



eSPORTS PLAYER FINISH PLACEMENT PREDICTION USING MACHINE LEARNING

PRAJYOT TADAS

This dissertation was submitted in partial fulfilment of the requirements for the
degree of MSc Artificial Intelligence and Applications.

DEPARTMENT OF COMPUTER AND INFORMATION SCIENCES
UNIVERSITY OF STRATHCLYDE

AUGUST 2022

DECLARATION

This dissertation is submitted in part fulfilment of the requirements for the degree of MSc of the University of Strathclyde.

I declare that this dissertation embodies the results of my own work and that it has been composed by myself. Following normal academic conventions, I have made due acknowledgement to the work of others.

I declare that I have sought, and received, ethics approval via the Departmental Ethics Committee as appropriate to my research. – N/A

I give permission to the University of Strathclyde, Department of Computer and Information Sciences, to provide copies of the dissertation, at cost, to those who may in the future request a copy of the dissertation for private study or research.

I give permission to the University of Strathclyde, Department of Computer and Information Sciences, to place a copy of the dissertation in a publicly available archive. (please tick) Yes [☒] No[]

I declare that the word count for this dissertation (excluding title page, declaration, abstract, acknowledgements, table of contents, list of illustrations, references and appendices is 10,689 words.

I confirm that I wish this to be assessed as a Type 1 2 3 4 5 Dissertation (please circle)

Signature: 

Date: 15th August, 2022

ABSTRACT

The eSports sector has grown enormously in terms of both viewership and income throughout the years. The growing viewership was the key cause of revenue increase. As this sector is relatively new and attracts younger audiences, there is lot of scope for expansion. As the eSports sector has developed into the giant that it is today, the utilisation of data analytics has also expanded in importance. Because eSport games create a large amount of in-match and post-match data, this data may be utilised for organising tournaments, creating teams, streaming, marketing, and gambling.

The goal of this project is to study various supervised learning algorithms. The idea behind doing this is to apply these techniques for training models on a very large dataset obtained from Kaggle to make predictions. The dataset provided by Kaggle has more than 4 million training points and around 2 million testing points. As this dataset was made available as a Kaggle competition, we do not have the target variable on test dataset. Hence, for this project only the training part was used for training and validation.

The dataset consists of post-match stats of the players in a PUBG match with the player finishing position, 0 being eliminated first and 1 being winner of the match. The task was to build a machine learning model that predicts the final placement of the player. The project involves Exploratory Data Analysis (EDA) to understand and visualise the data, effects of outliers and multi-collinear variable on the model, and feature engineering.

The analysis is done by comparing different supervised machine learning models and the metrics used in this project are mean absolute error (MAE) and R2 score.

ACKNOWLEDGEMENTS

The author would like to thank his supervisor, Dr. Michael Cashmore, for all of his help and support from the beginning of this study until its conclusion.

I'd like to thank the CIS staff and the professors I had the pleasure of studying under for all the kind assistance they gave me throughout my time at the university.

I'd like to give a shout-out to my friends and family for bearing with me and providing moral support as I worked to finish this project.

TABLE OF CONTENTS

<i>DECLARATION</i>	2
<i>ABSTRACT</i>	3
<i>ACKNOWLEDGEMENTS</i>	4
<i>TABLE OF CONTENTS</i>	5
<i>LIST OF FIGURES</i>	7
<i>LIST OF TABLES</i>	9
<i>CHAPTER 1: INTRODUCTION</i>	10
1.1 BACKGROUND.....	11
1.2 AIM AND OBJECTIVES	12
<i>CHAPTER 2: BACKGROUND ANALYSIS</i>	13
2.1 PRIOR AND RELATED WORK.....	14
2.2 WHAT IS eSPORTS	15
2.3 MACHINE LEARNING IN eSPORTS	16
2.4 LITERATURE REVIEW	17
2.5 eSPORTS ETHICS.....	19
2.6 eSPORTS GAMBLING	20
2.7 PLAYERUNKNOWN'S BATTLEGROUNDS (PUBG).....	21
<i>CHAPTER 3: MACHINE LEARNING AND BIG DATA</i>	22
3.1 BIG DATA.....	23
3.2 MACHINE LEARNING	24
3.3 SUPERVISED VS UNSUPERVISED VS REINFORCEMENT LEARNING	25
3.4 SUPERVISED LEARNING MODELS	26
3.5 REGRESSION.....	27
3.6 SOFTWARE	28
<i>CHAPTER 4: IMPLEMENTATION AND ANALYSIS</i>	29
4.1 Kaggle.....	30
4.2 Dataset Description	31
4.3 Target Column	33
4.4 Exploratory Data Analysis (EDA)	34
4.5 Outliers and Multi-collinearity	37
4.6 Scaling and Normalising the data.....	41
4.7 Train Test Split	42
4.8 Cross Validation	43
4.9 Feature Engineering.....	44

4.10 Training and Testing Procedure	45
4.11 Model Comparison	48
<i>CHAPTER 5: CONCLUSION AND DISCUSSION</i>	<i>50</i>
Conclusion of Analysis.....	51
Discussion	52
Further Work	53
<i>APPENDIX</i>	<i>54</i>
<i>BIBLIOGRAPHY</i>	<i>58</i>

LIST OF FIGURES

Figure 1: using info() method on the dataset

Figure 2: Checking for missing values

Figure 3: Distribution of target column

Figure 4: Exploring object type features

Figure 5: Exploring int64 and float64 type features

Figure 6: Different match types

Figure 7: Players per match

Figure 8: Group formation

Figure 9: Count plot for 'kills'

Figure 10: Count plot for 'roadKills'

Figure 11: Plot for longest kill

Figure 12: Plot for weapons acquired

Figure 13: Plot for DBNOs

Figure 14: Plot for distance travelled

Figure 15: Correlation matrix

Figure 16: New features

Figure 17: Removing features showing collinearity

Figure 18: Removing outliers

Figure 19: Keras Sequential model

Figure 20: RandomisedSearchCV best parameters

Figure 21: Scatterplots for 'walkDistance', 'rideDistance' and 'swimDistance'

Figure 22: Scatterplots for 'kills', 'damageDealt' and 'DBNOs'

Figure 23: Scatterplot for 'heals' and 'boosts'

Figure 24: VIF for collinearity detection

Figure 25: Count plot after final modifications on 'matchType' column

Figure 26: Random Forest Regressor

Figure 27: Kaggle submission

LIST OF TABLES

Table 0: Dataset Description

Table 1: First analysis (Baseline)

Table 2: Second analysis (without collinear features)

Table 3: Third analysis (dropping outliers)

Table 4: Fourth analysis (feature engineering)

Table 5: Fifth analysis (cross-validation)

Table 6: Neural networks results

Table 7: Hyperparameter tuned results

CHAPTER 1: INTRODUCTION

1.1 BACKGROUND

eSports, which stands for "electronic sports," are tournaments that use video games. eSports, often known as competitive online gaming, typically take the shape of organised, multiplayer video game tournaments, with the participants typically consisting of professional gamers competing either alone or in teams. Even though structured contests were a part of video game culture for many years, until the late 2000s, most of the people who took part in those contests were not professional gamers but rather casual gamers. Simultaneously, there was an exponential rise in the number of spectators tuning in to these events online. Multiplayer online battle arena (MOBA), first-person shooter (FPS), fighting, card, battle royale, and real-time strategy (RTS) games are the most common types of video games that are played in eSports (Tassi, 2012).

Over the past few years, the number of people who watch others play competitive video games has grown a lot. eSports are popular among player and spectators, supporting a worldwide entertainment sector. eSports analytics focuses on cyber-athlete analysis, planning and forecasting to provide data-driven information (Hodge, *et al.*, 2017). The use of big data analytics is crucial for the future of eSports. Over the years, the eSports community has turned their hobby into a legitimate enterprise, one that has thrived in a nearly entirely digital world and generated enormous quantities of data. Collecting enormous volumes of structured data is no easy process, but it's well worth it in the eSports industry, which depends on publicity as much as the traditional sports industry.

eSports can make a lot of money thanks to big data. Since its introduction, big data analytics has had a significant influence on the eSports industry. In the ever-changing digital economy, highly scalable business models are made possible by data collecting, aggregation, and analytics. The value of the game market has gone up a lot in the last ten years, and it is expected that by 2023, the gaming industry will be worth more than \$200 billion. Mobile games are also expected to bring in the most money, while console games make up almost a third of the money made around the world (Judge, 2022).

It is clear that the need for data science projects in the esports industry will continue to rise. Massive data sets enable the development of powerful analytical tools for players, teams, and marketers, as well as AI-powered prediction algorithms that seem poised to alter the future of professional video gaming (Orejas, 2020).

1.2 AIM AND OBJECTIVES

The aim of this report is to solve a regression problem on an extremely large dataset by designing a machine learning model which accurately predicts the final positions of the players. The goal is to explore the dataset and use various data pre-processing, feature engineering and data wrangling techniques before feeding the data to various machine learning models, visualise the results, and compare different models and methods. The question being answered is which model works better on this type of dataset.

In the end, a good machine learning model would be evaluated based on a lower value of Mean Absolute Error (MAE) and higher value of R^2 score. The objective of this report includes a good exploration of the dataset by doing Exploratory Data Analysis (EDA), checking for multi-collinearity among the features and dealing with outliers which can cause problems to the models. The other goal was to look at some of the research that has been done in this area and see if there were any links to the conclusions of this report.

It is possible that underfitting will become an issue because of the enormous amount of data points in the dataset relative to the number of features. Hence, significant amount of feature engineering would be required. The challenge of running models on such a large dataset would be challenging. Additionally, the process of cross validation of the features becomes difficult.

The desired variable has a value between 0 and 1. Trying other settings and hyper parameter adjustment would result in much lower error terms, hence upscaling the target variable is essential.

CHAPTER 2: BACKGROUND ANALYSIS

2.1 PRIOR AND RELATED WORK

This dataset was a part of a Kaggle competition which was hosted nearly 4 years ago. The PUBG Developer API allowed Kaggle to obtain the data that was necessary for the competition. The competition was based on Kaggle kernels only and had 1772 competitors who made 12,747 entries. The goal of this competition was to develop a machine learning model that is evaluated on the metric of mean absolute error (MAE) between predicted target value and observed target value. As of today, the best MAE observed on the leader board is 0.01385. But the reason of this analysis is not to beat this score but to evaluate different models based on MAE and R2 score.

One research in particular grabbed my attention since it makes use of the same dataset. The statistic employed was RMSE and R2 score. The models utilised for comparison were Linear Regression, Decision Trees and SVM. The author utilised a different strategy where he separated the full training set into 5 smaller sets each comprising 6000 data points. The idea of upscaling the target variable came after reviewing the table of findings which revealed RMSE in the range of 0.1005 to 0.1194 throughout the whole analysis. Because of this, reading and comprehending the model's performance while adjusting the hyperparameters is a challenging task. The author's findings revealed that SVMs are better suited for this dataset. The author's choice of RMSE was problematic since it penalises errors more than MAE (Ghazali, et al., 2021).

2.2 WHAT IS eSPORTS

eSports are competitive video games. Different leagues or teams compete in games like Fortnite, League of Legends, Counter-Strike, Call of Duty, Overwatch, PUBG, and Madden NFL. Millions of fans watch these gamers at live events, on TV, and online. Streaming platforms like Twitch let users watch their favourite gamers play in real time, which is how successful players create fanbases (Willingham, 2018).

The rise in this sector is a critical moment for the introduction and development of new data-driven technologies and applications meant to enhance both the user experience and that of all organisations and individuals in the eSports sector. The use of data in this industry stands out most prominently in two areas: firstly, technologically, with the development of various applications and systems aimed at enhancing and improve the gaming experience, and secondly, economically, with the evolution of this new market being sought (Orejas, 2020).

Brands invest in eSports marketing directly and indirectly to reach a huge, interested audience. This has led to strong financial growth in the business, only hampered by COVID banning public eSports tournaments. Things look to be back to normal in 2022 (Geyser, 2022). According to stats and reports from Newzoo, in 2020, the esports market was expected to bring in more than \$1.1 billion worldwide. Even though COVID19 didn't change the growth rate, which stayed at about \$950.3 million, the predictions for the next few years still look good (Takahashi, 2020). The total worth of the international eSports industry was estimated at \$1.38 billion in 2022. It was also predicted that by 2025, the worldwide eSports market will generate as much as \$1.87 billion in revenue. Asia and North America are now the most lucrative regions for eSports, with China alone contributing about 20% of the total (Gough, 2022).

According to some interesting facts and stats by (Howarth, 2022)

- The total prize pool for Dota 2 in 2021 came close to \$50 million.
- Team Liquid, the highest-earning esports organisation, has earned in over \$40 million.
- The highest-paid player in the industry has won almost \$7 million.
- Prize money for Esports competitions including PUBG is projected to reach \$27.53 million by 2023. That's an increase of more than 300 percent in only five years.

When it comes to gaming, a well-defined analytics approach gives teams and individuals the ability to make choices based on hard evidence. A company's employees should always be aware of how their current efforts directly affect revenue generation. One area where big data is becoming more useful is in the gaming industry, where it is being used to monitor trends, solve issues, and enhance game play. For gaming businesses, the need of a solid plan for gathering and analysing client data is growing. The article by Indicative Team gives an example on how data analysts helped in solving a problem and users were retained.

2.3 MACHINE LEARNING IN eSPORTS

As the popularity of eSports continues to rise, so does the need for statistical analysis of individual and collective performances. Machine learning is a powerful tool that can be used for eSports analytics. Machine learning algorithms can be used to automatically extract features from data, build models to predict players or team performance, and identify patterns in data (Hodge, et al., 2017).

The application of machine learning in eSports analytics is exciting, but it is vital to keep in mind that machine learning is only a tool, not a perfect solution. Machine learning algorithms will only be good as the data that they are given. In order to get the most out of machine learning, it is important to have high quality data. But the data generated by eSports and video games is high dimensional and sparse. This means that traditional machine learning algorithms may not be well suited to eSports data. Instead, newer methods such as deep learning maybe more effective. Deep learning is type of machine learning that is well suited for high-dimensional data. Deep learning algorithms can automatically extract features from data and build complex models to make predictions (Judge, 2022).

A lot of obstacles must be overcome before machine learning can be used for eSports analytics. One challenge is lack of publicly available data. Most eSports data is proprietary and is not available to the public. This makes it difficult for researchers to obtain the data needed to train and test machine learning models. Another challenge is lack of standardisation in data that is available. This makes it difficult to compare results across different datasets and generate findings. Despite these challenges, machine learning is a promising tool for eSports analytics. With the rapid growth of competitive gaming industry, more data is becoming publicly available, which will help overcome some challenges associated with machine learning. As machine learning technology continues to evolve, it is likely that its role in eSports analytics will continue to grow (Cruz, 2015).

Some applications of machine learning and data analytics in eSports are:

- Monitor player and team performance
- Identify match-fixing and cheating while improving the fairness of the game
- Improve accuracy of predictions
- Improve quality of commentary with insights and visualisations
- Assess the popularity of teams
- Improve production value of events and assess the impact of sponsorship
- Improve the organisation of leagues
- Improve gambling

(Shivang, 2020)

2.4 LITERATURE REVIEW

According to the author, the fast rise of the eSports sector in recent years has garnered the interest of researchers, but relatively little study has been conducted on the growth of eSports literature. The author analysed more than 250 research published between 2010 and 2021. In his study, the author identified a lack of cooperation across disciplines, such as technology and management, and indicated that such cooperation would be crucial for the long-term growth of eSports research. The author also indicated that studies in sectors like as management, technology, and health might aid in comprehending an individual's behaviour and views towards eSports and people. The author concludes with a discussion of the negative aspects of eSports, as individuals, particularly adolescents, get very preoccupied and develop inappropriate behaviour such as gambling and mental disorders such as addictions (Weisheng, et al., 2021).

In the next study, the author showed a way to break up matches into parts called "encounters" that are defined by their location and time. The author did this analysis of DOTA2, which is the most popular game. By looking at data from 412 games, the author came up with an algorithm that includes some game features and a way to measure performance. He observed, "In DOTA2, encounter results may be anticipated based on the initial circumstances, and match results can be predicted using encounter results". The author included examples of several well-known tactics to demonstrate his point of view. For the author, resources in-game are critical to a team's chances of winning an encounter. Because game creators are always adding new features and reworking existing ones, it is critical that eSports analytics be flexible and adaptive. With a few simple tweaks in the algorithms' beginning states and metrics, this encounter-based strategy may be applied to any team competition, says the author (Mahlmann, et al., 2016).

eSports, like conventional sports, provide access to both real-time and historical data regarding each activity done in the virtual world. Because the information is hidden deep inside massive volumes of data that are invisible to the naked eye, it might be utilised to improve live broadcast summaries of events. The author provided a large-scale case study using Echo, a production tool. Echo detects exceptional player performance using current and historical data and instantly transforms relevant data points to audience-facing visualisations. The authors' study was based on data gathered by installing Echo in one of the DOTA2 tournaments, which resulted in over 40 hours of footage, live discussions, and audience response. The graphs and findings demonstrated that Echo had a measurable impact on the breadth of narrative, enhanced audience engagement, and elicited an emotional reaction from viewers. The audience's enthusiastic comments suggested that additional data-driven visuals might be useful in eSports tournaments and live streams (Hodge, et al., 2017).

Because the advancement of computers and the creation of the internet have resulted in significant changes in eSports, the author performed a thorough examination of the evolution of the gaming industry and the number of players in Asia, Europe, and America. The data utilised from 2017 to 2019 revealed that the bulk of the participants in this industry were from the United States and China, and these two nations led in gaming market revenue. The authors' analyses revealed a link between these variables and economic metrics such as GDP per capita and internet population. The author anticipated his projections for the following years using

Microsoft's Power BI application. Because of their large youthful population, China and South Asian nations will see an increase in players and income, with an expected game market value of \$77 billion by 2025 (Ruiz, et al., 2022).

The author of this research suggested doing away with human coaches while training eSport athletes. According to the author, AI and reinforcement learning can solve this problem since human training doesn't always provide the best approaches to victory. Reinforcement learning was used to create an AI based on the author's examination of the game Flappy Bird. Both AI and human performance data were used to compile the study's findings. The findings showed that the AI began delivering higher scores after increasing the training rate. It also demonstrated how to win in instances when human players were unable to do so. The author advised that AI anticipated outcomes be employed in MOBA and FPS games since it reveals a lot of undiscovered possibilities (Du, et al., 2021). Also found during the famous AlphaGO versus Lee Sedol game, when the AI made an unexpected move that led the 18-time world champion to forfeit the match.

During the COVID-19 epidemic, all competitions were suspended owing to physical distance, which hampered sports and gambling. During this time, Indonesian internet news portals began covering esports. This author utilised three online news sources and this study examined press coverage of the PUBG Mobile Pro League Indonesia Season 1 lockdown period. The author aims that this study will provide esports stakeholders fresh and generalised viewpoints. In a pandemic that impacts many aspects of life, the media typically portrays a distorted reality. Esports are unaffected by the COVID-19 pandemic as players no longer need to be physically present to play a game through ICT (information and communications technology), which is optimistic for the COVID-19 social distancing method. This study revealed the opinions and reporting systems utilised to keep the public engaged in esports (Marta, et al., 2021).

2.5 eSPORTS ETHICS

Ethics are important in any field, but they are especially important in eSports. With so much money and so much at stake, it is vital that everyone involved in eSports act ethically (Pereira, et al., 2012).

There are a few different aspects to eSports ethics. The first is cheating. Cheating is strictly prohibited in eSports and can result in a player being banned from play. Cheating includes any action that gives a player an unfair advantage over their opponents, such as using hacks or exploits. The second aspect of eSports ethics is match-fixing. Match-fixing is when a player or team deliberately loses a game in order to gain an advantage in another game. This is a serious offence and can lead to a lifetime ban from eSports. The third aspect of eSports ethics is doping. Doping is the practice of taking medications that improve athletic performance. This is also a serious offence and can lead to a ban from eSports. Finally, the fourth aspect of eSports ethics is harassment. Harassment is any type of behaviour that is intended to harm or intimidate another person. This can include but is not limited to hate speech, threats, and stalking. Harassment is not tolerated in eSports and can lead to a ban from play.

eSports ethics are important because they ensure that the competition is fair and that everyone is treated fairly. Cheating, match-fixing, doping, and harassment are all serious offences that can lead to a ban from eSports. By adhering to the eSports ethical code, players can ensure that they are competing on a level playing field and that they are treating their opponents with respect (Chee & Karhulahti, 2020).

2.6 eSPORTS GAMBLING

Since the turn of the century, eSports have gained popularity. In 2018, there were 2.2 billion eSports players globally, and this figure is likely to rise. Along with its popularity, eSports betting has grown. Like regular sports, esports betting is varied. Bet on the result of a contest, such a football game. There are various methods to gamble on esports, including as betting on individual player performances, tournament winners, and a player's kill count.

The eSports betting market is currently dominated by a small number of specialist bookmakers, however, there is also a growing number of traditional bookmakers who are offering eSports betting markets. In addition, a number of cryptocurrency-based bookmakers have also emerged in recent years, utilising the decentralised nature of blockchain technology to offer a more secure and transparent betting experience (Gainsbury & Blaszczynski, 2017).

Given the relatively young age of eSports industry, there is still a lack of regulation surrounding eSports betting. This has led to a number of issues, including the use of skin betting, which has been linked to problem gambling behaviour. However, as the industry matures, it is expected that more regulations will be introduced in order to protect consumers and ensure fair and transparent betting experience (Richard, et al., 2021).

Data science and analytics can be used to track and predict player performance, to understand which teams are likely to win or lose, and to identify emerging trends. This information can help gamblers place their bets more effectively and can also help bookmakers minimise their exposure to risk. Data analytics can also help identify problem and prevent them from placing bets they cannot afford to lose. By understanding patterns of behaviour, bookmakers can target interventions and support services to those who need them the most. In sum, data analytics can help improve the efficiency and accuracy of gambling on eSports and can also help protect vulnerable people from falling into problem gambling (Liu, et al., 2021).

2.7 PLAYERUNKNOWN'S BATTLEGROUNDS (PUBG)

Developer and publisher PUBG Studios is a sub-division of Krafton responsible for the battle royale video game previously known as PlayerUnknown's Battlegrounds. PUBG is a first-person shooter in which as many as one hundred players engage in a "battle royale," essentially a large-scale variation of the traditional "last man standing" deathmatch. Anyone may sign up for the match on their own, or with a team of up to three others. Winners are those who are the last members of their team or the last members of their team overall (Carter, 2017).

Since its release in 2017, PUBG has taken the world by storm, particularly in the world of eSports. The game has seen a meteoric rise in popularity with over 1.1 billion players (including PUBG mobile) worldwide as of March 2022. The popularity has inevitably translated into the competitive scene, with PUBG tournaments being held worldwide with huge price pool. The game has also been trailblazer in the world of eSports, with its innovative 'Battle Royale' format becoming hugely popular. This popularity has led to other games such as Fortnite and Apex Legends adopting similar formats. PUBG has also had a significant impact on the way that eSports are broadcasted. It is evident that PUBG has had a huge impact on the world of eSports and its popularity looks set to continue to grow in coming years (Apolinario, 2022).

The player's ability to make decisions and execute strategies in real time is essential to their success in the game. This generates a significant quantity of data both during and after the game, which aids in the decision-making process for analysts and coaches. The information that was gathered is also helpful in determining the teams' and opponents' relative strengths and weaknesses. PUBG has been a big contributor to data analytics. The data generated has also been used to analyse player behaviour, map design, game balance and matchmaking. The game has also been used to test new data analytics techniques and algorithms.

CHAPTER 3: MACHINE LEARNING AND BIG DATA

3.1 BIG DATA

The field of Big Data is growing exponentially, resulting in organizations across the globe to re-evaluate how they store, process and analyse data. In addition, data is becoming more complex and unstructured, making it difficult to glean insights using traditional methods. Big data has the potential to help organizations overcome these challenges and make better, faster decisions (Ishwarappaa, 2015).

According to a study, 90% of the world's data has been created in the last two years alone. This deluge of data is being generated by everything from social media and sensors to mobile devices and video cameras. The sheer volume of data is quickly becoming unmanageable for traditional data processing and storage systems. In addition to volume, big data is also characterized by its velocity, variety and veracity. Velocity refers to the speed at which data is generated and collected. For example, the data generated by social media platforms is near-instantaneous. Variety refers to the different types of data that are being collected, such as text, images, audio and video. Veracity refers to the accuracy and completeness of data (Oriani, et al., 2020).

Organizations are beginning to realize that big data can be a powerful tool for competitive advantage. By harnessing the power of big data, organizations can gain insights into their customers, their business processes and even the behaviour of their employees. For example, a retail organization can use big data to track the real-time purchase behaviour of its customers. This information can be used to improve inventory management, customer service and marketing efforts. A manufacturing organization can use big data to monitor the performance of its assembly line in real-time. This information can be used to improve quality control and prevent downtime. An eSports organisation can use big data to track popularity of specific games and in-game items. This information can be used by tournament organizers to determine which games to feature and by developers to gauge the interest in new content. Also, big data can be used to understand demographics of eSports audience, and this information can be valuable to sponsors and advertisers who want to reach the most engaged and enthusiastic gamers.

In the future, big data will become even more important as organizations strive to stay ahead of the competition. Those who are able to harness the power of big data will be able to make better, faster decisions and gain a competitive advantage.

3.2 MACHINE LEARNING

Machine learning is the field of computer science that deals with the development of algorithms that allow computers to learn from data and improve their performance at specific tasks. The term “Machine Learning” was coined in 1959 by Arthur Samuel, an American computer scientist who is widely considered to be the father of this field. Machine learning algorithms can be divided into three broad categories: supervised learning, unsupervised learning, and reinforcement learning (Geron, 2019).

When it comes to analysing data, machine learning is a technique that may automate the process of creating analytical models. It's a subfield of AI where the central aspect is that computers can figure out what to do on their own by sifting through data and noticing trends and patterns. Machine learning's goal is to develop algorithms that, given some data as input, can make more accurate predictions about a target output through repeated use. Machine learning is widely used in a number of industries today, from retail to healthcare. In retail, machine learning is used for things like product recommendations and fraud detection. In healthcare, machine learning is used for things like diagnosis and prognosis. The benefits of machine learning include increased accuracy, efficiency and speed. Machine learning can also be used to make predictions about things that are difficult or impossible for humans to do, such as weather patterns or stock market trends. The downside of machine learning is that it can be difficult to understand how the algorithms work. Also, if the data used to train the algorithm is not good, the predictions made by the algorithm will not be accurate (Langley & Simon, 1995).

The future of machine learning is very exciting. We are on the cusp of new era where machine learning will become increasingly important and central to our lives. There are many exciting and potentially game changing applications of machine learning that are being developed and explore.

3.3 SUPERVISED VS UNSUPERVISED VS REINFORCEMENT LEARNING

Machine learning algorithms may be categorised into three broad categories: supervised learning, unsupervised learning, and reinforcement learning. To teach an algorithm to correctly anticipate labels for new data, it is "trained" using a dataset that already contains those labels. Unsupervised learning is where the algorithm is not given any labels and must learn to find structure in the data itself. Reinforcement learning is where the algorithm is given a "reward" for making correct predictions, so that it can learn to optimize its predictions (Geron, 2019).

Supervised learning is the most common type of machine learning algorithm. It is used for tasks such as classification (predicting which category a new data point belongs to) and regression (predicting a continuous value). Supervised learning algorithms are "trained" on a dataset that includes the correct answers (labels). The algorithm learns to map the input data to the correct label (Geron, 2019).

Unsupervised learning is used for tasks such as clustering (grouping data points that are similar to each other) and dimensionality reduction (finding a smaller representation of the data that captures the important features). Unsupervised learning algorithms are not given any labels and must learn to find structure in the data itself (Geron, 2019).

Reinforcement learning is used for tasks such as playing a game or controlling a robot. Reinforcement learning algorithms are given a "reward" for making correct predictions. The algorithm learns to optimize its predictions to maximize the reward (Geron, 2019).

3.4 SUPERVISED LEARNING MODELS

Linear models like **Linear Regression** and **Logistic Regression** are a good choice for data that is linearly separable. When the dependent variable is continuous, linear regression is used, when the dependent variable is categorical, logistical regression is used. Linear models work by finding a line or plane that best separates the data into classes. The line or plane is then used to classify new data.

Decision trees are good for data that is not linearly separable, and can be used for both classification and regression tasks. Decision trees work by dividing the data into sections, and then making a decision about which class each section belongs to. Decision trees can be very accurate, but they can also be overfit to the training data.

Random forest is another supervised classification and regression approach. The "forest" refers to uncorrelated decision trees that are combined to minimise variation and improve data predictions. Random forests are a good choice for data that is not linearly separable and has a lot of features. Random forests work by creating a bunch of decision trees, and then averaging the results of all the trees. This averaging process helps to reduce overfitting.

Vladimir Vapnik created the **Support Vector Machine (SVM)** for data classification and regression. It's used to generate a hyperplane where the gap between two data classes is greatest for classification challenges. This hyperplane separates classes of data points on each side.

Neural networks handle training data by simulating brain connections via layers of nodes. Every node has inputs, weights, a bias, and an output. If the output value exceeds a threshold, the node "fires," transmitting data to the next layer in the network. Neural networks learn this mapping function via supervised learning, changing depending on the loss function. When the cost function is close to 0, we can trust the model to be accurate.

(IBM, 2020)

Using supervised learning, (Borg, 2021) analysed a classification issue of churn prediction and customer retention in the online gaming business. A number of machine learning algorithms, including Logistic Regression, Random Forests, Decision Trees, LGBM, and XGBoost, were compared using the ROC-AUC measure. The author adds some observations on the usefulness of these models, pointing out, for example, that linear models need less time to train but have lower accuracies than non-linear models like LGBM and XGBoost.

3.5 REGRESSION

A regression is a statistical technique that is used to predict a dependent variable based on one or more independent variables. The technique can be used to assess the strength of the relationship between the dependent and independent variables, and it can also be used to identify which independent variables are most important in predicting the dependent variable. Linear regression can be used to predict future values of the dependent variable, based on past values of the independent variable. For example, linear regression could be used to predict the future value of a stock price, based on past values of the stock price or predicting the final placement of an eSports player or game. Linear regression can also be used to determine the cause-and-effect relationship between the independent variable and the dependent variable. For example, linear regression could be used to determine the effect of advertising on sales (Beers, 2022).

Regression is a supervised learning technique, which means that you need to have training data in order to use it. The training data is used to build a model that can then be used to make predictions on new data. There are many different types of regression, but the most common is linear regression. Linear regression is used to find the line of best fit for a set of data points. The line of best fit is the line that minimizes the sum of the squared errors (SSE). The SSE is the sum of the squared differences between the predicted values and the actual values.

Linear regression can be used for both single-variate and multi-variate prediction. Single-variate linear regression is used to predict a single value, such as the future value of a stock market index. Multi-variate linear regression is used to predict multiple values, such as the future values of multiple stock market indices. There are many different ways to perform linear regression, but the most common is ordinary least squares (OLS). OLS is a method of solving for the line of best fit that minimizes the SSE (Beers, 2022).

3.6 SOFTWARE

Python is a versatile language that is widely used in many different fields, including web development, scientific computing, and artificial intelligence. In recent years, Python has become increasingly popular for machine learning due to its ease of use and rich set of libraries.

While there are other languages that are used for machine learning (such as R), Python is currently the most popular choice among practitioners. Python's popularity in machine learning is largely due to the fact that it is relatively easy to learn compared to other languages, such as Java or C++. Additionally, Python has a large and active community that contributes a variety of useful libraries for machine learning, such as the popular Scikit-learn library (pythonbasics.org, 2021).

Python's **Scikit-learn** is a free, open-source machine learning package. Support vector machines, random forests, gradient boosting, k-means, and DBSCAN are just some of the many classification, regression, and clustering algorithms present, and it's made to work with Python's NumPy and SciPy libraries (Pedregosa, et al., 2011).

Jupyter notebooks are a web-based application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text. **Google Colab** is a free Jupyter notebook environment that runs in the cloud and doesn't require installation. Both Jupyter notebooks and Google Colab can be used for data analysis, machine learning, and scientific computing. Jupyter notebooks are popular among data scientists because they allow you to combine code, visualizations, and narrative text in one document. Google Colab is popular among machine learning engineers because it allows you to use free GPUs for training your models (Singh, 2019).

CHAPTER 4: IMPLEMENTATION AND ANALYSIS

4.1 Kaggle

Kaggle is a platform for predictive modelling and machine learning competitions. In a Kaggle competition, participants are given a data set and a goal, and they compete to produce the best model for predicting the outcome. These competitions are usually well-defined and the data is clean and well-organised. The dataset used for this project was downloaded from Kaggle.

URL: <https://www.kaggle.com/competitions/pubg-finish-placement-prediction>

This dataset was released four years ago by Kaggle as part of the Kernels only competition. The data was acquired through the PUBG Developer API, according to Kaggle. The purpose of the competition was to assess models based on their MAE, with the lower the better. The model's goal is to create accurate predictions about the player's ultimate placing based on the in-game metrics provided. The scale used here ranges from 1 (first place) to 0 (last place).

Kaggle offers two datasets: a training dataset and a testing dataset. The training dataset has about 4 million rows, whereas the testing set contains over 2 million rows. Because the target variable is not included in the test set, a meaningful assessment of the model on the test set is impossible. As a result, just the training set would be utilised and split for training and validation.

I prepared a submission utilising Kaggle kernels and a Linear Regression model to evaluate the model's performance on the testing set. Because the dataset was so vast, the time it needed to run and evaluate the model was excessive. The unranked MAE was 0.1018 in the end. I tried many models, but they all ran out of CPU and RAM. Therefore, I solely utilised the training set to generate the visuals and findings for this report.

4.2 Dataset Description

There are 4,446,966 individual data points spread over 29 different categories in the dataset. The game's description claims that one hundred players may participate in each match (matchId). Each player is part of a group (groupId), and at the conclusion of the game, each team's final placement in the standings is determined by how many other teams are still in the running. Players may do anything from drive cars, swim, run, shoot, and even die if they fall too far or are driven over in the game, including pick up and use a variety of weapons, revive knocked-out comrades, and more. So, the target is 'winPlacePerc', and the other 28 are features.

```
# information about dataframe
train_data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4446966 entries, 0 to 4446965
Data columns (total 29 columns):
 #   Column              Dtype
---  -
 0   id                   object
 1   groupId              object
 2   matchId              object
 3   assists              int64
 4   boosts              int64
 5   damageDealt          float64
 6   DBNOs                int64
 7   headshotKills        int64
 8   heals                int64
 9   killPlace            int64
10  killPoints            int64
11  kills                 int64
12  killStreaks          int64
13  longestKill           float64
14  matchDuration         int64
15  matchType             object
16  maxPlace              int64
17  numGroups             int64
18  rankPoints            int64
19  revives               int64
20  rideDistance          float64
21  roadKills             int64
22  swimDistance          float64
23  teamKills             int64
24  vehicleDestroys       int64
25  walkDistance          float64
26  weaponsAcquired       int64
27  winPoints             int64
28  winPlacePerc          float64
dtypes: float64(6), int64(19), object(4)
memory usage: 983.3+ MB

There are 29 columns in this dataframe. 6 columns are of float64 type, 19 columns are of int64 type and 4 columns are of object type.
```

Figure 1: using info() method on the dataset

A run through the info() function on the dataset revealed that, out of the total of 29 columns, 6 were of float64 type, 19 were of int64 type, and 4 were of object type.

S.No	Feature	Type	Description
1	id	object	player-specific ID
2	groupId	object	ID specific to a team (duo or squad)
3	matchId	object	ID specific to a match (approx. 100 players)
4	assists	int64	Helping team to secure a kill
5	boosts	int64	Total boost items used
6	damageDealt	float64	Total damage given to enemies
7	DBNOs	int64	Number of enemy players knocked but not killed
8	headshotKills	int64	Total enemies killed by shooting on head
9	heals	int64	Total heal items used.
10	killPlace	int64	Match kills ranking.
11	killPoints	int64	Kills-based player ranking.
12	kills	int64	Total enemies killed.
13	killStreaks	int64	Total enemies killed in short time period.
14	longestKill	float64	Death distance between players.

15	matchDuration	int64	Seconds played.
16	matchType	object	Type of match played.
17	maxPlace	int64	Match data maximum placement.
18	numGroups	int64	Match statistics for how many groups.
19	rankPoints	int64	Ranking of players.
20	revives	int64	Total times player revived a teammate after getting knocked.
21	rideDistance	float64	Vehicles' meter-long distance.
22	roadKills	int64	Vehicle-related kills.
23	swimDistance	float64	Swimming meter-long distance.
24	teamKills	int64	Player's teammate kills.
25	vehicleDestroys	int64	Total vehicle destructions.
26	walkDistance	float64	Meters walked.
27	weaponsAcquired	int64	Weapons found.
28	winPoints	int64	Win-based player rating.
29	winPlacePerc	float64	Prediction's goal, 1 is 1st position, 0 is last.

Table 0: Dataset Description

There is a brief explanation of the feature's data kinds and its purpose in the table above. We can exclude three object type variables (id, groupId, and matchId) since they make no difference to the accuracy of our predictions. The 'matchType' object, the last object type, is particularly intriguing since it details the sort of match that was actually played. One-hot encoding and other pre-processing methods may be employed to test whether or not this factor has a role in the final positioning.

```
# checking for missing values
train_data.isna().sum()

id          0
groupId     0
matchId     0
revives     0
boosts      0
damageDealt 0
time        0
headshotKills 0
kills       0
killPlace   0
killPoints  0
kills       0
killStreaks 0
longestKill 0
matchDuration 0
matchType   0
maxPlace    0
numGroups   0
rankPoints  0
revives     0
rideDistance 0
roadKills   0
swimDistance 0
teamKills   0
vehicleDestroys 0
walkDistance 0
weaponsAcquired 0
winPoints   0
winPlacePerc 1
dtype: int64
```

There is 1 missing value in the target variable winPlacePerc. As the dataset is very large, dropping this missing value will not affect it overall.

Figure 2: Checking for missing values

Only one value in the target column was found to be missing throughout the whole dataset. The dataset is big enough that omitting a single missing value has little to no effect on the reliability of the model's predictions. The median or mean may be used as a substitute for a missing number.

4.3 Target Column

'winPlacePerc' is the dataset's target column. The values in this column range from 0 to 1. The original coding and models were designed with this range in mind, but since the output MAE and R2 scores were so fractional, it was exceedingly challenging to experiment with hyperparameter tuning, feature engineering, and model comparisons. The figure below displays the distribution of the target column.

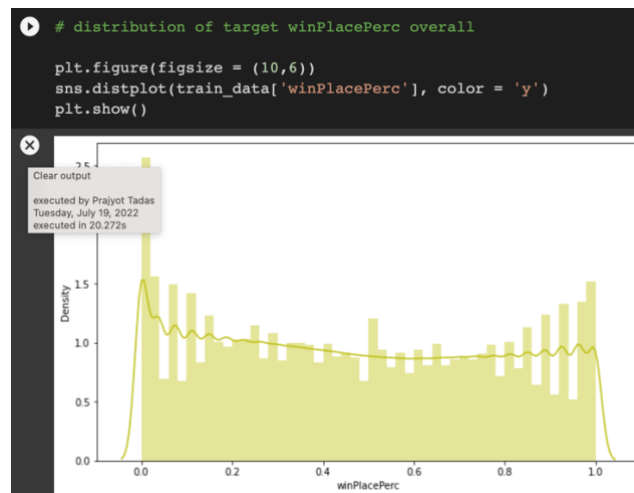
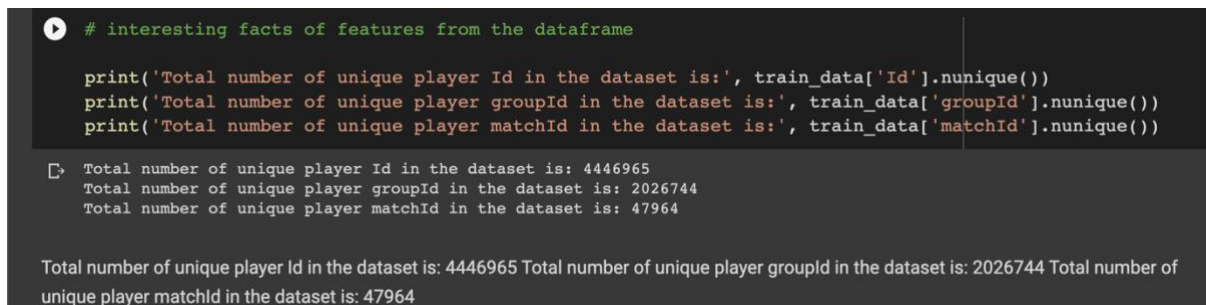


Figure 3: Distribution of target column

The target variable was thus upscaled in order to better understand the models and analyse the impacts of hyperparameter tuning and feature engineering. Each entry was scaled up by a factor of 100. This translates to 0 meaning first eliminated and 100 indicating last eliminated, i.e., the match winner. Because of the significant difference in errors caused by this upscaling, it was possible to identify best parameters for training the models. It also assisted in identifying the features that had the greatest influence on the model's performance.

4.4 Exploratory Data Analysis (EDA)

Exploratory data analysis is an approach to data analysis that is used to summarize data, identify patterns and relationships, and test hypotheses. This approach is often used to understand data sets that are new or unfamiliar, and can be used to find insights that may not be apparent from other methods.



```
# interesting facts of features from the dataframe

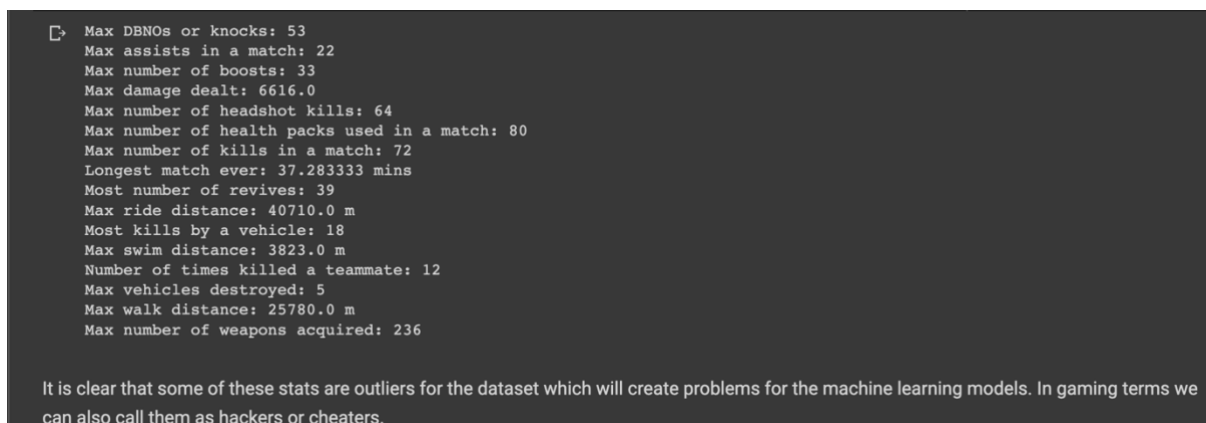
print('Total number of unique player Id in the dataset is:', train_data['Id'].nunique())
print('Total number of unique player groupId in the dataset is:', train_data['groupId'].nunique())
print('Total number of unique player matchId in the dataset is:', train_data['matchId'].nunique())

In [ ]: Total number of unique player Id in the dataset is: 4446965
        Total number of unique player groupId in the dataset is: 2026744
        Total number of unique player matchId in the dataset is: 47964

Total number of unique player Id in the dataset is: 4446965 Total number of unique player groupId in the dataset is: 2026744 Total number of unique player matchId in the dataset is: 47964
```

Figure 4: Exploring object type features

Despite having no bearing on the model's predictions, these three object type features provide useful insight into the dataset. Since each player has a unique player id, we may infer that the data comes from 4,446,965 individual players. There were 2,026,744 unique teams that played in 47,964 matches.



```
In [ ]: Max DBNOs or knocks: 53
        Max assists in a match: 22
        Max number of boosts: 33
        Max damage dealt: 6616.0
        Max number of headshot kills: 64
        Max number of health packs used in a match: 80
        Max number of kills in a match: 72
        Longest match ever: 37.283333 mins
        Most number of revives: 39
        Max ride distance: 40710.0 m
        Most kills by a vehicle: 18
        Max swim distance: 3823.0 m
        Number of times killed a teammate: 12
        Max vehicles destroyed: 5
        Max walk distance: 25780.0 m
        Max number of weapons acquired: 236

It is clear that some of these stats are outliers for the dataset which will create problems for the machine learning models. In gaming terms we can also call them as hackers or cheaters.
```

Figure 5: Exploring int64 and float64 type features

It's unusual to have such stellar numerical metrics. For instance, it would be fantastic to get 72 kills in a match, but it won't happen in every game. Cheaters and hackers may be exposed via analysis of this kind. The ability to take measures against such people to establish a level playing field might be a huge boon to game developers. The aforementioned figure suggests that the dataset has several outliers. A machine learning model's performance may be drastically impacted by outliers. Overfitting the data due to outliers might make a model unreliable when applied to fresh data. A model may potentially converge to a less-than-ideal answer if it encounters outliers.

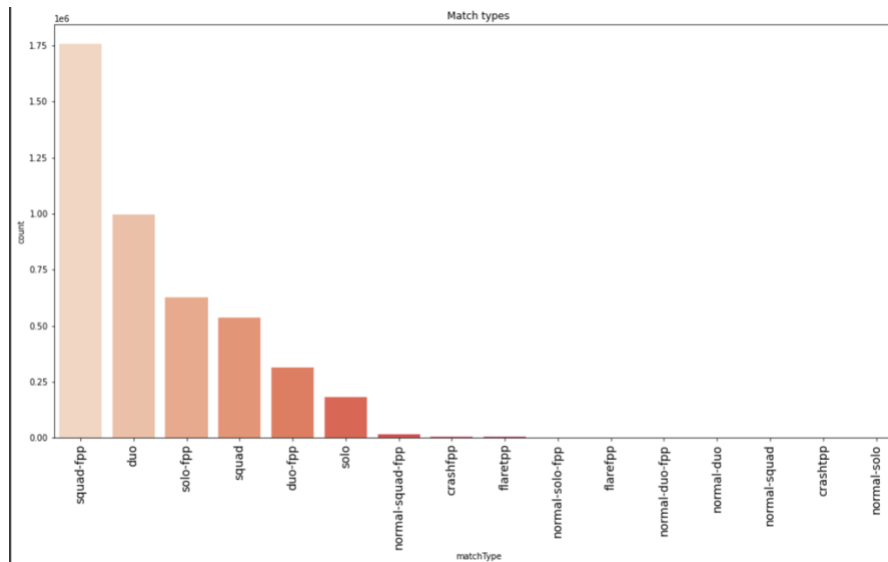


Figure 6: Different match types

It is evident from the above count plot that most of the players prefer playing 'squad-fpp', 'duo', 'solo-fpp', 'squad', 'duo-fpp' and solo matches.

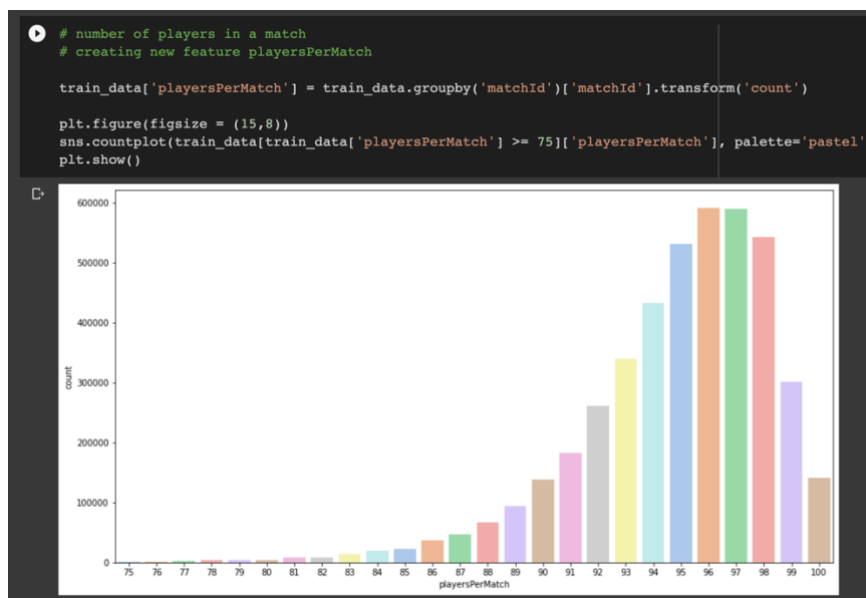


Figure 7: Players per match

The optimal number of participants in a single match is 100, as stated in the description. The above count plot, however, shows that some players do exit the match before it begins. There are typically around 94 people present at any one game.

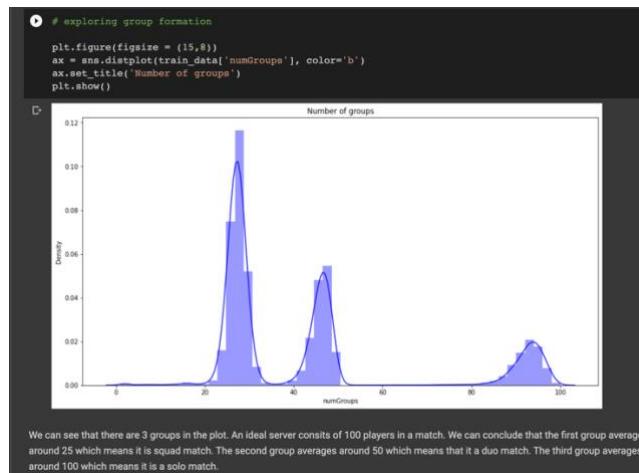


Figure 8: Group formation

The accompanying distribution shows that the data may be roughly divided into three categories. There is a roughly 25 averages in the first group, a roughly 50 averages in the second, and a roughly 100 averages in the third. The three distinct brackets represent squad, duo, and solo games.

Additional EDA and visualisations are provided in the appendices of this report.

4.5 Outliers and Multi-collinearity

As discussed in the previous section, outliers can be a problem to the machine learning model. Let's look at some anomalies in the features.

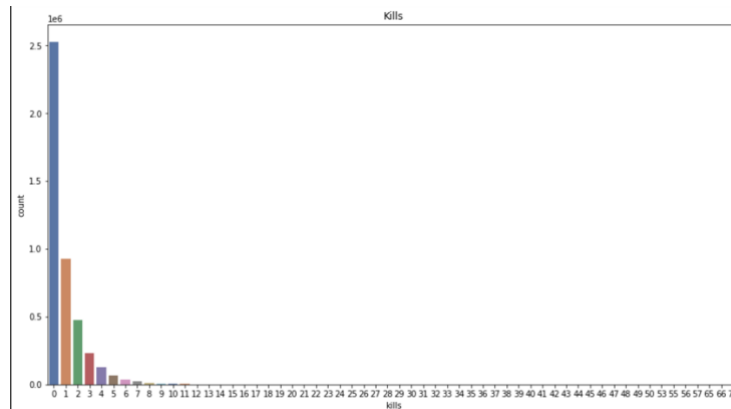


Figure 9: Count plot for 'kills'

The accompanying graph shows that most players die without scoring a single kill. When the kill totals were tallied, it was discovered that only a small percentage of players achieved over 20 kills. It suggests that players with more than 20 kills are an extreme case and are outliers for this dataset.

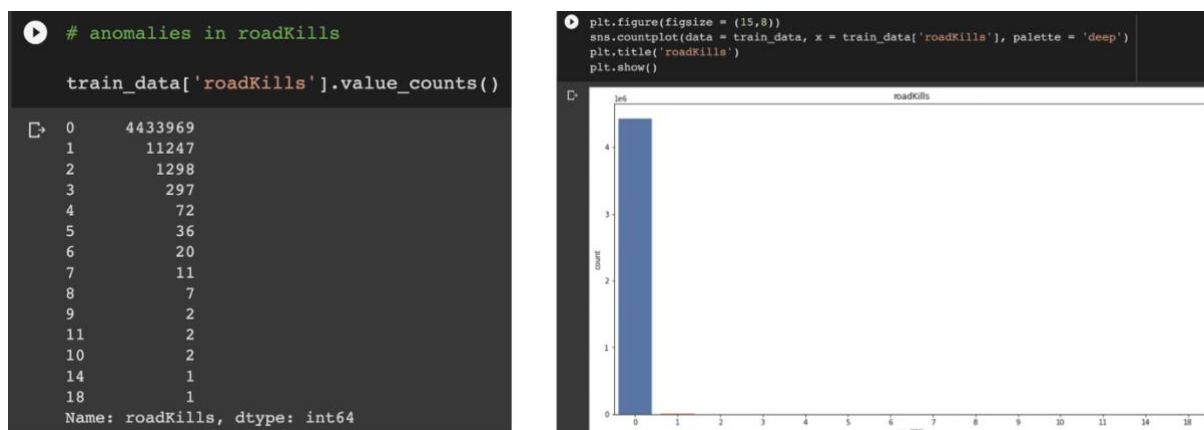


Figure 10: Count plot for 'roadKills'

The enemy's precise location must be known before attempting a roadkill, making it a challenging task. This explains why so many players were unable to get a roadkill. Six players have more than ten road kills, which may suggest they have access to hacks that reveal their opponents' whereabouts. This is one another way to spot players that are trying to win unfairly.

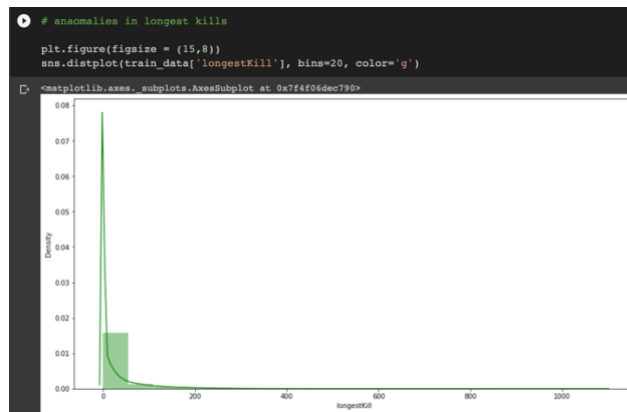


Figure 11: Plot for longest kill

According to the graph above, most of the killings take place between 0 and 200 metres, indicating a short to medium range. Sniper activities may result in kills at distances of more than 200 metres. Further investigation revealed that 261 players had kills that surpass 800 metres. 20 of the 261 players managed to tally kills beyond 1000 metres. This study leads us to believe that these gamers are using a 'aimbot' hack.

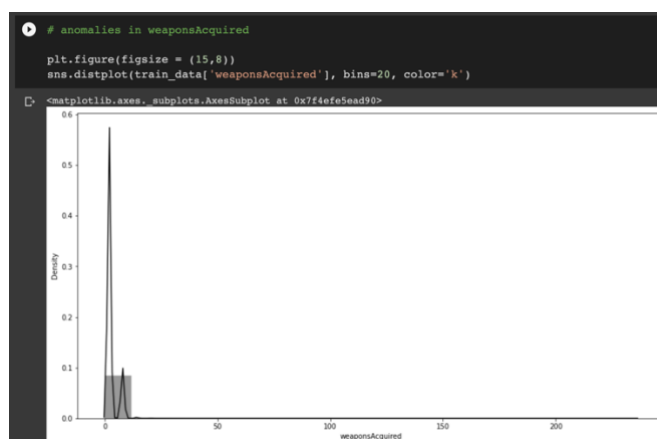


Figure 12: Plot for weapons acquired

According to the above plot, the majority of gamers employ 0 to 20 weapons. There is no purpose in getting additional weaponry since each player has their own choice of weapons. Further investigation revealed that 670 players obtained more than 35 weapons in a single match. The largest number of weapons obtained by a single player was 236. This dataset's outliers may be identified as such.

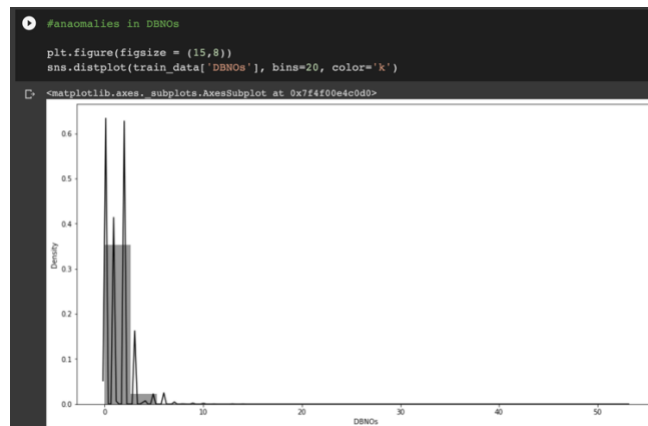


Figure 13: Plot for DBNOs

Down but not out (DBNO) is shorthand for "knockout." Those who have been knocked out of the game may be reanimated. According to the graph, on average a player could knock 0-10 opponents. In addition, statistics showed that 557 players averaged more than 15 knocks each game. It is possible to classify these players as outliers in this dataset.

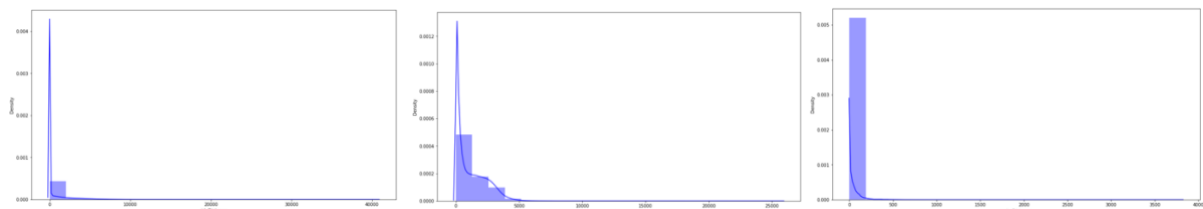


Figure 14: Plot for distance travelled

While the vast majority of participants did not travel more than five thousand metres, there were 1077 who walked over eight thousand. While the vast majority of riders covered less than 10 kilometres, 498 accomplished the feat of covering more than 15 kilometres. While the vast majority of swimmers cover less than 500 metres, there are 138 who cover more over 1,000. It makes little sense for a regular player to go to such great lengths.

Multi-collinearity

Multi-collinearity is a statistical phenomenon in which two or more predictor variables in a regression model are highly correlated, meaning that one can be linearly predicted from the others with a substantial degree of accuracy. This can create problems in interpreting the results of the model, because each predictor variable can be thought of as a "shadow" of the others.

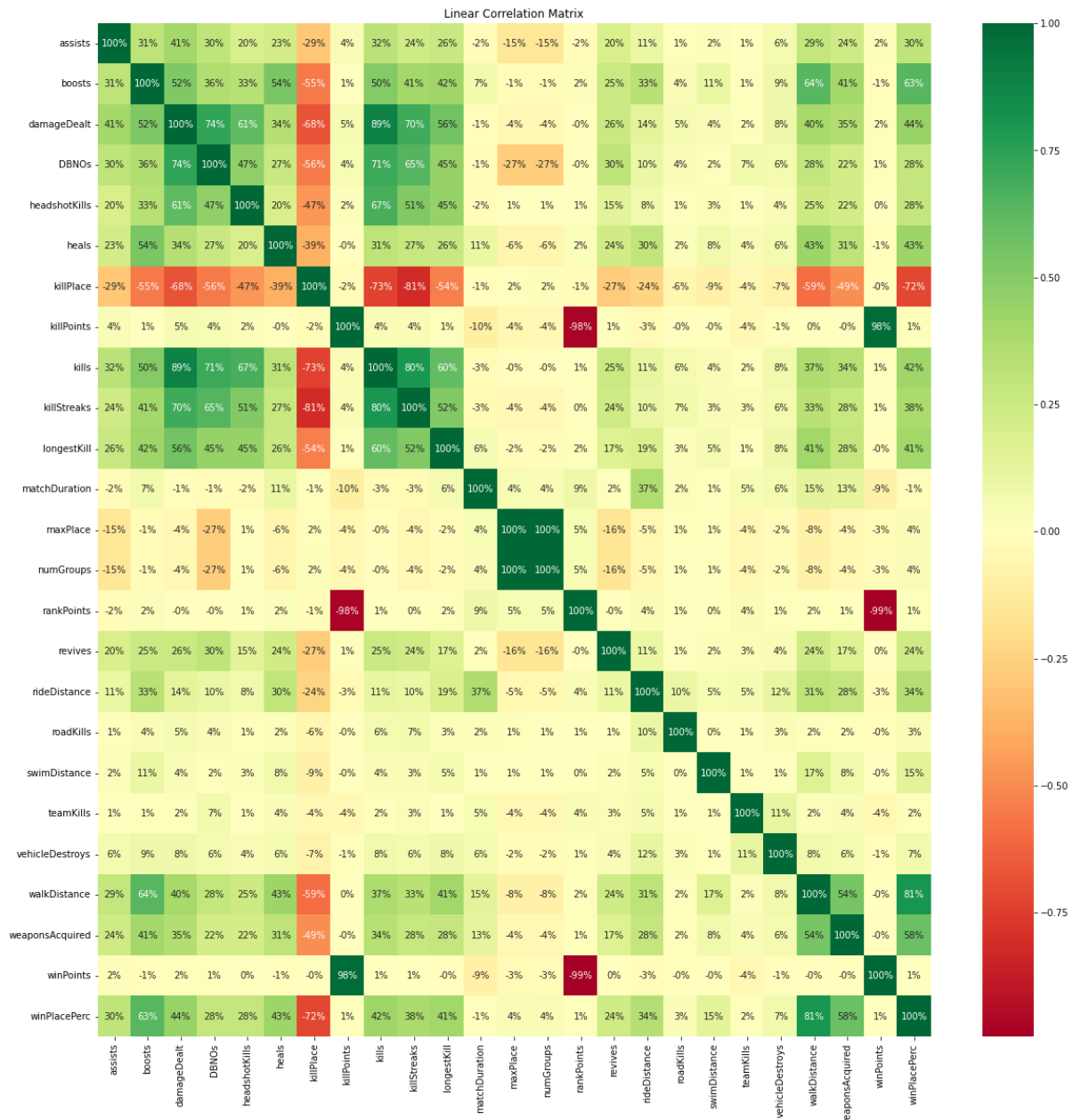


Figure 15: Correlation matrix

From the correlation matrix above, we may determine that several characteristics exhibit collinearity. Anything more than 80 percent or less than -80 percent is connected. We see a 98 percent positive correlation between 'winPoints' and 'killPoints', a -99 percent negative correlation between 'winPoints' and 'rankPoints', and a perfect 100 percent correlation between 'maxPlace' and 'numGroups'.

Another method called variable inflation factor was also used to check for multi-collinearity. In a multiple regression, the **variable inflation factor (VIF)** is a way to measure how much the predictor variables are alike. It is found by dividing the variance of all the betas in a given model by the variance of a single beta if it were fit on its own. $1/(N-p)$, where N is the number of observations and p is the number of predictor variables, gives the variance of beta.

4.6 Scaling and Normalising the data

There are a few reasons for normalizing and scaling your data before running machine learning models. Firstly, it can help improve the performance of the model by making the training process more efficient. Secondly, it can help prevent overfitting by providing a more consistent input to the model. Finally, it can make it easier to compare results across different models and data sets.

In this report, Standard Scaler pre-processing method was used. To normalise features, the `sklearn.preprocessing` package provides a class called **Standard Scaler**, which implements the Transformer API by zeroing off the mean and scaling to unit variance.

4.7 Train Test Split

Sklearn train test split is a way to divide data into sets that are used for training and sets that are used for testing. To do this, a random subset of the data is chosen for training, and the rest of the data is used for testing. This method can be used to measure how well machine learning models work.

The test size in this report was set at 0.2, which means that 20% of the data was allotted for testing purposes. The remaining 80% of the data was used to train the models. This avoids overfitting on training data. For the sake of model simplicity, the random state was set to 15 for each model, implying that the model would be assessed on the same testing data each time. The model would provide different results each time for the same split of data if no random state was set.

4.8 Cross Validation

Cross validation is a method for evaluating the effectiveness of machine learning models. This is accomplished by dividing the data into a training set and a test set, training the model on the training set, and then evaluating its performance on the test set. This procedure is done several times, and the model's ultimate score is based on its average performance.

In this report, predictions were made and compared with 5 splits. The **KFold cross-validation** process in the sklearn programming language is used to divide a dataset into k divisions (k folds), and then repeatedly train and test the model on each fold. There are k repetitions of the whole process, such that each division is tested once. The model's final score is the average of the scores from each of the k folds in which it has been tested.

4.9 Feature Engineering

The technique of using domain expertise in order to extract features from data, which may then be utilised to develop machine learning models, is referred to as feature engineering. In this report 8 new features were created from the existing features.

```
## feature engineering
train2['totalDistance'] = train2['rideDistance'] + train2['swimDistance'] + train2['walkDistance']
train2['healthItems'] = train2['heals'] + train2['boosts']
train2['teamWork'] = train2['assists'] + train2['revives']
train2['headshotRate'] = train2['headshotKills'] / train2['kills']
train2['playersJoined'] = train2.groupby('matchId')['matchId'].transform('count')
train2['playersPerTeam'] = train2.groupby('groupId').groupId.transform('count')

train2['totalTeamDamage'] = train2.groupby('groupId').damageDealt.transform('sum')
train2['totalTeamKills'] = train2.groupby('groupId').kills.transform('sum')
```

Figure 16: New features

The 'totalDistance' feature was made by adding up all the distances for ride, walk, and swim. The number of heals and boosts used by the player were added together to make the 'healthItems' feature. 'teamWork' was made by adding a player's assists and revives together. 'headshotRate' is the number of headshot kills divided by the total number of kills. 'playersJoined' is a feature that shows how many people are in a match. The number of players in a group is shown by the 'playersPerTeam' feature. The 'totalTeamDamage' and 'totalTeamKills' features are group statistics.

4.10 Training and Testing Procedure

The models used in this report were Linear Regression, Decision Trees, LGBM and XGBoost. The metrics to evaluate the model performance was mean absolute error (MAE) and R2 score. Measurement of how accurate the model's predictions are is known as the **Mean Absolute Error (MAE)**. The MAE is determined by taking the average of all the predictions and dividing it by the number of absolute errors. Models are judged on the **R2 score**, or coefficient of determination, which is a measure of how well they account for the variation in a set of measurements. How closely the data match the fitted line is a statistical metric. The better the model matches the data, the higher the R2 score will be. So, the main goal of the model is to get the MAE as low as possible while keeping the R2 score high.

Initially, each model was evaluated by dividing the data into an 80/20 split between training and testing. For the purpose of making predictions, the model parameters were left unchanged, and the training data was scaled. The predicted outcomes are listed below.

Model	Linear Regression	Decision Tree	XGBoost	LGBM
Test MAE	9.201	11.879	6.936	5.994
Train MAE	9.193	11.870	6.926	5.985
R2 score	0.831	0.749	0.899	0.926

Table 1: First analysis (Baseline)

After deleting features having multi-collinearity from the data, the following analysis was performed. The characteristics that were deleted were determined by the correlation matrix heatmap and the VIF. Six characteristics with significant collinearity were removed. These are 'killStreaks,' 'killPoints,' 'killPlace,' 'rankPoints,' 'maxPoints,' and 'winPlace'. Some of the model parameters were changed like max_depth and learning_rate.

```
train1 = train.copy()
train1 = train1.dropna()
train1 = train1.drop(['killStreaks', 'killPoints', 'killPlace', 'rankPoints', 'maxPoints', 'winPlace'], axis=1)
train1.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 417794 entries, 0 to 417793
Data columns (total 20 columns):
#  Column  Dtype
---  --
0  assists  int8
1  boosts  int8
2  damageDealt  float32
3  damage  int8
4  headshotKills  int8
5  kills  int8
6  kills  int8
7  longestKill  float32
8  matchDuration  int16
9  matchType  int64
10  numFrags  int8
11  revives  int8
12  sideWinLoss  float32
13  teamKills  int8
14  teamWinLoss  float32
15  teamKills  int8
16  vehicleDestructs  int8
17  winLoss  float32
18  weaponsAcquired  int16
19  winPlace  float32
dtypes: float32(6), int16(2), int64(1), int8(11)
memory usage: 232.7 MB
```

Figure 17: Removing features showing collinearity

The predicted outcomes are listed below.

Model	Linear Regression	Decision Tree	XGBoost	LGBM
Test MAE	11.533	11.895	9.234	9.099
Train MAE	11.527	3.297	9.231	9.089
R2 score	0.752	0.698	0.824	0.828

Table 2: Second analysis (without collinear features)

Data that has been cleaned of outliers was used for the following step in the study. Outliers have a significant influence on the models, hence 13,042 items were omitted from the datasets. Players with more than 20 kills, players with more than six road kills, players with more than 800 metre kills, players who walked more than 8000 metres, players who rode more than 15,000 metres, players who swam more than 1000 metres, and players with more than 15 DBNOs were included in the data that was dropped. Some of the models' parameters were tweaked to get better results.

```
[ ] train5 = train.copy()

train5.drop(train5[train5['kills'] > 20].index, inplace=True)
train5.drop(train5[train5['roadKills'] > 6].index, inplace=True)
train5.drop(train5[train5['longestKill'] > 800].index, inplace=True)
train5.drop(train5[train5['walkDistance'] > 8000].index, inplace=True)
train5.drop(train5[train5['rideDistance'] > 15000].index, inplace=True)
train5.drop(train5[train5['swimDistance'] > 1000].index, inplace=True)
train5.drop(train5[train5['weaponsAcquired'] > 35].index, inplace=True)
train5.drop(train5[train5['DBNOs'] > 15].index, inplace=True)
```

Figure 18: Removing outliers

The predicted outcomes are listed below.

Model	Linear Regression	Decision Tree	XGBoost	LGBM
Test MAE	9.166	6.154	6.229	5.858
Train MAE	9.160	5.889	6.219	5.831
R2 score	0.833	0.919	0.920	0.929

Table 3: Third analysis (dropping outliers)

The following study was conducted on newly developed features. Section 4.8 and Figure 15 illustrate the newly developed and analysed features. The findings of this analysis are shown in the table below. Some of the models' parameters were changed for better fitting.

Model	Linear Regression	Decision Tree	XGBoost	LGBM
Test MAE	9.970	6.371	6.156	5.784
Train MAE	9.974	6.315	6.155	5.766
R2 score	0.814	0.913	0.923	0.931

Table 4: Fourth analysis (feature engineering)

The next analysis was done with cross validation. KFold cross validation was used with 5 splits and 10 splits. The cross validation was performed on the data which has its outliers removed and having new features. The findings of this analysis with 5 splits are shown in the table below.

Model	Linear Regression	Decision Tree	XGBoost	LGBM
MAE	9.973	6.840	6.214	5.787
R2 score	0.814	0.900	0.921	0.931

Table 5: Fifth analysis (cross-validation)

The sixth analysis was done by creating a very basic neural networks with three layers. The keras Sequential model was trained for 25 epochs.

```
Model: "sequential_14"
```

Layer (type)	Output Shape	Param #
dense_43 (Dense)	(None, 128)	3072
dense_44 (Dense)	(None, 64)	8256
dense_45 (Dense)	(None, 1)	65

```

Total params: 11,393
Trainable params: 11,393
Non-trainable params: 0

```

Figure 19: Keras Sequential model

As the dataset is very huge it takes several hours to train a single model. The MAE obtained on the test set was 5.475.

Another analysis was made by using MLPRegressor class from sklearn's neural_networks. The activation function used was 'relu' and max_iter was set to 100. Again, this was a very basic model. The results are shown in the table below.

MAE test	5.896
MAE train	5.903
R2 score	0.926

Table 6: Neural networks results

Until now, the model's parameters were updated manually, and the model's parameters were determined by trial and error to get a desired outcome. However, since there are hundreds of settings and factors, this method might take a long time. The final analysis was carried out by performing cross validation on the LGBM model using RandomisedSearchCV. RandomisedSearchCV is a hyperparameter tuning approach that is used to discover the best combination of hyperparameters while building a model. Hyperparameters are settings that may be changed to improve the performance of a machine learning model. The results are shown in the table below.

MAE test	5.542
MAE train	5.517
R2 score	0.936

Table 7: Hyperparameter tuned results

4.11 Model Comparison

The baseline models outperformed the models that had collinear features removed. The models were not overfitting since the training and testing errors were almost identical, and the R2 score demonstrated that it predicts the new data extremely well. LGBM was the best model for the baseline. Other models, such as Random Forests and SVM, were examined as well, but there was a considerable difference in training and testing errors, thus those models were not chosen.

The second analysis, in which collinear features were deleted, revealed to be worse than the baseline since the errors were substantial. When experimenting with Decision Tree parameters, a remarkable observation was found. The model performed well until max depth reached 12, when a significant difference in errors was noted. In this analysis, LGBM performed better than others with MAE of 9.099 on test data with R2 score of 0.828.

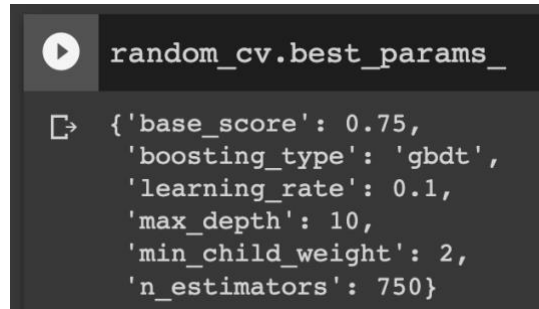
As seen in Tables 1 and 2, the baseline performed better; the third analysis was performed by eliminating just the outliers while preserving all of the features. With smaller MAEs, Linear Regression, XGBoost, and LGBM fared marginally better. The MAE of Decision Trees decreased significantly, falling from 11.879 to 6.154. When the learning rate was adjusted to 0.5, XGBoost performed better. When learning rate was set to 0.2, LGBM produced the best overall performance.

With feature engineering, new features helped improve the performance of XGBoost and LGBM, while Linear Regression and Decision Trees' performance went down. Decision Tree was a good fit until max depth was 15. After that, the MAE and R2 scores dropped by a fair bit. With learning rates of 0.5 and 0.6, respectively, XGBoost and LGBM both got the best results. Overall, LGBM gave the best performance.

For the next test, a simple KFold cross validation was used. The results were looked at using both 5 splits and 10 splits, but neither showed a significant difference. The results of the simple KFold cross validation were similar to those of analysis 3 (outliers removed) and analysis 4 (feature engineering). Stratified KFold cross validation was also tried, but the MAE and R2 scores weren't much better than KFold cross validation.

Neural networks came next in the analysis as it went on. A particularly effective method for tackling regression problems is the neural network. The keras API, which is built upon tensorflow, was used to create a neural network. This neural network employed the keras sequential model and has three layers. The neural network only had 25 training epochs, yet it still managed to get an MAE of 5.475 on the test set. Due to the size of the dataset, training for merely 25 epochs took a long time. The model would have produced a superior MAE if it had been trained on even more epochs. The same holds true for the hidden layers and activation functions, since training takes a long period. Using the neural networks from the Sklearn library, another experiment was conducted. It also yielded results that were comparable to those of the Keras neural networks, demonstrating its enormous potential for handling regression challenges on big datasets.

All of these analysis revealed that neural networks and LGBM are effective models for this dataset. Thus, the LGBM model was used in the final experiment to test hyperparameter selection and tuning. The pre-processing of the model was comparable to that of earlier testing. Outliers were eliminated, scaling was carried out using the Standard Scaler, the 'matchType' variable was transformed using the Label Encoder, and new features were generated. To get the optimum hyperparameters, RandomisedSearchCV was employed. The figure below displays RandomisedSearchCV results.

A screenshot of a Jupyter Notebook cell. The cell's title is 'random_cv.best_params_'. The output is a dictionary of hyperparameters for the LGBM model. The parameters and their values are: 'base_score' (0.75), 'boosting_type' ('gbdt'), 'learning_rate' (0.1), 'max_depth' (10), 'min_child_weight' (2), and 'n_estimators' (750).

```
random_cv.best_params_  
[>] {'base_score': 0.75,  
     'boosting_type': 'gbdt',  
     'learning_rate': 0.1,  
     'max_depth': 10,  
     'min_child_weight': 2,  
     'n_estimators': 750}
```

Figure 20: RandomisedSearchCV best parameters

These hyperparameters for the LGBM model were used to get the outcomes shown in Table 7. After hyperparameter tuning, it was noted that the MAE produced had the lowest value and the highest R2 score.

CHAPTER 5: CONCLUSION AND DISCUSSION

Conclusion of Analysis

Throughout the course of this research, a wide range of machine learning techniques and models were put through their tests. As was noted previously, the method that performs the best with this dataset is the one in which had outliers removed, and feature dimensions are extended by adding extra features. Additionally, LGBM is the most effective algorithm for huge datasets because to its speed and efficiency.

After doing preliminary research and examining the results of several models on this dataset, it became clear that there are non-linearities in the data. The MAE is somewhat high when a standard Linear regression model is applied to the data, but it drops dramatically when a non-linear model is employed, such as Decision Trees and LGBM. Due to the limited number of features available, feature engineering is crucial. Attempts to reduce the number of features by discovering correlations between them were made, however this method proved unsuccessful. Grouping and enhancing features in the dataset were shown to be useful in reducing the dataset size.

What I found most unexpected was that a simple linear regression model including all the features still produced a respectable R^2 score, suggesting that underfitting may be common. Because of this, we needed to use non-linear models and trim the dataset to cope with it. Simple KFold cross validation produced identical results to the data which had all the features for every model.

Neural networks and hyperparameter tuning displayed their efficiency by producing the most desirable results even without having fewer layers and fewer parameters tweaked. In the conclusion, despite the fact that this research is centred on PUBG, the method may be applied to any multiplayer game by making use of features that are compatible.

In regression analysis, a good fit to the data is indicated by a low MAE and a high R^2 score. Data scientists aim to have the errors as low as possible when dealing with a regression problem.

Discussion

From Chapter 2, it's clear that data science, machine learning, and AI have a lot to offer the world of eSports. A lot of people are spending time and energy to get a better understanding of the best eSports games, like (Mahlmann, et al., 2016) and (Hodge, et al., 2017) did for DOTA2. eSports is a new market with a lot of potential that hasn't been fully explored yet. Even though the COVID-19 pandemic caused big problems in every industry, eSports was still very profitable, according to (Marta, et al., 2021) and other sources. (Shivang, 2020) and (Judge, 2022) have talked about how eSports can be used in many ways and using machine learning techniques will only help it grow. As (Pereira, et al., 2012) and (Chee & Karhulahti, 2020) point out, this is a relatively new industry, so there are a lot of problems that aren't regulated yet. Data analytics can help deal with these problems.

The gambling industry is the one that can make the most money. As the analysis in this report shows, even a simple model could predict the results of matches with less error. According to (Gainsbury & Blaszczyński, 2017) and (Liu, et al., 2021), data analytics has the potential to improve and regulate this sector. The study by (Sweeney, et al., 2019) about the structure of the eSports gambling market can be combined with the study by (Macey, et al., 2020) about eSports gambling and demographic factors. Combining the results and using data analytics can help this industry grow exponentially.

Regression models with fewer mistakes may contribute to corporate growth in a few key ways. Regression models may improve firms' resource allocation choices by offering more precise forecasts. Regression models may aid in the development of niche marketing and product strategies by revealing the interplay between a number of previously unknown variables. Regression models may improve corporate operations by eliminating guessing from decision-making (Teeboom, 2019).

The growth of mobile eSports is another area that looks good. Game makers promoted their games by making games that could be played on mobile devices. After 2017, popular mobile games like PUBG mobile, Call of Duty mobile, and Fortnite mobile came out. This industry grew very quickly because these games didn't need expensive equipment and could be played anywhere. Since mobile esports are just getting started, there are a lot of kinks to work out before the business can really take off. The absence of standardised mobile devices, cross-platform play between mobile and console/PC games, and specialised mobile esports venues are just a few of the obstacles in the way of mobile gaming's full potential. Even with these obstacles, the mobile esports market is predicted to expand even more in the years to come. It's just a matter of time until mobile esports become a dominant force in the gaming business, what with the proliferation of smartphones and tablets.

To sum up, I believe that Big Data, Data Science, and Machine Learning can have a good impact on the eSports sector. Finding and punishing cheaters, match fixers, and harassers in eSports is a step toward a more ethical scene. Increased regularisation, improved chances, and protection for at-risk populations are all ways this may benefit the gambling industry. It may aid teams and coaches in practising more effectively and coming up with superior game plans. The future of the eSports sector with data playing a significant role is exciting for many reasons, and this has a lot to offer.

Further Work

As with any endeavour, there is room for improvement and development here. One way to expand this project is to test out a number of different Neural Network models and play about with the parameters until you find the one that produces the best results, to increase the quantity of work spent on feature engineering in order to provide additional functionality to the system. This research is centred on PUBG, the method may be applied to any multiplayer game by making use of features that are compatible.

As was previously said, this strategy is equally applicable to mobile eSports predictions. Each month, a plethora of mobile eSports competitions are held. The mobile version of PUBG commands the vast bulk of the market. Due to a lack of prior research and publications, mobile eSports data analytics have promising future.

APPENDIX

EDA

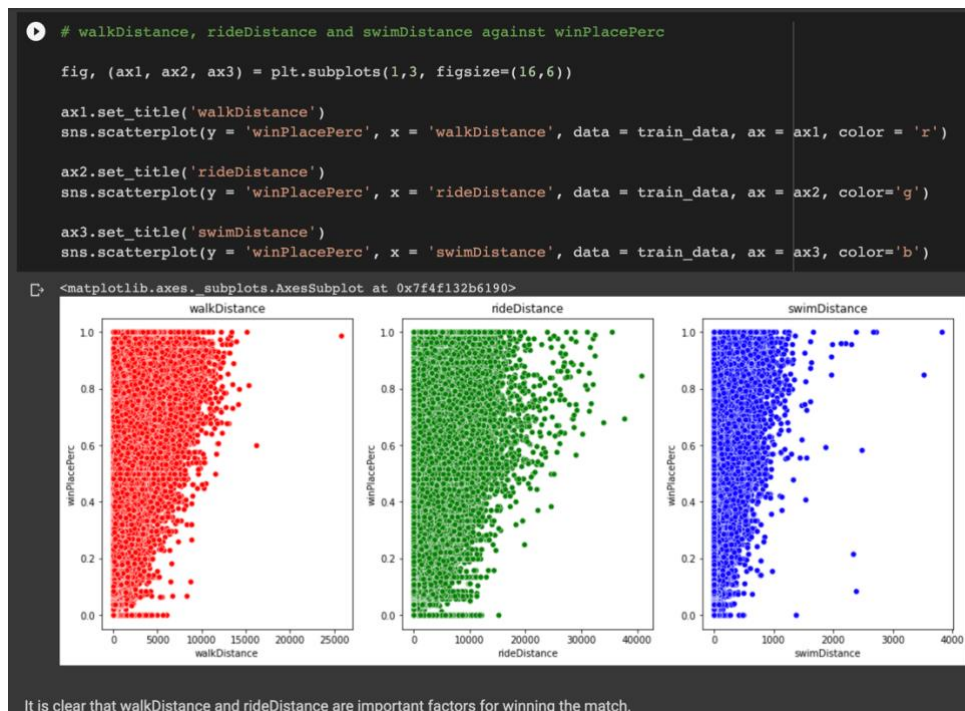


Figure 21: Scatterplots for walkDistance, rideDistance and swimDistance

It would seem that one's odds of success improve in direct proportion to the amount of time spent travelling by foot and car. When playing, it is true that players must constantly be on the move to take advantage of cover and avoid taking damage in the playzone.

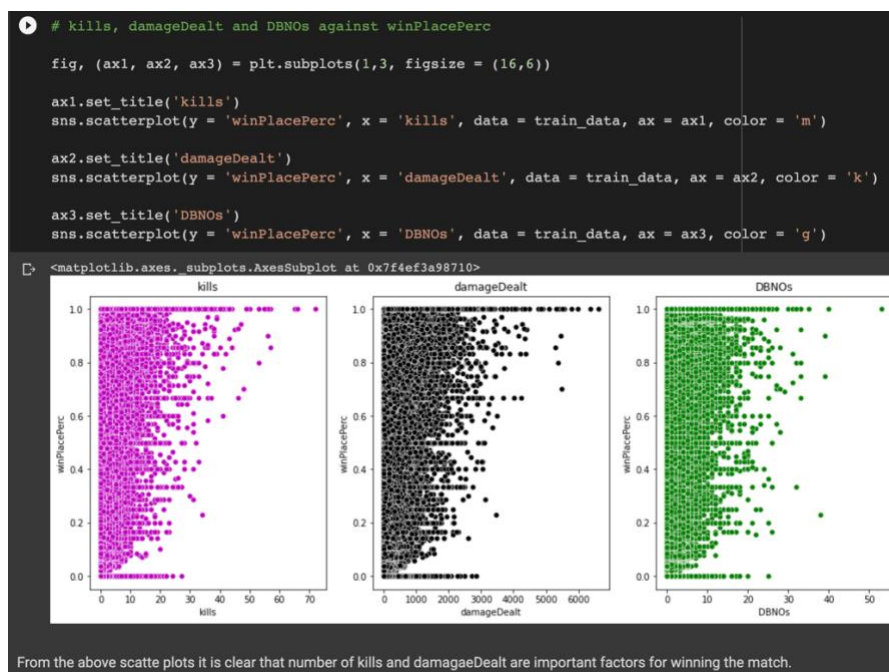


Figure 22: Scatterplots for kills, damageDealt and DBNOs

The scatterplots make it obvious that increasing your kill count, damage output, and opponent knockout rate will improve your odds of winning. As the game advances, you'll need to prove your shooting skills if you want to claim the victory.



Figure 23: Scatterplot for heals and boosts

From the scatterplot, it's clear that you need more heals and boosts to win. From a gaming perspective, we can say that as the game goes on, you will face other players, get into gunfights, and get hurt. So, if you want your chances of winning to be as high as possible, you will need to heal and keep yourself strong.

	variables	VIF
0	assists	1.500051
1	boosts	3.343797
2	damageDealt	9.726983
3	DBNOs	3.777864
4	headshotKills	2.128548
5	heals	1.885656
6	killPlace	31.426528
7	killPoints	53.407424
8	kills	10.968927
9	killStreaks	8.997456
10	longestKill	2.078803
11	matchDuration	47.572565
12	maxPlace	1143.186563
13	numGroups	1122.287475
14	rankPoints	100.003129
15	revives	1.315702
16	rideDistance	inf
17	roadKills	1.022025
18	swimDistance	inf
19	teamKills	1.047684
20	vehicleDestroys	1.041792
21	walkDistance	inf
22	weaponsAcquired	5.342024
23	winPoints	123.753501
24	winPlacePerc	20.665557
25	playersPerMatch	180.679843
26	totalDistance	inf

Figure 24: VIF for collinearity detection

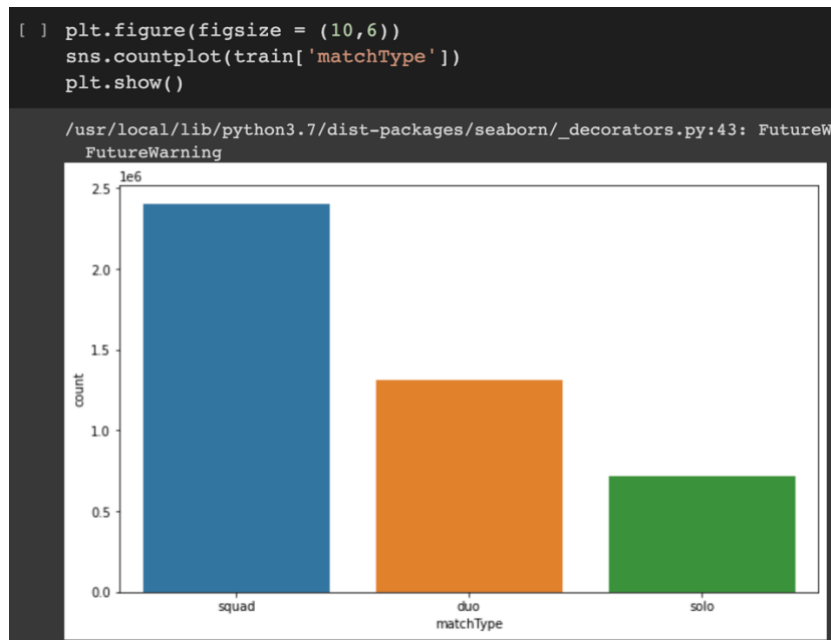


Figure 25: Count plot after final modifications on 'matchType' column

The 'matchType' feature has a lot of information. There were 16 types of matches, but each one was either a team match, a duo match, or a solo match. So, the whole column was changed into groups of three, two, and one. After that, Label Encoder was used to transform the whole column.

Machine Learning Models

```
[ ] ### Random forest

from sklearn.ensemble import RandomForestRegressor

model_rf = RandomForestRegressor(n_estimators=80, min_samples_leaf=3, max_features='sqrt', n_jobs=-1)
model_rf.fit(x_train, y_train)
y_pred_RF = model_rf.predict(x_test)

mse_RF = mean_squared_error(y_test, y_pred_RF)
mae_RF = mean_absolute_error(y_test, y_pred_RF)

print('Mean squared error using Random Forest: ', round(mse_RF,4))
print('Mean absolute error Using Random Forest: ', round(mae_RF, 4))

Mean squared error using Random Forest:  0.0072
Mean absolute error Using Random Forest:  0.0604
```

Figure 26: Random Forest Regressor

The image above shows why the target variable should be scaled up. We can see that both the MAE and the RMSE are very small numbers, which made it hard to figure out which hyperparameter was right for any model. Random Forest Regressor was also used to analyse data, but it took a lot longer to train than other models, so it wasn't used.

Submission and Description	Private Score	Public Score	Use for Final Score
CS958_sub2 Version 1 (version 1/1) 20 days ago by PrajyotTadas Notebook CS958_sub2 Version 1	0.10181	0.10181	<input type="checkbox"/>

Figure 27: Kaggle submission

I made a submission using Kaggle's kernel. The model used was a simple Linear Regression and the MAE for that model is shown in figure above.

BIBLIOGRAPHY

Shivang, 2020. *Data Analytics in ESports – Future Prospects – Jobs – Everything You Should Know*. [Online]

Available at: <https://www.scaleyourapp.com/data-analytics-in-e-sports-future-prospects-jobs-everything-you-should-know/>

Education, I. C., 2020. *Supervised Learning*. [Online]

Available at: <https://www.ibm.com/cloud/learn/supervised-learning>

Tassi, P., 2012. 2012: The Year of eSports. *Forbes*.

Hodge, V. et al., 2017. *Win Prediction in Esports: Mixed-Rank Match Prediction in Multi-player Online Battle Arena Games*, s.l.: s.n.

Judge, J., 2022. The Importance Of Data Analytics And Machine Learning In Gaming.

Orejas, M., 2020. *Medium*. [Online]

Available at: <https://medium.com/bedrockdbd/data-in-esports-47dd3b7040d>

[Accessed 21 July 2022].

Willingham, A., 2018. What is eSports? A look at an explosive billion-dollar industry.

Takahashi, D., 2020. *Newzoo: Global esports will top \$1 billion in 2020, with China as the top market*. [Online]

Available at: <https://venturebeat.com/2020/02/25/newzoo-global-esports-will-top-1-billion-in-2020-with-china-as-the-top-market/>

Geyser, W., 2022. *The Incredible Growth of eSports [+ eSports Statistics]*. [Online]

Available at: <https://influencermarketinghub.com/esports-stats/#toc-0>

Gough, C., 2022. *eSports market revenue worldwide from 2019 to 2025*. [Online]

Available at: <https://www.statista.com/statistics/490522/global-esports-market-revenue/>

[Accessed 21 July 2022].

Howarth, J., 2022. *27 Mind Blowing Esports Stats (2022)*. [Online]

Available at: <https://explodingtopics.com/blog/esports-statistics>

[Accessed 21 July 2022].

Team, Indicative, n.d. *Gaming Analytics: How to Leverage Your Customer Data for Sustained Business Growth*. [Online]

Available at: <https://www.indicative.com/resource/gaming-analytics/#:~:text=With%20this%20context%20in%20mind,%2C%20monetization%2C%20and%20business%20impact.>

Cruz, J., 2015. *Factors Affecting Data Collection*. [Online]

Available at: <https://prezi.com/ahfgd687klk/factors-affecting-data-collection/>

Weisheng, C., Fan, T., Nam, S. & Sun, P., 2021. *Knowledge Mapping and Sustainable Development of eSports Research: A Bibliometric and Visualized Analysis*, s.l.: s.n.
Mahlmann, T., Schubert, M. & Drachen, A., 2016. *Esports Analytics Through Encounter Detection*, s.l.: s.n.

Ruiz, J., Toukourmidis, A. & Moreno, S., 2022. *An overview of the gaming industry across nations: using analytics with power BI to forecast and identify key influencers*, s.l.: s.n.

Du, X., Fuqian, F., Hu, J. & Wang, Z., 2021. *Uprising E-sports Industry: machine learning/AI improve in-game performance using deep reinforcement learning*, s.l.: s.n.

Marta, R. et al., 2021. *Gaining public support: Framing of esports news content in the COVID-19 pandemic*, s.l.: s.n.

Pereira, G. et al., 2012. *Serious Games for Personal and Social Learning & Ethics: Status and Trends*. [Online]
Available at: <https://www.sciencedirect.com/science/article/pii/S1877050912008204>
[Accessed 25 August 2022].

Chee, F. & Karhulahti, V., 2020. *The Ethical and Political Contours of Institutional Promotion in eSports: From Precariat Models to Sustainable Practices*. [Online]
Available at:
https://www.researchgate.net/publication/346663166_The_Ethical_and_Political_Contours_of_Institutional_Promotion_in_eSports_From_Precariat_Models_to_Sustainable_Practices

Gainsbury, S. & Blaszczyński, A., 2017. *HOW BLOCKCHAIN AND CRYPTOCURRENCY TECHNOLOGY COULD REVOLUTIONIZE ONLINE GAMBLING*. [Online]
Available at:
https://www.researchgate.net/publication/319945691_HOW_BLOCKCHAIN_AND_CRYPTOCURRENCY_TECHNOLOGY_COULD_REVOLUTIONIZE_ONLINE_GAMBLING
[Accessed July 2022].

Richard, J., Ivoska, W. & Derevensky, J., 2021. *Towards an Understanding of Esports Gambling: Demographic and Clinical Characteristics of Youth Esports Bettors*. [Online]
Available at:
https://www.researchgate.net/publication/355411526_Towards_an_Understanding_of_Esports_Gambling_Demographic_and_Clinical_Characteristics_of_Youth_Esports_Bettors
[Accessed July 2022].

Liu, M., Dong, S. & Zhu, M., 2021. *The application of digital technology in gambling industry*. [Online]
Available at: https://www.emerald.com/insight/content/doi/10.1108/APJML-11-2020-0778/full/html?casa_token=WJnNrO7PRuQAAAAA:CNJ9iEuof4VHIdLQOAeX-CJN-QCjayQXK4ENGaaJARXCmStxz5MPPr26Ta-J4UjVodHaSg87wL2JhF5kHkXSYW-sTFIQfj7H4L7KxbtJkf-wiX9qDQu8
[Accessed 26 July 2022].

- Apolinario, T., 2022. *How Many People Play PUBG?*. [Online]
Available at: <https://fictionhorizon.com/how-many-people-play-pubg/>
[Accessed 26 July 2022].
- Carter, C., 2017. *Wrapping your mind around a popular, confusing game*. [Online]
Available at: <https://www.polygon.com/playerunknowns-battlegrounds-guide/2017/6/9/15721366/pubg-how-to-play-blue-wall-white-red-circle-map-weapon-vehicle-inventory-air-drop>
[Accessed 21 July 2022].
- Ishwarappaa, J., 2015. *A Brief Introduction on Big Data 5Vs Characteristics and Hadoop Technology*, s.l.: s.n.
- Oriani, R., Peruffo, E. & McCarthy, I., 2020. *Big Data for Creating and Capturing Value in the Digitalized Environment: Unpacking the Effects of Volume, Variety, and Veracity on Firm Performance**, s.l.: s.n.
- Geron, A., 2019. *Hands-on Machine Learning with Scikit-Learn, Keras and TensorFlow*. In: s.l.:s.n.
- Langley, P. & Simon, H., 1995. *Applications of machine learning and rule induction*. [Online]
Available at: <https://dl.acm.org/doi/abs/10.1145/219717.219768>
- IBM, 2020. *Supervised Learning*. [Online]
Available at: <https://www.ibm.com/cloud/learn/supervised-learning>
- Beers, B., 2022. *Regression Definition*. [Online]
Available at:
[https://www.investopedia.com/terms/r/regression.asp#:~:text=Investopedia%20%2F%20Joules%20Garcia-What%20is%20a%20Regression%3F,\(known%20as%20independent%20variables\).pyhtonbasics.org](https://www.investopedia.com/terms/r/regression.asp#:~:text=Investopedia%20%2F%20Joules%20Garcia-What%20is%20a%20Regression%3F,(known%20as%20independent%20variables).pyhtonbasics.org), 2021. *Why Python for Machine Learning?*. [Online]
Available at: <https://pythonbasics.org/why-python-for-machine-learning/>
- Pedregosa, F. et al., 2011. *Scikit-learn: Machine Learning in Python*. [Online]
Available at: <https://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html>
- Singh, D., 2019. *All About Using Jupyter Notebooks and Google Colab*. [Online]
Available at: <https://www.datasciencecentral.com/all-about-using-jupyter-notebooks-and-google-colab/>
- Sweeney, K., Tuttle, M. & Berg, M., 2019. *Esports Gambling: Market Structure and Biases*, s.l.: s.n.
- Macey, J., Abarbanel, B. & Hamari, J., 2020. *What predicts esports betting? A study on consumption of video games, esports, gambling and demographic factors*, s.l.: s.n.

Borg, L., 2021. *Investigating Training Set and Learning Model Selection for Churn Prediction in Online Gaming*, s.l.: s.n.

Ghazali, N., Sanat, N. & Asaril, M., 2021. *Esports Analytics on PlayerUnknown's Battlegrounds Player Placement Prediction using Machine Learning Approach*, s.l.: s.n.

Teeboom, L., 2019. *The Advantages of Regression Analysis & Forecasting*, s.l.: s.n.

<https://scikit-learn.org/stable/>

https://xgboost.readthedocs.io/en/stable/python/python_api.html

<https://lightgbm.readthedocs.io/en/latest/pythonapi/lightgbm.LGBMRegressor.html>

<https://pandas.pydata.org/docs/>

<https://keras.io/api/>