VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belgaum-590018

A PROJECT REPORT **(15CSP85)** ON

"LEAGUE OF LEGENDS MATCH OUTCOME PREDICTION"

Submitted in Partial fulfillment of the Requirements for the Degree of

Bachelor of Engineering in Computer Science & Engineering

By

RAKSHA S MANGASULI (1CR16CS128)

SANMUKH LODHAVIA (1CR16CS149)

SHIVAKUMAR (1CR16CS158)

SHRISTY SAUMYA (1CR16CS160)

Under the Guidance of,

Prof. Savitha S.

Asst. Professor, Dept. of CSE

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR INSTITUTE OF TECHNOLOGY

#132, AECS LAYOUT, IT PARK ROAD, KUNDALAHALLI, BANGALORE-560037

CMR INSTITUTE OF TECHNOLOGY

#132, AECS LAYOUT, IT PARK ROAD, KUNDALAHALLI, BANGALORE-560037

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the project work entitled **"LEAGUE OF LEGENDS MATCH OUTCOME PREDICTION"** carried out by **Ms. RAKSHA S MANGASULI**, USN **1CR16CS128**, **Mr. SANMUKH LODHAVIA**, USN **1CR16CS149**, **Mr. SHIVAKUMAR**, USN **1CR16CS158, Ms. SHRISTY SAUMYA**, USN **1CR16CS160,** bonafide students of CMR Institute of Technology, in partial fulfillment for the award of **Bachelor of Engineering** in Computer Science and Engineering of the Visveswaraiah Technological University, Belgaum during the year 2019-2020. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library.

The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said Degree.

.

Prof. Savitha S.	**Dr. Prem Kumar Ramesh**	**Dr. Sanjay Jain**
Asst. Professor	**Professor & Head**	**Principal**
Dept. of CSE, CMRIT	**Dept. of CSE, CMRIT**	**CMRIT**

External Viva

Name of the examiners — Signature with date

1. ________________

2. ________________

DECLARATION

We, the students of Computer Science and Engineering, CMR Institute of Technology, Bangalore declare that the work entitled "**LEAGUE OF LEGENDS MATCH OUTCOME PREDICTION**" has been successfully completed under the guidance of Prof. Savitha S, Computer Science and Engineering Department, CMR Institute of technology, Bangalore. This dissertation work is submitted in partial fulfillment of the requirements for the award of Degree of Bachelor of Engineering in Computer Science and Engineering during the academic year 2019 - 2020. Further the matter embodied in the project report has not been submitted previously by anybody for the award of any degree or diploma to any university.

Place:

Date:

Team members:

RAKSHA S MANGASULI (1CR16CS128) ____________________

SANMUKH LODHAVIA (1CR16CS149) ____________________

SHIVAKUMAR (1CR16CS158) ____________________

SHRISTY SAUMYA (1CR16CS160) ____________________

ABSTRACT

We employ logistic regression to predict which team will win a League of Legends match based upon the historical game play statistics of match participants. The most successful model achieves greater than 95% win prediction accuracy on the testing data. In this project, the online game League of Legends, developed by Riot Games in 2009, will be our case study. The main reason is everyone in our group had played and enjoyed the game and would like to bring a more passionate and informed view to this project than they would with another case study. An important aspect was also Riot Games' API, which allows the public to easily make queries and pull information from their servers. Another great selling point is the fact that League of Legends is one of the most popular modern day E-sports, meaning its balance updates affect people's lives on a professional level.

ACKNOWLEDGEMENT

I take this opportunity to express my sincere gratitude and respect to **CMR Institute of Technology, Bengaluru** for providing me a platform to pursue my studies and carry out my final year project.

I have a great pleasure in expressing my deep sense of gratitude to **Dr. Sanjay Jain,** Principal, CMRIT, Bangalore, for his constant encouragement.

I would like to thank **Dr. Prem Kumar Ramesh,** Professor and Head, Department of Computer Science and Engineering, CMRIT, Bangalore, who has been a constant support and encouragement throughout the course of this project.

I consider it a privilege and honor to express my sincere gratitude to my guide **Prof. Savitha S, Asst. Professor,** Department of Computer Science and Engineering, for the valuable guidance throughout the tenure of this review.

I also extend my thanks to all the faculty of Computer Science and Engineering who directly or indirectly encouraged me.

Finally, I would like to thank my parents and friends for all their moral support they have given me during the completion of this work.

TABLE OF CONTENTS

LIST OF FIGURES

TABLE INDEX

LIST OF ABBREVIATIONS

ADC	**Attack Damage Carry**
API	**Application Program Interface**
DOTA	**Defense of the Ancients**
DTC	**Damage to Champions**
KDA	**Kills, Deaths, Assists**
LOL	**League of Legends**
MOBA	**Multiplayer Online Battleground Arena**
RTS	**Real-Time Strategy**

CHAPTER 1

INTRODUCTION

1.1 An Introduction to Data & Analytics in the Gaming Industry

Just as it did with traditional sports, the collection, analysis, and use of all kinds of data is starting to change the way that competitive games are played and understood. Nowadays, as the E-sports became more and more popular, data analyses on E-sports games are also become much more common. Because of the foundations of multiple commercial E-sport Leagues, data analyses is playing a signifificant role, just like the role they play in NBA and other sport leagues. In this project, the online game League of Legends (LoL), developed by Riot Games in 2009, will be our case study.

The main reason is everyone in our group had played and enjoyed the game and would like to bring a more passionate and informed view to this project than they would with another case study. An important aspect was also Riot Games' API, which allows the public to easily make queries and pull information from their servers. Another great selling point is the fact that League of Legends is one of the most popular modern day E-sports, meaning its balance updates affect people's lives on a professional level.

1.2 Related Work

As League of Legends is a popular game, there are already a couple of applications that exist in the data analysis sphere. Websites like **League of Graphs and MetaSrc** collect basic stats and information from the League of Legends API and use it to construct simple analyses like what the most popular champion and what the win rate of two champions when played together are. What these existing applications do not currently do is prediction of a currently ongoing game with all of the features of that game in mind, and that is the hole that our project is attempting to fill.

1.3 Background on League of Legends as a Game

1.3.1 Game overview

League of Legends is a Multiplayer Online Battleground Arena (MOBA) style game developed by Riot Games. Each game consists of two teams with 5 players each. Each team starts the game at opposite sides of the arena in a base that contains their Nexus. The goal of each game is to overcome obstacles such as minions, structures and enemy players in order to destroy the enemy Nexus. Each team generally consists of 5 standard positions divided across the map: Top, Middle, Jungle, Attack Damage Carry(ADC)/Marksman and Support. Each player in a match, usually lasting between twenty minutes and an hour, controls a unique champion chosen from a pool of more than a hundred with differing characteristics and abilities. The game boasts 100 million monthly players and a flourishing competitive scene with millions in tournament prize pools as well as online viewers.

Fig 1.1. Snapshot of a League of Legends Match

With so many variables at play within each game it is impossible to say that one variable is going to determine the success or failure of any particular player. Also, with such a large community behind League of Legends, predicting match outcomes for casual players and tournament games would be interesting and valuable for players and fans.

1.3.2 Game information

League of Legends is an intricate game that requires knowledge of many elements to be played proficiently. In the game, two teams of five players each battle against each other on a battlefifield called Summoner's Rift. There are other battlefields with different objectives and layouts, but they will not be studied in our project. Summoner's Rift is comprised of each team's base, three lanes, and the jungle. The goal of League of Legends is to destroy the opposing team's Nexus, a structure located in the enemy base.

Fig1.2 League of Legends Game Map

The "gold differential" is an example of a raw statistic that can provide a sense of the state of a match at a glance. Usually when a team has more gold than the other team, it means their champions are carrying more powerful items and are more likely to win if they get into a scrap. Combined with individual KDA (Kills, Deaths, Assists) numbers and the overall number of kills a team has accrued, it's an easy way to let the viewers know who is winning. When we turn on an LCS broadcast, gold differential is one of the first numbers we see at the top of the screen. But statistics like these don't tell the whole story.

1.3.3 Game API

The collection of Data is made significantly easier when the game has an open API, or tools for scraping the game’s code to find relevant data points. Riot Games provides a public API endpoint to access nearly all kinds of data that would be available to see in the offificial game client. The API gives access to match data which includes the champions selected and the roles of the players playing in the match and a lot of other useful information.

1.4 Problem statement

To predict the outcome of a League of Legends match in real time, played by two teams against each other, consisting of a randomly generated set of players.

1.5 Relevance of the Problem

As League of Legends is a popular game, there are already a couple of applications that exist in the data analysis sphere. Websites like League of Graphs and MetaSrc collect basic stats and information from the League of Legends API and use it to construct simple analyses like what the most popular champion and what the win rate of two champions when played together are.What the existing applications do not currently do is prediction of a currently ongoing game with all of the features of that game in mind, and that is the hole that our project is attempting to fill.

1.6 Objectives

1. To create a better gaming experiences for every gamer that plays the game by giving better results.

2. To enable the viewers to find out if a particular team that they're supporting is going to win or not.

3. To give a better idea to the players to decide which objectives are more relevant and important for winning the game.

4. To help the team management to perform a better analysis of the game.

CHAPTER 2

LITERATURE SURVEY

2.1 Reference paper 1: MOBA GAMES

Some 205 million people watched or played eSports in 2014. Among them, MOBA games are in the lead, both in players and watchers. The League of Legends Championship sold out Staples Center in 2013, then sold out the 40,000-seat World Cup Stadium in Seoul a year later while drawing an online audience of 27 million. However ,even though many players are drawn every day to MOBA games, re-search on the subject is still at an early stage. This paper aims to reflect the current research landscape focused strictly on MOBA games, in order to provide a summary of the topics covered and to open venues for new lines of investigation.

Existing studies acknowledge the early stage of current research on MOBA games. “Despite its vast, enthusiast community and influence on contemporary game designers, the MOBA remains under-explored by academics. But few games exhibit a greater need for socially-aware services than this relatively new genre which brings new ways of collaboration and competition on the table, gender and cultural chal-lenges and even new social networks which need to deal with the in-herent toxic behaviour that arises in these contexts. In essence, MOBA games are a subgenre of real-time strategy games in which two teams, typically consisting of five players each, compete against each other with each player controlling a single character.

Contrary to real-time strategy games, there is no unit or building con-struction in a MOBA game, so “much of the strategy revolves around individual character development and cooperative team play in combat. Two of the three most played PC games are MOBA games: League of Legends and DOTA 2. And yet, it all started from a small and niche fan-made custom map for Blizzard’s real-time strategy (RTS) game StarCraft, Aeon of Strife (AoS), back in 1998, by a modder (an in-dividual who deliberately modifies games to his advantage or for fun)called Aeon64. While AoS set the basics, it wasn’t until Defense of the Ancients (DOTA) when the MOBA genre was born as we know it today.

In 2009, Riot Games released League of Legends with a completely different pricing approach: it was free to play. Anyone could download it and play with a rotating selection of heroes and some limitations –extra content could be purchased in-game, though.Besides LoL, the second most popular MOBA game as of today (with more than 11 million unique players per month according to their website) is DOTA 2, which was released in 2013 through Valve's STEAM digital store. Both games are similar in concept but they differ in execution; overall, League of Legends simplified DOTA while DOTA 2 is often considered more demanding and strategically complex than LoL.

2.2 Reference paper 2: Machine Learning to analyze League of Legends

When League of Legends was released in 2009, few people could have predicted what was to follow. The undeniable increase of eSports has been led by the ever-popular platform produced by Riot Games. The last benchmark that was made public by the creators indicated over 100 million monthly users. These figures have given League of Legends the top spot amongst the MOBA. We will firstly examine the market before the huge success of LoL and it's competition. For those who are unaware of how the game works, the classic version of the game consists of two teams of five competitors aiming to destroy the base of the competition whilst defending their own.

Then we will dig a little deeper. We have set out to characterize the team play and predict the results of some of the professional matches that will be played in the future. We have based our insights on data published by Tim Sevenhuysen using Oracle's Elixir. This data set includes information on each competition (both on an individual and group level). The data covers all facets of the game including: gold obtained, damage caused, and "farming." This data set forms the foundation of our analysis.

Data and Planning

Our analysis uses data taken from 7 different leagues that were distributed worldwide with the results of each game. What we are trying to predict for future games is whether it will be a victory or not using a simple data classifier. We have used statistics from the 2017 Spring Split, which started with an extensive data set. We have access to the all the variables (player, winner, team, gold, damage etc.) from each match which is then further divided per team and finally per player from the ten from each team.

2.3 Reference paper 3: Win Prediction in Multi-Player Esports: Live Professional Match Prediction

Esports is the term used to describe video games that are played competitively and watched by, normally large,audiences. Esports is an important research field across academia and industry just in terms of size .Goldman Sachs predicted a compound annual growth rate of 22% with the market worth $1.1 billion by 2019 and Superdata estimated there will be 330 million spectators by 2019. The availability of detailed data from virtually every match played coupled with this huge expansion has introduced the field of esports analytics .Esports analytics is defined by as: "the process of using esports related data to find meaningful patterns and trends in said data and the communication of these patterns using visualization techniques to assist with decision-making processes".

This definition highlights a fundamental challenge in esports: making the matches comprehensible to the audience. Many esports are complex and fast-paced, making it hard to fully unpack thelive action with the naked eye. MOBAs also provide a fertile testing ground for machine learning due to the availability of high-dimensional, high-volume data .Esports analytics has focused on the Multi-player Online Battle Arena (MOBA) genre, which is arguably the most common esports format. MOBA titles such as League of Legends,DotA 2 and Heroes of Newerth attract hundreds of millions of players.

Within esports, but not ablyin MOBAs, win prediction has formed the focal point of analytics research across industry and academia, even if that research is somewhat fragmented.

However, previous work has several limitations, including the fact that it is mainly focused on pre-match predictions, which informs betting ,rather than models that can integrate live data streams, and seek to inform and engage the audience. There is also a lack of research at the professional level, despite differences in player behavior as a function of skill being documented.

Furthermore, no previous prediction models have been adapted for and tested in actual esports tournaments . However, this previous work has limitations. In many ways this is due to esports analytics being an emergent field of inquiry. We detail these limitations in under-prioritizing data from professional play-ers, building models from data across the entire skill set which lowers the accuracy for professional match win prediction, only predicting historical data rather than real-time (live) prediction, and using data generated overlong time periods across significant game updates and changes.

Unlike traditional sports, in which the game rules are mostly stable, in esports major updates can significantly alter the core characteristics of the game mechanics. These major updates could render previous data obsolete.The focus of this paper is to use live game state (e.g.positions of players, performance metrics etc.) to predict the likely winner for the popular MOBA game DotA 21.This paper builds on and significantly expands a preliminary feasibility report which demonstrated prediction on a small data set and established some data features to use from an initial set of possibilities.

2.4 Reference paper 4: Data Analytics Applications in Gaming and Entertainment

Use of data science (e.g. artificial intelligence) techniques has spread over many fields, with a wide range of purposes. The huge, and constantly growing, amounts of data being captured allow complex techniques to provide insights which are potentially deeper than those that can be found by applying only traditional, and often simpler, methods. The application of data science techniques perfectly fits interactive environments, where multiple data can be generated.

One of these environments that allows for multiple interactions is games. In particular, the use of games with purposes beyond entertainment (e.g. learning, raising awareness or changing attitudes and behaviors), that is, so called serious games, has also increased in the last years. These types of games are especially popular in domains such as medicine or the military, and have proven their effectiveness for children 2 and adolescents, as the familiarity of these users with gaming environments and the characteristics of games (interactivity, motivation, engagement) facilitate their interactions with serious games.

The collection and analysis of data has reached a great number of fields: in education, the fields of educational data mining and learning analytics, sometimes used interchangeably, are widely spread. Their aim is to understand learners and their environments and improve the learning process through analysis of the data collected from students' interactions with the learning environment. As with any other highly interactive system, a lot of data can also be gathered from serious games to guide data-based decision-making.

Building up from the fields of educational data mining and learning analytics, which focus in education in general, game learning analytics is defined as the collection, analysis and extraction of information from data collected from serious games. The aim of the current paper is to conduct a systematic literature review on the applications of data science techniques to analyze game analytics data and/or learning analytics data from serious games.

2.5 Reference paper 5: Predicting Customer Lifetime Value in Free-to Play Games

As game companies increasingly embrace a service-oriented business model, the need for predictive models of player behavior becomes more pressing. Multiple activities, such as user acquisition, live game operations, or game design, need to be supported with information about the choices made by the players and the choices they could make in the future. This is especially true in the context of free-to play games, where the absence of a pay wall and the erratic nature of the players playing and spending behavior makes predictions about the revenue and allocation of budget and resources extremely challenging. We present an overview of customer lifetime value modeling across different fields.

We will introduce the challenges specific to free-to-play games across different platforms and genres, and we will discgy the state-of-the-art solutions with practical examples and references to existing implementation. Customer lifetime value refers broadly to the revenue that a company can attribute to one or more customer over the length of their relationship with the company . The process of predicting the lifetime value consists in producing one or more monetary values that correspond to the sum of all the different types of revenues that a specific customer, or a specific cohort, will generate in the future.

The purposes of this prediction are manifold: for example, having an early estimation of a customer's potential value allows more accurate budgeting for future investment; moreover, monitoring the remaining potential revenue from an established customer could permit preemptive actions in case of decreased engagement

CHAPTER 3

SYSTEM REQUIREMENTS SPECIFICATION

3.1 Hardware requirements

1. Processor - I3/Intel processor

2. RAM - 4GB (min)

3. Hard disk - 160GB

4. Keyboard - Standard Windows keyboard

5. Mouse

6. Monitor - SVGA

3.2 Software requirements

1. Operating system - Windows 7/8/10

2. R

3. R studio

4. Shiny libraries

5. Additional libraries

CHAPTER 4

SYSTEM ANALYSIS AND DESIGN

4.1 Proposed system

We will explain our proposed system, what it is and how we are using it in our project specifically as well.

Fig.4.1 Predictive Analysis using R

There are 7 stages:

1. Define problem statement

Firstly we start by describing the objectives for our project which is to predict the outcome of the match.

2. Data collection

We chose one of the websites to get all the data available for each game played during a particular season.

3. Data cleaning

We use the correlation function to get an idea about the variables that are highly correlated with the variable result.

4. Data analysis

We complete the process of cleaning, transforming, inspecting and modeling data so that the data can be analysed.

5. Build predictive model

We create models using logistic regression. We choose a model that is most accurate.

6. Validate model

Then the model is validated with the test samples and checked for its accuracy.

7. Deployment

Based on the most accurate model, we create a web application using R Shiny.

4.2 The data

4.2.1 Data collection

Many websites present data gathered using the public Riot Games API , which allows collection of data about summoners (the LoL term for players), champions, and past matches. Some of these websites analyze the data or present it visually, while others just display the data itself. We chose one of those websites to get a fifile in .xlsx format containing all the data available for each game played during the 2017 season. The downloaded dataset contained more than 75 variables, some of them were not meaningful for us. So we decided to get rid of those variables that won’t be helpful in our analysis.

gameid	league	split	week	game	playerid	side	position	player	team	champi	ban1	ban2	ban3	ban4	ban5	gamele	result	k	d	a	teamkil
1,002E+09	NALCS	2017-2	9,3	3	1	Blue	Top	Lourlo	Team Liqu	Gnar	LeBlanc	Zac	Shen	Tristana	Kog'Maw	25,38333	0	0	2	0	3
1,002E+09	NALCS	2017-2	9,3	3	2	Blue	Jungle	Dardoch	Team Liqu	Gragas	LeBlanc	Zac	Shen	Tristana	Kog'Maw	25,38333	0	0	5	1	3
1,002E+09	NALCS	2017-2	9,3	3	3	Blue	Middle	Mickey	Team Liqu	Ekk	LeBlanc	Zac	Shen	Tristana	Kog'Maw	25,38333	0	0	3	0	3
1,002E+09	NALCS	2017-2	9,3	3	4	Blue	ADC	Piglet	Team Liqu	Vayne	LeBlanc	Zac	Shen	Tristana	Kog'Maw	25,38333	0	0	3	1	3
1,002E+09	NALCS	2017-2	9,3	3	5	Blue	Support	Matt	Team Liqu	Lulu	LeBlanc	Zac	Shen	Tristana	Kog'Maw	25,38333	0	3	5	0	3
1,002E+09	NALCS	2017-2	9,3	3	6	Red	Top	Ssumday	Dignitas	Jarvan IV	Caitlyn	Kalista	Thresh	Blitzcrank	Bard	25,38333	1	3	0	9	18
1,002E+09	NALCS	2017-2	9,3	3	7	Red	Jungle	Shrimp	Dignitas	Maokai	Caitlyn	Kalista	Thresh	Blitzcrank	Bard	25,38333	1	2	1	15	18
1,002E+09	NALCS	2017-2	9,3	3	8	Red	Middle	Keane	Dignitas	Taliyah	Caitlyn	Kalista	Thresh	Blitzcrank	Bard	25,38333	1	8	0	8	18
1,002E+09	NALCS	2017-2	9,3	3	9	Red	ADC	Altec	Dignitas	Varus	Caitlyn	Kalista	Thresh	Blitzcrank	Bard	25,38333	1	4	1	9	18
1,002E+09	NALCS	2017-2	9,3	3	10	Red	Support	Adrian	Dignitas	Janna	Caitlyn	Kalista	Thresh	Blitzcrank	Bard	25,38333	1	1	1	14	18
1,002E+09	NALCS	2017-2	9,3	3	100	Blue	Team	Team	Team Liqu		LeBlanc	Zac	Shen	Tristana	Kog'Maw	25,38333	0	3	18	2	3
1,002E+09	NALCS	2017-2	9,3	3	200	Red	Team	Team	Dignitas		Caitlyn	Kalista	Thresh	Blitzcrank	Bard	25,38333	1	18	3	55	18
240067	LCK	2017-2	1,1	1	1	Blue	Top	ADD	MVP	Sejuani	Xayah	Zac	Lee Sin	Nidalee	Rengar	40,25	0	0	4	3	11
240067	LCK	2017-2	1,1	1	2	Blue	Jungle	Beyond	MVP	Graves	Xayah	Zac	Lee Sin	Nidalee	Rengar	40,25	0	4	2	4	11
240067	LCK	2017-2	1,1	1	3	Blue	Middle	Ian	MVP	Taliyah	Xayah	Zac	Lee Sin	Nidalee	Rengar	40,25	0	2	3	4	11
240067	LCK	2017-2	1,1	1	4	Blue	ADC	MaHa	MVP	Ashe	Xayah	Zac	Lee Sin	Nidalee	Rengar	40,25	0	4	0	5	11
240067	LCK	2017-2	1,1	1	5	Blue	Support	Max	MVP	Thresh	Xayah	Zac	Lee Sin	Nidalee	Rengar	40,25	0	1	3	7	11
240067	LCK	2017-2	1,1	1	6	Red	Top	ikssu	Jin Air Gre	Jayce	Galio	Syndra	Elise	Kennen	Fiora	40,25	1	5	2	4	12
240067	LCK	2017-2	1,1	1	7	Red	Jungle	UmTi	Jin Air Gre	Ivern	Galio	Syndra	Elise	Kennen	Fiora	40,25	1	0	1	9	12
240067	LCK	2017-2	1,1	1	8	Red	Middle	Kuzan	Jin Air Gre	Orianna	Galio	Syndra	Elise	Kennen	Fiora	40,25	1	2	3	6	12
240067	LCK	2017-2	1,1	1	9	Red	ADC	Teddy	Jin Air Gre	Varus	Galio	Syndra	Elise	Kennen	Fiora	40,25	1	3	1	6	12
240067	LCK	2017-2	1,1	1	10	Red	Support	SnowFlow	Jin Air Gre	Zyra	Galio	Syndra	Elise	Kennen	Fiora	40,25	1	2	4	8	12
240067	LCK	2017-2	1,1	1	100	Blue	Team	Team	MVP		Xayah	Zac	Lee Sin	Nidalee	Rengar	40,25	0	11	12	23	11
240067	LCK	2017-2	1,1	1	200	Red	Team	Team	Jin Air Gre		Galio	Syndra	Elise	Kennen	Fiora	40,25	1	12	11	33	12
240080	LCK	2017-2	1,1	2	1	Blue	Top	ikssu	Jin Air Gre	Jarvan IV	Syndra	Galio	Ashe	Kennen	Graves	29,63333	0	0	4	4	5
240080	LCK	2017-2	1,1	2	2	Blue	Jungle	UmTi	Jin Air Gre	Lee Sin	Syndra	Galio	Ashe	Kennen	Graves	29,63333	0	2	4	2	5
240080	LCK	2017-2	1,1	2	3	Blue	Middle	Kuzan	Jin Air Gre	Taliyah	Syndra	Galio	Ashe	Kennen	Graves	29,63333	0	2	1	0	5
240080	LCK	2017-2	1,1	2	4	Blue	ADC	Teddy	Jin Air Gre	Varus	Syndra	Galio	Ashe	Kennen	Graves	29,63333	0	0	2	1	5
240080	LCK	2017-2	1,1	2	5	Blue	Support	SnowFlow	Jin Air Gre	Zyra	Syndra	Galio	Ashe	Kennen	Graves	29,63333	0	1	5	0	5
240080	LCK	2017-2	1,1	2	6	Red	Top	ADD	MVP	Sejuani	Zac	Elise	Thresh	Jayce	Fiora	29,63333	1	3	2	5	16

Fig 4.2. Dataset file in .xlsx format

So, we decided to delete 25 variables that won't be helpful in our prediction. Now, we have a dataset which contains 50 variables.

4.2.2 Data Dictionary

In order to interpret downloaded match data file, we tried to make a sort of dictionary containing the variable names and descriptions below:

Table 1.1 Data Dictionary

Variable	Description
gameid	Game identifier from Riot's server.
league	League
split	Time period covered, denoted by year and suffixes (1 spring, 2 summer, po playoffs, r regionals, w worlds)
week	"week within season," decimal, "day within week".
game	Game number within the series. T tiebreaker.
playerid	Player/team identifier, for differentiating map side and position.
side	Map side.
position	Player position.
player	Player name.
team	Team name.
champion	Champion name.
ban1	Team's first ban.
ban2	Team's second ban.
ban3	Team's third ban.
ban4	Team's fourth ban.
ban5	Team's fifth ban.
gamelength	Game length, in minutes. (Seconds measured in hundredths of a minute.)
result	Game result (1 win, 0 loss).
k	Total kills.
d	Total deaths.
a	Total assists.
teamkills	Total kills by team.
teamdeaths	Total deaths by team.
fb	First blood kill (1 yes, 0 no).
fbassist	First blood assist (1 yes, 0 no).
fbvictim	First blood victim (1 yes, 0 no).
fbtime	First blood time, in minutes. (Seconds measured in hundredths of a minute.)
kpm	Kills per minute (individuals and teams reported separately).
okpm	Opponent kills per minute (for players, reflects opponent in same position).
ckpm	Combined kills per minute (own kills and opponent kills combined).
fd	First dragon of game killed (1 yes, 0 no).

fdtime	First dragon time, in minutes. (Seconds measured in hundredths of a minute.)
teamdragkills	Total dragons killed by team.
oppdragkills	Total dragons killed by opposing team.
ft	First tower of game killed (1 yes, 0 no).
fttime	First tower kill time, in minutes. (Seconds measured in hundredths of a minute.)
teamtowerkills	Total towers killed by team.
opptowerkills	Total towers killed by opposing team.
dmgtochamps	Total damage dealt to champions.
dmgshare	Share of team's total damage dealt to champions.
earnedgoldshare	Share of team's total gold, with starting gold and inherent gold generation removed.
totalgold	Total gold earned from all sources.
goldspent	Total gold spent.
cspm	Creep score per minute. All creep score variables include minions and monsters.
goldat10	Total gold earned at 10:00.
oppgoldat10	Opponent's total gold earned at 10:00.
gdat10	Gold difference at 10:00.
goldat15	Total gold earned at 15:00.
oppgoldat15	Opponent's total gold earned at 15:00.
gdat15	Gold difference at 15:00.

4.3 Use Case Diagram

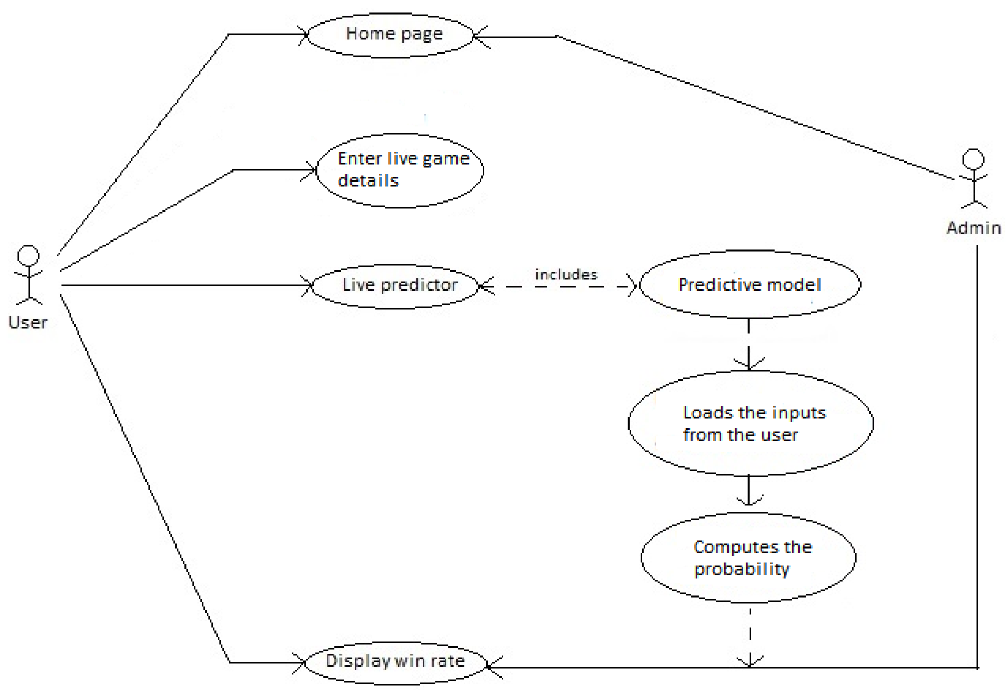

Fig 4.3 Use case diagram

As shown in Fig 4.3, there is a home page where the user can enter live game details at some point of time. There is a live predictor which includes predictive model. The model loads the inputs from the user. Then it computes the probability of winning. Thus the win rate is displayed. The user and admin can both access it.

4.4 Activity Diagram

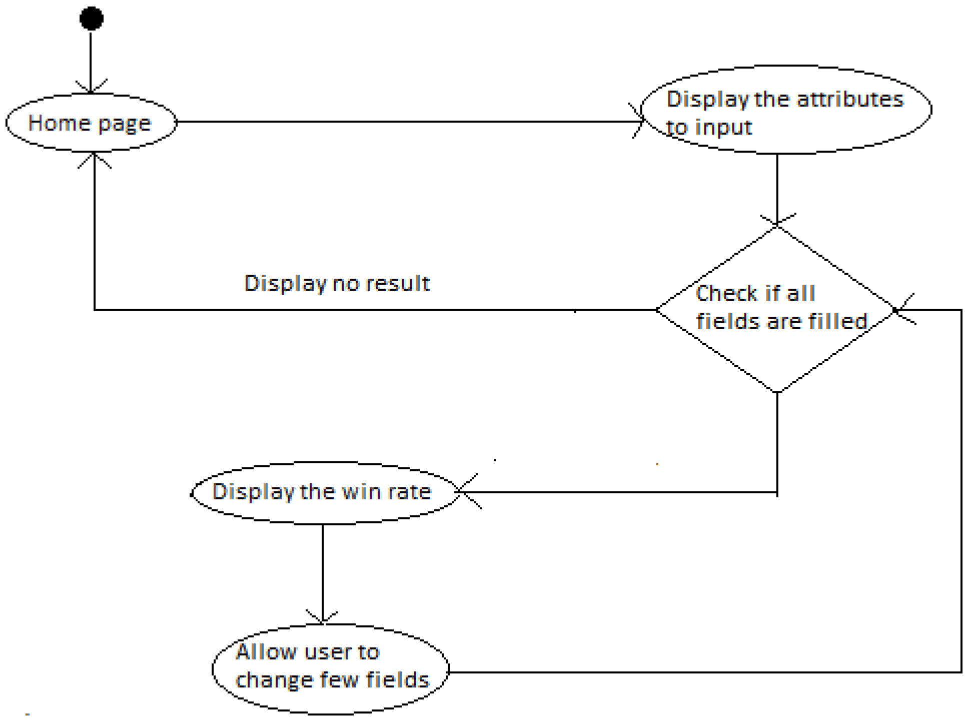

Fig 4.4 Activity Diagram

As can be seen in Fig. 4.4, the first entry point is Home Page. We display the attributes to input. It needs to be checked if all the fields are filled or not. If not, then no result is displayed. If they are filled, then the win rate is displayed automatically. We allow the user to change the values of a few fields and again checked if the fields are filled or not, if yes then win rate is displayed.

4.5 Flowchart

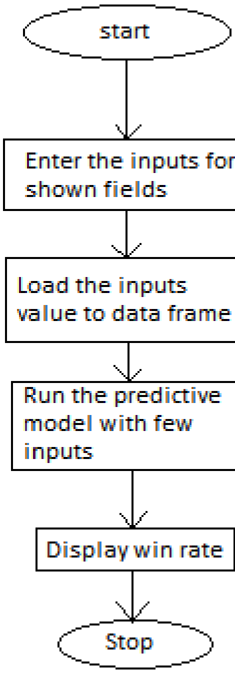

Fig. 4.5 Flowchart

As can be seen in Fig. 4.5, first we enter the inputs for shown fields. Then to the dataframe we load the input values. Then we run the predictive model with few inputs. Thus the win rate is displayed.

CHAPTER 5

IMPLEMENTATION

5.1 Logistic regression

Logistic regression is a method to predict a dependent variable given a set of independent variables , such that dependent variable is categorical.

Dependent variable(Y):

The response binary variable holding values like 0 or 1, Yes or No.

Independent variable (X):

The predictor variable used to predict the response variable.

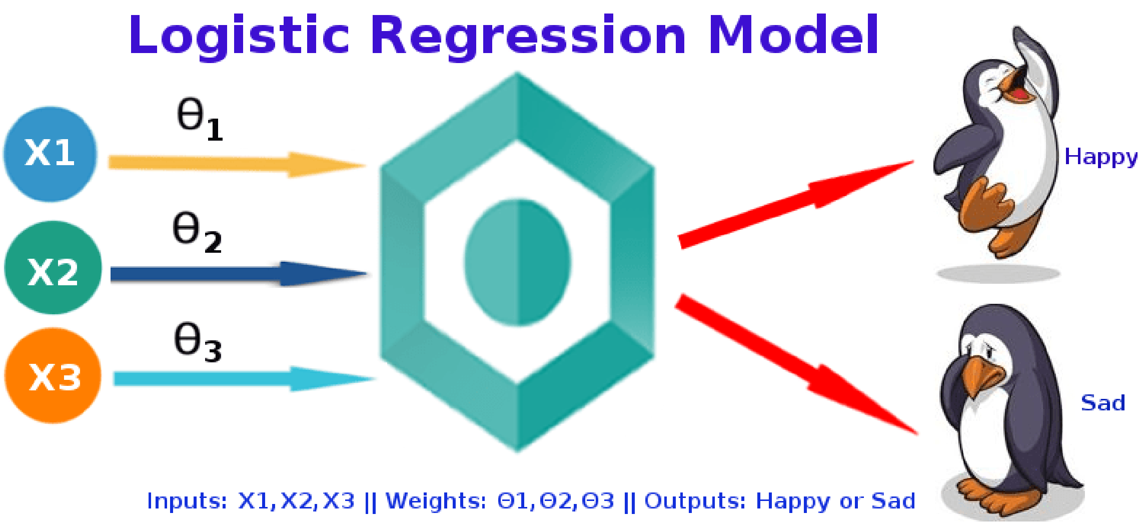

Fig.5.1 Logistic regression model

The following equation is used to represent a linear regression model:

$$\log(Y/1-Y) = C + B1X1 = B2X2 + \ldots \qquad (5.1)$$

Eqn 5.1 represents a logistic regression model

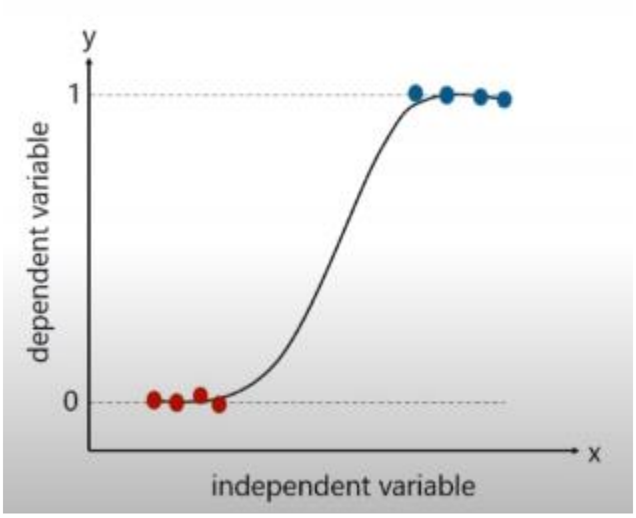

Fig.5.2 Graph representing logistic regression

Fig 5.2 is the graphical representation of logistic regression.

5.2 Data Analysis

We complete the process of cleaning, transforming, inspecting and modeling data so that the data can be analysed.

5.2.1 Winner Prediction

Actually, using only prematch knowledge (champions, masteries, roles, spells) from the very start is only a weak predictor of match outcome but using in-game statistics, the model becomes a strong predictor. That's why we will try to build our model based on in-game statistics. In fact, we're not sure which of our variables are useful in predicting a result outcome. In fact, it's often helpful to build bivariate models, which are models that predict the outcome using a single variable. But the problem is which of the 28 variables is a signifificant predictor of the Result variable in a bivariate logistic regression model?

In order to avoid building 28 different models, we will use the correlation function to get an idea about the variables that are highly correlated with the variable Result.

Code snippet for cleaning data:

```
# Removing Rows with all NAs (missing values) in the dataframe
data_clean=data_lol[complete.cases(data_lol), ]

# Correlation with result
i1 <- sapply (data_clean, is.numeric)
y1 <- “result”
x1 <- setdiff (names(data_clean)[i1],  y1)
cor (data_clean[x1], data_clean[y1])
```

Now that we have prepared the dataset, we need to split it into a training and testing set. We set the random seed to 1 and use the sample.split function to select the 50% of observations for the training set (the dependent variable for sample.split is Result), and we name the data frames train and test.

Code snippet for splitting data into training and testing set

```
# Splitting into a training and testing set
set.seed(10)
library (caTools)
split = sample.split (data_clean$result, SplitRatio = 0.5)

train = subset (data_clean, split == TRUE)
test = subset (data_clean, split == FALSE)
```

Now, we use logistic regression trained on the training set in order to predict the dependent variable Result using all the independent variables. To determine signifificance, we look at the stars in the summary output of the model. We defifine an independent variable as signifificant if there is at least one star at the end of the coeffificients row for that variable (this is equivalent to the probability column having a value smaller than 0.05).

5.2.2.1 Logistic Regression Model

In fact, we need to build different models so as to fifind the best one with the highest accuracy. First, we will exclude the variables referring to stats at the end of the game, because it would be too late to predict a win in this case. Also, we will omit the variables related to individual stats since it is a team game and one player can’t win by himself.

Model 1: Using goldat10, goldat15, oppgoldat10 and oppgoldat15 :

Model1 <- glm (result ~ goldat10+oppgoldat10+goldat15+oppgoldat15, data=train, family = binomial)
summary(Model1)

Model 2: Using gdat10 and gdat15:

Model2 <- glm (result ~ gdat10+gdat15, data=train, family = binomial)
summary (Model2)

Since it is more convenient to work with the second model in order to make predictions. We also compute the confusion matrix using a threshold of 0.5.

Prediction using Model 2

testPredict = predict (Model2, type="response", newdata=test)
table (test$result, testPredict>0.5)

Confusion matrix for Model 2

	FALSE	TRUE
0	4196	2228
1	2291	4155

Accuracy for Model 2: 0.64887 or 64.88%

We can deduce that the model has poor accuracy.

Model 3: We use logistic regression trained on the training set to predict the dependent variable result using fb, teamkills, teamdeaths as independent variables.

Model3 <- glm (result ~ fb+teamkills+teamdeaths, data=train, family = binomial)
summary (Model3)

Prediction using Model 3

testPredict = predict (Model3, type="response", newdata=test)
table (test$result, testPredict>0.5)

Confusion matrix for Model 3

	FALSE	TRUE
0	6016	408
1	300	6146

Accuracy : 0.94499 or 94.99%

Clearly,this model has much more accuracy than Model 2.

Model 4:

We use logistic regression trained on the training set to predict the dependent variable **result** using **ft, teamtowerkills, opptowerkills** as independent variables.

Model4 <- glm (result ~ ft+teamtowerkills+opptowerkills, data=train, family = binomial)
summary (Model4)

Prediction using Model 4

testPredict = predict (Model4, type=”response”, newdata=test)
table (test$result, testPredict>0.5)

Confusion matrix for Model 4

	FALSE	TRUE
0	6273	151
1	146	6300

Accuracy for Model 4: 0.9763 or 97.63%

So its logical that building a predictive model based on variables related to towers statistics is more reasonable, since achieving a victory condition, typically destroying the core building(the Nexus) , teams need to bypass all the towers in a line.

Model 5: Final Model, using all significant variables

Model_final <- glm (result gdat10+gdat15+fb+teamkills+teamdeaths+ft+teamtowerkills+ opptowerkills +fd+teamdragkills, data=train, family=binomial)

Summary (Model_final)

Prediction using final model:

testPredict = predict (Model_final, type=”response”, newdata=test)
table (test$result, testPredict>0.5)

Confusion matrix for final model

	FALSE	TRUE
0	6318	106
1	95	6351

Confusion Matrix Explained

• Correct : Predicted Red Team to win and actually Won : 6318 times

• Correct : Predicted Blue Team to win and actually Won : 6351 times

• Wrong: Predicted Red Team to win but Red team Lost : 95 times

• Wrong: Predicted Blue Team to win but Blue team Lost : 106 times

Accuracy : 0.98438 or 98.43%

The final model has the desired accuracy.

5.3 Front end / Back end implementation details:

We used R shiny to build the website.At its base level, Shiny is an R package that brings R to the web. Shiny is based on a reactive programming model, similar to a spreadsheet. Spreadsheet cells can contain literal values, or formulas that are evaluated based on other cells. Whenever the value of the other cells change, the value of the formula is automatically updated.

Shiny apps behave the same way. However, unlike a spreadsheet which requires a spreadsheet program to use reactively, a Shiny app is simply a web application created in R. With Shiny you can make your data analyses reactive and accessible to anyone with a web browser, without having to know anything about web programming. We used the shiny package to design the front end and back end or model or code is written in R.

Therefore, we decide to build our web application containing: Our best prediction model which has 98% as accuracy; you can fill a form and get the percentage of your win chance.

CHAPTER 6

RESULTS AND DISCUSSION

Using our best prediction model which has 98% as accuracy; you can fill a form and get the percentage of your win chance.Once the game has started, at 10 mins and 15 mins, the viewers can enter the gold difference, gold earned and find out whether their team is winning or not. The players can also use the web application at the end of the game and check what were their chances of winning at 15 mins after the game started, where they went wrong, what they could have done better.

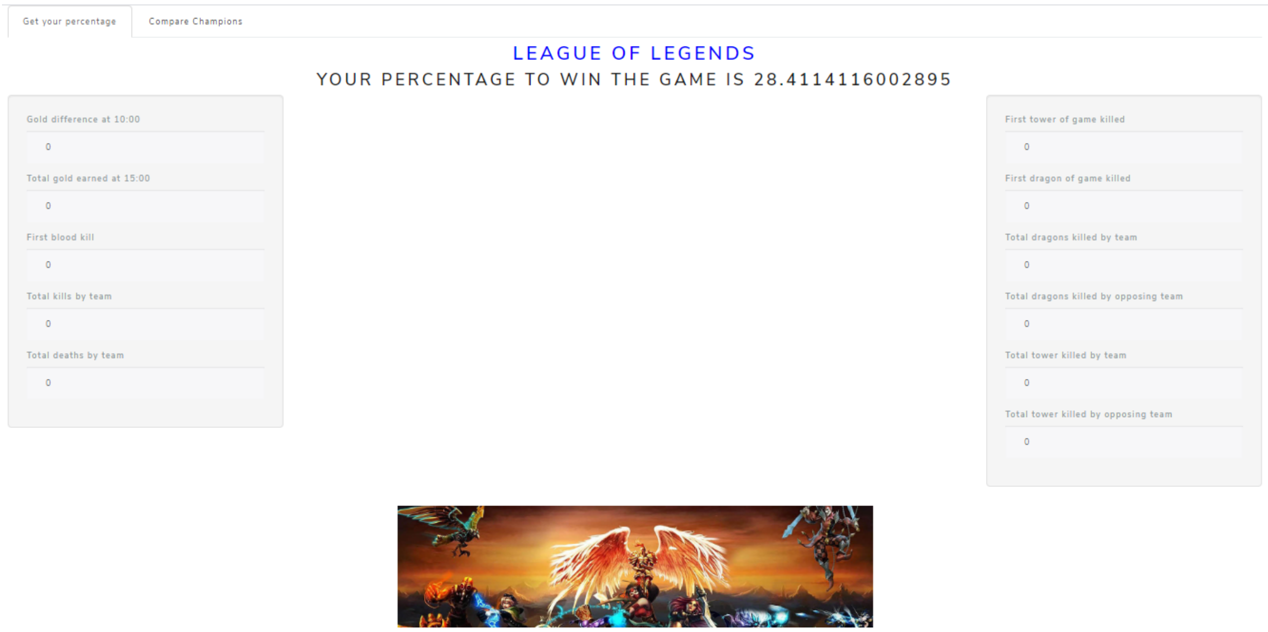

Fig.6.1 Preview of our web application

Objectives like total gold earned by a particular team at 15 mins is a decisive factor about which team wins.We have used the most basic and the most output-yielding factors that we can for calculating the win rate.These factors are interdependent so it might biased upto an extend. Through various permutations and combinations , we have tried to select the most affecting factors. Our all different models have accounted for the same thing . However, we have realized that the gold at 15 mins is the most important and affecting feature contributing the win rate .

Also, our prediction does not however guarantee the win due to the various luck based and time dependent factors. The result is up to as accurate s we can get but it also relies on how much every factor that we did not take in consideration in our game like team tower kills etc.

We concluded the following data from the Apii collection from the Riot games:

i) The average Win Ratio for a Pro Team is 45%.

ii) Team WE is the most appeared team in all the Leagues with 143 games

iii) SK Telecom T1 has the highest Win Ratio. Out of 139 games they managed to win 97 leading to an outstanding 69,78% Win Ratio .

Despite there being many existing sites that mined data through the Riot API and release interesting, personalized statistics, none can effectively predict the winner team during a game. Our project takes on this task using different variables that are related to both team stats in order to predict the winner.

Here we consider two cases in which the values of different factors are entered during a game to our web application .

Case I)Win rate :99.99%

i) Gold difference after first 10 mins of game:-2400 (the opponent has 2400 gold more than the current team)

ii)Total gold earned at the end of 15 mins: 3500

iii)First blood killed after: 7mins

iv)Total deaths by team: 7

v)Total kills by the team: 6

vi)Time when first tower was killed: 5mins

vii)Time of first dragon was killed: 5 mins

viii)Total dragons killed by the team: 1

ix)Total dragons killed by the opposite team: 1

x)Total number of towers killed by the team: 1

xi)Total towers killed by the opposing team: 0

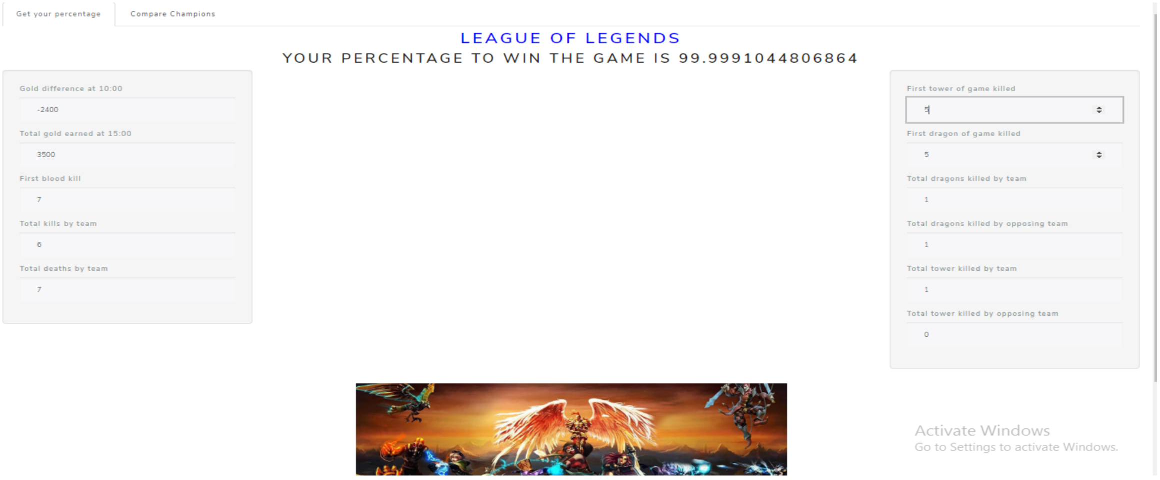

Fig 6.2 Snapshot for case I

We see that the win percent changes to 99.99(approximately), which is much closer to winning the game .

Case II)Win rate :71.14%

i) Gold difference after first 10 mins of game: 3000

ii) Total gold earned at the end of 15 mins: 4000

iii) First blood killed after: 7 mins

iv) Total deaths by team: 8

v) Total kills by the team: 7

vi) Time when first tower was killed: 0 mins

vii) Time of first dragon was killed: 5 mins

viii) Total dragons killed by the team: 2

ix) Total dragons killed by the opposite team: 1

x) Total number of towers killed by the team: 1

xi) Total towers killed by the opposing team: 0

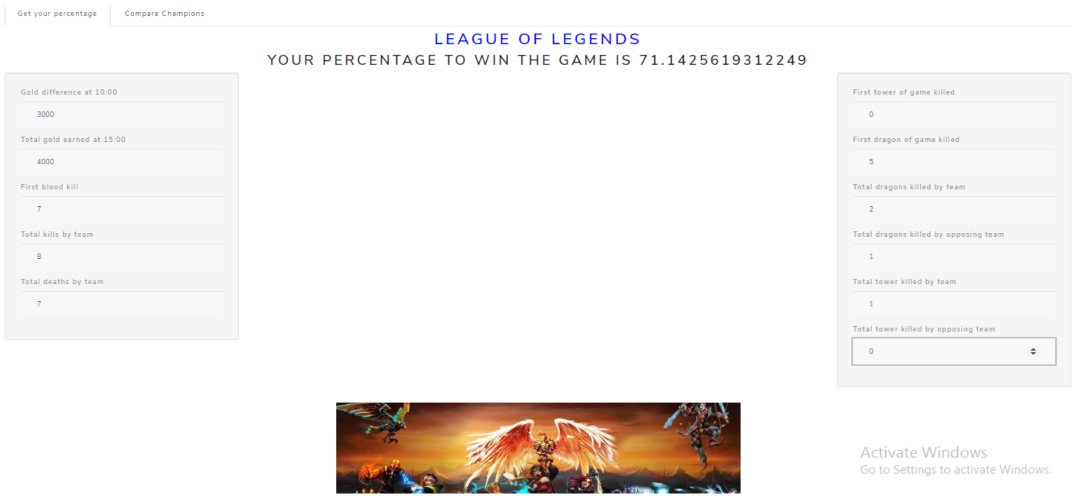

Fig 6.3 Snapshot for Case II

As shown in Fig 6.3, we see that the win percent changes to 71.14(approximately),which is less closer to winning the game . Which means further the damage can be controlled by this time too win this game.

We also concluded the various relations between the factors in our game:

We also tried to use the linear regression method to see the correlation between Gold earned and Damage made. As shown below, there is a strong linear correlation between Gold and Damage, which means that players make good use of the gold earned in the games, and convert the gold into damage to the enemies.

cor(data_lol$dmgshare, data_lol$earnedgoldshare)

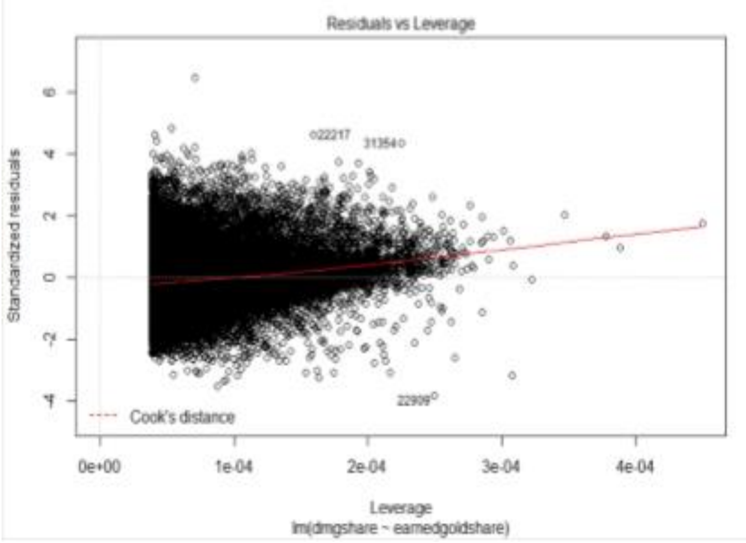

Fig 6.4 Leverage(dmgshare ~ earned goldshare) vs Standardized residuals

Fig 6.4 shows the leverage of earned gold share against the results which are standardized. Moreover, we wanted to see if the Damage To Champions had a strong correlation with the Kills. Thus,we used the linear regression method to see there relationship between Kills and DTC.As we expected, DTC did have a strong correlation with kills. As more damage made would result in more kills, the game seemed fair and encouraging for players.

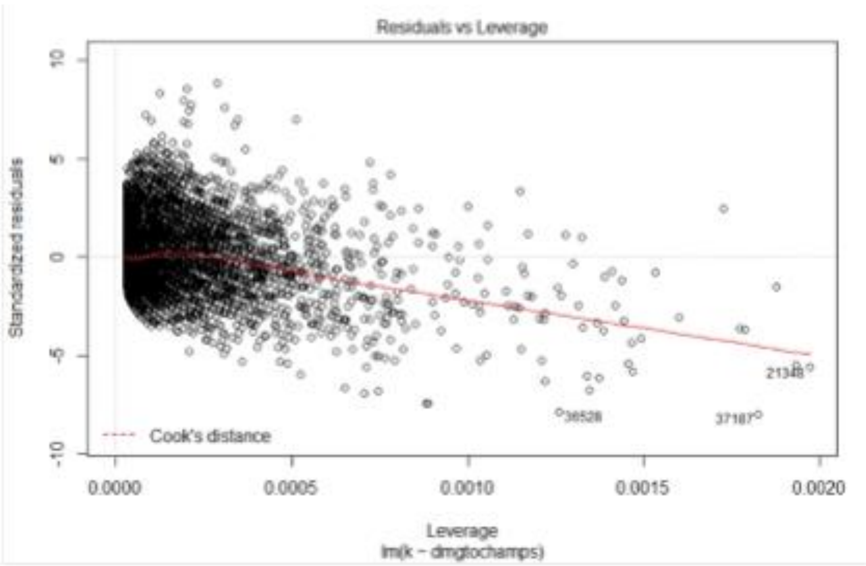

Fig 6.5 Leverage lm(k ~ dmgtochamps) vs Standardized residuals

Fig 6.5 reperesents a graph of leverage(Damage to champions) against standardized residuals. Additionally, we want to know if the game length is an important factor to win the game. As shown in the chart below, the win rate is apparently lower after 40 minutes, and when it was before 23 minutes, the win rate is zero. This gives us an idea that most of the games won have between 25 and 40 minutes.

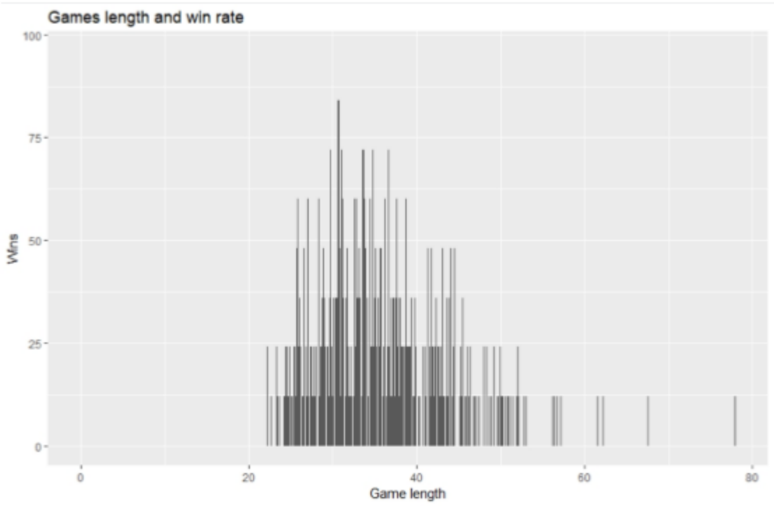

Fig 6.6 Game length vs Wins

Fig 6.6 is a graph that depicts the length of the games against the number of wins. This has been debated for numerous times through the years. We can see that the Blue - Red Team win percentage is steady around 55%-45% and some years a bit more, in favor of the Blue Team.

tapply(data_lol$result, data_lol$side, sum)/6

The average game length on a Blue Team Win is around 36,4 mins but when Red team wins the average game length is around 37,3 mins.

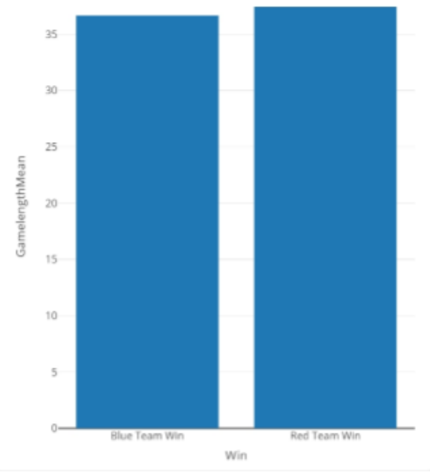

Fig 6.7 Win vs GamelengthMean

From Fig 6.7 we can conclude that when Red team wins, games tend to have bigger duration than when Blue Team wins.

We looked for the top 10 champions with the highest pick rate :

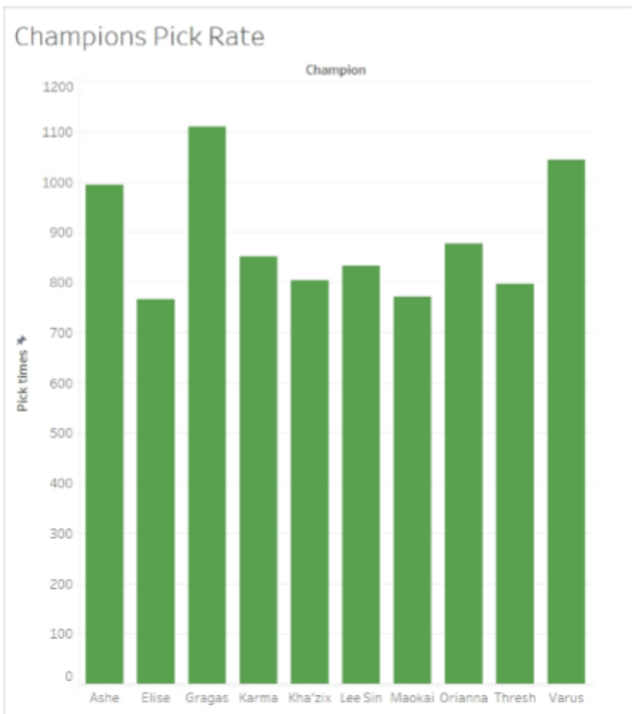

Fig. 6.8 Graph depicting Champions Pick Rate

Fig 6.8 is a representation of champion performance. Win rates typically are associated with the actual strength and performance of a champion. Pick and ban rates typically are associated with how popular a champion is or strong it is perceived to be. These statistics were essential for the final analysis as they represent a champion's "success".

CHAPTER 7

TESTING

For testing our web application, we are using three test cases for using various values for different parameters:

Case 1: To check the accuracy of the model, first we enter values of a team when they won.

Values entered are:-

i) Final gold difference at 10:00 = 114

ii) Total gold earned at 15:00 = 520

iii) First blood kill = 0

iv) Total kills by team = 3

v) Total deaths by team = 10

vi) First tower of game killed = 1

vii) First dragon of game killed = 2

viii) Total dragons killed by team = 4

ix) Total dragons killed by opposing team = 2

x) Total towers killed by team = 2

xi) Total towers killed by opposing team = 0

Percentage to win the game predicted by model in this case: 90.896%

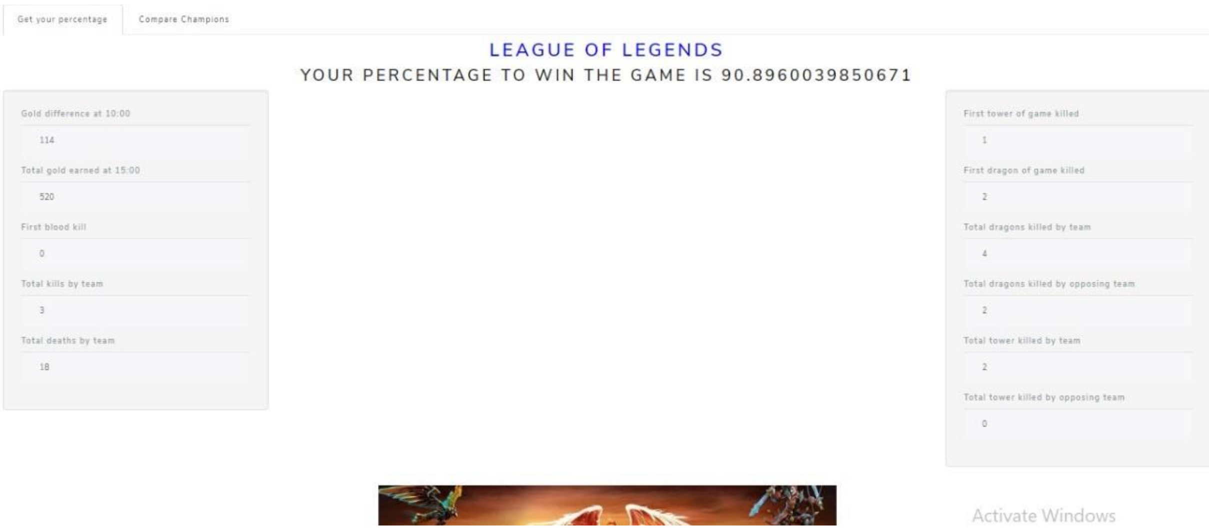

Fig. 7.1 Case 1 screenshot of application

Case 2: Many values are similar to the first case. The last two parameters are changed to check the effect on win percentage.

Values entered are:

i) Final gold difference at 10:00 = 114

ii) Total gold earned at 15:00 = 520

iii) First blood kill = 0

iv) Total kills by team = 3

v) Total deaths by team = 18

vi) First tower of game killed = 1

vii) First dragon of game killed = 2

viii) Total dragons killed by team = 4

ix) Total dragons killed by opposing team = 2

x) Total towers killed by team = 5

xi) Total towers killed by opposing team = 1

Percentage to win the game predicted by model in this case: 62.611%

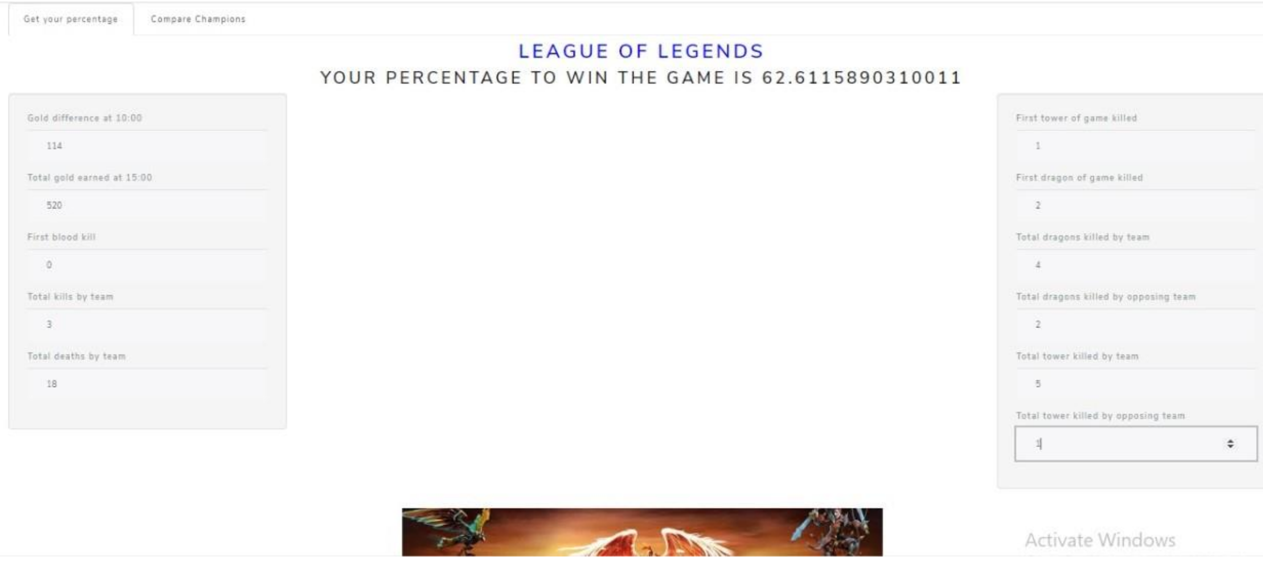

Fig 7.2 Case 2 screenshot of application

The win percentage which has reduced compared to Case 1 shows that even if one or two values are changed, there is a big change in the predicted win percentage.

Case 3: We enter various values to see in what scenarios and what values a team would lose.

Values entered are:

i) Final gold difference at 10:00 = 200

ii) Total gold earned at 15:00 = 1000

iii) First blood kill = 0

iv) Total kills by team = 3

v) Total deaths by team = 18

vi) First tower of game killed = 1

vii) First dragon of game killed = 1

viii) Total dragons killed by team = 4

ix) Total dragons killed by opposing team = 1

x) Total towers killed by team = 8

Total towers killed by opposing team = 1

Percentage to win the game predicted by model in this case: 14.187%

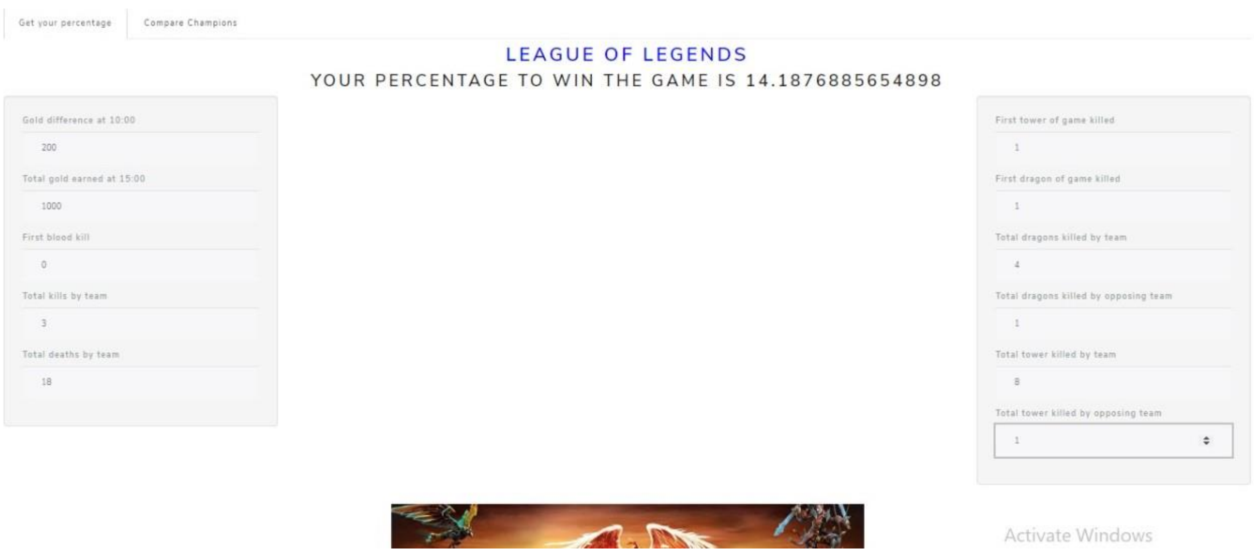

Fig 7.3 Case 3 screenshot of application

CHAPTER 8

CONCLUSION AND FUTURE SCOPE

8.1 CONCLUSION

We did a data project on League of Legends since we are fans of the game and its use of data. The project is focused on the question of whether or not we can create a model to predict winning teams based on performances. We use logistic regression models to predict the match outcomes. Features are extracted from the data that Riot Games API exposes– including champions picked for the game, player role information, and mastery levels for the players' champions (pre-game knowledge) as well as in-game player statistics. We fifind that using only prematch knowledge (champions, masteries, roles, spells) from the very start is only a weak predictor of match outcome but using in-game statistics, the model becomes a strong predictor.

We hope that we'll be able to look at more current data in the future, but this is a great demonstration of how data can power anyone from normal players, to Riot Games professional broadcasts, and even teams looking for ways to best analyze the strengths and weaknesses of enemy teams to make informed decisions. We can even use data analysis to improve matchmaking algorithms and game balance to the benefit of everyone that plays League of Legends around the world.

Despite there being many existing sites that mine data through the Riot API and release interesting, personalized statistics, none can effectively predict the winner team during a game. Our project takes on this task using different variables that are related to both teams stats in order to predict the winner. In the future, it would be interesting to develop a finer sense of how accurate the prediction system becomes given the amount of data at a certain point in a match. In other words, how much more accurate would the prediction becomes if data extracted from the first X minutes of each match is used as part of the feature vector as compared to the first Y minutes as well as how the accuracy changes with respect to X. We also have the data to be able to implement many more features if we desired. The Riot API also contains much more information which we ignore but could use for many purposes.

8.2 FUTURE SCOPE

In the future, it would be interesting to develop a finer sense of how accurate the prediction system becomes given the amount of data at a certain point in a match. In other words, how much more accurate would the prediction becomes if data extracted from the first X minutes of each match is used as part of the feature vector as compared to the first Y minutes as well as how the accuracy changes with respect to X. Furthermore, the overfitting in the pre-match data could potentially be dealt with by modifying the way the champion selection is used as a feature as it currently accounts for 1330 of the features in the prematch feature vector.

We also have the data to be able to implement many more features if we desired.The Riot API also contains much more information which we ignore but could use for many purposes. Moreover the results we are achieving can be more accurate if we made the discussions tier wise for the game. Some non-technical factors can also be taken in account while calculating the win rate of a player. One such example is enjoyability of the player.

We hope that we'll be able to look at more current data in the future, but this is a great demonstration of how data can power anyone from normal players, to Riot Games professional broadcasts, and even teams looking for ways to best analyze the strengths and weaknesses of enemy teams to make informed decisions. We can even use data analysis to improve matchmaking algorithms and game balance to the benefit of everyone that plays League of Legends around the world.

REFERENCES

[1] G. Palm. Warren mcculloch and walter pitts: A logical calculus of the ideas immanent in nervous activity. Brain Theory, page 229–230

[2] League of legends overview. URL https://na.leagueoflegends.com/en/.

[3] LoL Stats. Stats straight from the source. URL https://developer.riotgames.com/.

[4] RiotWatcher. Riotwatcher api. URL http://riot-watcher.readthedocs.io/en/latest/.

[5] Thomas Huang, David Kim, and Gregory Leung. League of legends win prediction. URL http: //thomasythuang.github.io/League-Predictor/abstract.pdf.