure 1.1: E-sports market revenue yearly growth

This also strongly correlates with the amount of money e-sports athletes make. What was just a hobby a few years back has changed into a viable career path with more and more prize money being offered for placing highly in tournaments – the most prestigious events are already on par with the most prestigious events of classic sports by this metric. Figure 1.2 shows cumulative prizes awarded to players of different e-Sport titles during the year 2018.

It is definitely not only about money, the crowd following e-sports has been increasing year by year as seen in figure 1.3, growing from around 130 millions viewers in 2012 up to 400 million viewers by 2018 with further audience growth being predicted in the years to come. With current numbers, e-sports as a whole can be compared to popular traditional sports such as

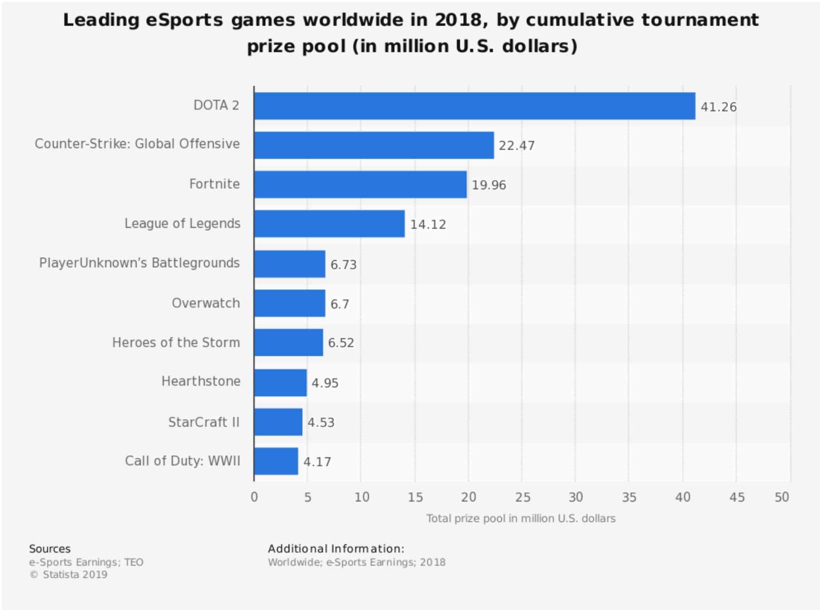

Figure 1.2: E-sports games ordered by cumulative prize pool in 2018

golf and rugby - sitting at ranks #9 and #10 in the list of most watched sports worldwide, with around 450 million viewers.

Following such crowd engagement, there has been a massive growth in the betting market, leading to incorporation of e-sports events into established betting companies and the rise of brand new ones, focusing primarily on e-sports together with specialized companies focusing on predicting match results and trying to exploit the initial inefficiencies as the companies used to predict classic sports struggled to successfully calculate the odds for e-sport matches.

Even now, it still appears to be highly possible to profit in the betting market using correct algorithms to predict the match outcomes. This is however more challenging than originally expected, as the scene is extremely volatile thanks to a plethora of factors such as frequent player transfers, changing match conditions and even changes to the games being played as they are being updated over time.

While there were several attempts at predicting 'live' results based on what is currently happening in the matches such as predicting the results of a Dota match based on what heroes were selected [2] or based on metrics computed throughout the game [3], little publicly available research has been made into predicting the results of specific e-sports matches.

1.2 Objectives

Exploring the existing literature, it was evident that there were basically no attempts to model the outcomes for e-sports matches with only a few papers discussing other e-sports

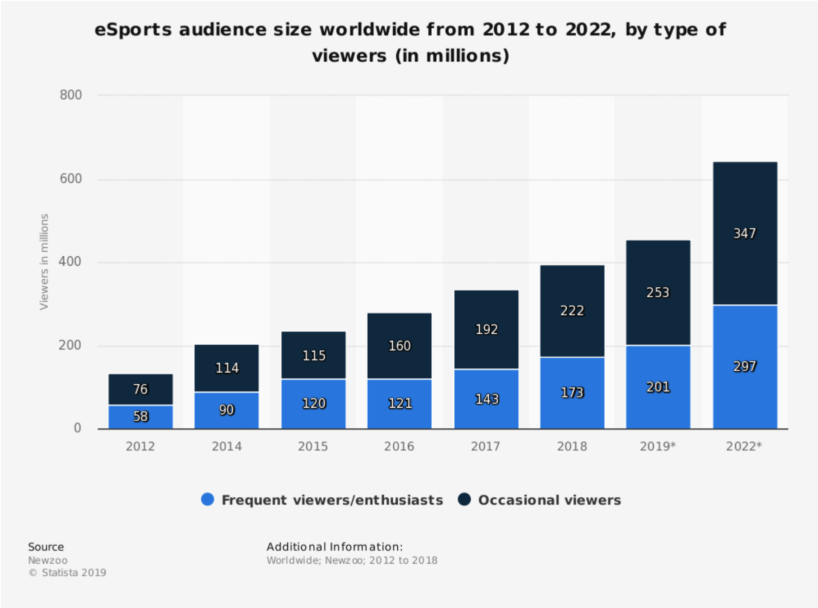

Figure 1.3: E-sports audience yearly growth

matters, while the are several established models used to predict classic sports. A complete analysis of existing literature can be found in Chapter 2. In an attempt to fill the gap, we try to tailor some of the models which show potential to yield interesting results to e-sports use. In particular, the focus of this thesis is on generalized autoregressive score (GAS) model [4], which was successfully used in predicting soccer results [5] and an approach named the common opponent model, which utilizes only the most relevant subset of available data, proposed as a method to model tennis matches [6] and transformation of these models to be used to predict results of e-sport matches in either Dota 2, or Counter-Strike.

Objectively, the aim of the thesis is to:

- Present a selection of prediction models for e-sports match results and try to improve the prediction accuracy by choosing valid methodology.
- Experiment with multiple different sets of historical data and quantify the impact they have on prediction accuracy of presented models.
- Introduce possible further modifications that should lead to increased accuracy of models in the field.

1.3 Thesis structure

The thesis is structured in 6 chapters including this one.

- **Chapter 2**, with the goal to introduce most of the topics touched upon in the thesis, is divided into three sections. The first one provides background information on the topic of e-sports and specifically the game of Counter-Strike, which is being analyzed in the thesis. Second one briefly introduces the theoretical methods used later on, while the third one provides a more detailed review of literature already published on the subject or relevant to it.

- **Chapter 3** describes the data set used to validate presented models and derives some mathematical formulas used further in the thesis.
- **Chapter 4** introduces the general autoregressive score model, the theory behind it, then discusses possible applications and alternatives when working with our data set and then evaluates the performance of the individual approaches.
- **Chapter 5** follows the structure of the previous chapter, but is centered around the common opponent approach, explaining the principles behind it, illustrating the possible variants and evaluating them.
- **Chapter 6** concludes the thesis by comparing the results achieved and discussing the work done on the subject. It also includes discussion on possible improvements and further work on the topic.

2. Background

This chapter introduces the game of Counter-Strike by describing the rules, the competitive match system and the way professional tournaments are being run and explains the inherent variables that might affect the predicted outcome. Then it introduces the theory required in the understanding of the methods used in the thesis and provides a comprehensive review of literature covering the methods used.

2.1 On the game of Counter-Strike

Counter-Strike is a team-based first person shooter game, originally introduced in 1999. Two teams, each consisting of five players, compete in multiple rounds of objective-based game modes with the goal of winning enough rounds to win the match. Under standard tournament rules, two teams of five players each compete in a best-of-30 match. The game is asymmetrical, with each team having slightly different objective and unlike traditional sports, every match can be played on a different map, which leads to a variable win probability for a round, affected by the selected map and the side the team is currently playing as.

2.1.1 Game mechanics

As mentioned, the game is classified as a first-person shooter and was inspired by special forces anti-terrorism missions. In every game, a team can either play as the terrorist side (also called the attacking side, as their objective is to successfully plant an explosive and blow up one of two key targets), or the counter-terrorist side (or defending side, as their objective is to defend both of the key targets by either preventing an explosive from being planted or defusing in before it explodes). A round is won either by completing the major objective of your side, or eliminating all opponents.

The game has an economy-management layer, where players receive money for either winning rounds, completing objectives or killing their opponents. This money can be used to purchase better gear in the form of either weapons or tactical equipment, which increases your chance to win the round. If a player survives the round, he keeps all the gear he had on him while if he was taken down, he needs to purchase everything again. This implies that winning a round with enough players surviving increases the probability to win subsequent rounds and there are several common strategies to utilize economy disparity or to try and prevent it.

2.1.2 Match rules

One game is being played as a best-of-30 rounds and can therefore end as soon as one team reaches 16 points. After 15 rounds have been played, the teams switch sides and everything besides score effectively resets to initial stage, removing all acquired gear and saved money from players. In the case where the game ends tied at $15 - 15$, an overtime is triggered.

Overtime consists of separate phases, each lasting exactly 6 rounds, with the teams switching sides after three rounds have been played. In every overtime stage, 6 rounds are being played with the game ending when one team wins four rounds and therefore wins the whole game. If it again ends in a $3 - 3$ tie, another overtime stage starts immediately. This process repeats until a winner is decided.

2.1.3 Map selection

As mentioned before, a major distinction compared to traditional sports is that the game can be played on one of several possible maps. This adds an additional layer to strategy, as each map is designed differently and therefore the probabilities of winning a round while playing one side differ across maps, with some of them having over 55% probability for one side to win. Also, given the amount of maps, it is given that not every team will have the same strength on all of them, giving a possibly major advantage to a team that is able to avoid playing maps that their opponent prefer.

For every tournament, there is a prepared selection of seven eligible maps the teams can choose from. This set is usually the same for most of the important tournaments in one season and changes are being made when a new season starts.

There is an official process through which maps are selected to be played in the match. With one team, chosen randomly, starting, both alternately eliminate one map from the eligible set until the number of maps remaining for selection is equal to the number of games the match consists of – in the vast majority of cases, the match consists of either one or three games. This method allows both teams to influence the selection and avoid playing on maps they consider being the most disadvantaged at. After a map is selected, just before the game starts, the teams play one shortened round with special rules to determine the initial side selection, with the winner deciding which team starts on which side.

2.1.4 Tournaments

The tournament scene is unofficially split into four levels, with Premier being the highest one, then Major, Minor and Regional. Premier Tournaments offer an outstanding prize pool, are almost exclusively played offline at well-established venues, and feature the best teams from all over the world. Major Tournaments also feature a large prize pool and are mostly attended by top-tier teams, that were either invited based on previous performances or qualified for the event. Minor Tournaments offer a smaller prize pool and less prestige than Major Tournaments but still draw a high level of competition, especially since winning these tournaments usually leads to either qualifying or getting an invite to a Major one.

2.2 Theoretical methods

This section provides a brief introduction to the theoretical methods being either used or expanded upon throughout this thesis. We introduce the basics of probability theory and show the probabilistic distributions used in parts of the thesis. After that, we describe the fundamentals of significance testing and the concept of a discrete-time Markov chain.

2.2.1 Basic probability theory

This section provides an overview of the basic probability theory formulas and terminology which is expanded upon later in the thesis.

A sample space S, is the full set of outcomes of an experiment. If we flip a coin, it consists of two options, heads and tails. Two events are mutually exclusive when their intersection is an empty set

$$A \cap B = \emptyset. \tag{2.1}$$

Two events are independent when the probability of both happening is the probability of one event multiplied by the probability of the other, denoted as

$$P(A \cap B) = P(A)P(B). \tag{2.2}$$

Conditional probability is the probability of an event occurring given that another event is known to have occurred. The probability of event A occurring given that event B occurred is indicated as $P(A|B)$

$$P(A|B) = \frac{P(AB)}{P(B)}. \tag{2.3}$$

Combining the above-mentioned, we can calculate the conditional probability of two independent events A and B as

$$P(A|B) = P(A) \tag{2.4}$$

2.2.2 Probability distributions

Bernoulli distribution

Bernoulli distribution is used to describe a random variable which takes the value of 1 with probability p and the value of 0 with probability $1-p$. The most common use is as a binary success – failure variable, where a random event succeeds with the probability p. Equations (2.5) and (2.6) describe the probability mass function and cumulative distribution function of the Bernoulli distribution

$$f(k) = \begin{cases} p & \text{for } k = 1 \\ 1-p & \text{for } k = 0 \end{cases} \tag{2.5}$$

$$F(k) = \begin{cases} 0 & \text{for } k < 0 \\ 1-p & \text{for } 0 \leq k < 1 \\ p & \text{for } k \geq 1. \end{cases} \tag{2.6}$$

The mean and variance of Bernoulli distribution are then respectively given by equations 2.7 and 2.8.[7]

$$E(x) = p \tag{2.7}$$

$$Var(x) = p(1-p) \tag{2.8}$$

Binomial distribution

The binomial distribution is related to the Bernoulli's. It represents the discrete probability distribution of achieving k successes in n independent and identically distributed binary experiments with probability of success equal to p. From there it is easily visible that the distribution is equal to Bernoulli's for $n = 1$. The binomial distribution mass density function is defined as

$$f(k) = \binom{n}{k} p^k (1-p)^{n-k} \tag{2.9}$$

where n stands for the number of trials, k the number of successes, p the probability of success and

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \tag{2.10}$$

The cumulative distribution function is a sum of all the discrete values up to k as seen in equation (2.11)[7]

$$F(k) = \sum_{i=0}^{k} \binom{n}{i} p^i (1-p)^{n-i} \tag{2.11}$$

and mean and variance are given by

$$E(x) = np \tag{2.12}$$

$$Var(x) = np(1-p). \tag{2.13}$$

Negative binomial distribution

The negative binomial distribution is closely related to both distributions discussed above. If we observe a sequence of independent Bernoulli trials with probability p for success and $1-p$ for failure until a predefined amount of failures r has occurred. Then the random number of successes will follow the negative binomial distribution.

The mass density function with number of failures r, number of successes k, and p as the probability of success is defined as:

$$f(k) = \binom{k+r-1}{k} p^k (1-p)^r \tag{2.14}$$

with mean and variance respectively defined by equations (2.15) and (2.16)

$$E(x) = \frac{pr}{1-p} \tag{2.15}$$

$$Var(x) = \frac{pr}{(1-p)^2}. \tag{2.16}$$

2.2.3 Significance testing

Z-score

The Z-Score or standard score is the number of standard deviations a data point x is from the mean, μ, considering a normal distribution with standard deviation σ. Z-Score can be calculate using equation (2.17)

$$z = \frac{x-\mu}{\sigma}. \tag{2.17}$$

The Z-Score can be used in conjunction with the standard normal distribution to find the p-value of the Z-Test. The p-value in this case is the probability of getting a sample average with a specific deviation or more from the mean of the population distribution.

A/B testing

The A/B test, also called split test, checks whether the results of two experiments are significantly different from one another. It is designed to distinguish between differences in results occurring to natural variation of the samples and results occurring because the sample means are in fact different from one another. It has been adapted by businesses worldwide to test features of websites and applications.

In this thesis we use two-sample Z-tests for A/B testing as our sample sizes are sufficiently large in all our experiments. This allows for the variances of distributions to be estimated with a high amount of accuracy.

2.2.4 Discrete-time Markov chain

Discrete-Time Markov Chain is a stochastic process which consists of a countable number of states in its state space, is sampled at discrete points in time and the probability of a random variable moving to the next state depends only on the present state and is therefore independent from any other previous states - this feature is called the Markov property.

Therefore, a stochastic process is considered a discrete-time Markov chain if the following holds true:

$$P[X_{n+1} = X_{n+1}|X_n = X_n, X_{n-1} = X_{n-1}, \ldots, X_0 = X_0] = P[X_{n+1} = X_{n+1}|X_n = X_n] \tag{2.18}$$

2.3 Literature overview

E-sports as a new phenomenon has very few papers written on it and only a percentage of those concerns statistics and result forecasting. For this reason, we have tried to convert some approaches used in forecasting results in traditional sports into e-sports usage.

2.3.1 E-sports statistics and result forecasting

As e-sports only became a subject of study in the last couple of years, only a handful of papers were published in this area with highly varying areas of focus and quality of the paper itself, but two main categories can be differentiated between.

First category are usually mostly surface level publications, trying to explore the field and draw parallels toward classical sports. This is a case of Parshakov and Zavertiaeva [8] who perform regression analysis on the effect of nationalities on success in e-sports and report results similar to studies done on the Olympics with certain nations having significantly better performance than others. Parshakov et al.[9] study diversity among Counter-Strike teams and build regression models with the goal to explain the amount of prize money won by a team based on the diversity in languages, cultures and skill of the individual team members, arriving at the conclusion that language diversity, if it does not exceed a particular level, actually leads to better results. This finding can however be contested with the hypothesis that increasing language diversity is strongly correlated with increasing team strength as team tries to find the five strongest players regardless of nationality. Makarov et al.[10] use logistic regression to forecast Counter-Strike matches and discusses the viability of using TrueSkill, an adaptation of the traditional Elo ranking model for multiple player matches as a forecasting tool.

The second category is more focused around the application of machine learning methods, either supervised or unsupervised, in order to predict match results. Multiple papers were published in this area with majority of them focusing on the game Dota 2, known to be the e-Sport title with the highest market revenues. Yang et al.[11] use neural networks and logistic regression to predict Dota 2 match result from live data after every minute passes in the match, using several derived variables to describe the current state of the match. Hodge et al.[12] build up on this approach and forecasts professional matches using a random forest algorithm with a combination of both pre-match and in-game data. Drachen et al.[13] investigated differences in the spatio-temporal behavior in Dota 2 matches based on player skill and experience. The authors found that higher-ranked players tend to move around more often while managing to stay closer to their teammates, indicating increased understanding of teamwork. For collecting information, they used a spatial division of the Dota 2 map into several square zones and investigated movement between those zones. Schubert et al. [14] try to predict the result of the ongoing match by splitting it into defined encounters in which one team directly interacts with the other and using their outcomes as explanatory variables towards match result and the contributions of individual players and encounters towards it, leading to not only trying to forecast the result itself but also rank the players in the match.

2.3.2 Generalized autoregressive score models

Generalized autoregressive score (GAS) models, also known as dynamic conditional score models were originally proposed in 2008 by Creal et al.[4]. At the same time, Harvey and Chakravarty [15] developed a score driven model optimized to handle increased volatility, called the Beta-t-(E)GARCH model, build upon exactly the same philosophy. Creal et al. further developed the model in the following years [16] and illustrated practical applications in analyzing stock volatility. GAS models found usage in several areas, including stock market diversification, discussed by Avdulaj and Barunik [17], Zhao and van Wijnbergen's [18] application in decision making in incomplete markets or risk assessment modelling [19].

Koopman and Lit[20] proposed using GAS models for forecasting the Premier league soccer matches, building upon Reep and Benjamin [21] who proved that both the number of scored goals and completed passes in a soccer match follow the negative binomial distribution to an extent as well as Rue and Salvensen [22] who proposed a dynamic model to represent the varying strength of teams at different points in time. Later they extended their study and included several variants of their model, experimenting with bivariate Poisson distribution as well as Skellam distribution. The possibilities of applying a GAS model on forecasting Counter-Strike matches were presented by Pikhart and Holy[23] with mixed results.

2.3.3 Common opponent modelling

Common opponent approach to result forecasting is rare and not many papers were published on it since it was suggested by Knottenbelt et al. in 2012 [6]. Following papers by Spanias[24], Madurska[25] and Sipko[26] who proposed further expansion of the model via the usage of machine learning to optimize its results were all building upon the foundation provided by Knottenbelt, who is also listed as a co-author in these papers. Validation of the model and comparison to other well performing models was done by Kovalchik[27].

3. Data set

This chapter describes all the data set related tasks within the project, including the origin and description of data used to train and test the model on. It introduces the data set and explains the transformations and changes applied.

3.1 Data set

Currently, detailed data about Counter-Strike matches are not publicly available, while basic information about matches including the result, time of match and on which map it was played are accessible through several websites.

3.1.1 Data acquisition

The original data set consists of roughly 17,000 professional matches of all levels played between the years 2016 and February 2019. It was downloaded from a website [28] using a Python HTML-parsing library called BeautifulSoup.

3.1.2 Data exploration

The downloaded data set consists of 16,829 matches. Out of those, we selected 11,303 matches in which at least one of the two teams was included among the top 50 teams in the respective year, which excluded mostly matches irrelevant to either one of the approaches modelled in this thesis.

The parsed data include only a few selected variables, as an advantage of both discussed models is being tailored to work with only match results, which are public knowledge:

- date – the date on which the match was played
- map – which map the match took place on
- team1 – the name of first team in the match
- t1score – how many rounds the first team won in the match
- team2 – the name of second team in the match
- t2score – how many rounds the second team won in the match

From these, we can derive several custom variables for further use in the discussed models:

- rw1 – % of rounds won in a match by the first team

- rw2 – % of rounds won in a match by the second team, these two sum up to 1
- winner – the team that won the match (e.g. the team with more rounds won)
- loser – the team that lost the match

All of the matches in the data set have been played on one out of eight maps, their distribution can be found in figure 3.1. We can also look at their distribution by years, showcased in figure 3.2. It is visible that the number of matches is increasing as the scene is still growing.

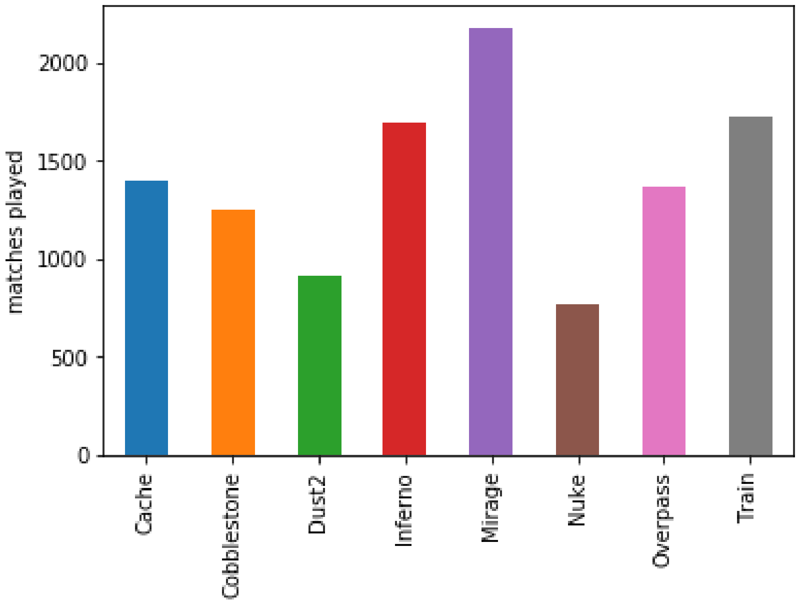

Figure 3.1: Number of matches played by map

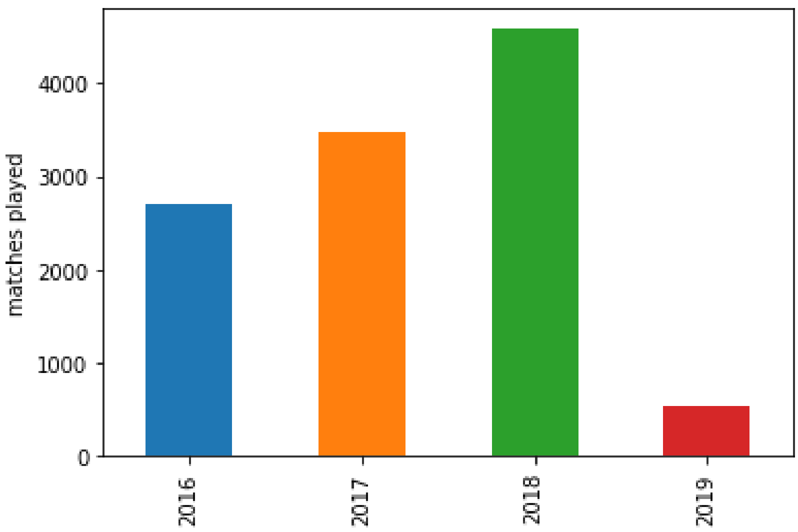

Figure 3.2: Number of matches played by year (2019 only contains January and February data)

4. General autoregressive score model

This section loosely follows the approach and methodology proposed by Pikhart and Holy [23].

4.1 Introduction

Generalized autoregressive score model is a time series model with time-varying parameters. The dynamics of parameters which are variable in time are captured by the autoregressive parameter and the score of the conditional observation density. When considering unscaled score, the time-varying parameter f_t follows the recursion

$$f_{t+1} = C + Bf_t + A\nabla(x_t, f_t), \tag{4.1}$$

in which C represents the constant parameter, B the autoregressive parameter, A the score parameter and $\nabla(x_t, f_t)$ the score

$$\nabla(x_t, f_t) = \frac{\partial \ln \mathrm{P}[X_t = x_t | f_t]}{\partial f_t}. \tag{4.2}$$

Forecasting accuracy of score-driven models was proven to be similar or highly competitive when compared to their parameter-driven counterparts by Koopman et al.[29] with a significant advantage of simple estimation through the maximum likelihood method. The log-likelihood is given by

$$L(\psi) = \sum_{t=\tau}^{T} lnP(X_t = x_t | f_t) \tag{4.3}$$

with τ being an integer for initialization purposes – as not every team starts playing at the very beginning of our data set, we introduce the index τ which indicates the time t on which a new team enters the model. The initial elements of f_τ are set to $\frac{C}{1-B}$.

4.2 Possible variants of the model

We consider three possible approaches – a naive model with skill that is calculated once and from there on is constant in time. While this is the simplest variant to model, just from the definition it should not be the perfect solution, but more of a benchmark for the dynamic models. Considering dynamic models, we propose two variants, one of them considering overall strength of a team and the second one differentiates between team strengths based on which map the match is played on.

All variants are based on the GAS specification with Bernoulli distribution as the underlying statistical distribution. The time-varying probability of winning a match is transformed via logistic link to ensure its values belong to the interval (0,1), corresponding to the logistic regression model.

As the matches are usually being broadcasted online in real time and only extremely rarely are multiple matches played at once, we are able to order the matches by the time when they were played and assign them values of t, where the first match in the data set has $t = 1$ and then we are able to construct an ordered sequence of matches where each match has assigned an unique value of t and for all matches except the first and last one in the sequence, there exists exactly one match with $t - 1$ and exactly one match with $t + 1$.

4.2.1 Map-independent variant

This model assumes the strength of teams to change over time either thanks to external effect such as player substitution, changes in actual form and others or thanks to simply improving the team strength overall. Once again, we construct the probability function for team i to defeat team j similarly to the static approach. As this model is dynamic, this probability differs with changes in time t and is obtained as

$$p_{i,j,t} = \frac{1}{1 + e^{-(S_{i,t} - S_{j,t})}} \tag{4.4}$$

With skill $S_{i,t}$ variable over time, we calculate it through the following recursion. The change in skill S_i of team i after winning a match, based on the skill S_j of the opponent can be observed in 4.2.

$$\begin{aligned}
S_{i,t+1} &= C_i + B_i S_{i,t} + A_i \frac{e^{S_{k,t}}}{e^{S_{i,t}} + e^{S_{k,t}}} && \text{if team } i \text{ defeated team } k \text{ in time } t, \\
S_{i,t+1} &= C_i + B_i S_{i,t} - A_i \frac{e^{S_{i,t}}}{e^{S_{i,t}} + e^{S_{k,t}}} && \text{if team } i \text{ lost to team } k \text{ in time } t, \\
S_{i,t+1} &= S_{i,t} && \text{if team } i \text{ did not play in time } t.
\end{aligned} \tag{4.5}$$

The changes to perceived strength of teams are illustrated by figure 4.1. We can see the skill values changes in time for three representative teams – Astralis, considered to be the top team in the world for the last few years, fnatic, a team usually ranked around tenth place and Immortals, a team that has very strong 2016 showing where they managed to win three major tournaments but started to slip ever since.

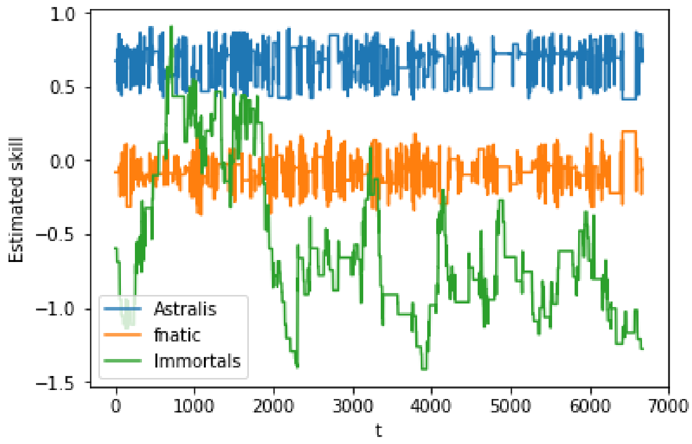

Figure 4.1: Comparison of estimated skill in time for three selected teams

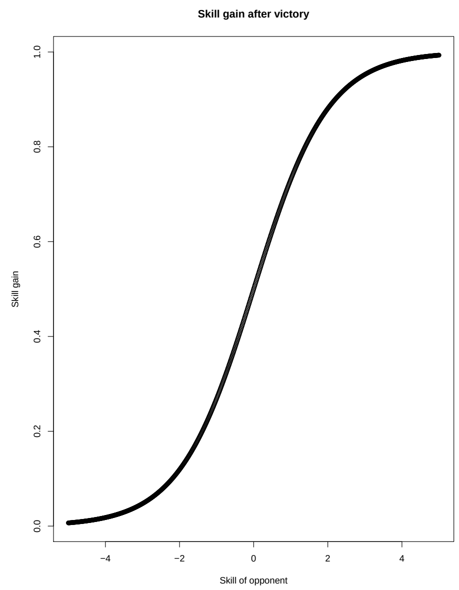

Figure 4.2: Skill increase after winning a match based on skill of opponent

4.2.2 Map-dependant variant

Since Counter-Strike can be played on more than one map, an alternative approach building upon the presumption that the choice of a map has either positive or a negative effect on the strength of either team.

Under this assumption we consider the teams to have different strength on each map, and therefore the probability of team i defeating team j on map m in time t is denoted by

$$p_{i,j,m,t} = \frac{1}{1 + e^{-(S_{i,t}+M_{i,m}-S_{j,t}-M_{j,m})}} \tag{4.6}$$

with the skill $S_{i,t}$ of team i once again following recursion

$$\begin{aligned}
S_{i,t+1} &= C_i + B_i S_{i,t} + A_i \frac{e^{S_{k,t}+M_{k,o}}}{e^{S_{i,t}+M_{i,o}} + e^{S_{k,t}+M_{k,o}}} && \text{if } i \text{ defeated } k \text{ on map } o \text{ in time } t,\\
S_{i,t+1} &= C_i + B_i S_{i,t} - A_i \frac{e^{S_{i,t}+M_{i,o}}}{e^{S_{i,t}+M_{i,o}} + e^{S_{k,t}+M_{k,o}}} && \text{if } i \text{ lost to } k \text{ on map } o \text{ in time } t,\\
S_{i,t+1} &= S_{i,t} && \text{if team } i \text{ did not play in time } t,
\end{aligned} \tag{4.7}$$

where $\theta_i = (C_i, B_i, A_i, M_{i,1}, \ldots, M_{i,n_M})'$ are the parameters for team i. The parameter $M_{i,m}$ denotes the relative strength of team i on map m with the value of $M_{i,1}$ set to 0 due to the issue of identifiability.

In table 4.1, we can see the adjustments to skill of two example teams on all eight maps, with Cache as the reference category. Interestingly, we learn that both team have Train as their estimated best map. This learning might be useful to the teams themselves, because if the teams were to face each other in this situation, Immortals should not try select their best map Train, which is incidentally the map their opponent is strongest at, but instead opt for Nuke, where the difference between the two teams appears to be the greatest.

Map	**Map-specific skill** Astralis	**Map-specific skill** Immortals
Cache	0.000	0.000
Cobblestone	0.668	0.236
Dust2	0.481	0.083
Inferno	0.217	-0.680
Mirage	3.574	2.716
Nuke	-2.801	1.043
Overpass	0.359	-1.063
Train	3.763	3.697

Table 4.1: The performances of two selected teams on all eight availible maps relative to the reference category

4.2.3 Static model

In the simplest variant, we model the probability of team i defeating team j as

$$p_{i,j} = \frac{1}{1 + e^{-(S_i - S_j)}} \tag{4.8}$$

where S_i and S_j are the strengths of respective teams i and j, computed from training data and staying constant for the testing data set.

This model assumes that over the time period considered, the strength of every team is constant. This approach is very naive and its aim is mostly to provide a good benchmark for previous models to compare against.

4.3 Evaluating model performance

The main metric used in this section to evaluate models is classification accuracy, derived from the confusion matrix as seen in figure 4.3, consisting of four outcomes. True positive when the model correctly forecasts the value of 1, false positive when the model forecasts 1 but the actual value was 0, true negative when the model correctly forecasts the value of 0 and false negative when the model returns 0 while the actual value was 1. Accuracy is then given by

$$\text{Accuracy} = \frac{\text{TP} + \text{TP}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4.9}$$

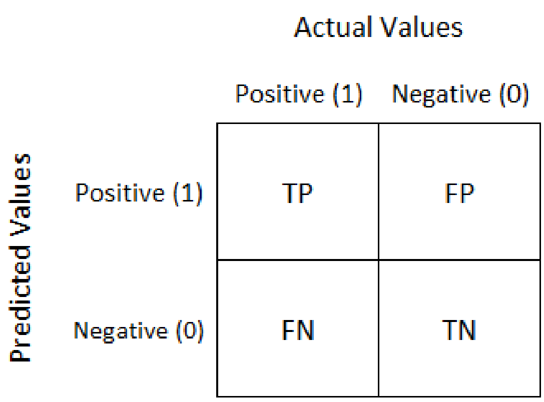

Figure 4.3: Confusion matrix schema

4.3.1 Dynamic versus static modelling

For this testing session, we used 6673 matches played between the top 50 teams from the beginning of 2016 to the end of year 2018. Year 2016 was designated as training period, while the remaining two years were used to test the prediction accuracy. The results of all three models are shown in table 4.2.

Model	**Predicted matches**	**Accuracy**
Static	4543	0.5460
Map-specific GAS	4543	0.6122
Map-independent GAS	4543	0.5784

Table 4.2: Overview of GAS and static models accuracy and the number of predictions made

Based on these results, we can compute pairwise two-sample Z-tests to confirm whether the improved accuracy between these models is statistically significant. Starting with confirmation that the static model is truly under performing when compared to the other two, which can be seen in table 4.3. As expected, with the p-value approaching zero, the dynamic model possesses significantly stronger predictive power.

Model	**n**	**p**	**s.e.**	**p-value**
Static	4543	0.5460	0.05675	0.000
Dynamic	4543	0.5817	0.05675	–

Table 4.3: A two-sample Z-test between static and dynamic model using results from matches played between top 50 teams in 2017 and 2018

Also we need test whether the additional increase in accuracy obtained from expanding the dynamic model to consider the adjustment to performance based on what map the match was played is truly because of superior predictive power or attributed to variance. The results are displayed in figure 4.4. We can see that once again, there has been an increase in predictive power when comparing these two models. However, with the time needed to execute the model compared to the map-independent variant, it is questionable whether the improvement would be worth it for repetitive usage.

Model	**n**	**p**	**s.e.**	**p-value**
Map-specific	4543	0.6122	0.05675	0.000
Map-independent	4543	0.5784	0.05675	–

Table 4.4: A two-sample Z-test between static and dynamic model using results from matches played between top 50 teams in 2017 and 2018

In figure 4.4 we illustrate how the skill of teams over time is modelled. We show the three top teams from the modeled era, Astralis, which was considered the top team in the world for the majority of the observed period. SK, a team that only assembled their roster in late 2016 and had a very solid run during the year 2017 and Natus Vincere, which started to rise with some squad changes and was considered the top team for the second half of 2018.

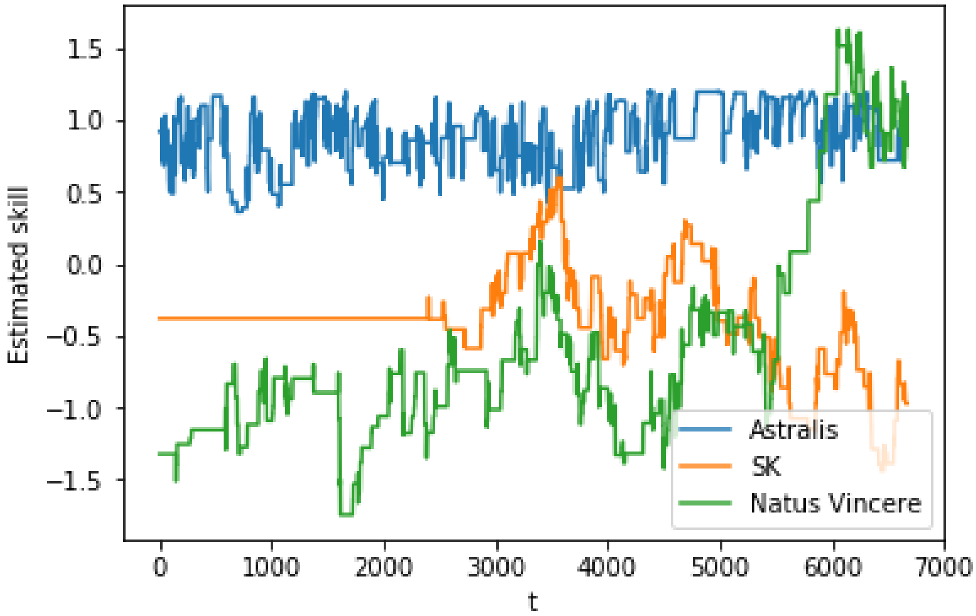

Figure 4.4: The perceived skill of the top 3 teams from the data set over time

We can see that the model correctly managed to represent all of the major events in the timeline – the improvement of Natus Vincere, the late start and early success followed by a drop in performance by SK and the dominance of Astralis, even though it seems to strictly overrate the Astralis team, which while being one of the most consistent teams in the scene, still had weaker showings in the fall and winter of 2017 and their actual results placed them into the top four at best.

4.4 Conclusions

Our findings confirmed the initial hypothesis we had, that is that adding more information and building a more complex model does lead to an increase in predictive power. In this case, significance testing confirmed that the map-specific model yields the most accurate predictions on our data set, followed by the map-independent model and the static one. However the time it takes to execute these models, which is counted in hours for the map-specific variant gives some arguments towards using other approaches.

We tried out several different weights for the score-driven and autoregressive parameters and reported better forecast accuracy the more weight shifted towards the score parameter, which makes the model slightly quicker to adapt in case the team experiences a big change performance such as transferring multiple players.

An issue which leads to decrease in overall accuracy is that the model tends to overestimate the difference in skill between the best team and the second best. As we used Bernoulli distribution to forecast the match winner, tt effectively predicts Astralis to win almost all of their matches.

5. Common opponent model

5.1 Introduction

As the professional scene in Counter-Strike and other e-sports titles is split into several tiers, called the Premier, Major and Minor circuit (loosely similar to the concept of leagues in football for example) and also considering the geographical regions, teams play against teams similar to them in skill and strategy more often than not. When using basic statistics such as percentage of won matches, we might run into several issues thanks to this bias.

Considering a strong team that plays in the highest-level events and a weak one that mostly plays in regional-level tournaments it can be easily seen that the notion of an 'average' opponent is significantly different for these two teams. If we computed an array of variables to predict the strength of these two teams, it would be highly probable that the 'weak' team, which is actually dominating in it's region would be deemed more likely to win a heads-on match against the 'strong' team.

The goal of the common opponent approach is to overcome exactly this obstacle. The model is designed to take advantage of the concept of transitivity and it's presence in the e-sports scene by basing the outcome prediction only on data which is common for both teams modelled. This is achieved simply by finding a subset of opponents that have played a game against both modelled teams and then use only the statistics from matches played against this subset of opponents. This approach ensures that the 'average' opponent is approximately of the same skill for both modelled teams.

5.2 Match as a discrete-time Markov chain

The hierarchical approach is popular when modelling sports which use the point scoring system, where the first one to n points wins the match. Originally introduced by Kemeny and Snell [30], who modelled a tennis match using a Markov chain. The idea behind it is simple – in order to win a game, the player must win a number sets, for which he needs to win a number of games, for which he needs to win a number of rallies. This method was further developed for sports using this method of scoring, such as volleyball [31] or tennis [32]. The same logic can be directly applied to Counter-Strike matches.

Using this approach and under the assumption that points are independent of each other and identically distributed, it is possible to model the whole match using only a single parameter – the probability of winning a round.

We can visualize such a Markov chain in figure 5.1, where every game starts in the initial state where both teams have zero points. Then every time team 1 wins a round with probability p, we move one step to the left and when team 2 wins a point with probability $1 - p$, we move one step to the right. This way we are able to

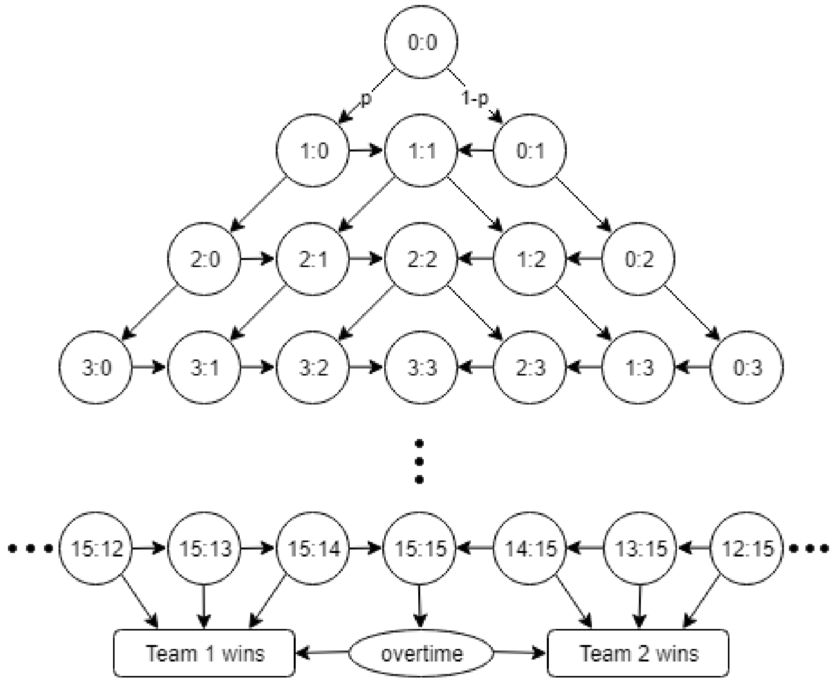

Figure 5.1: Markov chain of a Counter-Strike match

5.3 Calculating probability to win a match

Let $M(p)$ be a simple function which estimates a probability of a team winning a Counter-Strike match, commonly denoted as a best-of-30, where the first team to reach 16 points wins, or in the case of a 15 – 15 tie, a team needs to win by two points in overtime. Parameter p represents the probability of the team winning a round.

In order to calculate the probability to win a match, we need to sum the p of every possible chain of events which leads to the team winning a match. As the match can be modelled as a discrete-time Markov chain, this can be easily done.

In this case, the match can have 16 possible outcomes, fifteen of them being with the team winning 16 – n and the last one an overtime victory with a score of $(n + 2)$ to n. In order to be able to calculate the probability of victory, we need to sum up all of these possibilities, where the respective possible outcomes can be calculated as:

$$
\begin{aligned}
[16-0]: &\quad p^{16} \\
[16-1]: &\quad p\binom{16}{1}p^{15}(1-p) \\
[16-2]: &\quad p\binom{17}{2}p^{15}(1-p)^2 \\
&\vdots \\
[16-14]: &\quad p\binom{29}{14}p^{15}(1-p)^{14}
\end{aligned} \tag{5.1}
$$

These are all pretty straightforward, we simply calculate the probability of getting to a match state just before the winning point is scored and multiply that value by p. With the last outcome this approach needs to be adjusted, first we calculate the $15-15$ outcome and then the probability of winning from there as

$$
\binom{30}{15}p^{15}(1-p)^{15}\frac{p^2}{1-2p(1-p)} \tag{5.2}
$$

To derive the value from M(p) from these building blocks, we then calculate the sum of these 16 equations

$$
M(p) = \Pr(16{:}0) + \Pr(16{:}1) + \cdots + \Pr(n+2:n). \tag{5.3}
$$

5.4 On the concept of transitivity

Complete transitivity applied in this area would mean that if Team A defeats team B and Team B defeats team C it must hold true that Team A would defeat team C if these two were matched against each other. In other words, with complete transitivity in effect, there must exist a perfect ranking order of all teams where the first team is the strongest one and will always win against any lower ranked opponent.

It is very obvious that complete transitivity does not hold true in almost any kind of sport, e-sports being no exception to the rule. Nevertheless, it is safe to assume that some form of transitivity exists. This transitive aspect is what allows us to make any kind of prediction based on previous performances and/or results against an opponent of a similar strength.

This reasoning has the potential to yield interesting results, as the professional Counter-Strike scene is small, consisting of less than 100 teams. Even though this means there is a limited number of encounters in the relevant time-frame for two specific teams, the set of common opponents should be large enough to allow for a robust prediction.

5.5 Possible variants of the model

As the model was originally designed to be used for predicting tennis matches, there are several possible adjustments to be made when modelling a different game.

5.5.1 Basic variant

A straightforward adaptation of Knottenbelt's model [6], where we consider teams A and B be the teams whose match we wish to model. Then let C_i for $1 < i < N$ be the set of N common opponents these two teams both faced in the past. For each C_i we can compute $rw(A{,}C_i)$ as the proportion of rounds won by A against C_i and $rw(B{,}C_i)$ as the proportion of rounds won by B against C_i.

In the case where either team A or team B has played multiple matches against one opponent C_i in the learning data set, there are two possible avenues to take. Either compute $rw(A{,}C_i)$ and $rw(B{,}C_i)$ as averages over all matches against C_i or model those encounters as multiple instances of common opponent matches.

As discussed earlier, it is possible to compute the probability of a team winning a match against an opponent based on the difference in rounds won. In order to model how A and B would play against each other through results against their common opponents C_i, we first need to calculate the differences between $rw(A{,}C_i)$ and $rw(B{,}C_i)$ against the set of common opponents. Then we can use those differences to predict the result of a match between A and B.

This way we can compute Δ_i^{AB}, which represents the advantage or disadvantage team A has over team B when considering the proportion of rounds won against common opponent C_i as $\Delta_i^{AB} = rw(A{,}C_i) - rw(B{,}C_i)$.

Let $M(p)$ be a function which estimates a probability of a team winning a match where parameter p represents the probability of the team winning a round. Using this function and the value Δ_i^{AB}, we can approximate the probability of team A defeating team B in a match based on their previous results against common opponent C_i:

$$\Pr(\text{A beats B via } C_i) \approx M(0.5 + \Delta_i^{AB}). \tag{5.4}$$

Important to note that this equation can become invalid if we consider two teams with so significantly different strength that Δ_i^{AB} falls outside the interval $-0.5 < \Delta_i^{AB} < 0.5$ In order to prevent our model failing on this exception, we need to cap the possible values within the boundaries $(-0.5, 0.5 >)$.

To get the probability of team A defeating team B, we calculate the average Pr(A beats B via C_i) across all common opponents, combining the data into a single value:

$$P_{\text{avg}}^{AB} = \frac{\sum_{i=1}^{n} \Pr(\text{A beats B via } C_i)}{N} \tag{5.5}$$

In order to illustrate the common opponent approach, we model an example match. Table 5.1 shows the rw statistic against common opponent for both teams. In the case where one team faced the common opponent multiple times, the value represents the arithmetic mean across all encounters.

Opponent	**fnatic** rw	**G2** rw
AGO	46.3%	76.4%
Astralis	35.7%	19.7%
BIG	47.2%	42.1%
FaZe	47.7%	50.5%
HellRaisers	34.3%	45.8%
Heroic	53.2%	64.9%
LDLC	45.9%	44.4%
Liquid	47.6%	33.3%
Natus Vincere	44.2%	41.9%
NiP	53.8%	65.2%
North	48.6%	35.5%
OpTic	66.7%	57.7%
Virtus.pro	41.6%	39.5%
Windigo	42.6%	34.6%
compLexity	53.3%	52.7%
forZe	59.6%	54.5%
mousesports	45.5%	31.0%

Table 5.1: Proportion of rounds won from matches played against common opponents for teams fnatic and G2. The data includes all common opponent matches played in the last three months before the modelled match.

From the table 5.1 we can estimate the relative advantage or disadvantage of fnatic against G2 and using the function $M(p)$ calculate the probability of fnatic winning the match given the data against common opponent C_i. These results are presented in table 5.2. Taking the average we get an estimate of 0.581 for fnatic to win the match. If we consider the negative binomial distribution as the underlying distribution for number of rounds won, the outcome with maximum probability would be $16 - 10$. This prediction was fairly accurate, as the actual match ended up being $16 - 11$ in favor of fnatic.

It's worth noting that in table 5.2 there is significant variation between the estimated win probabilities which suggests that the model needs a sufficiently populated set of common opponents in order to produce stable result estimates.

Opponent	Δ	**Pr (fnatic defeats G2)**
AGO	-0.30	0.0%
Astralis	0.16	97.2%
BIG	0.05	72.2%
FaZe	-0.03	37.5%
HellRaisers	-0.11	9.0%
Heroic	-0.12	8.4%
LDLC	0.01	56.7%
Liquid	0.14	95.4%
Natus Vincere	0.02	60.4%
NiP	-0.11	8.9%
North	0.13	93.9%
OpTic	0.09	85.3%
Virtus.pro	0.02	59.9%
Windigo	0.08	82.4%
compLexity	0.01	53.1%
forZe	0.05	72.4%
mousesports	0.15	95.8%
Average		**58.1%**

Table 5.2: Probability of fnatic defeating G2, showing contribution data from every common opponent C_i in its own row.

5.5.2 Map dependant model variant

Since Counter-Strike can be played on more than one map and team performances are known to vary across different maps, a model that considers what map is going to be played should theoretically lead to an improved predictive ability compared to the more naive basic variant.

We build upon the model introduced in this section, however several changes are needed to proceed. As every map has slightly different advantage or disadvantage for the attacking team, we need to expand the function $M(p)$ into $M(p,q)$ where the parameter p now conveys the probability of winning a round while playing as the attacking side on a given map and the added parameter q as the probability of winning a round as the defending side on the same map.

With m being the coefficient denoting advantage or disadvantage for the attacking side, we get this equation:

$$\Pr(\text{A beats B on map M via } C_i) \approx \frac{M(m+\Delta_i^{AB},(1-m)) + M(m,(1-(m-\Delta_i^{AB})))}{2} \tag{5.6}$$

In Equation (5.6), we calculate the match probabilities twice, using the method suggested by Spanias for tennis matches, where players have different probability to score a point if they serve opposed to receiving [24]. Once we positively influence team A's probability of winning a round and once we negatively influence team B's probability of winning a round, then return the average of the two values.

As there are usually seven eligible maps, only considering matches played on a specific one effectively removes about 85% of the data set, therefore in order to avoid having too few results to predict we treat every encounter against the same common enemy as a data point this time.

Opponent	**Astralis**	**Natus Vincere**	**Δ**	**Pr (Astralis defeats Na'Vi)**
FaZe	80.0%	15.8%	0.50	100.0%
FaZe	80.0%	15.8%	0.50	100.0%
Liquid	61.5%	20.0%	0.42	100.0%
Heroic	76.2%	64.0%	0.12	89.5%
North	84.2%	53.3%	0.31	100.0%
North	5.9%	53.3%	-0.47	0.0%
MIBR	100.0%	76.2%	0.24	99.7%
MIBR	69.6%	76.2%	-0.07	17.2%
HellRaisers	55.2%	84.2%	-0.29	0.0%
Average				**67.4%**

Table 5.3: Proportion of rounds won from matches played against common opponents for teams Astralis and Natus Vincere, together with win probability.

In table 5.3 we can see the modelled results of a match between teams Astralis and Natus Vincere. In this example we can notice that the Δ on the first two rows have been capped to 0.5 as denoted before. Also, we can see that for example team North appears twice with the same statistics against Natus Vincere in both cases, since there was only one match, while Astralis faced the same opponent twice.

The modelled match ended with a one-sided victory where Astralis won 16 – 4. The model correctly identified the winning team in this case, suggesting 16 – 7 as the most probable outcome.

5.5.3 Time-weighted model variant

In the literature published on this approach there were discussions about whether to use 3, 6 or 12 months of data before the simulated match, the outcomes compared by Spanias, [24] but there is was only a mention of utilizing weighted data, using data from a longer period with the most significance placed on recent data and diminishing with increasing time.

This kind of approach was already considered in the Dixon and Coles paper on forecasting soccer matches [33] which proposed a weighting function:

$$\phi(t) = exp(-\xi t) \tag{5.7}$$

which exponentially decreases the weights of all previous results according to the parameter $\xi > 0$.

Figuring out the optimal value of ξ can be a complicated task. Correct but computationally very expensive approach would be to write an optimized which iterates through all possible values of ξ and selects the one leading to the highest prediction accuracy. Another approach, which possibly sacrifices some accuracy for drastically shorter computational times is to iterate only over a specified subset of values ξ.

In figure 5.2 we can see the visualization for three different values of parameter ξ. As our hypothetical modelled match took place on the first of January, all weights from following dates are set to zero.

As the matches are often not played at the same time and there might have even be weeks or months between when the two modelled teams encountered the common opponent, it is necessary to perform some additional steps for the method to provide relevant results.

First, when looking at multiple matches against one common opponent C_i, we transform the weights $\phi(t)$ through

$$v_{i,n} = v_{i,n} / \frac{v_{i,n}}{\sum_{n=1}^{N} v_{i,n}} \tag{5.8}$$

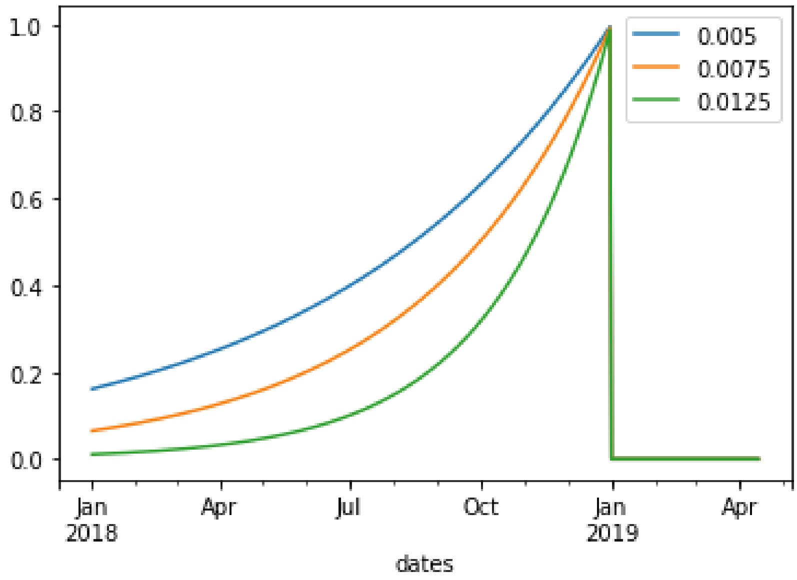

Figure 5.2: Weights for matches played up to one year before the modelled match based on value of ξ

where the sum of transformed weights v_n of all N matches against opponent i is equal to 1. In the case of multiple encounters against a common opponent, this allows us to assign more weight to more recent matches.

Also, after iterating through multiple matches against the same common opponent C_i, we also need to weigh the final information taken from each common opponent. There are two approaches, which both have some merit. As the two teams haven't most probably played the common opponent at the same time, we select either the minimum or the average of the match dates played against that opponent and then weigh information from that common opponent using equation 5.7.

5.6 Evaluating model performance

5.6.1 Map-specific and general models

We tested the models in several iterations, changing the amount of historic data available to the model for predicting the match result and arrived at the results observable in table 5.4, which shows that the general model performs better in every iteration. This can be explained by the average size of a set of common opponent encounters used to forecast the result, which is less than 10 on average for the map-specific variant and in some cases dropped as low as 2, which made it very sensitive to outliers.

monthsBack	**General** *accuracy*	**Map-specific** *accuracy*
3	0.592	0.545
4	0.596	0.532
5	0.601	0.537
6	0.595	0.542
7	0.595	0.544
8	0.591	0.560
9	0.590	0.561

Table 5.4: Prediction accuracy of general versus map-specific common opponent model depending on the amount of historic data

It is also worth noting that as we increase the amount of historic data available to the model, the accuracy of map-specific model goes up, while the accuracy of the general model starts to drop after we include over five months of data. This shows that in such a dynamic field as e-sports where teams can change their roster twice in a few months, accuracy drops with prolonging the historical learning data set. This is also the case of the map-specific model, however it's accuracy went up as we gain more encounters and the model itself becomes less vulnerable to outliers and correct itself in some cases where it used to predict incorrectly given smaller data samples.

This phenomenon can be illustrated in figure 5.3, where we look at the match between fnatic and G2 which was already used as an example earlier in the chapter and let the model predict it based on different historic data samples. When 6 or less months were included, the model correctly predicted fnatic winning the match, even though the accuracy decreased over time, but on more than 6 months of data, the model would actually predict the opposite result.

5.6.2 Values of ξ in the time-weighted model

We tested four different values of ξ and the results of running the model with those can be found in table 5.5. The best results were achieved using $\xi = 0.0125$ with accuracy 61.27%. With further increases in the value of ξ the model was basing the prediction on less and less matches, as the higher the value of ξ is, matches closer to the predicted one get assigned the weight of 0, shrinking the data set available to forecast the result from.

Also, we can look on the effect of ξ on prediction through figure 5.4 which again shows the predicted chance for fnatic to defeat G2. It is visible that with increasing value of ξ the model assigns more and more weight to the most recent encounters, which were positive for fnatic, while with $\xi = 0.1$ and less, forecasts the opposite result, based on the encounters from more

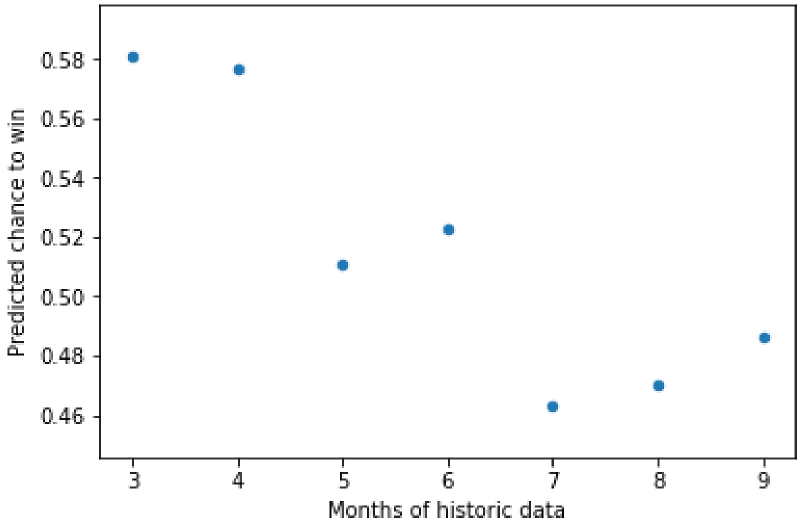

Figure 5.3: Predicted chance for fnatic to defeat G2 depending on the amount of historic data

ξ	**Accuracy**
0.0775	0.6059
0.0125	0.6107
0.0175	0.6015
0.2225	0.5978

Table 5.5: Time-weighted model accuracy based on values of xi

distant history.

5.6.3 Comparison of all three models

For this comparison, we select the variants of each model which led to the highest prediction accuracy and test whether there are any significant differences between their performances. These can be found in table 5.6. Worth noting is the number of predicted matches out of roughly 7000 in the testing set. The map-specific variant was only able to predict about 60% of all played matches because of the constraints placed on it.

Based on these models, we can compute pairwise two-sample Z-tests to confirm whether the improved accuracy between these models is statistically significant. First we compare the general and map-specific models, the results are shown in table 5.7 and can easily be interpreted. As the p-value is effectively equal to zero, the two samples have a probability close to zero to have the same mean and therefore the performance of the general model is significantly better than the map-specific one.

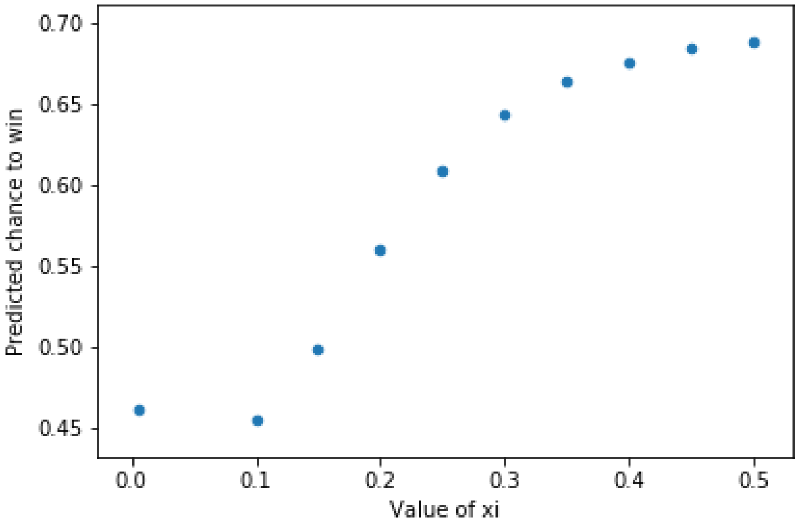

Figure 5.4: Predicted chance for fnatic to defeat G2 depending on the value of xi

Model	**Predicted matches**	**Accuracy**
General	6579	0.601
Map-specific	4184	0.561
Time-weighted	6608	0.611

Table 5.6: Overview of common opponent model variants' accuracy and number of predictions made

Also, we can test the general model, which was deemed to be significantly superior to the map-specific one against the time-weighted variant. The results are shown in table 5.8. With p-value of 0.0041, we can say that with certainty of 99.5%, the time-weighted model offers a significant improvement in accuracy compared to the general one and through transitivity also the map-specific variant.

5.7 Conclusions

We discussed and tested three model variants and found out that the time-weighted model has the strongest predictive ability. Contrary to our original hypothesis, the map-specific variant of the common opponent approach did not yield better results, unlike tennis where differentiating between clay, grass and hard surfaces had positive effect on the model. The reason behind this is that in tennis, the subset of available matches decreases by $\frac{2}{3}$, while in Counter-Strike it goes down by $\frac{7}{8}$ of the original size, leaving not enough matches in the common opponent subset to make robust and unbiased predictions.

Model	**n**	**p**	**s.e.**	**p-value**
General	6579	0.601	0.05611	0.000
Map-specific	4184	0.561	0.05689	–

Table 5.7: A two-sample Z-test between general and map-specific model using results from matches played by top 50 teams in 2018

Model	**n**	**p**	**s.e.**	**p-value**
General	6579	0.601	0.05611	0.0041
Time-weighted	6608	0.611	0.05603	–

Table 5.8: A two-sample Z-test between time-weighted and map-specific model using results from matches played by top 50 teams in 2018

Overall the adaptation of the model proved to be a feasible approach, reporting slightly less prediction accuracy than the tennis application, nevertheless this can be attributed to the more dynamic environment of e-sports and the frequent roster changes happening, which makes past results not just obsolete, but also detrimental to the forecast as a result.

Conclusion

In the thesis, we presented two different models which share the common trait of only requiring publicly available data in order to forecast future match results. Some conclusions from the applications of both models were already presented in their respective chapters, however the comparison between the two approaches is still due.

The common opponent approach is the easier model to build and test and is also able to process data faster. However it only offers us the probability for one team to defeat another given specific circumstances. When compared to the GAS approach, we are also able to learn the ranking of all modelled teams in a given point in time, or approximate how well they perform on different maps, which might be actually useful information for the teams to have, when they decide on which maps to select against which opponent.

Results comparison

Model	Variant	Matches predicted	Accuracy
Static	-	4543	0.546
GAS	general	4543	0.578
GAS	map-specific	4543	0.612
Common opponent	general	6579	0.601
Common opponent	map-specific	4184	0.561
Common opponent	time-weighted	6608	0.611

Table 5.9: Comparison of prediction accuracy of the best iterations of all discussed models

In the table 5.9 we can find the results of testing all the discussed models. From the six models presented, three of them report accuracy over 60%, two of the common opponent variants and the map-specific GAS model. We can do one final Z-test in order to compare the two models with the best results, the outcome of which is shown in table 5.10. With the p-value of 0.299, we can not proclaim the difference in means as statistically significant.

Therefore in the end, the best iteration of both discussed approaches reports similar predictive power.

Model	**n**	**p**	**s.e.**	**p-value**
GAS Map-specific	4543	0.612	0.05675	0.299
CO Time-weighted	6608	0.611	0.05603	–

Table 5.10: A two-sample Z-test between the best performing generalized autoregressive score model and the best performing common opponent model

Further research

Future research based on the approaches and models presented in this thesis may lead to improved performance and new findings. This section quickly outlines possible directions of future research in this area.

Solving the lack of data in common opponent approach

With the map-specific common opponent approach, one of the shortcomings was the reduction in available historical data due to the strict use of only matches played on the same map. One possible avenue, already proposed by Spanias [24] is to introduce a two-tier common opponent model.

Using this approach, we can look at a team $C_{a,1}$ that has played only against Team B but never against team B by utilizing the set of common opponents between teams $C_{a,1}$ and B. This is then repeated for all possible opponents $C_{a,i}$ that faced team A but not team B. The underlying transitive logic, where team A beats team $C_{a,i}$ who beats team $C_{b,i}$ who beats team B, then team A is likely to defeat team B.

Further research would be needed to understand whether this might enhance or diminish the model's performance.

Analyzing possible correlation between maps

In the map-specific model, we essentially assigned the weight $w = 1$ to past matches that shared a map with the modelled match and zero to the remaining ones. However another approach which would lead to an increased sample size is to explore possible similarities between maps. If we could prove that there is a correlation between match results played on map X and map Y, we might be able to allow the correlated map into the model with a non-zero weight.

Including player transfers into the model

When compared to classic sports, there are cases of major roster changes in e-sports, such as teams replacing the whole squad, which leads to a sudden drastic change in the team performance. An alternative avenue of modelling would not to model the performance of a team as a whole, but a team as a combination of five distinct players.

References

1. STATISTA.COM. *eSports market - Statistics & Facts*. 2019. `https://www.statista.com/topics/3121/esports-market/`.
2. ANDONO, Pulung Nurtantio; KURNIAWAN, Nanang Budi; SUPRIYANTO, Catur. DotA 2 bots win prediction using naive bayes based on adaboost algorithm. In: *Proceedings of the 3rd International Conference on Communication and Information Processing*. 2017, pp. 180–184.
3. WANG, Nanzhi; LI, Lin; XIAO, Linlong; YANG, Guocai; ZHOU, Yue. Outcome prediction of DOTA2 using machine learning methods. In: *Proceedings of 2018 International Conference on Mathematics and Artificial Intelligence*. 2018, pp. 61–67.
4. CREAL, Drew; KOOPMAN, Siem Jan; LUCAS, Andre. *A General Framework for Observation Driven Time-Varying Parameter Models*. 2008. Available also from: `http://www.tinbergen.nl/discussionpaper/?paper=1416`.
5. KOOPMAN, Siem Jan SJ; LIT, Rutger, et al. *Forecasting Football Match Results in National League Competitions Using Score-Driven Time Series Models*. 2017. Technical report. Tinbergen Institute.
6. KNOTTENBELT, William J; SPANIAS, Demetris; MADURSKA, Agnieszka M. A common-opponent stochastic model for predicting the outcome of professional tennis matches. *Computers & Mathematics with Applications*. 2012, vol. 64, no. 12, pp. 3820–3827.
7. PAPOULIS, Athanasios; PILLAI, S Unnikrishna. *Probability, random variables, and stochastic processes*. Tata McGraw-Hill Education, 2002.
8. PARSHAKOV, Petr; ZAVERTIAEVA, Marina. Success in eSports: Does Country Matter? *Available at SSRN 2662343*. 2015.
9. PARSHAKOV, Petr; COATES, Dennis; ZAVERTIAEVA, Marina. Is diversity good or bad? Evidence from eSports teams analysis. *Applied Economics*. 2018, vol. 50, no. 47, pp. 5064–5075.
10. MAKAROV, Ilya; SAVOSTYANOV, Dmitry; LITVYAKOV, Boris; IGNATOV, Dmitry I. Predicting Winning Team and Probabilistic Ratings in “Dota 2” and “Counter-Strike: Global Offensive” Video Games. In: *International Conference on Analysis of Images, Social Networks and Texts*. 2017, pp. 183–196.
11. YANG, Yifan; QIN, Tian; LEI, Yu-Heng. Real-time esports match result prediction. *arXiv preprint arXiv:1701.03162*. 2016.
12. HODGE, Victoria; DEVLIN, Sam; SEPHTON, Nick; BLOCK, Florian; DRACHEN, Anders; COWLING, Peter. Win Prediction in Esports: Mixed-Rank Match Prediction in Multi-player Online Battle Arena Games. *arXiv preprint arXiv:1711.06498*. 2017.

13. DRACHEN, Anders; YANCEY, Matthew; MAGUIRE, John; CHU, Derrek; WANG, Iris Yuhui; MAHLMANN, Tobias; SCHUBERT, Matthias; KLABAJAN, Diego. Skill-based differences in spatio-temporal team behaviour in defence of the ancients 2 (dota 2). In: *2014 IEEE Games Media Entertainment*. 2014, pp. 1–8.
14. SCHUBERT, Matthias; DRACHEN, Anders; MAHLMANN, Tobias. Esports analytics through encounter detection. In: *Proceedings of the MIT Sloan Sports Analytics Conference*. 2016, vol. 1, p. 2016.
15. HARVEY, Andrew C; CHAKRAVARTY, Tirthankar. Beta-t-(e) garch. 2008.
16. CREAL, Drew; KOOPMAN, Siem Jan; LUCAS, André. Generalized Autoregressive Score Models with Applications. *Journal of Applied Econometrics*. 2013, vol. 28, no. 5, pp. 777–795. ISSN 0883-7252. Available from DOI: `10.1002/jae.1279`.
17. AVDULAJ, Krenar; BARUNIK, Jozef. Are benefits from oil “stocks diversification gone? New evidence from a dynamic copula and high frequency data. *Energy Economics*. 2015, vol. 51, pp. 31–44.
18. ZHAO, Lin; WIJNBERGEN, Sweder van. Decision-making in incomplete markets with ambiguity ”a case study of a gas field acquisition. *Quantitative Finance*. 2017, vol. 17, no. 11, pp. 1759–1782.
19. BERNARDI, M; CATANIA, Leopoldo. Switching-GAS copula models for systemic risk assessment. *arXiv preprint arXiv:1504.03733*. 2015.
20. KOOPMAN, Siem Jan; LIT, Rutger. 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)*. 2015, vol. 178, no. 1, pp. 167–186.
21. REEP, C.; BENJAMIN, B. Skill and Chance in Association Football. *Journal of the Royal Statistical Society. Series A (General)*. 1968, vol. 131, no. 4, pp. 581–585. ISSN 00359238.
22. RUE, Havard; SALVESEN, Oyvind. Prediction and retrospective analysis of soccer matches in a league. *Journal of the Royal Statistical Society: Series D (The Statistician)*. 2000, vol. 49, no. 3, pp. 399–418.
23. PIKHART, Miroslav; HOLY, Vladimir. Modelling e-Sports matches using generalized autoregressive score models. In: *Proceedings of the MME 2018 Conference*. 2018, vol. 1.
24. SPANIAS, Demetris. Professional Tennis: Quantitative Models and Ranking Algorithms.
25. MADURSKA, Agnieszka M. A set-by-set analysis method for predicting the outcome of professional singles tennis matches. *Imperial College London, Department of Computing, Tech. Rep.* 2012.
26. SIPKO, Michal; KNOTTENBELT, William. Machine learning for the prediction of professional tennis matches. *MEng computing-final year project, Imperial College London*. 2015.

27. KOVALCHIK, Stephanie Ann. Searching for the GOAT of tennis win prediction. *Journal of Quantitative Analysis in Sports*. 2016, vol. 12, no. 3, pp. 127–138.

28. HLTV.ORG. *CS:GO Statistic database*. 2019. `https://www.hltv.org/stats`.

29. KOOPMAN, Siem Jan; LUCAS, Andre; SCHARTH, Marcel. Predicting Time-Varying Parameters with Parameter-Driven and Observation-Driven Models. *Review of Economics and Statistics*. 2016, vol. 98, no. 1, pp. 97–110. Available from DOI: `10.1162/rest_a_00533`.

30. KEMENY, JG; SNELL, JL. Finite Markov Chains, Van Nostrand. *Princeton, New Jersey*. 1960.

31. SEPULVEDA, David; HILENO, Raúl; GARCIA DE ALCARAZ, Antonio. Analysis of Volleyball Attack from the Markov Chain Model. In: 2017.

32. RIDDLE, Lawrence H. Probability models for tennis scoring systems. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*. 1988, vol. 37, no. 1, pp. 63–75.

33. DIXON, Mark J; COLES, Stuart G. Modelling association football scores and inefficiencies in the football betting market. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*. 1997, vol. 46, no. 2, pp. 265–280.