The Impact of Toxic Behavior on Match Outcomes in DotA

Arjen Traas
ANR: 487051

HAIT Master Thesis

Thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Arts in Communication and Information Sciences,
Master Track Human Aspects of Information Technology,
at the School of humanities
of Tilburg University

Thesis committee:

Pieter Spronck
Sander Bakkes

Tilburg University
School of Humanities
Department of Communication and Information Sciences
Tilburg center for Cognition and Communication (TiCC)
Tilburg, The Netherlands
June 2017

Acknowledgments

First of all, a word of appreciation is in order for Märtens et al. [2015], who provided the raw data and Mark Verschoor, author of Verschoor [2016], who provided a Python script for data extraction. The well-structured raw data helped tremendously with building the subsets and without Mark's Python script, the process of pre-processing would probably take days or weeks longer.

While writing this thesis, dr. ir. Pieter Spronck (my thesis supervisor) was always available for feedback. Pieter, thanks for the structural guidance and flexibility. I would also like to thank the second reader, dr. ing. Sander Bakkes.

Abstract

Toxic behavior, a form of anti-social behavior, is a common occurrence in online games. While the nature and definition of toxic behavior remains vague and context dependent, this thesis tries to help understand the in-game consequences of toxic behavior. Our goal was to investigate the possible relation between toxic behavior and match outcomes in the popular MOBA DotA. We identified predictive variables of winning a match and used those variables to build a Prediction Model to predict match outcomes. We found that toxic teams, that is, a team for which a player initiates toxic behavior, have significantly lower chances of winning the game.

Contents

1 Introduction

1.1 Motivation

Video gaming has substantially increased in popularity the last decade: the number of people who play video games has reached 1.8 billion worldwide [McKane, 2016]. Multiplayer Online Battle Arena (MOBA) is a game genre that is particularly popular. Two common examples are League of Legends (LoL) and Defense of the Ancients (DotA). With respectively 100 million and 15 million monthly players [Wolmarans, 2016], these online games are both in the top five of player bases of all games [Paul, 2017]. A typical MOBA game consists of two opposing teams of five players who need to destroy their opponent's base. Every player controls one character in one of the two teams. These games are very team centric; team performance and communication are key to victory.

In order to communicate with other players, players can send messages in different chat channels. These channels were meant to facilitate team coordination [Märtens et al., 2015], but are often used to send 'toxic' messages. Toxicity is a form of anti-social behavior [Verschoor, 2016] and is a common situation in MOBA games ([Verschoor, 2016], [Märtens et al., 2015]).

Both from a societal perspective as from a business perspective toxic behavior is undesirable. Players who experience toxic behavior may be discouraged to play again or more frequently. Therefore, it is important that toxic behavior is discouraged and minimized.

While playing League of Legends myself, I encountered a message from the developer (Riot Games) stating that toxic players are 20 percent more likely to lose a game. This fascinated me, the outcome of a match is dependent on many variables and toxicity may be one of them. This made me curious about the relationship of toxicity and match outcome and that curiosity is one reason why this thesis will try to shine some light on the possible relationship between toxicity and match outcomes.

Moreover, the impact of toxic behavior has not often been subject of research, in contrast with the term cyberbullying, which has been investigated more frequently. Cyberbullying and toxic behavior bear great resemblance and therefore, the impact of toxic behavior is also related to the impact of cyberbullying. The latter is associated with both serious mental health issues (such as depression, anxiety, lack of self-esteem, emotional distress, substance (ab)use and suicidal behavior) and physical health issues [Nixon, 2014]. Because of the impact that toxic behavior can have on an individual, the impact of toxic behavior needs to be investigated thoroughly. This thesis will not examine health related consequences of toxic behavior, but in-game consequences. Once players are more aware of the in-game consequences, they might be discouraged to exhibit toxic behavior, which could ultimately help reducing the amount of toxic behavior and limit its impact.

1.2 Prior Research

Prior research about toxicity in online games often involve automatic identification of toxic behavior. Some of this research is done by analyzing player reports [Kwak et al., 2015], others analyzed chat messages ([Märtens et al., 2015]; [Verschoor, 2016]) to automatically label toxicity. Märtens et al. [2015] also tried to predict the winner of a match using chat

messages. Using the messages that were labeled as a precondition for toxicity, they did find some predictive power of the winning team, but they report a weak link.

Another type of research is the linguistic analysis of toxic messages. Kwak and Blackburn [2014] used over half a million toxic reported cases and identified several linguistic components to help understand how toxic messages typically are formed.

The third type of research involves the willingness to report another player in such a system. Kwak et al. [2015] found that only a small portion of toxic behavior is reported. This is an interesting finding, because it might implicate that players generally tolerate a certain amount of toxicity. A full, more detailed literature review is given in chapter 2.

1.3 Problem statement and research questions

As stated before, toxicity is a common occurrence in MOBA games [Verschoor, 2016]. Gaining more insight in the effects of toxicity on the outcome of a game may help players understand why it is undesirable to exhibit toxic behavior and ultimately help to improve the online experience of players.

This thesis will examine the possible relation of toxic behavior with match outcomes of the MOBA DotA. More specifically, it will try to identify variables that are predictive of winning a match, to evaluate the values of those variables at the moment a toxic message occurs and compare them to the values of the same variables at the end of the game. Assuming that players know what the current state of the game is, it will provide some insight in what the effect of toxicity is on game outcome.

Märtens et al. [2015] already trained a classifier based on pure text messages, and found a weak link between toxic messages and winning and losing a game. This thesis will have the same topic, but will have a different approach.

The problem statement of this thesis is:

To what extent is there a relationship between toxic behavior and match outcomes?

The following research questions will be investigated:

RQ1: What variables are predictive for winning a match of DotA?

In order to do this, match statistics at the end of DotA matches will be gathered. Via different metrics, correlations will be determined between a number of variables and match outcomes.

RQ2: What is the status of a DotA game, regarding the variables that are found in RQ1, at the moment a toxic message is sent?

To investigate this research question, the script of Verschoor [2016] will be used to retrieve and automatically identify toxic messages. Afterwards, the variables that were found in RQ1 will be evaluated at the moment a toxic message occurs. This way, we can gain some insight on the effect of toxicity on those variables.

RQ3: To what extent does the occurrence of toxic messages have an impact on match outcomes in DotA?

RQ3 is formed to evaluate the effect of toxic messages on match outcomes of DotA matches. The analyses that correspond with RQ1 and RQ2 are evaluated to determine a possible relationship.

1.4 Outline

The outline of this thesis will be described in this section. In chapter 2, the background of this thesis is given in the form of related work, in particular with respect to the work of Märtens et al. [2015] and Verschoor [2016]. In chapter 3, the methods that were applied regarding the pre-processing and analyzing of the data is presented. The results will be presented in chapter 4. In chapter 5, the results will be evaluated and placed in a broader perspective. Shortcomings and directions for further results will also be discussed in chapter 5. Finally, in chapter 6, an overall conclusion will be drawn and the research questions and problem statement will be answered.

2 Background

Relevant concepts and previous research will be discussed in this chapter. In the first section, we will briefly explain DotA. In 2.2, toxic behavior and common examples of toxic behavior are discussed. In 2.3, we will discuss motivations to play online games and why toxic behavior originates from those motivations to some extent. In 2.4, motivations for toxic behavior will be elaborated, which are based on motivations to play games. The background chapter will be concluded with a section about automatic detection of toxic messages, which has been researched more frequently the last years. In this thesis, we will make use of an algorithm that automatically detects toxic messages. Two papers will be discussed in particular: Märtens et al. [2015] and Verschoor [2016].

2.1 Defense of the Ancients (DotA)

In this thesis the MOBA Defense of the Ancients (DotA, not to be confused with DotA 2) is used to investigate toxic behavior. Before discussing the theoretical background, a short description of the game is given here.

DotA is an online game, where players play individual games which typically last about 30 to 50 minutes. Every player can enter a game, where there are two teams of five players each. The two teams are named after two races, namely *Scourge* and *Sentinel*. Each player selects one hero that he will be playing. The two teams fight each other to ultimately destroy the enemy's base called the Ancient. The map in which most games take place is divided into two parts, one for each team, separated by a river. The Ancients are at the center of each team's base and from there, three lanes form paths to the enemy's base: the top, middle and bottom lane. Between the lanes is a neutral area called the *Jungle*. In figure 1, a schematic representation of the map of DotA is given[1].

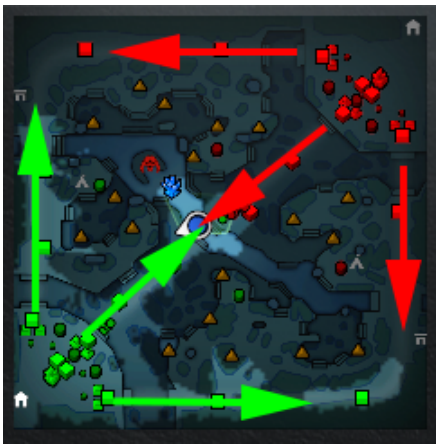


Figure 1: The map of DotA 2, virtually the same as the map of DotA.

[1]The map of DotA 2 is showed here, because we were unable to find copyright-free images of the DotA map. Differences between the maps of DotA and DotA 2 are minimal and for the purpose of this discussion inconsequential.

In order to destroy the Ancient, each team has to complete several smaller objectives. For example, the teams have to destroy turrets, which are structures that are placed in the lanes to defend the base. During the game, waves of creeps spawn regularly. Creeps are neutral AI bots that spawn at the base of both teams at exactly the same time and walk up the three lanes. Players can kill these creeps, as well as other players and turrets in order to gain gold and experience. Gold is used to buy and upgrade items, such as weapons and armor, to enhance the hero defensively, offensively or tactically for the duration of the game. Experience is used to level up the heroes. Leveling up the hero gives a boost in the base stats of the hero, such as damage and health.

2.2 Toxic behavior

In MOBA games such as DotA, the in-game chat function is useful in many ways. Players are able to communicate with their teammates to plan out a strategy and their opponents to chit-chat. Players could however, also verbally assault other players using the same chat system. These assaults often include heavy insults and are regarded as 'toxic'. Toxicity is a form of antisocial behavior [Verschoor, 2016]. Toxic behavior is generally considered as a group of negative types of behavior, such as cyberbullying, [Kwak et al., 2015], trolling and flaming [Verschoor, 2016]. However, the definition of toxic behavior often remains vague, because it largely depends on the type of game and its features, rules, customs and ethics [Kwak et al., 2015] as well as cultural differences [Warner and Ratier, 2005]. Cyberbullying (2.2.1), trolling (2.2.2) and flaming (2.2.3) will be explained in the remainder of this section.

2.2.1 Cyberbullying

In order to get a clear definition of cyberbullying, the definition of bullying in general should be determined. Bullying is usually defined as an aggressive and intentional act towards an individual that cannot easily defend himself, carried out by an individual or a group and occurs repeatedly [Olweus, 1993]. Physical, verbal and relational bullying are used to refer to 'traditional' bullying [Smith et al., 2008].

Cyberbullying is a form of bullying that has risen as information technology has evolved. Smith et al. [2008] describe cyberbullying as bullying via electronic means, specifically via mobile phones or the Internet and online games would fit in that definition as well. Cyberbullying is a form of habitual toxic behavior, meaning that it happens repeatedly. Cyberbullying can be seen as a serious problem of the online world. Over half the people active on the Internet are bullied online[2]. Moreover, as gaming is getting more popular with younger people, cyberbullying can cause far reaching problems [Kwak et al., 2015].

2.2.2 Trolling

Trolling is a more investigated term [Verschoor, 2016] and holds multiple definitions. Thacker and Griffiths [2012] describe trolling as an intentional act that provokes other

[2]Source: http://www.bullyingstatistics.org/content/cyber-bullying-statistics.html

users in an online environment to create a certain outcome, often not desirable for the troll or its teammates. Buckels et al. [2014] used the following definition: "the practice of behaving in a deceptive, destructive, or disruptive manner in a social setting on the Internet with no apparent instrumental purpose" [Buckels et al., 2014, p.97]. Herring et al. [2002] identified three types of messages that trolls often use:

1. Seemingly sincere messages
2. Messages that are purely sent to provoke predictable responses
3. Messages that are purely sent to waste the time of the reader by using futile arguments

In MOBA games, trolling often includes not playing to the best of your ability to provoke teammates.

2.2.3 Flaming

Flaming is sending a hostile message of strong emotions and can include swearing, insulting and name-calling [Lee, 2005]. Flaming is also described as "uninhibited behavior in computer-mediated communication (CMC)" [Kayany, 1998, p.1135]. While previous research stated that flaming is an effect of CMC [Kayany, 1998], Kayany [1998] states that social context is the primary determinant of flaming. This indicates that the social setting (the online game) is often the cause of flaming. Flaming in online gaming is almost exclusively via chat or voice-chat. As this thesis will analyze chat-messages, a substantial part of the toxic behavior analyzed will be flaming.

2.3 Motivations for online gaming

Toxic behavior is often present in online games. Motivations to play those games need to be discussed when exploring toxic behavior. Why people enjoy playing games is (among others) explained by Yee [2007]. Yee [2007] mentions three factors of why people play games, namely achievements (2.3.1), social factors (2.3.2) and immersion (2.3.3). While there are numerous theories about why people play games ([Csikszentmihalyi, 2014]; [Vorderer and Ritterfeld, 2004]; [Sherry et al., 2014]; [Przybylski et al., 2010]; [Tekofsky, 2017]), this thesis focuses on the well-known, understandable factors formed by Yee [2007]. The motivations to play games are crucial to understand why people exhibit toxic behavior. The motivations for toxic behavior (discussed in 2.4) originate from the motivations to play games.

2.3.1 Achievements

Yee [2007] formed three main components to why people enjoy playing games, by building on Bartle's Player Types [Bartle, 1996]. Yee [2007] specifically focused on Massively Multiplayer Online Role-Playing Games (MMORPG), but is also relevant for MOBA games. First of all, Yee [2007] states that people who play online games want to achieve things. These goals vary in form and complexity. Leveling up a character (quickly) could

be a goal, as well as accumulating in-game resources (gold, weapons, etc.). In DotA, the accumulation of resources is important, as more gold equals more items, which makes a character stronger. Players like to have constant improvement and to gain power in numerous ways.

Also, in some games, there is a high level of 'mechanics'. This is defined as the "interest in analyzing the underlying rules and system in order to optimize character performance" [Yee, 2007, p. 773]. For example, in some games it is important to estimate how much damage a certain character at a certain time in the game does, in order to make a decision. Improving these mechanics can give the players a feeling of advancing in the game.

Competition is another important aspect of the achievement factor. Some players play games because they like to compete with others, to achieve a higher ranking or just for recognition. The desire to challenge others is an important reason to play games, especially in DotA. While DotA itself is not available any more, DotA 2 is played competitively, up to the level of professional play. Annual championships are organized, with prize pools that exceed 20 million dollars. Trying to get to high (or professional) level of play is an important reason to play online games.

2.3.2 Social Factors

Social factors are known to have serious impact on behavior of an individual and forms the second factor of Yee [2007]. We will first discuss the three ways in which social factors are reasons for people to play games. Subsequently, we describe two ways in which *social influence* is apparent in online games: social norms and critical mass [Hsu and Lu, 2004].

Yee [2007] describes that there are three reasons why people play games for social purposes. The first is socializing: users want to chat with other players, build op friendships and possibly help them achieve their goals. The second is that people want to form long-term relationships with other users, to share personal interests, ideas and thoughts. The last reason is that people derive satisfaction from collaborating with others and being part of a group effort.

Hsu and Lu [2004] describe the term social influence and that there are two types of social influences that are apparent in online games: *social norms* and *critical mass*. In social norms, two general influences are distinguished: informational influence, which occurs when the reality of an individual is shaped by information gained from a social group, and normative influence, which means that an individual conforms to the expectations of others to gain something or not to lose something. In other words, people behave like others ('play game X') to belong in a certain group. Critical mass entails the fact that "the value of technology to a user increases with the number of its adopters" [Hsu and Lu, 2004]. When more people play a certain game, it attracts more people. In present research this is quite applicable, as the high number of users of DotA attract more users. This is highly related to social norms, as high player bases form a reason to belong to the large group who plays the game.

2.3.3 Immersion

Immersion forms the third factor of Yee [2007] that describe why people like to play games. Learning new things is important while playing games [Vorderer and Ritterfeld, 2004]. Players enjoy diving in the world of the game and learning new things about it in terms of exploring the world and discovering hidden places, items or other in-game features [Yee, 2007]. Moreover, players like to be immersed in a story about the game or their character. Furthermore, escapism is an important aspect of immersion. Players like to escape from real life, just for a short period of time. This is known to have a therapeutic effect [Hromek and Roffey, 2009]. Players are able to remove some measure of stress when they play a game. However, escapism can also have negative effects [Verschoor, 2016]. Instead of relieving the stress and personal problems, they try to avoid dealing with these difficult situations.

Immersion is an aspect that is less relevant for DotA. Players play individual games, where there is less role-playing, lore (the story behind heroes, maps or events) and other game information available to go through than in a MMORPG game. The aspect of escapism however, is apparent in DotA.

2.4 Motivations for toxic behavior

There are various reasons for people to exhibit toxic behavior, which partly originate from the reasons to play online games, discussed previously. This section will mainly focus on reasons for toxic behavior in online games, and less on toxic behavior in general. The main aspects of the reasons for exhibiting toxic behavior will be discussed in 2.4.1, along with two common motivations. These different motivations are enabled by the type of game (in this case MOBA) and originate from motivations to play games. These enabling factors will be explained in 2.4.2.

2.4.1 Reasons for toxic behavior

We will first discuss the main aspects of reasons for all toxic behavior. Afterwards, we will discuss two specific reasons for players to exhibit toxic behavior: anger or frustration and revenge.

All forms of toxic behavior have some common motivational characteristics [Verschoor, 2016]. The three aspects players seek to achieve by exhibiting toxic behavior are interaction, confrontation and getting responses. Toxic players like to confront others and disagree with their views in order to get an emotional response. Getting that response is a large motivator for toxicity [Thacker and Griffiths, 2012]. In other words, people exhibit toxic behavior to upset other players.

Moreover, the first specific reason to exhibit toxic behavior is anger or frustration [Verschoor, 2016]. As stated before, MOBA is a game genre that is very competitive and team centric. This causes players to be frustrated towards other players if they do not perform as desired or make decisions that are not in line with the views of the player. The desire to achieve goals and to advance to a higher level of play causes the player to be frustrated if he or she thinks that other players cause the delay. Even simple aspects of the game, e.g. playing a certain hero, could lead to a player to *tilt*. Tilting is an online

term that is used often in the community of DotA and LoL and is used for someone to be frustrated by something that causes the player to play worse and worse, make more mistakes and ultimately become more frustrated[3]. While we have not found research that demonstrates it, we find it likely that tilting may cause toxic behavior.

The second reason why people are toxic is because of revenge [Thacker and Griffiths, 2012]. Revenge is often paired with some degree of anger or frustration. People who are the recipient of toxic behavior tend to exhibit toxic behavior themselves. Not necessarily towards the originally toxic actor, but also in general towards other people or nobody in particular. This could be due to the fact that people see that toxic people are not punished, so they do not see the harm to exhibit toxic behavior themselves. "Similarly, they see others who are non-toxic without reward" [Verschoor, 2016, p. 15].

2.4.2 Enabling factors

Characteristics of MOBA games and online games in general enable players to exhibit toxic behavior. One of the most prominent characteristics is dissociative anonymity [Suler, 2004]. Dissociative anonymity entails that it is hard to determine who is controlling a certain online character. Users interact with characters or avatars and these do not link to natural persons. Naturally, more advanced users can find out more using IP addresses, but these users are in relatively low numbers [Suler, 2004]. This anonymity makes users feel far less vulnerable than when they would interact with natural persons. Therefore, they are more likely to post extreme content; the content can not be easily linked to someone's personal life. In the unlikely event that the identity would be discovered, the risk of experiencing physical consequences is also very low.

Dissociative anonymity also implies that when communicating with each other, players call each other with the username, character, role or hero. Communicating using these avatar names desensitizes players [Weger and Loughnan, 2014], meaning that it is harder to realize that you are playing with human beings [Verschoor, 2016].

This notion is in line with the *online disinhibition effect* [Suler, 2004]. The online disinhibition effect describes an effect that occurs in online environments. "While online, some people self-disclose or act out more frequently or intensely than they would in person" [Suler, 2004, p. 321], in a positive and in a negative way. The positive disclosure is called the benign disinhibition. People tend to share personal things, such as emotions, fears and wishes, and "they show unusual acts of kindness and generosity, sometimes going out of their way to help others" [Suler, 2004, p. 321]. The negative side of the disinhibition effect is called toxic disinhibition. People also share rude language, harsh criticisms, anger, hatred and threats in the "dark underworld of the Internet" [Suler, 2004, p. 321].

2.5 Automatic detection of toxic messages

The automatic detection of toxic messages has been researched more frequently the last years. We use an algorithm that automatically detects toxic messages in this thesis. Therefore, two papers will be discussed below, which both use the same data set as this thesis.

[3]Source: http://www.urbandictionary.com/define.php?term=Tilting

Märtens et al. [2015] created an algorithm that automatically labels messages as toxic. The algorithm achieves this via analyzing messages and their context and comparing them to a pre-defined dictionary of words. Märtens et al. [2015] incorporated a way to detect different ways of spelling for one word. For example, the word 'noob' is formed with the letters *n*, *o* and *b*. With this combination, all sorts of spellings for the word 'noob' are detected ('NOOOOOOOOb', 'boon', 'nooobbbbb' or 'noonb'). It made use of n-grams with $n = 1, 2, 3, 4$. This means that they distinguished remarks as toxic if they had a sequence of n toxic words. Märtens et al. [2015] found that toxic messages are frequently used in matches of DotA. Furthermore, they found that toxic messages were more frequently preceded by kill-events than random messages; players are more toxic after someone is killed. They also found that the team that wins uses fewer toxic messages than the losing team. Lastly, they tried to predict match outcomes using messages and found a weak link between toxicity and match outcome. In conclusion, Märtens et al. [2015] provided us with insightful work that serves as a basis for this thesis.

Verschoor [2016] took the above mentioned research as a starting point and aimed to improve the algorithm used by Märtens et al. [2015]. He also added the research question whether it is possible to predict the type of message (toxic or non-toxic) with the use of in-game events. He also changed the game selection (selected all games, instead of the selection of Märtens et al. [2015]), player selection (selected all players, instead of the selection of Märtens et al. [2015]) and missing or extra letters (he did not use the system to detect different spellings of one word). The algorithm of Märtens et al. [2015] was used as a starting point, and Verschoor [2016] improved it by not only checking for word combinations of up to four words, but looking at sequences of any length. He found an increase of 82% in the number of toxic labeled sequences (62,301 versus 34,237) and a manual inspection of the newly labeled message showed an accuracy of 99% (99 out of 100 were actually toxic). In other words, Verschoor [2016] improved the algorithm. He also found a weak correlation in predicting toxicity with in-game events (62.7% compared to a baseline of 50%).

2.6 Summary of the background

The Background chapter started with a section about DotA. DotA is an online game where team communication is important. A negative aspect of the importance of team communication is that toxic behavior is a common occurrence. Toxic behavior is anti-social behavior and has different forms, including cyberbullying, trolling and flaming. There are various reasons for players to exhibit toxic behavior, which are based on the motivations to play the game. This chapter was concluded with two papers that form the foundation of this thesis. The algorithm introduced by Märtens et al. [2015] and improved by Verschoor [2016] automatically labels messages as toxic or not. The algorithm of Verschoor [2016] was used to obtain results in this thesis.

3 Method

This chapter will provide the methods that were applied in order to get results. First, the raw data will be described in 3.1. The methods that were applied to evaluate what variables are predictive for winning match will be discussed in 3.2. In section 3.3, we will explain the way we measured the impact of toxicity on match outcome. This chapter will end with a short description of the binary logistic regression analysis we performed to evaluate how newly introduced variables contribute to the prediction of toxicity.

3.1 The raw data

The raw data consisted of 12952 text files, and is the same data both Verschoor [2016] and Märtens et al. [2015] started with. Each text file contained data of one match of DotA, coded in XML format. The matches were played between February 2 and February 6 of 2012 ([Märtens et al., 2015]; [Verschoor, 2016]). Each file was structured in a certain way, and will be discussed below.

- General game information. Each file started with general game information, such as the date, game length, game type, winner of the match, etc.
- Player data. A list of 11 players, five players of each team and an observer. The observer does not participate in the match but is used to collect the raw data. For each player there is general information, such as name, race of the character and which team the player is in. There are also statistics about how the player performed in the match, including the total number of kills, deaths, creep kills, creep denies, assists, neutral objectives and the amount of gold.
- Event data. This is a list of events that were recorded during the match. The events are also structured, containing the time of the event, the type of the event, a text message containing a description of the event and the player who performed the event. The event types include player kills, where the killer is recorded and the player who was killed. This list also includes players who were disconnected from the game, players who left the game and game objectives that were secured by a team.
- Ping data. Pings are ways to communicate fast and effectively with your team. A player can leave a 'ping' anywhere on the map, which can have different meanings. For example, a player can leave a ping that a certain enemy is approaching, or can target an opponent who he wants to kill. Each ping is recorded, including a timestamp, the player who pinged and the X and Y coordinates on the map.
- Chat data. This contains messages that players sent to their own team or messages they sent to their opponents and their own teammates, meaning that there were three channels: the sentinel channel, the scourge channel and the all-chat channel. DotA also allows players to send private messages in the game, but these were not recorded.

3.2 Predictive variables for match outcomes

A data set was created with variables (match statistics) at the end of a match (discussed in 3.2.1). What analyses we have performed to measure the correlation between those variables and winning a match, are discussed in 3.2.2.

3.2.1 Creating a dataset with match statistics

The first step to retrieve results was to determine what variables are most predictive for winning or losing a game. To reach that objective, a data set is needed with match outcomes and corresponding statistics of each team at the end of the game. The player list contained such information for each player. These statistics were added together to get the team totals. All variables (kills, deaths, creep kills, creep denies, assists, gold, neutrals and actions per minute (APM)) were extracted. The winning team and the game length were added (obtained from the general game information).

Only the matches with a clear winner were selected for analysis. After a brief exploration of the data, it became clear that numerous matches were abandoned right after the match started. These matches did have a clear winner, but had no valuable game statistics, as nothing had happened that game. To filter those matches, a game length threshold was introduced. Looking generally at the data, a threshold of 1100 seconds (about 18 minutes) seemed to give appropriate results. This means that a match needed to last at least 18 minutes to be analyzed. This resulted in a data set of 5389 matches, each with the following variables:

- File name
- Winning Team; either 'Sentinel' or 'Scourge'
- Game Length; the length of the game in seconds
- Kills (both Sentinel and Scourge); the number of kills the entire team acquired
- Deaths (both Sentinel and Scourge); the number of deaths the entire team acquired
- Assists (both Sentinel and Scourge); the number of assists the entire team acquired
- Creep kills (both Sentinel and Scourge); the number of creep kills the entire team acquired
- Creep denies (both Sentinel and Scourge); the number of creep denies the entire team acquired
- Gold (both Sentinel and Scourge); the amount of gold the entire team acquired
- Neutrals (both Sentinel and Scourge); the number of structures, neutral monsters, etc. the entire team acquired
- APM (both Sentinel and Scourge); the number of actions per minute

New variables were then introduced via the programming language R. For each team, we added the ratio of each variable. Using domain knowledge, we expected that not only the raw numbers should be analyzed, also the ratios between the two teams. For example, the variables Kill.Ratio.Sentinel and Kill.Ratio.Scourge were introduced, and were calculate as follows:

$$\text{Kill.Ratio.Sentinel} = \text{Kills.Sentinel} \,/\, (\text{Kills.Sentinel} + \text{Kills.Scourge})$$
$$\text{Kill.Ratio.Scourge} = \text{Kills.Scourge} \,/\, (\text{Kills.Sentinel} + \text{Kills.Scourge})$$

This was done for all variables. This resulted in a data set with 5389 matches, each with 35 variables.

3.2.2 Analyzing match statistics

In order to evaluate what variables are predictive for winning a match, we explored the data by using a classifier. The classifier we used was a decision tree, due to its decent interpretability. The classifier was trained with the data exploration tool Weka. In the remainder of this thesis we will refer to this model as the Exploration Model.

The initial run of the Exploration Model showed that variables about gold income were highly predictive of winning a match. For the exact numbers, we refer to subsections 4.1.1 and 4.1.2. As we could not retrieve the amount of gold at the time messages were sent, we decided to remove gold variables. Also, all variables other than kill, death and neutral objective related variables, were removed as we could not retrieve more variables at the time a message was sent.

After exploring the data, the statistical correlation between the winning team and every other variable was calculated using SPSS Statistics. The results are reported in 4.1.2.

3.3 Measuring the impact of toxic messages on match outcome

To measure the impact of toxic messages on the match outcome, a data set with all messages, regardless of toxicity, was needed. We used the algorithm of Verschoor [2016] to retrieve all messages. The algorithm and the changes we made in the algorithm will be discussed in 3.3.1. In 3.3.2 we will cover what missing values we encountered, how we filtered our data and what variables we added. Using the variables that were predictive for winning a match, we made a model that predicts match outcomes based on those variables, which will be briefly explained in 3.3.3. This section will be concluded with 3.3.4, in which we will discuss what analyses we performed to measure the impact of toxic messages on match outcomes.

3.3.1 Changing the existing algorithm

In order to get a data set containing all messages from all the games that were analyzed, we used the algorithm of Verschoor [2016]. The algorithm retrieved all messages, including information about the message, the file name, the game length, and the chat channel. It also included a classification of toxicity; the message was labeled as 1 if it was toxic and

as 0 if it was non-toxic. The classification of the algorithm of Märtens et al. [2015] was also included. The algorithm checked for deaths and objectives within a period of time before and after the message. For more information about the original algorithm, we refer to Verschoor [2016].

The first change that was made was the time window. The original script applied a certain time window in which the algorithm checked for events. We changed the time window from a set number of seconds, to the time between the start of the game and the moment the message was sent. This was applied to get a clear understanding of how the situation of the game was at the moment a toxic message is sent.

We also changed the events that were retrieved within the time frame. The original algorithm only checked for deaths and secured objectives. The number of deaths was recorded for the individual, as well as the team and the other team. We removed the individual numbers, because we only focused on the status of the game of the two teams. The objective-related variables were also removed, because of the results we retrieved from the correlation analysis, discussed in section 3.2.2 and reported in 4.1.2. We added the number of kills of the two teams. We could not retrieve more events or variables at the moment of messages, due to how the data was composed.

This resulted in a data set of 529045 messages, with the following variables:

- file name - the name of the file that was analyzed.
- toxic - a classification of toxicity: 1 for toxic or 0 for non-toxic.
- toxic_delft - a classification of toxicity by the algorithm of Märtens et al. [2015]
- time - the time stamp of the message
- game_length - the total length of the game in seconds
- time_norm - the relative time of the message (number between 0 and 1)
- chat_channel - the channel in which the message was sent
- player - the name of the player who sent the message
- chat_message - the message
- deaths_own_team - number of deaths of the team of the player
- deaths_other_team - number of deaths of the other team
- kills_own_team - number of kills of the own team
- kills_other_team - number of kills of the other team

3.3.2 Finalizing the data

We explored the data briefly and noticed that we missed messages from numerous matches in our CSV file. After checking those matches, it became clear that in these matches, the players were not using the English language to send messages (many of them were Russian messages). As a result, these messages were not retrieved, as the pre-defined list with toxic words and phrases only contained English entries.

Numerous messages were sent in a match that did not have a winning team at the end of the game. Moreover, few messages were sent in an unknown chat channel. Removing the messages without a winner and the messages that were sent in an unknown chat channel almost halved the number of messages (regardless of toxicity). The figures are in table 1.

	Total	**Toxic**	**Non-toxic**
Before filtering	529045	10185	518860
After filtering	274456	5434	269022

Table 1: Number of messages before and after filtering missing values.

We were only interested in the first toxic message in the game, because the following toxic messages could be a reaction towards the first toxic message. We filtered out all toxic messages that were sent after the first toxic message. This way, we could measure the effect of the initiator of toxic behavior on match outcomes. The number of toxic messages was further reduced from 5434 to 565.

We added the kill and death ratios, in order to get the same variables as we analyzed at the end of the game for RQ1. This was done after the first analysis and was done the same way as described in 3.2.2.

We also added a variable that described the team that the player was in. There was a variable that described in which chat channel the message was sent, but this variable also included many messages that were sent in the all-chat. This team variable was also created to transform the variables at the end of the match from for example *Kills.Sentinel* to *Kills.Own.Team* or *Kills.Other.Team*, in order to make a separation between the team of the player who exhibits toxic behavior and the other team.

3.3.3 A model to predict match outcome

To evaluate how the two teams are performing at the time a message was sent, we constructed a model to predict match outcomes. A J48-decision tree was trained using 10-fold cross-validation to predict the winner of a match based on the variables that were predictive for winning a match (discussed in 3.2.2) at the end of a match. Only the variables that were both highly predictive of the match outcome and retrievable from the data, were used. The classifier was trained on all 269022 messages, regardless of toxicity and whether or not a toxic message was sent before. This model predicted the outcome of a match based on the relevant variables we found at the moment a message was sent, labeling them with a 1 if the model predicts that the sender of the message will win at the end and labeling them with a 0 if the model predicts the sender of the message will lose.

Moreover, we made this model in order to quantify the feeling of players whether they are in a winning, losing or neutral situation at a certain time in the game. To identify a neutral situation, we made a copy of each message that was processed by the model and reversed the ratio variables. Such a copy then represents a fictional player of the opponent team. When both predictions were the same, e.g. a win was predicted for both the original message and the copy with reversed ratios, we identified the situation as neutral. To elaborate the model further, results needs to be included. We will report the results in 4.1.3 and the model will further be discussed in 5. In the remainder of this thesis, we will refer to this model as the Prediction Model.

3.3.4 Comparing toxic and non-toxic teams

In order to compare the toxic and non-toxic messages, we made two selections of non-toxic messages. The first was a selection where the win-loss distribution at the end of the game was the same for both the toxic and the non-toxic set. From the 565 toxic messages, 338 were losses and 227 were wins at the end of the game (from the point of view of the toxic player). The same win-loss distribution was used to semi-randomly select 565 non-toxic messages. The second selection was a selection of non-toxic messages where the distribution of predicted wins and losses was the same for both the non-toxic messages and the toxic messages.

We then identified six groups to compare the group sizes for toxic and non-toxic messages. These groups were based on the predicted outcome, or in other words, the predicted type of situation the sender of the message was in, and the actual outcome.

Group	Predicted match outcome	Actual match outcome
A	Win	Win
B	Loss	Loss
C	Loss	Win
D	Win	Loss
E	Neutral	Win
F	Neutral	Loss

Table 2: The groups, based on the predicted match outcome at the time the message was sent and the actual match outcomes.

Due to the large number of non-toxic messages compared to the toxic messages, we repeated both the semi-random selections of non-toxic messages 20 times. We calculated the mean, standard deviation and standard error of the mean of those groups and compared them to the group sizes of the toxic selection.

3.4 Predicting toxicity

Aside from our research questions, we wanted to evaluate how the variables that were predictive of match outcome contributed to the prediction of whether a message is toxic or not. We performed a binary logistic regression to predict toxicity and to evaluate how

the variables contribute to the effect. Verschoor [2016] already performed such an analysis. We think it is interesting to see what the newly added variables, the kill and death ratios, contribute to the effect.

4 Results

In this chapter, the results will be discussed. In 4.1 we will start by discussing the Exploration model, the statistical correlations between match statistics at the end of a game and the match outcome and the accuracy of the resulting Prediction Model to predict match outcomes. The results of the group analysis in order to measure the impact of toxic behavior on match outcomes will be reported in 4.2. In the last section of this chapter, the results of the binary logistic regression will be reported.

4.1 Predictive variables for match outcomes

Our first goal was to investigate what variables were predictive of winning a match of DotA. We had a first look at the data using a classifier, which will be discussed in 4.1.1. In 4.1.2 we report the statistical correlations between the winning team and all retrievable variables at the end of a match. The model that was created based on the variables with the highest correlations (and retrievable at the time a message was sent) is discussed in 4.1.3.

4.1.1 Exploratory analysis: decision tree

A Exploration Model was trained in order to perform an exploratory analysis to predict the winner of a match. The majority baseline (ZeroR) scored an accuracy of correctly classified matches of 53.02% (Mean Absolute Error = 0.50). The Exploration Model we trained was a decision tree (J48), of which the upper part is displayed in appendix A.

The Exploration Model obtained an accuracy of 97.20% (Mean Absolute Error = 0.03). From the tree could be derived that Gold was a highly predictive variable. As discussed in 3.2.2, we removed the gold related variables because gold was not retrievable at the moment a message was sent. After removing the gold related variables, we tested the Exploration Model again. This resulted in an accuracy of 91.90% (Mean Absolute Error = 0.09). The tree illustrated that the death ratio variables and the neutral objectives variables were strong predictors of the winning team.

4.1.2 Statistical correlations

The results of the Exploration Model are merely an indication about the variables at hand. In order to conclusively state what variables correlate highly with match outcome, we calculated the statistical correlations between all variables and the winning team. Below are the results of this analysis.

Feature	Correlation strength	Sig. (1-tailed)	Sig. (2-tailed)
Gold.Ratio.Scourge	0.913	<.001	<.001
Gold.Scourge	0.819	<.001	<.001
Death.Ratio.Sentinel	0.797	<.001	<.001
Kill.Ratio.Scourge	0.793	<.001	<.001
Assist.Ratio.Scourge	0.748	<.001	<.001
Kills.Scourge	0.611	<.001	<.001
Deaths.Sentinel	0.605	<.001	<.001
Creep.Ratio.Scourge	0.554	<.001	<.001
Assists.Scourge	0.523	<.001	<.001
Neutral.Ratio.Scourge	0.395	<.001	<.001
APM.Ratio.Scourge	0.289	<.001	<.001
Neutrals.Scourge	0.283	<.001	<.001
Creeps.Scourge	0.232	<.001	<.001
Creep_deny.Ratio.Scourge	0.227	<.001	<.001
Creep.denies.Scourge	0.165	<.001	<.001
APM.Scourge	0.152	<.001	<.001
Game.Length	0.085	<.001	<.001
Creeps.Sentinel	-0.076	<.001	<.001
Creep.denies.Sentinel	-0.136	<.001	<.001
Neutrals.Sentinel	-0.198	<.001	<.001
APM.Sentinel	-0.221	<.001	<.001
Creep_deny.Ratio.Sentinel	-0.227	<.001	<.001
APM.Ratio.Sentinel	-0.289	<.001	<.001
Neutral.Ratio.Sentinel	-0.395	<.001	<.001
Assists.Sentinel	-0.496	<.001	<.001
Creep.Ratio.Sentinel	-0.554	<.001	<.001
Deaths.Scourge	-0.567	<.001	<.001
Kills.Sentinel	-0.576	<.001	<.001
Assist.Ratio.Sentinel	-0.748	<.001	<.001
Gold.Sentinel	-0.783	<.001	<.001
Kill.Ratio.Sentinel	-0.793	<.001	<.001
Death.Ratio.Scourge	-0.797	<.001	<.001
Gold.Ratio.Sentinel	-0.913	<.001	<.001

Table 3: Correlations with Winning.Team

The highest correlation with Winning.Team are the two gold ratio variables. The variable with the next highest correlation is the death ratio (0.797 and -0.797, $p < .001$) and the kill ratio (0.793 and -0.793, $p < .001$).

4.1.3 Model for predicting match outcomes

We created the Prediction Model that predicts match outcomes, based on the kill and death ratios of both teams at the end of a match, because the kill and death ratios

correlate strongly with match outcome and were retrievable. The Prediction Model that was created was a standard decision tree (J48) and correctly classified 99.74% of the matches (Mean Absolute Error = 0.003) as either a win or a loss. The entire Prediction Model is displayed in appendix B.

Using the Prediction Model on the subset of toxic messages, the model achieved a classification accuracy of 56.99%. This means that the Prediction Model rightfully predicted little more than half of the outcomes, based on the ratios at the moment a toxic message was sent. On the first selection of non-toxic messages (which was repeated 20 times) the Prediction Model correctly classified on average 59.30% of the matches. On the second selection of non-toxic messages the Prediction Model achieved an average classification accuracy of 61.75%. This indicates that in many games, the match outcome can change during the match. However, despite the significantly lower classification accuracy of the Prediction Model on the test sets, we expect that the model still accurately quantifies the feeling of the players of winning or losing at a certain time in the game. We find it likely that players (unknowingly) make use of the model, i.e. evaluate the current kill and death ratios, to check how teams are performing. Also, we expect that players can make a good guess whether they will eventually win or lose the game if the kill and death ratios do not change. We will discuss the model more in chapter 5.

4.2 The impact of toxicity on match outcomes

Using the Prediction Model, we made predictions of the match outcomes at the moment a toxic message was sent. Based on the predicted outcome and the actual outcome, we made six groups. The distribution of these groups for the toxic messages was as follows:

Group	Predicted match outcome	Actual match outcome	N
A	Win	Win	94
B	Loss	Loss	171
C	Loss	Win	78
D	Win	Loss	91
E	Neutral	Win	55
F	Neutral	Loss	76

Table 4: The number of toxic messages in each of the six groups.

The Prediction Model was then tested on two selections of non-toxic messages: (1) a selection with an identical distribution of the actual outcomes at the end of the game (win/loss) (2) a selection with an identical distribution of predicted outcomes (win/loss/neutral). The results are in the table below.

Group	Toxic	Non-toxic 1	SE	Non-toxic 2	SE
A	94	138.60 (9.90)	2.21	112.40 (6.50)	1.45
B	171	150.25 (9.16)	2.05	172.90 (6.60)	1.48
C	78	51.70 (7.79)	1.74	76.10 (6.60)	1.48
D	91	126.15 (6.79)	1.52	72.60 (6.50)	1.45
E	55	36.70 (6.29)	1.41	58.05 (6.95)	1.55
F	76	61.60 (6.69)	1.41	72.95 (6.95)	1.55

Table 5: The six groups: non-toxic messages versus toxic messages. Standard deviation between parentheses. For the non-toxic selections, the standard error of the mean (SE) is displayed at the right of the column.

The non-toxic figures are means of 20 repetitions of random samples of the selections discussed earlier. The sample means were close to the actual mean from the whole population, as the standard error of the mean (SE) is relatively low. Comparing the toxic and non-toxic group sizes, we made the assumption that the SE for toxic group sizes is more or less equal to the SE of the non-toxic group sizes.

We first compared the 565 toxic messages to 565 non-toxic messages with the same actual win-loss distribution. 227 wins and 338 losses were observed for the toxic messages, the same distribution was used to select non-toxic messages. From the 227 wins, 94 were predicted for the toxic teams versus 138.60 for the non-toxic teams. Therefore, toxic teams transit predicted wins into actual wins significantly less often than non-toxic teams (difference = 44.60 $>$ 2 * SE = 4.42). Moreover, from the 338 actual losses, 171 were predicted for the toxic teams versus 150.25 for the non-toxic teams. Therefore, we conclude that toxic teams transit predicted losses into actual losses significantly more often than non-toxic teams (difference = 20.75 $>$ 2 * SE = 4.10).

We then compared the toxic messages to 565 non-toxic messages with the same distribution of predicted outcomes (win, loss, neutral). From the 565 messages, 185 wins were predicted for both types of messages. From those 185 predicted wins, 94 were transited into actual wins for the toxic teams, versus 112.40 for non-toxic teams. This supports our first conclusion that toxic teams transit predicted wins into actual wins significantly more often (difference = 18.40 $>$ 2 * SE = 2.90). From the 249 predicted losses, 171 transited into actual losses for toxic teams versus 172.90 for the non-toxic teams, which means that there was no significant difference found here (difference = 1.90 $<$ 2 * SE = 2.96). From the 131 messages where a neutral situation was predicted, the toxic teams transited 55 in a win and 76 in a loss. Non-toxic teams transited 58.05 games in a win and 72.95 in a loss. Therefore, we conclude that there is no significant difference between the number of actual wins and losses when a neutral situation was predicted (difference = 3.05 $<$ 2 * SE = 3.10).

In conclusion, the following results were obtained:

1. A predicted victory for a toxic team turns into an actual victory at the game's end significantly less often for a toxic team than for a non-toxic team.

2. A predicted loss for a toxic team turns into an actual loss at the game's end signifi-

cantly more often for a toxic team than for a non-toxic team.

3. The number of wins and losses at the game's end, where a neutral situation was predicted, do not significantly differ for toxic and non-toxic teams.

4.3 Predicting toxicity

We performed a binary logistic regression to predict toxicity and evaluate how the kill and death ratios contribute to the prediction of a message being toxic. Toxicity was the dependent variable (0 = non-toxic, 1 = toxic), the variables were *killratio_own_team, deathratio_own_team, killratio_other_team* and *deathratio_other_team*. The results are in table 6.

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-4.32744	0.07341	-58.947	<.001
killratio_own_team	-0.69049	0.23146	-2.983	<.01
deathratio_own_team	0.17304	0.14713	1.176	.24
killratio_other_team	0.91144	0.16013	5.692	<.001
deathratio_other_team	0.50580	0.20312	2.490	<.05

Table 6: Unstandardized binary logistic regression coefficients

The estimates generally indicate either an increase (if estimate ≥ 0) or a decrease (if estimate ≤ 0) in the probability of the dependent variable. For instance, we see that an increase of the *killratio_own_team* decreases the probability of the message being toxic. Similarly, we see that *deathratio_own_team* does not correlate significantly. The kill ratio of the other team correlates significantly positive with toxicity, meaning that an increase of the kill ratio of the other team increases the probability of the message being toxic. This is also the largest estimate, so kill ratio_other_team has the largest effect on the probability of a message being toxic. Lastly, an increase of the death ratio of the other team is associated with an increase of the probability of the message being toxic.

5 Discussion

The results will be evaluated in this chapter, as presented in chapter 4. The shortcomings of this thesis will be discussed as well as some implications and directions for further research.

5.1 Evaluating results

The goal of this thesis was to investigate a possible relation between toxic behavior and match outcomes of the popular MOBA DotA. To achieve this, we first investigated what variables were predictive for winning a match.

We found that gold was an almost perfect predictor of winning a match of DotA. As we could not retrieve any data containing gold at the moment a message was sent, we used the kill and death ratios. The kill and death ratios correlated the most with winning or losing a match after deleting variables about gold. Those ratios were defined as, for example, the number of kills of team 1 divided by the total number of kills. A possible explanation for kill and death ratios having a high correlation with match outcome is that kills and deaths are ways to gain or lose significant amounts of gold. While a hero is dead, it can not accumulate the same amounts of gold as while the hero was alive. For instance, the hero can not kill creeps or other players. Therefore, a death could mean a significant discrepancy in gold and therefore, a significant discrepancy in chances of winning a game.

When comparing the kill and death ratios, we found a discrepancy. We noticed that a death on one side, did not necessarily mean a kill on the other side. This was due to the fact that players are able to die in other ways, e.g. by neutral monsters or in some cases, it is possible to kill teammates for tactical purposes. This would indicate that these two variables should not be treated as the same, but should be analyzed as two separate variables.

After analyzing which variables correlated the strongest with winning a match, we created a Prediction Model to predict the match outcome using only those variables at the end of the game that had a strong correlation with winning a match and were retrievable at the moment a message was sent, namely kill ratios and death ratios. This Prediction Model, a decision tree, correctly predicted the match outcome in almost all cases (99.74%). Assuming that players have a sense or general feeling about the game and how it is going to progress based on the kill and death ratios, this Prediction Model would quantify that feeling of winning or losing a game at the moment a toxic message is sent. In other words, players can estimate what the match outcome would be if they compare the kill and death ratios according to our model and when the ratios remain unchanged. Moreover, the Prediction Model predicts the match outcome based on kill and death ratios. Imagine the following example where team A is going to win eventually. At a certain time in the game, team A has 25 kills and 7 deaths and team B has 7 kills and 25 deaths. Based on these kill and death ratios, we strongly expect players to have a general sense that team A has the highest chances of winning, due to the strong correlation with observable data (kills and deaths). Additionally, the Prediction Model would predict that team A is going to win.

We tested the Prediction Model on both the toxic messages and two selections of

non-toxic messages. We found that toxic teams lose more when they were already losing and win less when they were at the winning hand. When a message is sent in a neutral situation, where both teams have somewhat equal chances of winning, the number of wins and losses do not significantly differ for toxic and non-toxic teams. In general, however, we conclude that sending a toxic message does not improve the chances of winning a game.

The actual causes of these results are hard to tell, but there are some possible explanations. We expect that toxicity has a negative impact on the mental state of teammates, due to the importance of team communication. Toxic players possibly target other players in the team and that could cause a decrease in confidence or less motivated players to win the game. Moreover, we expect that the frustration that is often part of toxic behavior, is a factor that causes a decrease in concentration of the toxic player itself. All in all, it is hard to tell what toxic behavior actually causes to decrease chances of winning. Future research could try to gain more insights in the direct effects of toxicity.

Lastly, we performed a binary logistic regression, to evaluate how the new variables introduced in this study (the kill and death ratios) would contribute to the prediction of toxicity. Verschoor [2016] performed a similar analysis and we wanted to evaluate how the new variables would contribute to the prediction. We found that the kill ratios significantly impact the probabilities of toxicity, where an increase in the kill ratio of the own team of the player who sends the messages decreases the probability of a message being toxic and an increase in the kill ratio of the other team increases the probability of a message being toxic. This is in line with the finding of Verschoor [2016] that toxic messages are most frequently preceded by a kill. Kills and deaths are possibly a measure that players of DotA use to indicate how well a player is performing. When this kill-death score of a team mate is lower, or decreases by a death, we expect that the majority of players finds that he or she is performing worse. As discussed before, the competitive atmosphere may enable players to exhibit toxic behavior in these situations. Moreover, the player who dies is also more likely to exhibit toxic behavior.

5.2 Weaknesses

One of the main weaknesses was the composition of the raw data. We had a limited number of toxic messages and could not retrieve as many variables as we want at the time a message was sent. As a consequence, we had to limit the number of non-toxic messages to be able to compare the two sets. We also had to limit our Prediction Model to a few variables, where variables that describe a gold score would be more effective at predicting match outcomes.

Another weakness is that the results are hard to generalize. As discussed before, toxic behavior depends largely on the type of game and the characteristics of the game. We only investigated the effect of toxic behavior on match outcomes in DotA. The number of kills and deaths and the ratios could be very different in other game genres. However, we expect that our findings are applicable on other MOBA games, such as League of Legends and DotA 2, due to many similar characteristics.

The algorithm that automatically labels the messages as toxic, only marked English remarks as toxic. Toxic behavior is also dependent on cultural differences [Warner and Ratier, 2005], meaning that the games that we had to delete due to its Russian language,

could affect the results. Moreover, the MOBA genre is popular in Asia, which could also give other results.

The question whether or not the cause of the higher number of losses at the toxic teams is actually the toxicity, remains unanswered. The conclusion that we can draw is that statistically, toxic teams have less chances of winning a game. We could not conclude that the actual cause of losing was toxicity.

5.3 Future Research

As described, above, a limitation of our research was our raw data. The number of toxic messages and number of variables are limited, but form interesting improvements for future research. In future research, where the number of toxic messages is larger and the event data could include some score of gold or other relevant variables at that time in the game, the effects of toxicity could become more clear.

Additionally, future research could include more variables to investigate the effect of toxic behavior, such as items, the level and experience of heroes and at what time of the game the toxic message is sent. Especially the time in the game could be of interest, as toxicity in relative early stages of the game could be of less importance then when toxic behavior is exhibited in the last few minutes of the game. These variables may also be 'out-game variables' [Verschoor, 2016], such as demographics.

6 Conclusion

The purpose of this thesis was to investigate a possible relationship between toxic behavior and match outcomes in the popular MOBA DotA. To summarize our study, we will discuss the first (6.1), second (6.2) and third research question (6.3). Finally, we will discuss our problem statement (6.4).

6.1 RQ1: Predictive variables for winning

We composed a data set of DotA matches with all team statistics at the end of the game, including the number of kills and deaths, the amount of gold, etc. A classifier was trained to give basic insights and statistical correlations with the winning team were calculated for all the variables. We found that the ratio of the amount of gold between the two teams was the most predictive, but we could not use that variable because we could not retrieve a gold score at the moment a toxic message was sent. After the variables about the gold score, kill ratios and death ratios were the most predictive.

6.2 RQ2: the status of the game at the moment of toxicity

The algorithm of Verschoor [2016] was adapted and used to get a large data set with messages of the analyzed games, with a classification of toxicity (0 was non-toxic, 1 was toxic). A model was created to predict the winner of a match, based on the variables we found while answering RQ1 (the kill and death ratios) at the end of the game. We expect that players can make an estimate about whether they will be winning or losing the game, using our model. Players can compare the kill and death ratios of both teams at a certain time in the game, and estimate whether they will win or lose if the kill and death ratios remain the same. This way, our model gives us an indication about how the players feel the match is progressing.

6.3 RQ3: impact of toxicity on match outcomes

Using our model that predicted the match outcome, we created six groups, based on the predicted outcome (win, loss and neutral) and actual outcome (win or loss). We found that toxic teams lose more matches if they were already losing and win less matches if they were already winning. When a message was sent in a neutral situation, the toxic and non-toxic teams do not differ in number of wins and losses at the end of the game.

6.4 Problem statement: the relationship between toxicity and match outcomes

Our problem statement was to investigate to what extent there exists a relationship between toxic behaviour and match outcomes. It remains hard to define an exact relationship between toxicity and match outcomes, but we did find that toxic behaviour does not enhance the chances of winning a game.

References

Bartle, R. (1996). Hearts, clubs, diamonds, spades: Players who suit muds. *Journal of Virtual Environments*. Retrieved from http://mud.co.uk/richard/hcds.htm.

Blackburn, J. and Kwak, H. (2014). Stfu noob! predicting crowdsourced decisions on toxic behavior in online games. *Proceedings of the 23rd International Conference on World Wide Web*, pages 877–888.

Buckels, E., Trapnell, P., and Paulhus, D. (2014). Trolls just want to have fun. *Personality and Individual Differences*, 67:97–102.

Csikszentmihalyi, M. (2014). *Toward a Psychology of Optimal Experience*. Springer.

Herring, S., Job-Sluder, K., Scheckler, R., and Barab, S. (2002). Searching for safety online: Managing "trolling" in a feminist forum. *The Information Society*, 18:371–384.

Hromek, R. and Roffey, S. (2009). Promoting social and emotional learning with games. *Simulation & Gaming*, 40(5):626–644.

Hsu, C.-L. and Lu, H.-P. (2004). Consumer behavior in online game communities: A motivational factor perspective. *Computers in Human Behavior*, 23(3):321–326.

Kayany, J. (1998). Contexts of uninhibited online behavior:flaming in social newsgroups on usenet. *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, 49(12):1135–1141.

Kwak, H. and Blackburn, J. (2014). Linguistic analysis of toxic behavior in an online video game. *6th International Conference on Social Informatics*. Retrieved from https://arxiv.org/abs/1410.5185.

Kwak, H., Blackburn, J., and Han, S. (2015). Exploring cyberbullying and other toxic behavior in team competition online games. *CHI'15 Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems,*, pages 3739–3748.

Lee, H. (2005). Behavioral strategies for dealing with flaming in an online forum. *The Sociological Quarterly*, 46(2):385–403.

McKane, J. (2016). There are 1.8 billion gamers in the world, and pc gaming dominates the market. Retrieved from https://mygaming.co.za/news/features/89913-there-are-1-8-billion-gamers-in-the-world-and-pc-gaming-dominates-the-market.html.

Märtens, M., Shen, S., Iosup, A., and Kuipers, F. (2015). Toxicity detection in multiplayer online games. *In 14th International Workshop on Network and Systems Support for Games (netgames)*.

Nixon, C. (2014). Current perspectives: the impact of cyberbullying on adolescent health. *Adolescent Health, Medicine and Therapeutics*, 5:143–158.

Olweus, D. (1993). *Bullying at school: What we know and what we can do.* Oxford: Blackwell.

Paul, J. (2017). By the numbers: Most popular online games right now. Retrieved from https://nowloading.co/posts/3916216.

Przybylski, A., Rigby, C., and Ryan, R. (2010). A motivational model of video game engagement. *Review of general psychology*, 14(2):154.

Sherry, J., Lucas, K., Greenberg, B., and Lachlan, K. (2014). Video game uses and gratifications as predictors of use and game preference. *Playing video games: Motives, responses, and consequences*, pages 248–262.

Smith, P. K., Mahdavi, J., Carvalho, M., Fisher, S., Russell, S., and Tippett, N. (2008). Cyberbullying: its nature and impact in secondary school pupils. *Journal of Child Psychology and Psychiatry*, 49(4):376–385.

Suler, J. (2004). The online disinhibition effect. *Cyberpsychology & Behaviour*, 7(3):321–326.

Tekofsky, S. (2017). *YOU ARE WHO YOU PLAY YOU ARE. Modelling Player Traits from Video Game Behavior.* PhD thesis, Tilburg University.

Thacker, S. and Griffiths, M. (2012). An exploratory study of trolling in online video gaming. *International Journal of Cyber Behavior*, 2(4):17–33.

Verschoor, M. (2016). Eating seeds as a pastime activity. predicting toxicity in online game chat using in-game events. Master's thesis, Tilburg University. http://arno.uvt.nl/show.cgi?fid=140685.

Vorderer, P. and Ritterfeld, U. (2004). Enjoyment: At the heart of media entertainment. *Communication Theory*, pages 388–408.

Warner, D. and Ratier, M. (2005). Social context in massively-multiplayer online games (mmogs): Ethical questions in shared space. *International Review of Information Ethics*, 4(7):46–52.

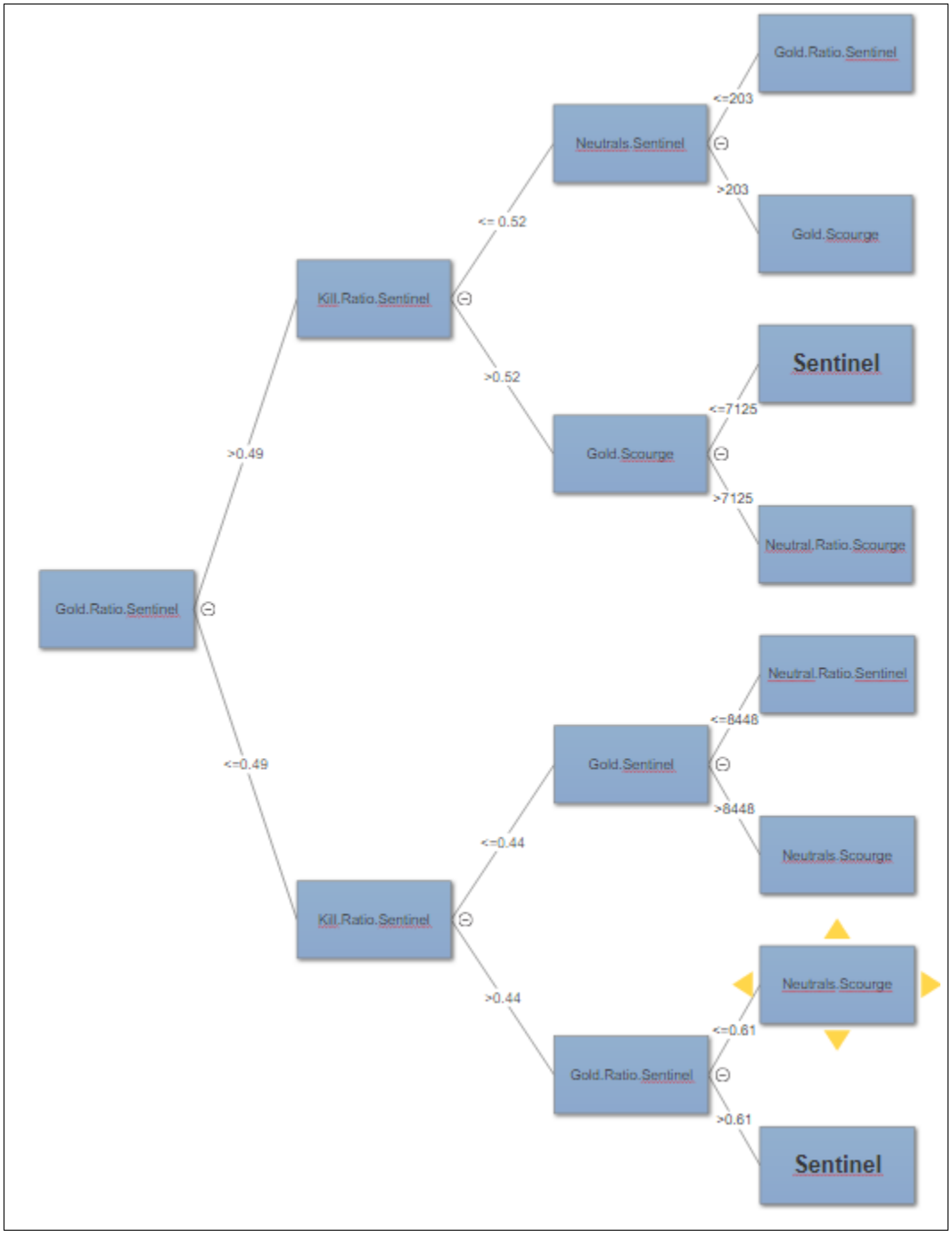
Weger, U. and Loughnan, S. (2014). Virtually number: Immersive video gaming alter real-life experience. *Psychonomic Bulletin and Review*, 21(2):562–565.

Wolmarans, K. (2016). Dota 2 vs. league of legends: Updating the numbers. Retrieved from http://www.criticalhit.net/gaming/dota-2-vs-league-legends-updating-numbers/.

Yee, N. (2007). Motivations for play in online games. *Cyberpsychology & Behavior*, 9:772–775.

Appendices

A The upper part of the Exploration Model



B The Prediction Model

```
=== Classifier model (full training set) ===

J48 pruned tree
------------------

DEATHR.OWN.END <= 0.505263
|   DEATHR.OWN.END <= 0.42735
|   |   DEATHR.OWN.END <= 0.393443
|   |   |   DEATHR.OWN.END <= 0.3625: 1 (31534.0)
|   |   |   DEATHR.OWN.END > 0.3625
|   |   |   |   DEATHR.OWN.END <= 0.363636
|   |   |   |   |   KILLR.OWN.END <= 0.644737: 1 (273.0)
|   |   |   |   |   KILLR.OWN.END > 0.644737
|   |   |   |   |   |   KILLR.OWN.END <= 0.653333: 0 (145.0)
|   |   |   |   |   |   KILLR.OWN.END > 0.653333: 1 (174.0)
|   |   |   |   DEATHR.OWN.END > 0.363636: 1 (11064.0)
|   |   DEATHR.OWN.END > 0.393443
|   |   |   DEATHR.OWN.END <= 0.393939
|   |   |   |   KILLR.OWN.END <= 0.61: 0 (364.0)
|   |   |   |   KILLR.OWN.END > 0.61: 1 (132.0)
|   |   |   DEATHR.OWN.END > 0.393939
|   |   |   |   DEATHR.OWN.END <= 0.411765
|   |   |   |   |   DEATHR.OWN.END <= 0.397959: 1 (2258.0)
|   |   |   |   |   DEATHR.OWN.END > 0.397959
|   |   |   |   |   |   KILLR.OWN.END <= 0.597561
|   |   |   |   |   |   |   DEATHR.OWN.END <= 0.41: 1 (3865.0)
|   |   |   |   |   |   |   DEATHR.OWN.END > 0.41
|   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.581395: 0 (185.0)
|   |   |   |   |   |   |   |   KILLR.OWN.END > 0.581395: 1 (553.0)
|   |   |   |   |   |   KILLR.OWN.END > 0.597561
|   |   |   |   |   |   |   DEATHR.OWN.END <= 0.402597: 0 (513.0)
|   |   |   |   |   |   |   DEATHR.OWN.END > 0.402597
|   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.62069: 1 (1323.0)
|   |   |   |   |   |   |   |   KILLR.OWN.END > 0.62069: 0 (192.0)
|   |   |   |   DEATHR.OWN.END > 0.411765
|   |   |   |   |   KILLR.OWN.END <= 0.569444
|   |   |   |   |   |   KILLR.OWN.END <= 0.568966: 1 (1637.0)
|   |   |   |   |   |   KILLR.OWN.END > 0.568966: 0 (176.0)
|   |   |   |   KILLR.OWN.END > 0.569444: 1 (9676.0)
|   DEATHR.OWN.END > 0.42735
|   |   KILLR.OWN.END <= 0.527778
|   |   |   KILLR.OWN.END <= 0.525641
|   |   |   |   DEATHR.OWN.END <= 0.472973: 1 (1600.0)
|   |   |   |   DEATHR.OWN.END > 0.472973
|   |   |   |   |   DEATHR.OWN.END <= 0.5
|   |   |   |   |   |   KILLR.OWN.END <= 0.511905
|   |   |   |   |   |   |   DEATHR.OWN.END <= 0.494624
|   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.493333
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.483333: 1 (1171.0)
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.483333
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.48913
|   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.486486: 0 (713.0)
|   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.486486
|   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.5: 1 (66.0)
|   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.5: 0 (160.0)
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.48913
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.480392: 0 (92.0)
```

```
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.480392
|   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.493151: 1 (930.0)
|   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.493151
|   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.495327: 1 (258.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.495327
|   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.5: 0 (145.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.5: 1 (232.0)
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.493333: 1 (1680.0)
|   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.494624
|   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.485714: 0 (725.0)
|   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.485714
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.495868: 0 (303.0)
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.495868
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.488095: 1 (227.0)
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.488095
|   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.490385: 0 (188.0)
|   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.490385
|   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.491803: 1 (334.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.491803
|   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.493333: 0 (283.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.493333
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.508475
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.507042
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.494624: 1 (136.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.494624
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.495327: 0 (265.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.495327
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.505747
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.50495: 1 (1455.0/609.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.50495: 0 (77.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.505747: 1 (99.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.507042: 0 (284.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.508475: 1 (167.0)
|   |   |   |   |   |   KILLR.OWN.END > 0.511905
|   |   |   |   |   |   |   KILLR.OWN.END <= 0.513514
|   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.487179
|   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.512821: 0 (262.0)
|   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.512821: 1 (176.0)
|   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.487179: 0 (1201.0)
|   |   |   |   |   |   |   KILLR.OWN.END > 0.513514
|   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.52
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.484536
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.483333
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.516949
|   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.513889: 1 (86.0)
|   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.513889: 0 (408.0)
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.516949
|   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.47619: 0 (164.0)
|   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.47619: 1 (285.0)
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.483333: 1 (179.0)
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.484536
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.493671: 0 (1163.0)
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.493671
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.516667: 1 (644.0)
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.516667: 0 (103.0)
|   |   |   |   |   |   |   |   KILLR.OWN.END > 0.52
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.476744: 1 (678.0)
```

```
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.476744
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.48: 0 (700.0)
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.48
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.521127
|   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.4875: 0 (200.0)
|   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.4875: 1 (425.0)
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.521127: 1 (718.0)
|   |   |   |   |   DEATHR.OWN.END > 0.5: 1 (930.0)
|   |   |   KILLR.OWN.END > 0.525641: 0 (819.0)
|   |   KILLR.OWN.END > 0.527778
|   |   |   DEATHR.OWN.END <= 0.428571
|   |   |   |   KILLR.OWN.END <= 0.564706: 0 (329.0)
|   |   |   |   KILLR.OWN.END > 0.564706
|   |   |   |   |   DEATHR.OWN.END <= 0.427481: 0 (306.0)
|   |   |   |   |   DEATHR.OWN.END > 0.427481: 1 (309.0)
|   |   |   DEATHR.OWN.END > 0.428571
|   |   |   |   KILLR.OWN.END <= 0.534351: 1 (3065.0)
|   |   |   |   KILLR.OWN.END > 0.534351
|   |   |   |   |   KILLR.OWN.END <= 0.571429
|   |   |   |   |   |   KILLR.OWN.END <= 0.567901
|   |   |   |   |   |   |   DEATHR.OWN.END <= 0.431193
|   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.55814
|   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.542857
|   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.537037: 0 (103.0)
|   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.537037: 1 (120.0)
|   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.542857: 0 (222.0)
|   |   |   |   |   |   |   |   KILLR.OWN.END > 0.55814: 1 (179.0)
|   |   |   |   |   |   |   DEATHR.OWN.END > 0.431193
|   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.471698
|   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.5375: 1 (1717.0)
|   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.5375
|   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.538462
|   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.459184: 0 (427.0)
|   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.459184: 1 (556.0)
|   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.538462
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.544444: 1 (3169.0)
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.544444
|   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.544872: 0 (347.0)
|   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.544872
|   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.46729
|   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.446602
|   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.442857
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.555556
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.554688
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.4375: 0 (137.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.4375: 1 (1434.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.554688: 0 (253.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.555556
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.566265: 1 (3610.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.566265
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.566667: 0 (33.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.566667: 1 (233.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.442857
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.556818
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.446154: 1 (690.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.446154: 0 (145.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.556818
```

```
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.444444: 0 (283.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.444444
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.5625: 1 (114.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.5625: 0 (222.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.446602
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.459677: 1 (5040.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.459677
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.554688
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.547009: 0 (55.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.547009: 1 (538.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.554688: 0 (178.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.46729: 0 (180.0)
|   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.471698
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.472973: 0 (367.0)
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.472973
|   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.538462: 0 (95.0)
|   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.538462: 1 (703.0)
|   |   |   |   |   |   |   KILLR.OWN.END > 0.567901
|   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.568966
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.434211: 1 (83.0)
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.434211: 0 (694.0)
|   |   |   |   |   |   |   |   KILLR.OWN.END > 0.568966
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.4375
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.431818: 1 (328.0)
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.431818: 0 (219.0)
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.4375: 1 (541.0)
|   |   |   |   |   KILLR.OWN.END > 0.571429
|   |   |   |   |   |   DEATHR.OWN.END <= 0.438776: 1 (1510.0)
|   |   |   |   |   |   DEATHR.OWN.END > 0.438776
|   |   |   |   |   |   |   DEATHR.OWN.END <= 0.44086: 0 (34.0)
|   |   |   |   |   |   |   DEATHR.OWN.END > 0.44086: 1 (555.0)
DEATHR.OWN.END > 0.505263
|   DEATHR.OWN.END <= 0.606061
|   |   DEATHR.OWN.END <= 0.527473
|   |   |   DEATHR.OWN.END <= 0.505376: 0 (752.0)
|   |   |   DEATHR.OWN.END > 0.505376
|   |   |   |   DEATHR.OWN.END <= 0.525
|   |   |   |   |   KILLR.OWN.END <= 0.471429
|   |   |   |   |   |   DEATHR.OWN.END <= 0.507937
|   |   |   |   |   |   |   KILLR.OWN.END <= 0.465753: 1 (176.0)
|   |   |   |   |   |   |   KILLR.OWN.END > 0.465753: 0 (150.0)
|   |   |   |   |   |   DEATHR.OWN.END > 0.507937: 0 (1773.0)
|   |   |   |   |   KILLR.OWN.END > 0.471429
|   |   |   |   |   |   KILLR.OWN.END <= 0.494624
|   |   |   |   |   |   |   DEATHR.OWN.END <= 0.506494: 1 (611.0)
|   |   |   |   |   |   |   DEATHR.OWN.END > 0.506494
|   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.507937: 0 (682.0)
|   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.507937
|   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.489362
|   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.487805
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.486486
|   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.485714
|   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.47541: 1 (326.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.47541
|   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.514286
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.510204
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.479592: 0 (232.0)
```

```
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.479592: 1 (163.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.510204: 1 (562.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.514286
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.523364
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.522727
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.516129: 0 (551.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.516129
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.482759
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.479592: 1 (74.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.479592: 0 (176.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.482759: 1 (549.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.522727: 0 (608.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.523364: 1 (193.0)
|   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.485714: 0 (387.0)
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.486486: 1 (344.0)
|   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.487805: 0 (690.0)
|   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.489362: 1 (761.0)
|   |   |   |   |   |   KILLR.OWN.END > 0.494624
|   |   |   |   |   |   |   DEATHR.OWN.END <= 0.516854
|   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.506494: 0 (624.0)
|   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.506494
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.506849
|   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.5: 1 (401.0)
|   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.5
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.518072: 0 (213.0)
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.518072: 1 (143.0)
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.506849
|   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.513514: 0 (695.0)
|   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.513514: 1 (204.0)
|   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.516854: 0 (701.0)
|   |   |   |   |   |   DEATHR.OWN.END > 0.525
|   |   |   |   |   |   |   KILLR.OWN.END <= 0.471264: 1 (644.0)
|   |   |   |   |   |   |   KILLR.OWN.END > 0.471264: 0 (180.0)
|   |   |   DEATHR.OWN.END > 0.527473
|   |   |   |   DEATHR.OWN.END <= 0.572519
|   |   |   |   |   DEATHR.OWN.END <= 0.570175
|   |   |   |   |   |   DEATHR.OWN.END <= 0.5375
|   |   |   |   |   |   |   KILLR.OWN.END <= 0.448718: 1 (220.0)
|   |   |   |   |   |   |   KILLR.OWN.END > 0.448718: 0 (5541.0)
|   |   |   |   |   |   DEATHR.OWN.END > 0.5375
|   |   |   |   |   |   |   KILLR.OWN.END <= 0.472222
|   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.462963
|   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.460784
|   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.455128
|   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.453488
|   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.553191
|   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.539474
|   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.444444: 1 (330.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.444444: 0 (162.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.539474: 0 (6473.0)
|   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.553191
|   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.422535: 0 (1111.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.422535
|   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.433333
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.432432
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.555556: 0 (253.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.555556
```

```
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.55914: 1 (241.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.55914
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.423077: 1 (187.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.423077
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.427083: 0 (519.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.427083
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.561644: 0 (381.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.561644
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.567568: 1 (598.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.567568: 0 (719.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.432432: 1 (302.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.433333
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.441558
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.555556
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.436782: 1 (142.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.436782: 0 (201.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.555556: 0 (3290.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.441558
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.44186: 1 (387.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.44186
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.561644
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.444444
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.556818: 0 (133.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.556818: 1 (266.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.444444
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.554217: 1 (143.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.554217: 0 (2855.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.561644
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.446809: 0 (187.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.446809: 1 (472.0)
|   |   |   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.453488
|   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.55: 1 (667.0)
|   |   |   |   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.55: 0 (167.0)
|   |   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.455128: 0 (2175.0)
|   |   |   |   |   |   |   |   |   KILLR.OWN.END > 0.460784
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.568966: 1 (446.0)
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.568966: 0 (245.0)
|   |   |   |   |   |   |   |   KILLR.OWN.END > 0.462963: 0 (1850.0)
|   |   |   |   |   |   |   KILLR.OWN.END > 0.472222
|   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.475728: 1 (403.0)
|   |   |   |   |   |   |   |   KILLR.OWN.END > 0.475728: 0 (149.0)
|   |   |   |   |   DEATHR.OWN.END > 0.570175
|   |   |   |   |   |   KILLR.OWN.END <= 0.434211
|   |   |   |   |   |   |   DEATHR.OWN.END <= 0.571429: 0 (417.0)
|   |   |   |   |   |   |   DEATHR.OWN.END > 0.571429: 1 (310.0)
|   |   |   |   |   |   KILLR.OWN.END > 0.434211: 1 (593.0)
|   |   |   |   DEATHR.OWN.END > 0.572519
|   |   |   |   |   DEATHR.OWN.END <= 0.587629
|   |   |   |   |   |   KILLR.OWN.END <= 0.430233: 0 (8743.0)
|   |   |   |   |   |   KILLR.OWN.END > 0.430233
|   |   |   |   |   |   |   KILLR.OWN.END <= 0.430556: 1 (158.0)
|   |   |   |   |   |   |   KILLR.OWN.END > 0.430556: 0 (1272.0)
|   |   |   |   |   DEATHR.OWN.END > 0.587629
|   |   |   |   |   |   DEATHR.OWN.END <= 0.590909
|   |   |   |   |   |   |   KILLR.OWN.END <= 0.418182
|   |   |   |   |   |   |   |   KILLR.OWN.END <= 0.378378: 1 (283.0)
|   |   |   |   |   |   |   |   KILLR.OWN.END > 0.378378: 0 (1608.0)
```

```
|   |   |   |   |   |   KILLR.OWN.END > 0.418182: 1 (590.0)
|   |   |   |   |   DEATHR.OWN.END > 0.590909
|   |   |   |   |   |   KILLR.OWN.END <= 0.4
|   |   |   |   |   |   |   KILLR.OWN.END <= 0.392857: 0 (2025.0)
|   |   |   |   |   |   |   KILLR.OWN.END > 0.392857
|   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.604938
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.6
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END <= 0.597222: 0 (176.0)
|   |   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.597222: 1 (350.0)
|   |   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.6: 0 (459.0)
|   |   |   |   |   |   |   |   DEATHR.OWN.END > 0.604938: 1 (318.0)
|   |   |   |   |   |   KILLR.OWN.END > 0.4: 0 (4480.0)
|   DEATHR.OWN.END > 0.606061
|   |   DEATHR.OWN.END <= 0.636364
|   |   |   DEATHR.OWN.END <= 0.635514: 0 (11986.0)
|   |   |   DEATHR.OWN.END > 0.635514
|   |   |   |   KILLR.OWN.END <= 0.353659
|   |   |   |   |   KILLR.OWN.END <= 0.346154: 0 (234.0)
|   |   |   |   |   KILLR.OWN.END > 0.346154: 1 (143.0)
|   |   |   |   KILLR.OWN.END > 0.353659: 0 (359.0)
|   |   DEATHR.OWN.END > 0.636364: 0 (42764.0)

Number of Leaves  :     180

Size of the tree :  359
```