IN3007/INM450 Individual Project

AI modelling using Neural Networks and deep learning to predict golfer performance valuations based on professional data.

Adam Khanzada, adam.khanzada@city.ac.uk

Consultant Name: Michael Garcia Ortiz

Clients: N/A

Word Count: 12,468

Contents

Abstract……3

Chapter 1: Introduction……4

1.1: Problem Solved……4

1.2: Project Objectives……4

1.3: Project Beneficiaries……5

1.4: Work Performed……6

1.5: Work Plan……7

Chapter 2: Output Summary……8

2.1: Data Collection……8

2.2: PDD……8

2.3: Feature Analysis……8

2.4: Model Analysis……9

2.5: Performance Rankings……9

2.6: Final Paper……9

Chapter 3: Literature Review……10

3.1: Machine Learning……10

3.2: Neural Networks……11

3.3: Regression……12

3.4: Optimisation……12

3.5: Models in Sports……13

3.6: Models in Golf……15

Chapter 4: Methodology……17

4.1: Feature Breakdown……17

4.2: Software Code……18

4.3: Data Collection……18

4.4: System Preparation……18

4.5: Initial Modelling……19

4.6: Evaluative Metrics……19

4.7: Tuning……20

4.8: Final Modelling……21

Chapter 5: Results……22

5.1: Feature Analysis……22

5.2: Initial Model Results……32

5.3: Tuning Results……34

5.4: Tuned Models……35

5.5: Tuned Model Testing……37

Chapter 6: Conclusions……38

6.1: Objectives Review……38

6.2: Impact……39

6.3: Evaluations……40

6.4: Final Conclusions……40

Chapter 7: Glossary……41

7.1: Golf Terms……41

7.2: AI Terms……42

Chapter 8: References……43

8.1: Literature References..43

8.2: Coding References……44

Chapter 9: Appendices……45

9.1: Appendices A……46

9.2: Appendices B……57

9.3: Raw Code……79

9.4: ReadMe……89

Abstract

The main goal of this research is to build a neural network capable of accurately predicting golfer skill to at least 50%. How well the model forecasts the performance metric to be chosen in a test set serves as a gauge of the project's success. By offering knowledge about various models that have been constructed and a dataset, the objective is to contribute to the sports data science sector. By doing this, the project hopes to help athletes, branding teams, and sports organisers better understand how to evaluate rookie or seasoned players using golf statistics.

Chapter 1: Introduction

1.1: Problem to be solved:

I am looking to solve the problem that is faced by many organisations involved in professional golf pertaining to the correct evaluation of a rookie/amateur or professional players performance valuation. This would be a useful statistic for advertisers, sponsors, and tournament managers to be able to effectively plan and decide on rookie/amateur players worthiness of sponsorship.

It is very hard for sponsors to predict the success and popularity of newcomers to the sport, so I seek to alleviate the burden of waiting on tournament results by offering a predictive model to help derive valuations for these new players. This will allow for more informed decisions to be made on the sponsorship investments on players companies need to make.

Overall, this is solving the problems of indecision and time-loss created by the uncertainty in a rookie players prospects for several beneficiaries in the golfing world.

1.2: Project Objectives

1. This project shall develop and AI model using Neural Networks that will be able to predict an effective metric for player performance valuation.

1.a) The performance valuation will be able to generate predictions when tested to have an accuracy of at least 0.5 based on the professional player test data.

1.b) The predictive model will be able to be used as insight into the potential performance of players outside of the training datasets scope and could apply to players from other tours or levels of the sport (College Golf, Olympic Golf)

1.c) The model developed will be subject to evaluation and improvement through comparison and optimisation.

2. This project shall research into the effective measures of value a player is deemed to have and to what effect features will determine the extent of the success of players.

2.a) derive an effective understanding for performance value based on statistical research and data analysis of the PGA Tour golfer data collection. Analysis should extensively cover a wide variety of performance statistics.

3. This project shall streamline the indecision process of selecting golfers for sponsorships and tournament planning by giving insight into the factors that make a good golfer and how this can be measured.

3.a) This measure of effectiveness of this projects goals towards helping indecision can be gauged by how well the models perform and the time saved in using this new method.

1.3: Project Beneficiaries:

I believe that my project will offer great benefit and insight to several parties.

Golf Sponsors

Sponsors will greatly benefit from being able to successfully model the potential success of up-and-coming players. The predictive model I hope to build will greatly assist them in deciding which players are worthwhile time and money investments for sponsorship deals. For a sponsor there is great uncertainty and indecision created by the lack of tournament wins or popularity metrics for new players. My model seeks to offer insight into the value these new players can offer sponsors from an exposure/money generation standpoint. Sponsors are always on the lookout for standout players in performance and audience reception and I believe the model I am developing will stream-line the currently conjectural ineffective system that wastes time and opportunity.

Golf Advertisers

Golf advertisers for events will be able to benefit from my research and modelling as they can come to better conclusions on which players are able to sell the tournaments and appropriately use them in advertisement campaigns. These campaigns typically feature the top players consisting of seasoned veterans of the sport but where the issue arises is deciding on which of the new generation of players to feature in these ad campaigns. This problem is especially prevalent in the golfing scene as the demographic for golfers is ageing and advertisers are always trying to tap into a new demographic by catering adverts to younger generations. The model I wish to create will seek to assist in deciding on the players to feature thus hopefully propelling increased viewership and therefore sales for advertisers.

Tournament Planners

Tournament planners work in a similar strain to the advertisers as they seek to promote tournaments in several different ways. A tournament planner is continually looking to select the best players to fulfil the role of creating the ideal sports viewing for an audience. To effectively do this the planner will need to understand how to pick and choose the playing field for a given tournament and how to best match up players in golf groups to create fierce competition. This can be a struggle at times, especially when trying to gauge where to slot in newer players. My AI model will help with this as the valuation derived by my research should offer insight into how a planner should manage the flow of new coming players and where to seed them in future tournaments to receive optimal returns (revenue)

Golf Players

Finally, the players themselves should be able to gain valuable insight from the model I hope to create as I believe it is important for an induvial, especially in the sports scene, to be aware of the value they bring to the sport. The players should be able to know what is affecting their future prospects when it comes to getting sponsorship deals, advertisements, and rankings in tournaments. I believe that the research and modelling I hope to do will be able to shed light on how a player can better their performance and individual value they have to offer.

1.4: Work Performed

All data will be analysed, and all objectives should be met within the timeframe.

Assumptions made.

All of the assumptions made regarding the project's objectives and data.

-I can assume that all the data collected is accurate and correct player data (no biases or errors)

-Assuming that the historical data can predict future data regardless of external factors such as age, disease etc..

-Assuming that my model will be able to accurately make judgements even though players will have varying amounts if historical data depending on when they joined the PGA Tour or became a professional

Chapter Overview

Chapter	Description
1.Introduction	Introducing the projects problem to be solved and establishing the background and objectives needed to be met for successful solution
2.Production Summary	Project output will be presented alongside all data and tables produced that can be found in appendices
3.Literature Review	A thorough review of the literature and articles that inspired and aided in the projects understanding in Neural networks and sports data
4.Methodology	This will cover how the project tasks were executed and completed
5.Results	A report on the outputs produced from the project. This will be many tables and graphs placed alongside objectives
6.Conclusions and Evaluations	Overview of the projects objectives and derived conclusions based on results as well as an in-depth analysis of the level of success the project achieved and how it could have improved and how shortcomings could have been mitigated or avoided with the benefit of hindsight
7.Glossary	Definition of key terms
8.References	References to external work utilised
9.Appendices	All data, graphs and diagrams referenced throughout the report. Will also include the project proposal

1.5: Work Plan

Plan	Resources required	Start Date	End Date
Research/Literature review into the statistics important to golfer success	Internet, books, statistical data, time	After PDD accepted, already begun	1 weeks after proposal accepted
Data collection of PGA golfer tournament data and earnings…etc	PGA tour data (easily accessible)	Already sourced out some useful data	Within 1 week of proposal accepted
Cleansing and filtration of data to create a coherent and understandable data set	Any data manipulation software (Jupyter notebook) using Python, Excel, time	Mid- February	20th February
Witten introduction of project report (draft)	Time, research	Mid-February	20th February
Initial steps were taken towards building an AI model, testing several methods and approaches	Python AI libraries, research into neural networks, GitHub, time	End of February	Start of March
Methodology draft and thought process behind decisions written	Time, research	End of February	March 10th
Coding aspect completed for AI modelling	Python AI libraries, research into neural networks, GitHub, time	Ongoing throughout	Middle of March
Generation of useful graphics and statistics based on results of AI model	Python libraries such as matplotlib and excel, research into other metrics	ongoing	End of March
Predictive testing based on sample rookie/amateur data conducted	Rookie test data, dummy data, volunteer golf data, could use own data	Start of April	April 10th
Written reports on coding process and testing process	Research, time	Start of April	April 15th
Evaluative write up to see areas that could improve (draft)	Research, time	Start of April	April 20th
Goals lined up against success in meeting requirements	Research, time	End of April	End of April

Chapter 2: Output Summary

2.1: Golfer historical data collection and collation	
Description:	A majority of the data will be collected through data scraping of the data on the official PGA Tour data website as well as downloading csv files directly from https://www.pgatour.com/stats. The data set collected and collated currently has a player base of over 2500 unique golfers' data. This number may vary depending on how the data is cleansed/filtered down the line. The CSV file within the GitHub repository and attached to the final code package contains all the data used.
Usage:	The data will be used for training and testing models. It will also serve to conduct analysis into golf performance statistics and how they impact the success of a golfer.
Output Type:	A CSV file containing all the relevant golfer data. Currently contains 1670 rows of unique data and breakdowns of seasonal golfer stats.
Recipient:	Author, Reader, Sports Analyst's
Appendix Link:	Appendices 9.4 ReadMe

2.2: Project Definition Document and Golf Glossary	
Description:	A document designed for the purpose of helping non-golfers understand the syntax and technical terminology used when describing many characteristics and parts of a golf game and performance. This document can be found in the Appendix and aims to assist readers in their understanding of the features chosen and specific part of the essay where knowledge of the sport will directly correlate to understanding of how models/graphs are affected by these factors.
Usage:	Provide readers with the necessary knowledge to effectively understand the various key terms used throughout the essay.
Output Type:	Word Document attached in Appendices.
Recipient:	Reader
Appendix Link:	Appendices Glossary (see page.41)

2.3: Data feature analysis and justification	
Description:	A section of the research paper that focuses entirely on understanding the various features through graphical analysis and table analysis. The hope is to shed light on how various variables correlate and help gauge the effective performance factors in a golfer's rounds.
Usage:	Develop a better understanding of golf data and support the justification for several features selected.
Output Type:	Graphs and Analysis in research paper. Also presented in coding package using several graphs developed through SkikitLearn and Seaborn python libraries.
Recipient:	Reader, Author
Appendix Link:	Appendices Results (see page.57)

2.4: Analysis/Evaluation of Predictive model	
Description:	The analysis and evaluation through several means (graphically and calculations bases) of the predictive models I hope to develop. These should provide insight regarding the effectiveness of my models and how they can be improved with further iterations and more time.
Usage:	Provides the reader with insight on how the predictive models came to be and what steps can be taken to improve upon them. It will also provide an interesting argument for or against the choices made when designing the models parameters and establishing a feature set.
Output Type:	Coded Models in python represented through graphs and calculations. All graphs can be seen in the Appendices and will be analysed in the results section of the paper.
Recipient:	Reader, Author
Appendix Link:	Appendices [4-A to 4-F]

2.5: Predictive Model of player Performance rankings	
Description:	This should provide the ability for users to enter custom data of any golfers in order to gauge an idea of the players proficiency. Statistics used in the model construction will be able to help generate predictive results based on entered data to provide users with an accurate measure of proficiency.
Usage:	The Usage if this feature aims to help the several beneficiaries in fulfilling their goals of measuring the skill level of golfers. This varies between helping players or potentially sponsors come to better informed decisions.
Output Type:	Python code file supplied in coding package. Referenced in Appendices.
Recipient:	Author, Reader, Beneficiaries
Appendix Link:	Appendices [4-A to 4-F]

2.6: Final code output and Research Paper	
Description:	The final produced document entailing the entire process of discovery and resultant predictive models, followed by a through analysis of work done and evaluations and conclusion formed based on findings. The hope is to fulfil the several requirements set at the start of the project and provide utility through research to the several benefactors of the statistical modelling based on golfing metrics.
Usage:	The project document can be used to effectively understand the process of development and thoughts that went into designing and interpreting the end product.
Output Type:	Several packages of Code (around 500 lines of code of which a majority was done by me and the rest re-used/ re-purposed, CSV files and written PDF Documents.
Recipient:	Reader
Appendix Link:	Appendices 9.4: Raw Code

Chapter 3: Literature review

The opening paragraph on the importance of analysing research that has come before my work and trying to learn from it and evaluate the areas which could be further explored. Talk about personal learning and skills that will need to be developed for this project to progress.

3.1: Machine Learning

Machine learning is a broad term used to describe the development of computer systems that can learn and adapt without having to follow direct instructions. These computer systems are designed and trained to make competent decisions on their own to solve problems and make processes self-sufficient. Machine learning has continued to grow in relevance and sophistication since its inception in 1959 and continues to stand as a cornerstone of modern enterprise today due to its versatile and adaptive nature. The original concepts behind this revolutionary technology stemmed from the works of psychologist Donald O. Hebb, who pioneered the initial ideas of attempting to merge our brains functionalities and inner workings to modern problems by describing a system of artificial neural networks and neurons in his book **'The Organization of Behaviour' [Hebb, D.O. (2002)**]. The technology and implementation proceeded to grow in scope as more mathematicians progressed the study until we reached the modern era, where machine learning is utilised in most facets of life and holds great responsibility for the improvement in understanding data analytics and insight.

Machine learning algorithms are used to make either prediction or classification decisions based on a set of input data provided. The algorithm will then determine patterns in the data and attempt to make decisions based on these samples. There are many different types of algorithms that can be used to map out said patterns such as Neural networks, linear regression, clustering, and decision trees. Due to the breadth and depth of study conducted into machine learning over the past fifty years I have access to a plethora of studies conducted into the utility and usage of various machine learning algorithms and will look to evaluate their usefulness in researching this field to benefit the needs and requirements of my project.

In The paper **[Pineau, J. et all (2021)]** conducted a thorough investigation into the effective methods of improving the precision and reliability of machine learning algorithms and techniques by trying to rationalise the reasoning behind the disparity in initial research results and the percentage of follow up researchers aiming to reproduce said outcomes. "**a 2016 survey in the journal Nature revealed that more than 70% of researchers failed in their attempt to reproduce another researcher's experiments, and over 50% failed to reproduce one of their own experiments (Baker, 2016)**". The areas investigated to correct this lack of consistency in results were narrowed down into several categories such as Selective reporting, Over-claiming, and under-specification of training models. The insight gained from this paper that I found helpful in planning my methodologies out came from the emphasis the paper places on carefully conducting a thorough analysis of data and features before creating machine learning models. To ensure that I don't leave any of my work up for misinterpretation, I must aim to provide in-depth specification of metrics and not draw conclusions based upon results that are not founded by concrete evidence.

Deep Learning/Neural networks (some stuff can be moved to method, keep somewhat brief)

3.2: Neural Networks

Deep learning is a subset of machine learning that focuses on building a neural network with three or more layers. A neural network is a type of machine learning algorithm as mentioned earlier that attempts to mimic the behaviour of the human brains decision making abilities using a system of neurons to generate predictions. All neural networks are comprised of three core layers. The input which takes in data, the processing layer which takes the data and attempts to generate expected outcomes based on training data and finally the output layer which displays the end product/results.

Neural networks have continuously developed as a subsection of deep learning throughout the years due to the discoveries made in the work "**A Logical Calculus of the Ideas Immanent in Nervous Activity**" **[McCulloch, W.S. (1990)]** where the analysis of the brain lead to the conclusion that the patterns formed by our brains could be abstracted into simple binary logic. This work was furthered by psychologist Frank Rosenblatt in 1953 who developed the logic behind perceptrons that introduced weights to the processing layer of neural networks. The expansion in understanding of the human brain and deep learnings has meant that there is no shortage of useful research and papers to gain insight from regarding the utility of neural networks in my project.

The procedures used within the paper "**Speech Recognition Using Deep Neural Networks: A Systematic Review" [Nassif, A.B. (2019)]** have shed light on the sorts of practices I should utilise in my paper when planning the methodology and in how to effectively prepare for the evaluation of my results. The key methodologies highlighted in the paper revolve around the usage of dividing the work into three different stages; Planning, conducting, and reporting which are all broken down into several smaller phases. I look to emulate similar practices when developing my models. Moreover, the research this paper conducted into other papers and their analysis of speech recognition using neural networks have allowed me to clearly understand the importance of setting evaluative precedence early on in the project as to not skew any results down the line so that objective requirements and later analysis retains consistency with the projects goals and the work of similar research papers**." Several evaluation techniques were used in the research papers to evaluate the overall performance of the developed system. Table 8 illustrates the different identified techniques, as well as the percentage of papers that used each technique. As it can be seen, 56% of the papers used Word Error Rate (WER) to evaluate the performance of their system" [Baker, M. (2016)].** Defined evaluative models and metrics used across similar papers serves to strengthen the validity of evaluations and results therefore I will look to use metrics such as Root Mean Square Error, F1 scores and precision when assessing my results.

3.3: Regression and supervised learning on sequential data

Supervised learning refers to the use of correctly labelled data as input and output to assist in the training of a model to generate predictions more accurately. Regression is a type of supervised learning that can use an instruction set to correlate associations between dependant and independent variables. Similarly, to the research I have found in neural networks, I have found great insight by researching papers focused on supervised learning.

The paper **2An overview of the supervised machine learning methods" [Nasteski, V. (2017)]** drove me to well informed conclusions regarding the importance of supervised data and its impact on desired results when working with neural networks. I was able to gauge a better understanding of how supervised data and the labelling of outputs would affect results and the different methods used for supervised learning such as linear regression, logistic regression, decision trees and Bayesian classifications. The conclusion founded in this paper "**for the supervised learning it may be concluded that is one of the dominant methodology in machine learning. The techniques that are used are even more successful than the unsupervised techniques because the ability of labelled training data provide us clearer criteria for model optimization**", has driven me towards ensuring that all my data is presented with correct labels, is reliable and reproducible and has enough utility to justify the chosen features selected for analysis and neural network modelling through regression. Furthermore, my research of this paper is supported by my readings and discoveries made in "**Pattern Recognition and Machine Learning" [Bishop, C.M. (2009**)] where he delves into linear models for regression through supervised learning and explains the significance of carefully curated input variables and the benefits of these models. "**They have nice analytical properties and form the foundation for more sophisticated models",** leading into my further delving into the utility I can find in regression-based models and the pursuit of large amounts of comprehensive data suitable for my project.

3.4: Model Optimisation and Hyper Parameter Tuning

A shared goal or theme presented across many of the papers and books I researched was the aim to enhance the performance of predictive models to be more precise and accurate when making judgements, whether for a classification or regression-based problem. A few the papers I carefully investigated referred to processes called **"Hyper Parameter Tuning".** This can be described as the process of searching for the ideal model architecture by tweaking various constraints the model uses to operate. The paper **"Classification of date fruits using Convolutional Neural Networks" [Alhamdan, W and Howe, J. M (2021)]** took a manual search approach to testing out the different results yielded from changing a list of several variables. This resulted in several versions of a completed model, each in turn improving upon the next across a variety of metrics. I believe that this could prove to be an effective method for optimising my models down the line and further exploration into optimization techniques such as Grid Search or Halving may prove to serve the betterment of my initial prototype builds.

3.5: Data and Predictive modelling in sports

Data has played a pivotal role in the progression of the sporting industry in the past twenty years. With tougher competition, tighter margins for error and a now globalised world, the need for athletes competing in sports and companies sponsoring them to receive and utilise reliable data to make important decisions has increased greatly. Data can be used in every facet of a sport to be of benefit to various stakeholders in both team and individual sports.

The need for data to assist in making informed decisions grows in parallel to the revenue the sports industry turns over yearly. The greater the money involved, the more risk taken on by general managers/sponsors and players when making big decisions. Data and analytics in sports helps to justify and gather useful insight behind making critical decisions. These decisions driven by data science can result in cost effective strategies and better overall desired outcomes for all parties.

A paradigm shift was occurred in all major sports after the publication of Earnshaw Cook's **'Percentage Baseball'** in 1964. The subsequent impact of this successful approach to sports management using data analytics to value players changed the sporting landscape forever and resulted in the refactoring of the mindset and practices of all major sporting clubs in the world to adopt a data driven approach to decision making.

There is now no shortage of data available to analyse in all major sports. This has meant that I have had access to a plethora of useful research papers delving into how predictive models can be created using machine learning and sports data to accurately predict sporting trends and thus make well informed decisions.

The research carried out in **"Developing a Data-Driven Player Ranking in Soccer Using Predictive Model Weights" [Brooks, J. and Kerr, M. et all (2016)]** regarding the ranking of football players offered me great insight as I was able to understand how to approach the problem of selecting which statistics took priority when choosing features for my predictive model I hope to build. This is a critical topic of discovery as the selected features will greatly influence the success of my models. The paper mentions **"To extract the features that we used to build our predictive models, we first segment each game into a discrete sequence of observations.".** The meaning behind this statement in the paper was to analyse a football game and see the various events that breakdown the structure and flow of a football match. Using this method of thinking they were able to conclude that a football game can be broken down by possessions and therefore used the positioning of players and the possession statistics to develop a set of features that would be used for their classification model. This effective way of thinking about how a game is played to inform decisions for feature choice will be invaluable in helping my decision making as I will look to breakdown a golfer statistics into several phases of play throughout the year and round. This approach can be applied to my work through suggesting that a golf round consists of tee-shots, approach shots and putts. With this I can try to pick values that contribute to these parts of the game. Moreover, this thought process can be carried over to a higher level of the game and I can focus on the features that make up the overall performance of a player such as the various tournament results throughout the year which may lead me to select figures like scoring averages or tournament difficulty. The work conducted in this paper has provided me with an interesting perspective on how to look at the structure of the game and I will look to apply this in my methodology.

This was furthered by the findings in the paper **"Player Rank: Data-driven Performance Evaluation and Player Ranking in Soccer via a Machine Learning Approach [Pappalardo, L.P. and Cintia et all, P. (2019)],** which supported the approach used by Brooks and helped to cement ideas of feature implementation and also introduced new ideas about how features can be created through combining useful performance stats. The paper looked into individual player performance rankings which similarly to my research paper will focus on the individual rather than a team and attempt to develop metrics to weigh and rank the success of players. The paper goes into a great deal of depth discussing how the rating system will be based around simple scalar features and other complex weighted features. **"Each feature weight models the importance of that feature in the evaluation of the performance quality of any player. Formally speaking, given the multi-dimensional vector of features pm u = [x1,..., xn] and their weights".** The split between calculated metrics being weighted based on the level of impact they have on performance and the face value stats that are used leaves me to think about how I can similarly effectively manage my feature set and decide on the best set of data to represent performance. The study derived an interesting value based on a combination of completed passes, shots taken, and cards achieved, each having a weighted point system attached to determine whether the value was at a detriment to performance or a benefit. I could apply a similar method to my project where I look into both negatively affecting stats alongside the positively affecting values to reach a more comprehensive set of data to base my models off. An example of this could be weighing the number of strokes gained for a golfer with a relatively high weighting and then taking into account that a higher value associated with average bogeys per round would skew a performance variable.

The research into **"Modelling analysis and prediction of women javelin throw results in the years 1946 – 2013", [Grycmann, P. and Maszczyk, A. et all (2015)]** brought a different outlook to my discoveries as unlike the previous two papers. This paper focused on an individual (a sport played by a single person, not team) sport which resonates more with the sort of models I will hope to produce in my work as golf is also a solo game and the results being affected by external factors (teammates) to the individual are severely mitigated. One important point that stuck out to me when reading this research paper was the mentioning and rationale behind data era selection as they had access to decades of female javelin data but chose to build models based around specific time-periods. The paper states **"Due to a relationship between throw results and changes in javelin construction, results variability, and trends in the years 1949 through 2011 are presented as a broken trend line. The predictive models, on the other hand, correspond to the period after changes of the centre of gravity of the javelin, i.e., the year 1999."** This brought up a fascinating point for consideration when being selective about the data I should omit from my models and analysis. Any data the paper found to be being affected by the period of time or external factors was either filtered out for consistency purposes or amended in format to suit the modernisation of the sport. My personal research and knowledge of golf could be used to apply similar techniques used in this paper as there will be periods of the sport where the data will be skewed due to external factors such as the covid-19 pandemic which will most definitely result in shortages of data. Also, the natural progression of technologies regarding the design and rulings around golf equipment will have a toll on all data based earlier than 2011 due to the changes in the mandates for conforming golf equipment by the USGA and R&A, which restricted the designs of grooves on clubs pertaining to their depth, width, and shape, which directly impacted player performance negatively.

3.6: Golfing Data and predictive modelling

The papers above are heavily focused on footballing and other sports but it would be of further benefit to find more focused research on golf papers and evaluate its relevance and value to my project/understanding of the field.

The paper "**PGA Tour Machine Learning Project" [Park, J. (2019)]** was very helpful in allowing me to learn from example as our projects overlap in many areas and also rely on similar data sets. The insight gathered from this paper has allowed me to better understand how I can set out analysing golf data and what steps should be taken to prepare my data for regression-based modelling. The paper also brought up interesting points about the nature of golfing data and the sorts of statistics I should look to predict. The research conducted by Park looked to utilise a mix of golfing statistics throughout a player's year such as Rounds, Driving Distance, average GIR and various other strokes gained based metrics to predict whether a player won a tournament and the total player earnings for the year. He has used a wide array of metrics to develop a model which he believes accurately predicts performance through wins and earnings. The papers use of data analysis through graphs to justify the set of features and explain trends in earnings provided an interesting insight and inspires an effective approach to understanding the features at their cores and how they vary between the higher performing players. He mentions "**From the distributions plotted, it appears that most of the graphs are normally distributed. However, we can observe that Money, Points, Wins, and Top 10s tend to are all skewed to the right. This could be explained by the separation of the best players and the average PGA Tour player. The best players have multiple placings in the Top 10 with wins that allows them to earn more from tournaments, while the average player will have no wins and only a few Top 10 placings that prevent them from earning as much.".** These sorts of critical evaluations of his data and fields collected have meant that he has a firm idea of the causation of his data distributions and understands how the various features interact and affect each other. I think there is great merit to this approach as this thorough analysis he has conducted justifies his feature choices later on and allows for better understanding of how the chosen metrics correlate to his picked labels for performance.

An area in which further research and more scrutiny for the metrics of golfer performance that I believe to disagree with his paper and conclusions derived are that wins and earnings are a valuable gauge of player success and whether these values are worth deriving above others to track performance. Wins in golf are a heavily dissected area of contention for seeking out who the top players are, but due to the so few tournaments in a year and the individual nature of the sport, looking at this result rarely gives true insight into the best golfers. Furthermore, the usage of the metric for earnings to signify a calibre of player has some merit but lacks consistency and foundation as towards the end of a golfing season typically purses are increased, and event bonuses skew this statistic. The FedEx cup rankings and tournaments is a prime example of causation for inflated player earnings. When developing my own models and deciding on features and labels I shall carefully consider the methods I can use to mitigate the misinterpretation of my data for a direct metric of proficiency. A feature such as number of **"top 10 placements"** could provide a more well-rounded and consistent base for any claims of player performance measure.

This research combined with the knowledge gained from another similar project "**PGA tour winner Classification Machine Learning Project" [Prater, D (2018)]** that looked to use machine learning to build a classification model to predict PGA Tour wins provided my research with some clarity relating to more ways in which I can explore my data. Also, his conclusions and portrayal of results graphically to evaluate his created models alongside the raw calculated metrics of precision, f-1 score and recall generated effective well-found insight. Some of the observations made in his evaluations could provide great utility in my design/method choices, especially when considering my features selection and modelling techniques. A key point made was **"The number of individuals that hit the ball over 300 yards jumped from 25 in 2016 to 41 in 2017. What is causing this huge spike in long drivers? Can we expect this to increase at the same rate? How will this impact future tour events?"** which brings up the natural progression in some golfing stats as the years go by most likely caused by a shift in player mentality created by the influx of successful big hitters on the field such as Brooks Koepka and Dustin Johnson. Also, the slow progression in the optimization of golf equipment. I was hoping to minimise the impact of changing clubs by drawing my data from areas unaffected by changes in policy or era, however it seems that these marginal gains in yardages as the years progress are unavoidable. This does raise an interesting point regarding the usage of statistics that don't discriminate with time as they are comparative between players of the time, so they don't fall victim to changes of times. An example of something like this would be tournament rankings, Strokes Gained or average positional percentages. This could be a point of great significance when I come to decide on my features and provide rationale for my choices.

Moreover, the papers usage of an iterative approach to modelling based on feature engineering to maximise evaluative results has proven successful in creating receptive/accurate models upon each progressive repetition. **"We can see that the ROC AUC score has increased by engineering domain and polynomial features."** Thus, presenting a compelling case to pursue a methodology that focuses on improving my neural network incrementally. Methods such as feature engineering and hyper parameter tuning could provide a reasonable means for model optimisation.

Finally, the articles **"Wise guys: Data golf is taking analytics to a whole new level" [Corcoran, M. (2019)]** and **"The increasing presence of data analytics in golf" [Arastey, G.M. (2020)]** were both able to make cases for the utility on statistical optimization of golfing data and were able to come to well-found conclusions regarding several areas of the game. They credit the strides made in golf insight through data to the development of reliable technologies such as Trackman simulators and high-resolution and framerate cameras, such as Foresight Sports' GC2 Smart Camera System. The technological leap created by said systems has meant that research teams at GolfTEC have been able to identify key components in a golf swing to benchmark professional standards**. "These agencies often provide golfers with tailored technical support and produce objective analysis of their game to identify trends and assess strengths and weaknesses"**. I was then able to gain a well-rounded impression of the numerous factors that effect a player's performance and hopefully will be able to make use of their wide range of discoveries.

Overall, the wide array of research conducted through the plethora of papers, books and articles at my disposal has amounted to my confident understanding of golfing data and the various analysis and machine learning models I can apply to my methodologies and solutions.

Chapter 4: Methodology

The following section details the planning and breakdown of the several phases of development of my project. The importance of understanding the project requirements and scope before beginning the prototype solutions will provide several benefits ahead of time such as the mitigation of potential oversights and the necessary thought processes applied beforehand can result in fewer iterations and an overall more efficient and productive work cycle.

4.1: Feature Breakdown

When developing a project that would look to project and predict a players performance based on their yearly statistics, it required a deep dive into the vast array of metrics currently used to determine the success of players. A breadth first search into these values lead my research towards following an approach similar to that of (Jong 2019) and (Prater 2019) who selected a feature set for their models based on industry standards; however, I came to different conclusions as their projects were looking at a more generalised understanding of golfing success an applied less focus to factors strictly effecting performance such as earnings, which does indeed play a part, but can be a misrepresentation of performance understanding due to the escalating nature of golf purses throughout a single season and the general inflation that occurs on a yearly basis. This is especially so with the current state of affairs caused by the introduction of competitor tours like LIVgolf. Furthermore, the values I would use to represent the effectiveness of a player would pay considerable thought to the natural variance in standards of the times created by rule changes and technological improvements. This can be helped via the selection and scrutiny of data from a single period of golf and the careful assortment of features to not overly skew created models to be products of their time.

The following metrics are the collection of data scraped from and downloaded through csv from the official PGA tour website which I look to use in my model creation.

Features:

-Year

-Rounds Played in Year

-Average Fairway Percentage

-Average Driving Distance

-Average GIR (Greens in Regulation)

-Average Putts

-Average Scrambling

-Average Score

-Strokes Gained Statistics (SG: APR), (SG: OTT), (SG: ARG), (SG: Putts)

-Average SG Total

-Label/Prediction Value: Number of Top 10 Placements

4.2: Software Coding Methodology

Coding adopted a waterfall approach where careful planning of methods for coding were constantly tested and trialled based on research and findings. Methods will either be refined or altered to suit the improvement of the models and carry out the necessary functions for project objectives to be met. This planned approach allowed for some freedom in development and gave opportunity for the testing models (prototypes) to be created and refined to make way for later iteration. The nature of the project and its need to improve through a level of trial and error gave way to a later found agile approach as the model optimisation would require more freedom in thought and development. Machine Learning often requires experimentation and time so the more research and learning, the better prepared I will be for future redesigns and optimisations.

4.3: Data Acquisition and Analysis

The collection of data from reliable PGA Tour website using the sites features to download csv combined with the usage of the code provided by (Prater 2019) to scrape the remaining inaccessible data from PGATourData.com using beautiful soup python library.

A then diverse set of data on players of varying skill levels on the tour will be collected joined with other sources to form a single csv file with all raw untreated data. The data will then be loaded into a Jupyter notebook file through the use of panadas and numpy to fit the data into a data frame.

From the data frame the data can be properly checked for any errors. This is where the data cleansing and filtration will take place. All of the data's null values will be converted to 0's or the rows will be removed entirely. Furthermore, data outside of the time period scope will be filtered out. This s due to the need to have modern data unaffected by external factors.

The thorough analysis of the data will be conducted through the use of several python libraries and commands built into numpy, pandas and seaborn. We will be able to gauge a better understanding of the data shape and see how the various selected features interact and correlate with each other. Several graphs will be generated to analyse the relevance of the potential variables used for training our model. These graphs will help to inform decision making ahead of time.

Once data is cleansed and filtered it can be reshaped to a format that can be better interpreted referring to the shaping of data to support the formatting of a Tensor for our open-source machine learning platform of TensorFlow to develop a neural network model.

4.4: Computing System Preparation

-Usage of Anaconda. Anaconda is a distribution of the Python programming languages for precise computing, that intends to streamline package supervision and utilization. The offerings include data-science packages and a machine learning environment suitable for Windows and the needs of my project. All coding and data work will be conducted in Python through JupterLab as this IDE is effective in running and developing code in useful snippets. Several python libraries will need to be installed beforehand to ensure the AI modelling and data analysis can be done. These libraries are Numpy, Pandas, Seaborn, Keras, TensorFlow and Sci-kit Learn. Furthermore, it is worth noting that the system running these processes will have an impact on processing/load times and that this will all be done on a personal computer system with the following specifications: GPU: Radeon RX5700XT, CPU: Ryzen 5 3600X, RAM: 32GB 3200MgHz GDDR4 RAM

4.5: Initial Modelling

I will be using Tensor Flow open-source platform for machine learning to conduct a majority of the development on my deep neural network modelling of the number of top 10 player rankings. This will work well with the Keras interface python libraries to allow me to develop graphical models to portray my results. My results will be evaluated through the use of several recognised regression evaluation metrics such as Absolute Root Mean Square Error, Accuracy and the visual representations of the true vs predicted values my models generate. All the evaluations can be portrayed to show their effectiveness and the ability for development by measuring them along Epoch independent variable graphs.

4.6: Evaluative Metrics to Be Used and the expected/desirable outcomes.

-Absolute Root Mean Square Error (RMSE). Looking to see this value drop with each epoch and not increase as it would suggest losses and reveal the either underfitting or overfitting of the models to the training data.

$$\text{RMSE} = \sqrt{\frac{\sum (y_i - y_p)^2}{n}}$$

-Accuracy. Looing to see increasing accuracy with each epoch. Ideally seeing an improvement over initial models after hyper parameter tuning. This will suggest a level of success in models learning ability and the projects outcomes being met.

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{All Samples}}$$

-True vs predicted comparisons.
The ideal output would be seeing a graph that portrays positive linearity and minimal width in scatter from the y=x regression line. An advancement in the graph's convergence of gradient toward 1 and the correlation to be stronger in the positive direction would suggest optimal performance.

4.7: Hyper Parameter Tuning/ Model Architecture

It is very apparent that the initial models developed will be underperforming due to a lack of well-set parameters to assist in the learning process of my models when training on my dataset. It is to be expected as the nature of my data is random and continuous. This is why I will most likely need to conduct some hyper parameter tuning to optimise my models and derive results that will provide more value to my beneficiaries and meet the accuracy objectives of my project.

There are several methods that can be used for hyper parameter tuning, all providing their own benefits and drawbacks. I plan to implement a grid search method using Sci-kit Learn library methods for implementation. This approach consists of dividing the domain of the hyperparameters into a discrete grid. Then, we try every combination of values of this grid, calculating several running metrics using cross-validation. This exhaustive search method will take considerable processing time as it runs through all possibilities; however, the value gained from the end result will provide a set of optimal parameters I will need to build an improved model.

The following hyper parameters have been chosen for the grid search and will be incremented at several values in an array being attached to a dictionary that will generate results scoring the best arrangement.

-Optimizer refers to the algorithm methods defined for improving the performance of a model. It does this by taking different values a neural network uses through epochs to adjust for better learning rates and weights. You can use different optimizers in the machine learning model but when production with thousands of lines of data, even a single epoch can take a substantial amounts of time. Therefore, randomly choosing an algorithm is risking wasted handling time. I have decided that the needs of my project would be best suited to the trial of adaptive moment estimation using Adam Optimizer and the pursuit of testing parameters around Root Means Square Prop Optimizer (RMS Prop). These optimizers provide a balanced approach to my grid search as they are well suited to smaller data sets and mini batches. This is ideal for my data set of 1600 rows.

-Learning Rate is used to control the responsiveness of the model with each estimation of error when the model weights are renewed. Selecting values for potential learning rates can have positive impacts on model performance but can be detrimental to processing when the value is set too low, and too high causing sub-par learning due to the training being conducted too fast. For the sake of developing a balanced model that is not too taxing on my processor and having the benefit of not being undertrained, I have chosen to use an array of learning rates of 0.001, 0.01 and 0.1. This should provide sufficient opportunity for the grid search to develop a model within the time scale of the project and still have the competency to generate optimal results.

-Batch Size or splitting the data into batches is a method for optimally dividing our dataset into smaller chunks to be run through the neural network. The divisions of data are pushed through our network in batches of our set size until an epoch is completed. There is a balancing act that must be adjusted when optimising the batch size as we must try to configure an arrangement that will coincide with the number of epochs our model will train for. Having a large batch size will mean that our model can complete each epoch faster, given that we have enough processing power; on the other hand, this high value could result in the models lack ability to generalise competently on data it hasn't seen before. Low batch sizes suffer similar issues with regard to undertraining and create much greater processing times. In order to minimise the developmental risks to the project's timescale and performance, I have designed the grid search array to hold a variety of batch size values that won't tend to either extreme. These values will be 50, 100 and 200.

-Hidden Layer Size describes the number of nodes between the input and output layer in a neural network. The more layers, the more weighted values and functions are applied to separate data into strands for modelling. This is an interesting area with open interpretation to what can be considered optimal layer sizes and many approaches can be taken to assessing the best values to apply here. I aim to go for a depth approach which looks at basing the number greater the more challenging the problem is. The nature of my project and the variety in random data makes a heuristic approach viable and will hopefully yield compelling results. I have decided to test values ranging from 16 to 64 with the intention of basing these figures off similar papers that used complex sporting data.

-Epochs are the number of times all the training data makes a pass through the neural network. An epoch is made up of one or more batches of the data. In our case the batches will be varying in size based on the array mentioned earlier. When trying to figure out the optimal values for epochs to iterate through for my models, it is important to understand the effects of high and low epochs. In essence the more epochs, the more training the model will undergo as the data set is continuously passed through, a low number of passes can result in the improper fitting of data and create an imprecise and inaccurate model. This differs from a high level which improves precision, at the expense of potential overfitting and increased processing times. I have decided to base my values on standard practices and avoid excess training, but still reap enough benefit from my data set to build a competent model. I will be looking to test epochs at 10, 50 and 100 and see how my evaluative metrics respond to these figures. Ideally I can expect a plateau on my graphs with lower epochs, if not it will signify the need for further training.

4.8: Final Modelling and Testing

Based on the results developed from the training of hundreds of models through the grid search hyper parameter tuning we should be able to build a final model with the optimal parameters which will result in our final deliverable machine learning product. The model will be analysed using several graphs denoting the success via the evaluation through the metrics discussed above. Based on the findings further optimisation may be permissible given more time. Furthermore, re-assessment of chosen variables and features could lead to more promising future developments. This is all conjecture and will have to be proven upon coding operation commencement.

Chapter 5: Results

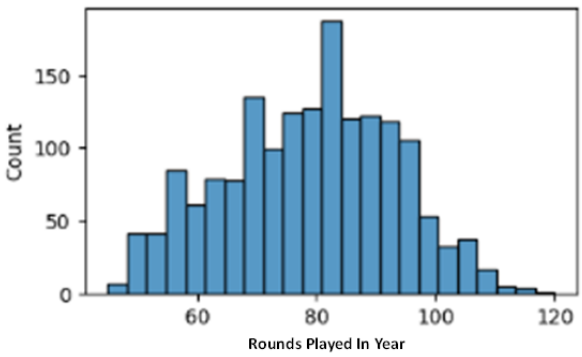
5.1: Feature Set Analysis:

Rounds Played in Year

Rounds Played in Year is a valuable metric for our model to process as a greater number of rounds played in a season (implying more tournaments played) directly relates to a greater number of opportunities for an individual to rank in the top 10's. This is further of note due to the nature of competitive golf since the halfway cut eliminates the lower half ranked competitors midway through a tournament where only the top half can continue play for the remaining rounds. Thus, correlating rounds played with better finishing positions as players with better scores would proceed to play all four rounds in a tournament unlike their lowlier ranked contemporaries.

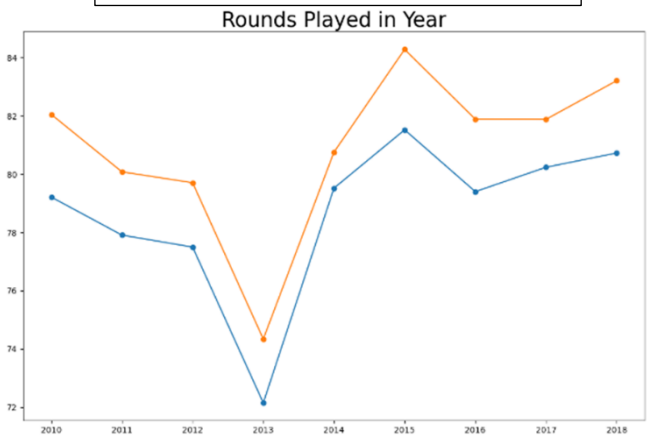
From the distribution graph (Fig 1.) we can observe that the majority of players are playing approximately 85 rounds per season. Generally, this pool of players will be comprised of some of the highest ranked due to their qualification to play in almost all events. These players, usually the top 50 in the world, are exempt to play in almost every available tournament in the season. Players on the tails of the distribution would be expected to be those ranked lower than the automatic qualifiers previously referred to. This is reasoned by the fact that lower ranked players would not qualify to play in many events so their rounds played will be low. Conversely the higher extremities would be players mid ranked, that would qualify to play in most tournaments but would choose to play a higher number to gather ranking points since their typical finishing positions are not consistently within the top 20. The best players, who will rank higher on average, will play a limited and selective schedule comprised of the most important competitions and fall within the 80-90 rounds played per season.

Fig 1. [1-B]



From the graph (Fig 2.) we can see the average of the top 10- players annually follows a very similar trend pattern to that of the average for the entire player pool. This is concordant with the notion that the best players play near the average or median rounds per season indicated as described above. The small discrepancy between the two trends is minimal and is most likely a result of better players making the cut more often, thus completing the four customary rounds as opposed to two.

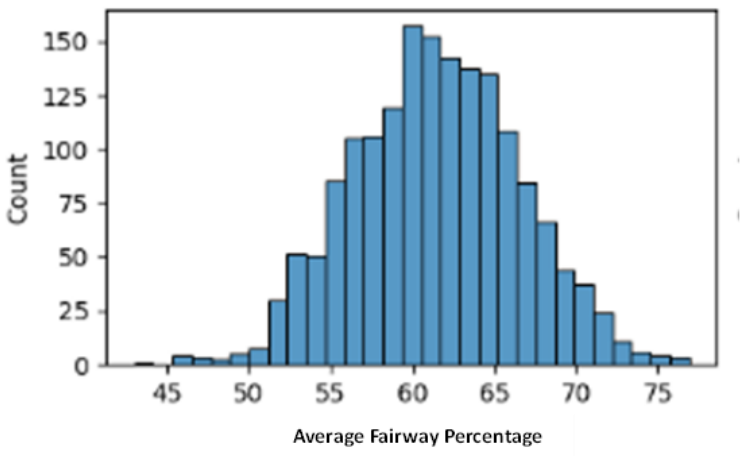
Fig 2. [2-A]



Average Fairway Percentage

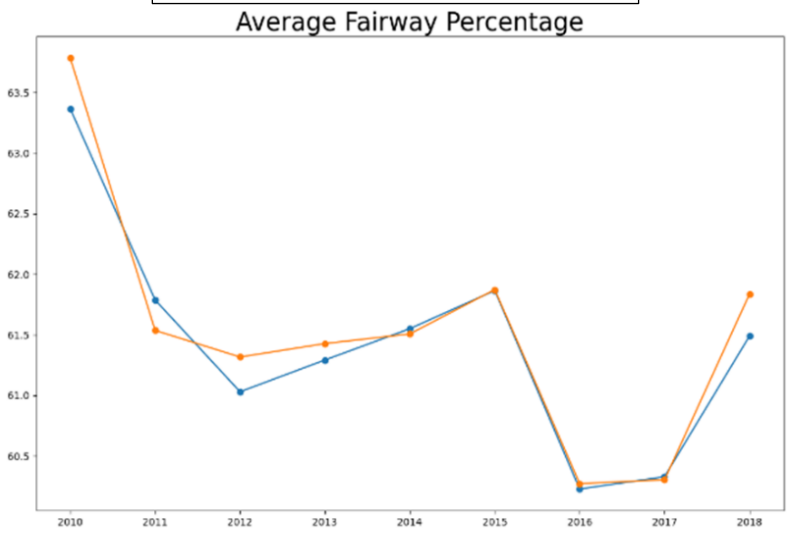
Average Fairway Percentage provides a statistic by which we can assess the effectiveness of a players shot making from the tee with regards to their accuracy. Due to the sport being played in a manner where the proceeding shot is played directly from a position determined by the prior stroke, it is typical that poor swings lead to greater difficulty with the upcoming attempt, therefore we have included a value denoting a player's accuracy to account for the added value to scoring from their precision off the tee.

Fig 3. [1-C]



As we can see from the frequency distribution from the graph (Fig 3.), the modal fairway percentage is approximately 60% for a tour player. The distribution follows a typical bell curve wit tails at either end and the bulk of the volume laying around the average.

Fig 4. [2-B]



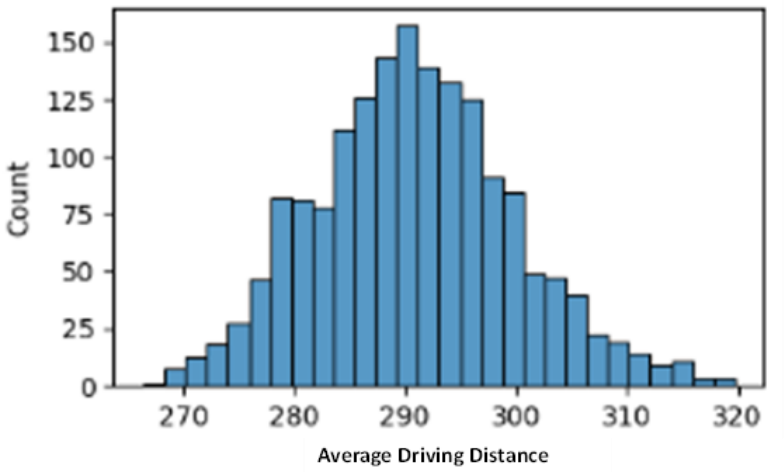
The graph (Fig 4.) here is clearly showing the similar trend pattern between top 10 players and the average player. This suggests that an isolated improvement in accuracy does not necessitate better performance alone as players with similar averages are ranking in many different positions as seen by top 10 players and average players following one another very closely. This difference can be reconciled by a recognisable difference in driving distance while maintain accuracy, as will be discussed in the next feature analysis.

Average Driving Distance

Average Driving Distance has been included as players who hit the ball farther are generally closer to greens on their approach shots. Closer shots are easier to control and thus would result in shorter proximity to the hole on average and a greater likelihood of completing the hole in fewer shots taken.

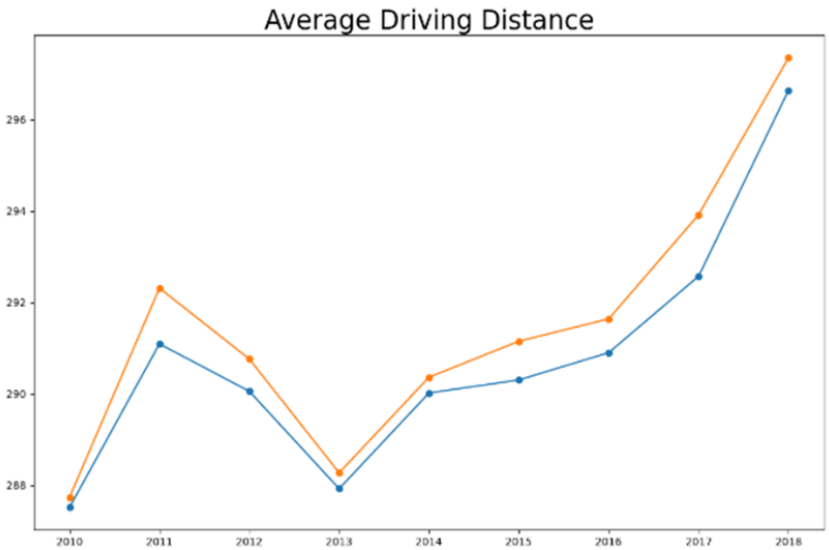
From the graph (Fig 5.) we can see a bell distribution where the greatest concentration of players sits at the 290 yards bracket. We can clearly see a mild positive skew demonstrating a greater spread of driving yardages longer than the average. The lower end of the tail is far more truncated suggesting the inability of players driving the ball below 260 yards to compete on the PGA Tour, whereas longer hitters almost indefinitely can compete and maintain their playing status.

Fig 5. [1-C]



As we can see from the graph (Fig 6.) higher performers drive the ball farther. Relating back to average fairway percentage discussed previously, we saw that better players were as accurate as the mean, but now additionally we can read that they drive the ball farther. This combination of hitting the fairway at the same rate as the average, but hitting it longer allows these players to be nearer target while simultaneously benefiting from the added control of being in the fairway, contributing to better scores.

Fig 6. [2-C]



Average GIR (Greens in Regulation)

Average GIR (Greens in Regulation) gives us a number by which we can ascertain a player's effectiveness in navigating a golf course from tee to green. More greens in regulation would directly relate to better scoring as it provides more scoring opportunities such as birdies, and a reduced likelihood of positive scores relative to par.

Fig7. [1-D]

The graph (Fig7.) shows an average greens in regulation percentage of 67.5%. The higher number of greens in regulation allows for better scores as more opportunity for bridies and a reduced likelihood of over par scores on holes. The better players are expected to be the ones with more greens in regulation per round.

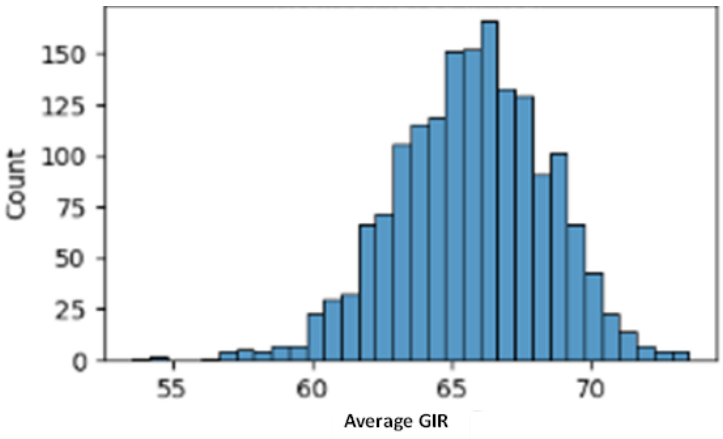
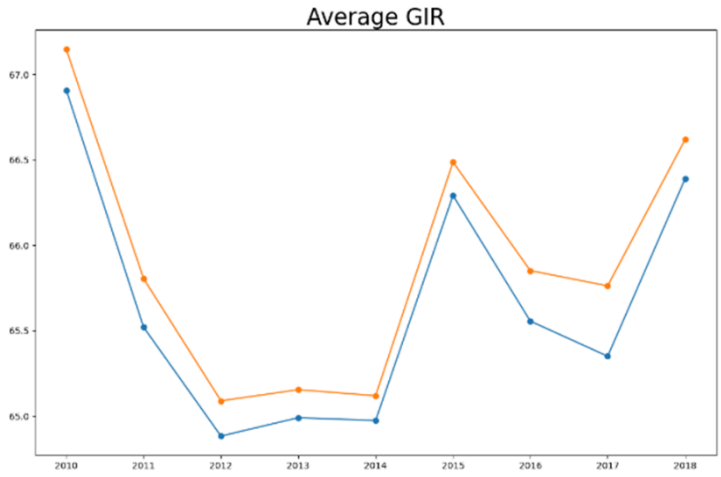


Fig 8. [2-D]

This trend mentioned beside (Fig 8.) can be seen by the difference between GIR percentage of the top 10 players and the average on the adjacent figure where the better players are seen to average a higher percentage. This trend is consistent as a 2 percent increase over the rest of the filed is witnessed yearly basis. This number may seem insignificantly small; however, across the many rounds and tournaments played a year. This number compounds to play a huge impact in a golfer's success.

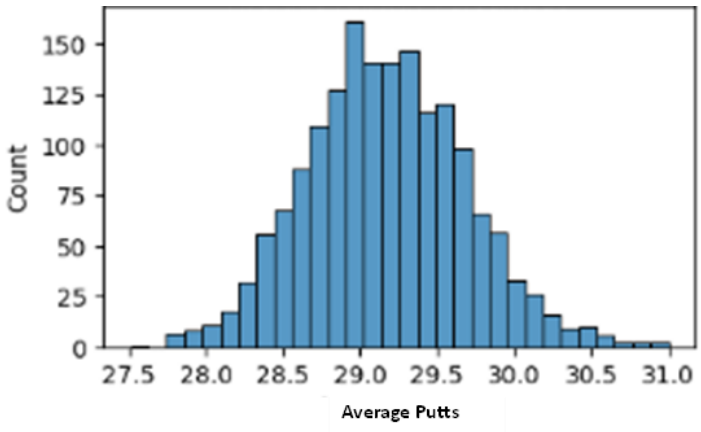


Average Putts

Average Putts has been included as it measures a player's effectiveness once on the green. Our previous metrics mentioned have all been variables describing a player long game, whereas putting average takes into account strokes made after. This is a significant part of the game as generally close to half all strokes made in a round of golf are putts for professionals.

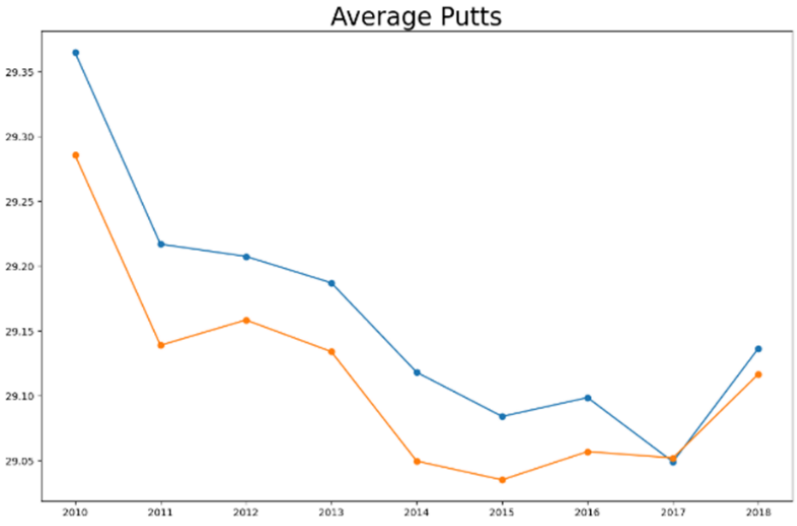
The distribution (Fig 9.) shows by average putts in the graph here also proceeds to follow the bell spread standard with similar tails to the previous graphs and no obvious skews. The better quality of play is denoted by the fewer putts taken in a round. The mean of 29.3 putts in a round being the standard for a PGA Tour professional suggests that good golfers are able to complete rounds in fewer shots made through putts than drives or approaches and therefore the requirement for excellent short game skills remains high. Players tending toward the mean will be producing more consistent results whereas the higher tail players will be unable to capitalise on chances effectively.

Fig 9. [1-E]



As we can see from the graph (Fig 10.) to the right, the players who have at least placed in the top 10 once on average perform better in putting consistently on a yearly basis. This is because putting numbers in a round are a direct contributor to end scores and the lower these values are, the better the overall performance. This tells us a great deal about the better players as they are able to manage the shorter aspects of the game significantly better and capitalise upon more birdie opportunities and recovery chances than the rest of the field. The average gap in the trends shows that better players perform 0.2 less putts per round.

Fig 10. [2-E]

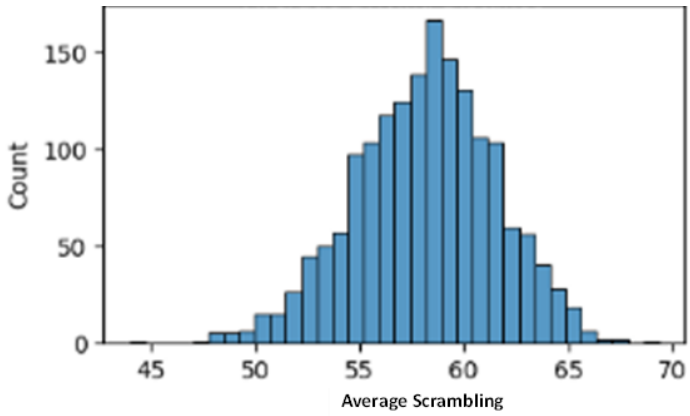


Average Scrambling

Average Scrambling is a descriptor for a player's capacity to recover scores of par or better after having missed a green in regulation. The ability to recover is directly related to lower scores and provides insight into a player's penchant for not compounding errors throughout a round of golf. The average GIR for a PGA tour player is approximately 65%, meaning a player would generally miss around six greens per round. These situations would require scrambling to salvage score and being able to hit the next shot close to the pin and attempt to whole the putt in one attempt. Although putting is already a section of the sport we have accounted for, scrambling provides a greater understanding of a player's ability to putt as a form of recovery rather than scoring opportunity (Birdies and eagles) as well as including a player's skill of adaptability to hit chip or pitches close to the hole.

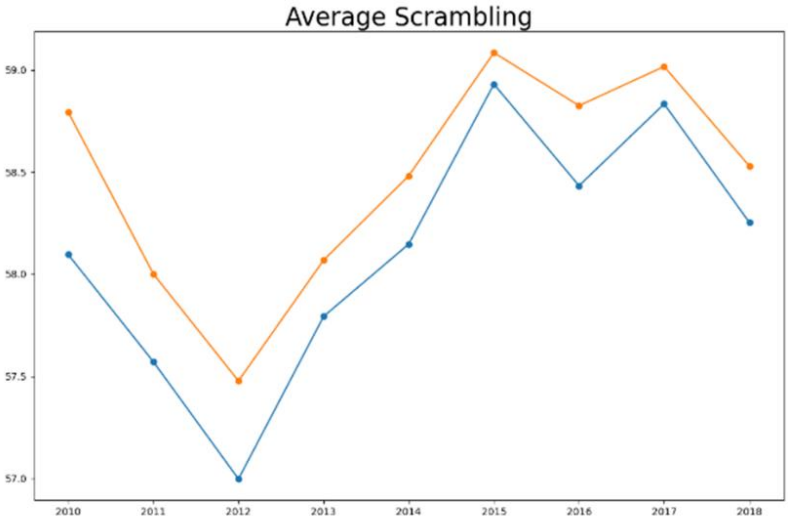
Once again we can see a bell curve distribution (Fig 11.) and this statistic has shown similar results to that of the putting. This is due to the close relations between scrambling and putting prowess. The mode average scramble percentage is 58% and this would suggest that a majority of the field are able to minimise the compounding of errors through recovery. The upper tail of this graph will be filled with better players as similarly to the putting metric, the short game plays a major role in overall performance.

Fig 11. [1-F]



The higher this percentage is, the better. This explains the trends we see on the graph (Fig 12.) to the right as generally the higher-ranking players who place in the to 10's is more proficient in recovery and short game. This disparity between the two lines on the graph is relatively consistent with the results on the putting section and maintains this roughly 0.8% gap on a yearly basis.

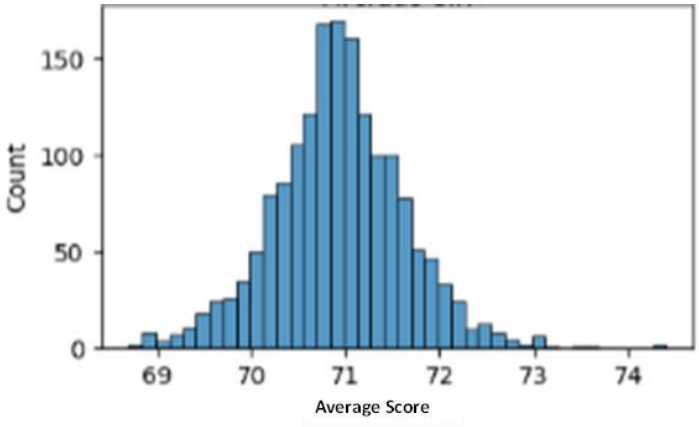
Fig 12. [2-L]



Average Score

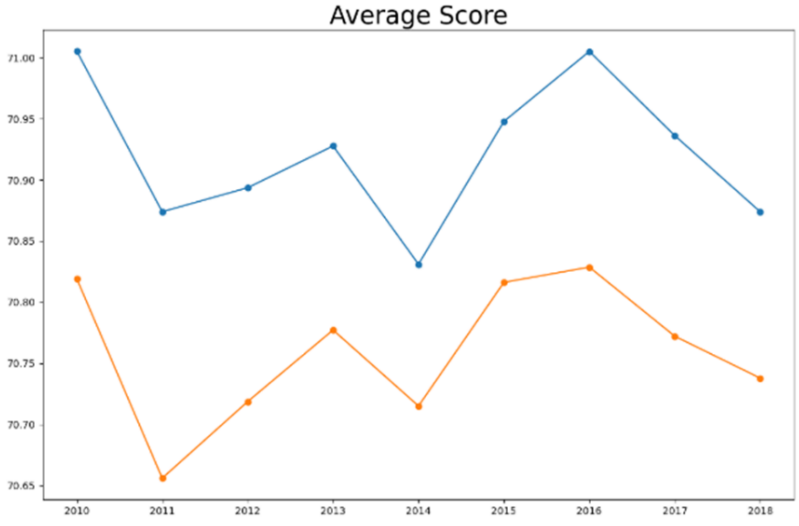
Average Score has been included in the list of features needed to train our model as lower scores would imply better rankings in tournaments. Scores in a tournaments may directly result in a finishing position; however, it is not true that necessarily lower average scores would strictly result in more top 10's as scoring standards between tournaments differ depending on the difficulty of the hosting course. Players may have lower scoring averages because they play tournaments held on easier courses. It is common for the most coveted tournaments to be held at more difficult venues ergo, players may have higher scores but still rank better in those championships as across the entire pool of players, scoring would be a depressed factor.

Fig 13. [1-G]



The distribution (Fig 13.) of average scores we would expect to see is in line with the resultant graph. PGA Tour professional should be producing under par scores on average every round which would typically mean a score of at most 72. The mean from the graph is suggesting this pattern is being met with an average and mode of 71. We would expect the dominant players to fill the counts created by the lower tail of the distribution.

Fig 14. [2-F]



Average score is the most direct metric we can use to see the effectiveness of a player's performance on a round as it directly denotes the ranking a player would achieve in a tournament. As we can see from the graph (Fig 14.) this expected trend of a significant gap between the scoring averages of the higher achieving players and the rest of the field remains prominent on a yearly basis. The mirrored pattern exhibited across the two groupings of players is an ideal result.

Strokes Gained Statistics:

Strokes Gained (SG) statistics are a measure of player performance in isolated aspects of their play relative to the average of their competitors. Since ranking in a tournament is dependent on one's performance compared to others, using strokes gained metrics give us insight into the effectiveness of a particular skill, and the relative gain garnered by the individual player from it against the rest of the field. I have recognised the following factors of play as of import: Average SG Putts, Average SG: OTT (Off the Tee), Average SG: APR (Approach) and Average SG: ARG (Around the Green).

Strokes Gained OTT

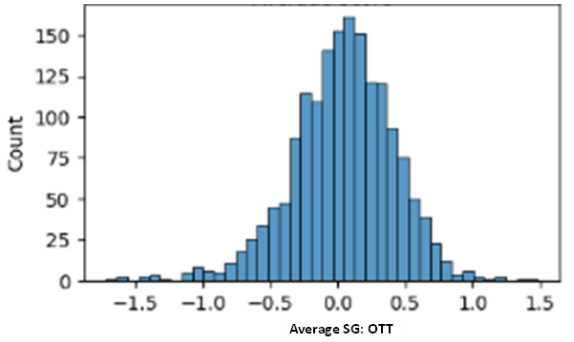


Fig 15

[1-J]

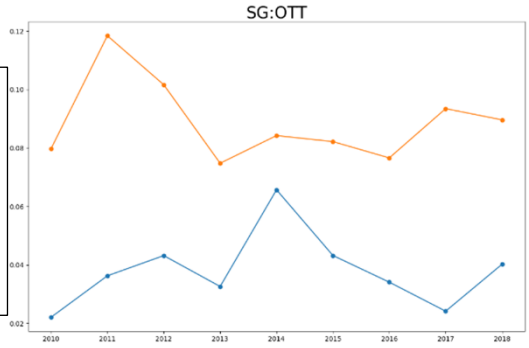


Fig 16.

[2-J]

Strokes Gained APR

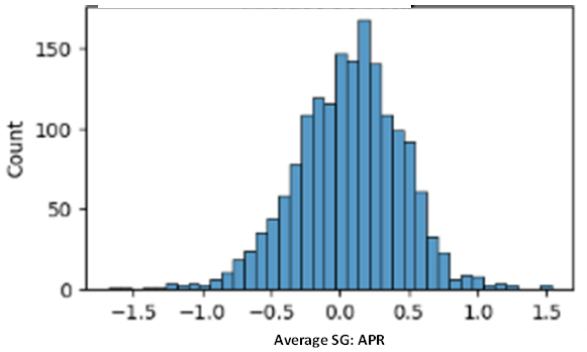


Fig 17.

[1-K]

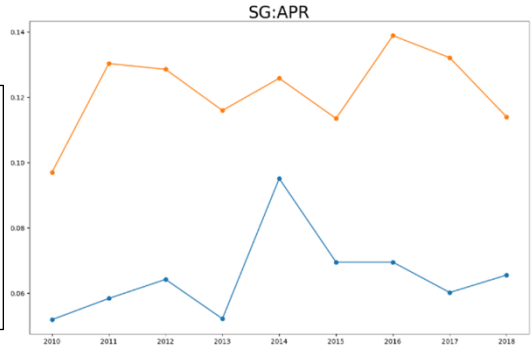


Fig 18.

[2-I]

Strokes Gained ARG

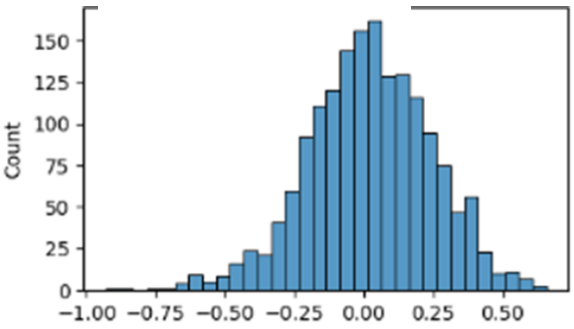


Fig 19.

[1-L]

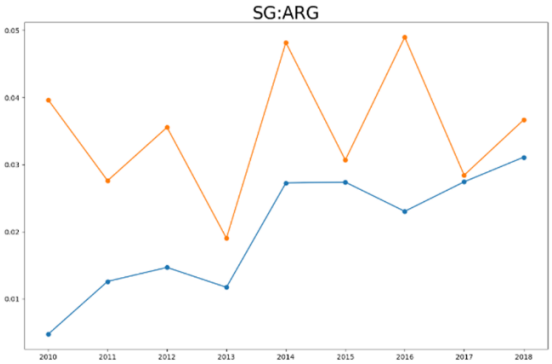


Fig 20.

[2-K]

Since strokes gained is a comparative metric to the average, most of the stats which show standard normal distribution about the mean of zero, zero indicating a proficiency in that skill exactly equal to the average. Players with greater strokes gained (positive) in any category would learn they are picking up scores against the average as a result of their better average ability in that category. The opposite is true for negative values. The line graphs above show this trend across all Strokes gained metrics which all denote the importance of their part they play in a professional players skill set in determining the number of top 10's they would get in a year.

Strokes Gained Total

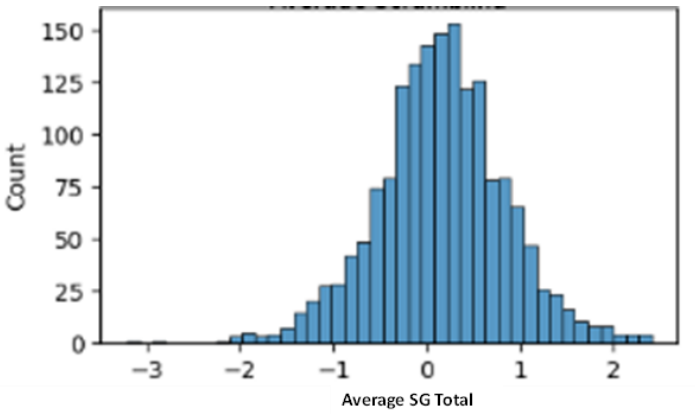


Fig 21. [1-I]

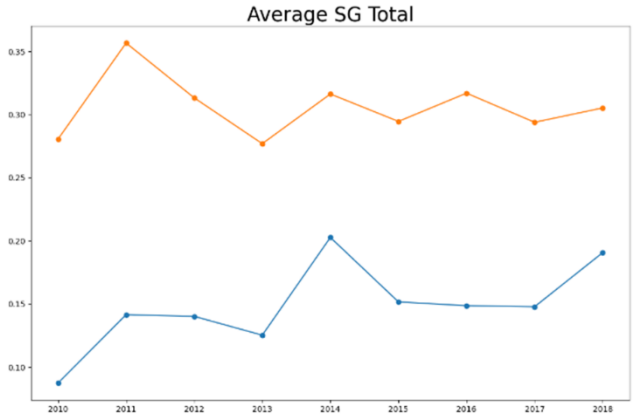


Fig 22. [2-G]

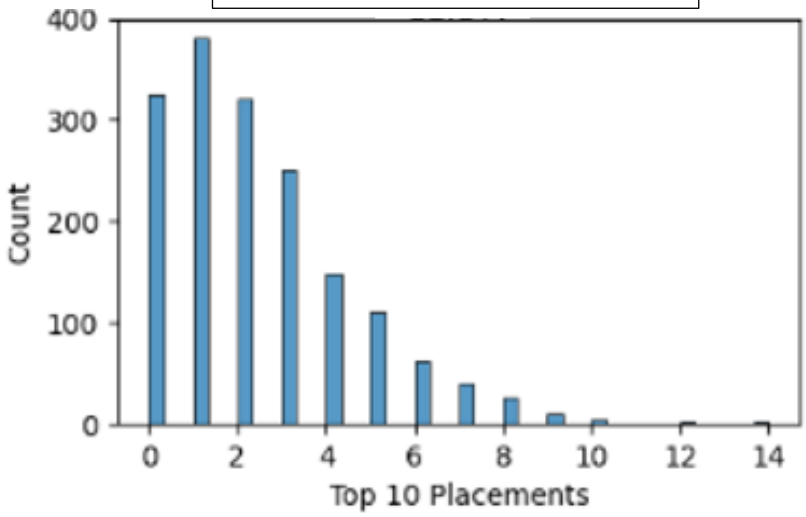
A similar logic applies to the metric of Strokes gained total as mentioned in the previous graphs analysis. As we can see from the graphs above, the strokes gained total difference between the higher performing players and the rest of field maintains a constant 0.15 differential and this is expected based on the direct relation this statistic has to the average score. The graph should have a nearly identical distribution and line separations as the in previously covered in Average Score.

Label/Prediction Value: Number of Top 10 Placements

I have selected Top 10 Placements as my metric for player performance. I have chosen this because it is a better indicator overall than wins or average finishing position. With regards to wins, I believe top 10's is a better value to utilise as golf as a sport does not necessitate the winning by the best players as consistently and repeatedly as can be observed in other sports. This is a result of the structure of the competitive landscape where tournaments may have fields up to 200 player all competing with each other simultaneously, unlike many other sports that are devised on a one-on-one competitive basis where far more often, the better player would be expected to win most of the time. Relating to the decision to choose top 10's rather than average finishing position, I felt that average finishing position was lacking in its ability to truly capture a player performance in a season as it is very sensitive to the volatility of a players scoring week to week which can be affected strongly by factors such as golf course style, course difficulty, weather conditions etc. where worse performances would drastically drag down the average but could potentially negate the impact of good finishes. This would therefore make using average ranking a poor judge of a golfer's proficiency as players who are consistently high on leatherboards could have unimpressive average placement despite performing well most of the time. This is more in line with the way golf performance is viewed as far more weight is placed on good finishes than a player's worst ones.

In light of these issues with the previously discussed monikers for performance I selected top 10's as I felt it had greater leeway in identifying good play and allows us to ignore very poor finishes with a greater focus on the weeks where they held a higher playing standard.

Fig 23. [1-M]



The distribution above (Fig 23.) has a positive skew and rightfully depicts the mean number of top 10 finishes for the average player being around 2. These players are middle ranking and make up a majority of the mean statistics reviewed prior. We can also see that there are a few outlier players every season that exceed 5 or more top 10 placements. This tail end of the distribution would be filled with the players expected to be ranked in the upper echelons of the sport and are triumphant in maintaining consistently high standards of play through all the metrics discussed above.

As the desired metric for our predictions and requirements, the distribution of this graph makes an excellent case for the need to derive top 10 placements as a metric for evaluating the success of golfers.

5.2: Initial Neural Network Model Results:

The following graphs were derived from the evaluative metrics produced from the initially built model before any optimisation or tuning.

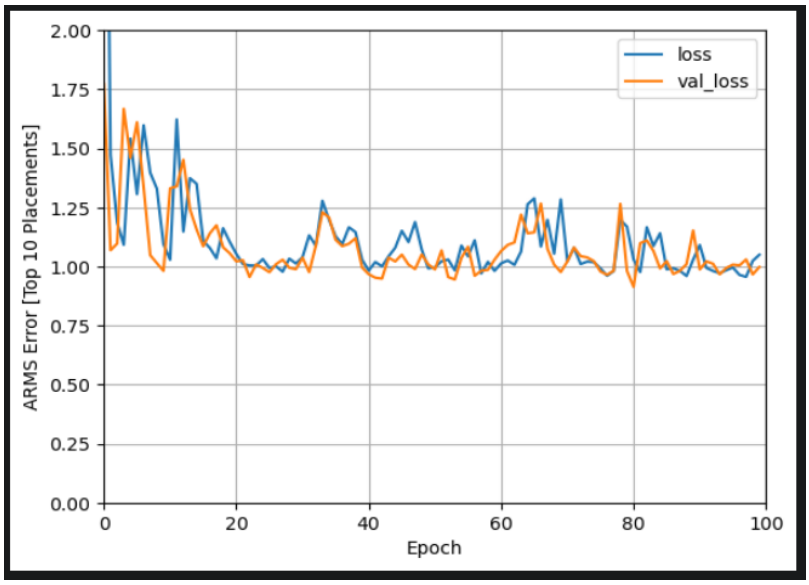


Fig 24.

[4-A]

The graph (Fig 24.) seen above has been used to show how the Absolute Root Mean Square error of the initial model varies through each iterative pass of the training data over a total of 100 epochs. The model seems to be producing somewhat erratic results with continuous peaks and troughs throughout. On the other hand, the initial trend up to 25 epochs suggests that the more training passes is resulting in fewer losses and more learning. Later results after the 25-epoch mark suggest overfitting to the training data and the need for better model optimisation. Overall, an expected result to be seen from an initial

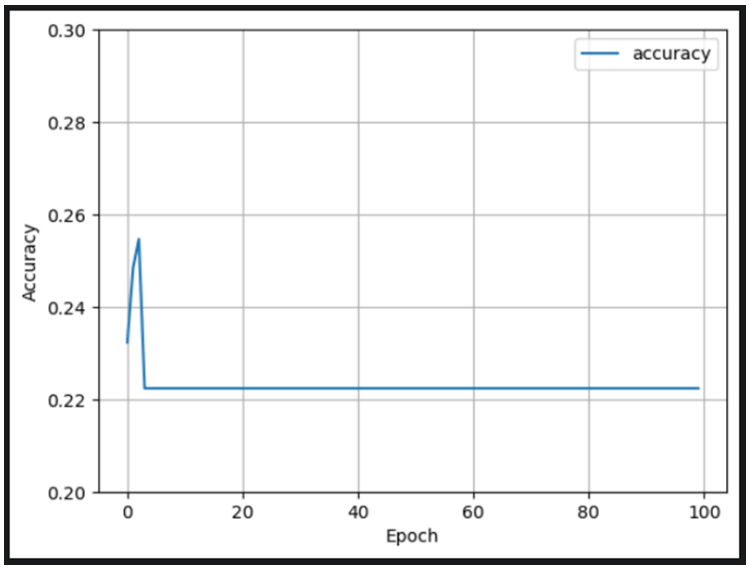


Fig 25.

[4-B]

The Graph (Fig 25.) above depicts our initial models changes in accuracy through each iterative pass of the training data over a total of 100 epochs. The model's accuracy shows great promise initially as we can see a spike in accuracy from 0.23 to 0.255 accuracy after only a few passes through the training data; however, this is short lived and the model proceeds to regress after a few more epochs. The model beyond 4 epochs is unresponsive and proceeds to maintain around a 0.22 accuracy level. This is a poor performance, even for a prototype model and currently falls well under the project's objectives.

The following graphical evaluation for our model is depicting the calculated prediction values our model has generated, compared to the true value presented in our testing data set. The more values tightly spread closer to the y=x line would indicate a well performing set of predictions.

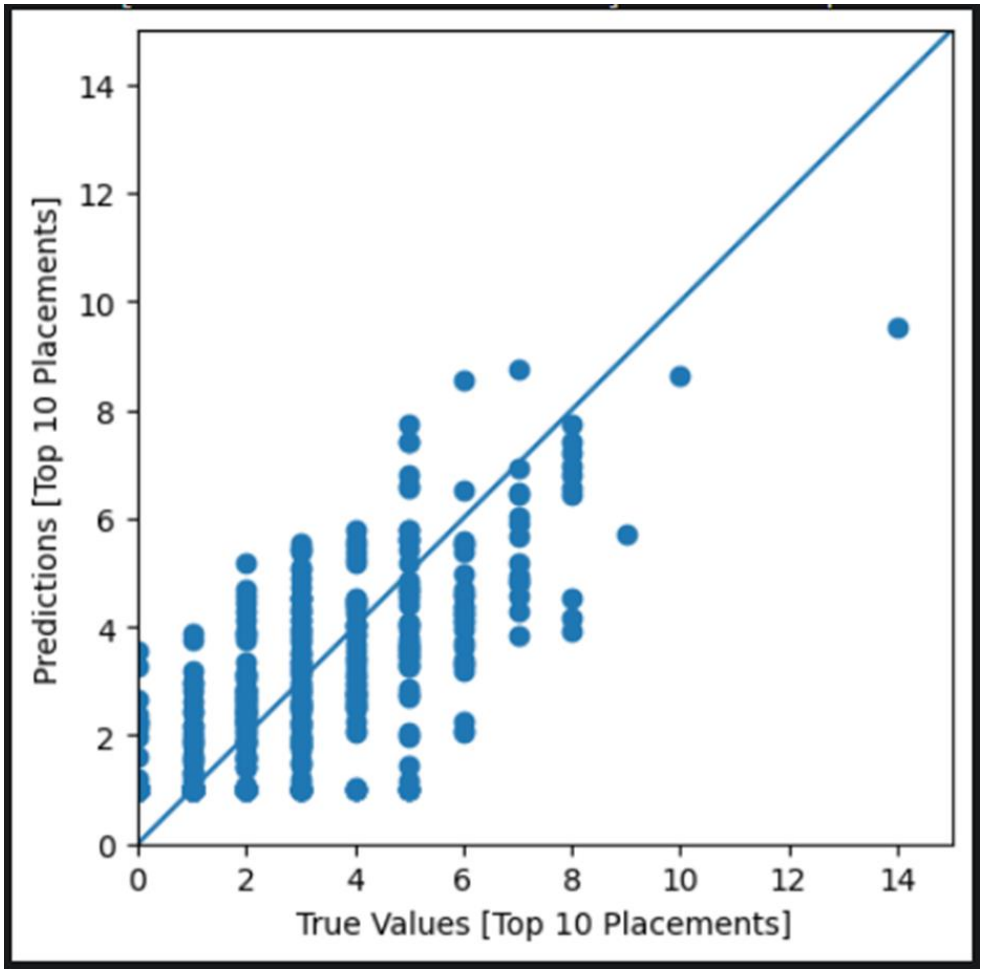


Fig 26.

[4-C]

As we can see from figure 26. above, the initial model does indicate a level of capacity for learning as the data from the graph above depicts a general trend towards showing a linear relationship between our predicted and true values. Interestingly we can take note of the fact that generally the model seems to understand the capabilities of upper echelon players with more precision and is able to distinguish the fact that their above average features denote a high number of Top 10 Placements. This can be seen by the plots closer to the top of the graph being relatively close between the predicted and true values. A clear drawback of the initial model is the sporadic performance in predicting the players between 2 and 4 true Top 10 Placements. There are a great number of plots that fall close to the ideal line; however, the overall spread suggests the underperformance in predictive ability here. This is most likely due to the high density of players who get 2-4 top 10 placements a year with similar feature values falling in this section, skewing the training, and thus impacting the model's ability to make accurate assumptions in these areas. For an initial model the resultant graphs show a steady but currently level of learning and promising progress towards meeting our end requirements.

5.3: Hyper Parameter Tuning Results:

This section explains some of the coding decisions and results that came from the Grid Search hyper parameter tuning that was mentioned before in the Methodology.

Fig 27. [Coding Block 19]

```
# Set up the hyperparameter grid for hyper parameter optimization using a grid search technique.
param_grid = {
#Decided to iterate through the following parameters using a dictionary array system.
    'learning_rate': [0.001, 0.01, 0.1],
    'batch_size': [100, 200],
    'epochs': [50, 100],
    'optimizer': ['Adam','rmsprop'],
    'hidden_layer_size':[64]
}
```

The code (Fig 27.) above was written to create a dictionary with the keys being the various hyper parameters used in my grid search and the arrays containing the values I would exhaustively run through in all combinations to derive the best possible amalgamation to give an optimal model. This process would take a great deal of time and processing power as each parameter and the several values applied would increase the number of iterations exponentially.

Fig 28. [Coding Block 20]

```
#Rebuilding a model for the hyper parameter tuning
#Could be made more effiecient but this allows me to easily see when each of my models is being made
#A big difference in this model is that it wil be taking its parameters from the param_grid made above and iterating through them.
def build_and_compile_model2(learning_rate, batch_size, epochs, optimizer,hidden_layer_size ):
    model2 = keras.Sequential([
        layers.Dense(hidden_layer_size, activation='relu'),
        layers.Dense(hidden_layer_size, activation='relu'),
        layers.Dense(1)
    ])

    optimizer = tf.keras.optimizers.get(optimizer)
    optimizer.learning_rate= learning_rate
    model2.compile(loss='mean_absolute_error', metrics='accuracy', optimizer=optimizer)

    return model2
#This model simmilar to the initally built one was built with the aid of (Basic regression | TensorFlow Core, 2023).
```

The following code (Fig 28.) applied shows the hyper parameters mentioned above being used as parameters for the building of the new models that will be iterated through using SKLearn GridSearchCV.

Fig 29. Grid Search Tuning V2 [5-B]

```
11/11 [==============================] - 0s 4ms/step - loss: 1.2180 - accuracy: 0.2771 - val_loss: 1.2708 - val_accuracy: 0.2265
Epoch 100/100
11/11 [==============================] - 0s 4ms/step - loss: 1.2035 - accuracy: 0.2800 - val_loss: 1.0938 - val_accuracy: 0.2444
Best hyperparameters:  {'batch_size': 100, 'epochs': 100, 'hidden_layer_size': 64, 'learning_rate': 0.01, 'optimizer': 'Adam'}
Best mean validation score:  -1.3847172260284424
```

The results of the grid search hyper parameter tuning have given us a detailed summary of the best assortment of values to use for the different parameters. These parameters produced a mean validation score of -1.3, suggesting a good composition was found that should improve upon our existing model. Typically, a significant change in the evaluative metrics between the second to last and last epoch would suggest overfitting of the model to the training data. This does not seem to be prevalent which bodes well for the success in our new set of tuned parameters to create a better, more optimised model.

5.4: Tuned Neural Network Models:

The following graphs were derived from the evaluative metrics produced from the iteratively built models after optimisation and hyper parameter tuning. The best parameter tuning values derived were used to generate the following results.

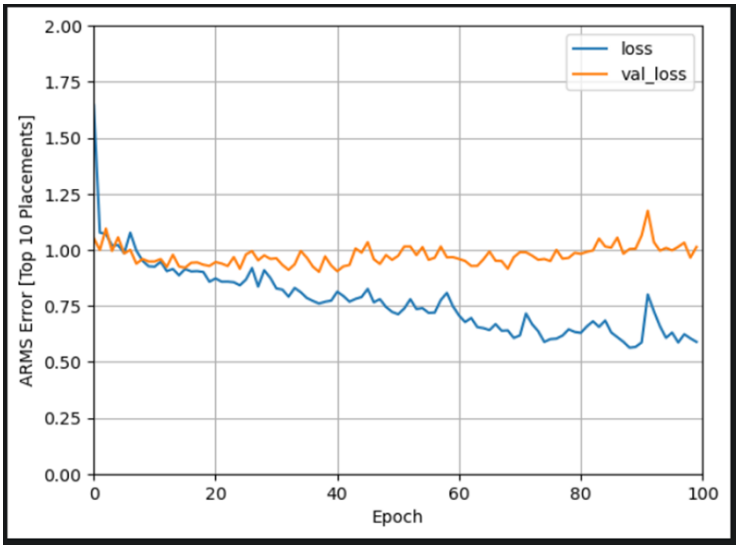


Fig 30.

[4-D]

The graph (Fig 30.) shown above displays the newly refined models Absolute root mean square error changes across the 100 epochs we set in our hyper parameters. This model has a much better shape and tells us a great deal more about the learning undertaken over time of our final build compared to the initial graph. Ideally we would see that both lines decrease as the epochs increase, this is prevalent until around 40 epochs. The continuous descent of the blue line shows promising learning as the losses are decreasing over time and the trend suggests this to continue with further training. The negative side of this graph shows after the 40 epochs with the rise in losses of our orange line suggesting some overfitting of the model to the training data. Overall, a significant improvement compared to the initial design due to the successful tuning of parameters.

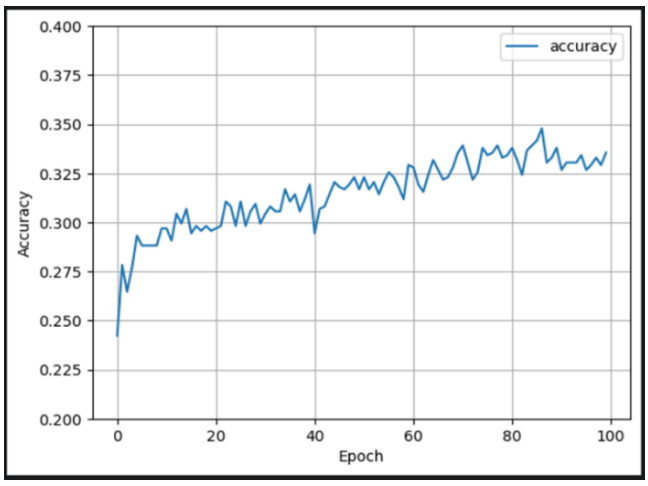


Fig 31.

[4-E]

From figure 31. we can see a clear improvement in accuracy and the increasing trend over epochs caused by the optimisation of hyper parameters. This graph stands in stark contrast to the one previously looked at in figure (). The sudden jump from 0.24 to 0.27 and the then gradual rise to a peak of 0.35 elicits optimism in the future for further training of the model and the overall success in building a competent neural network.

The following graphical evaluation (Fig 32.) for our newly refined model is depicting the determined prediction assessments our model has generated, compared to the true values depicted in our testing data set. The more values tightly spread closer to the y=x line would indicate a well performing model and good set of predictions.

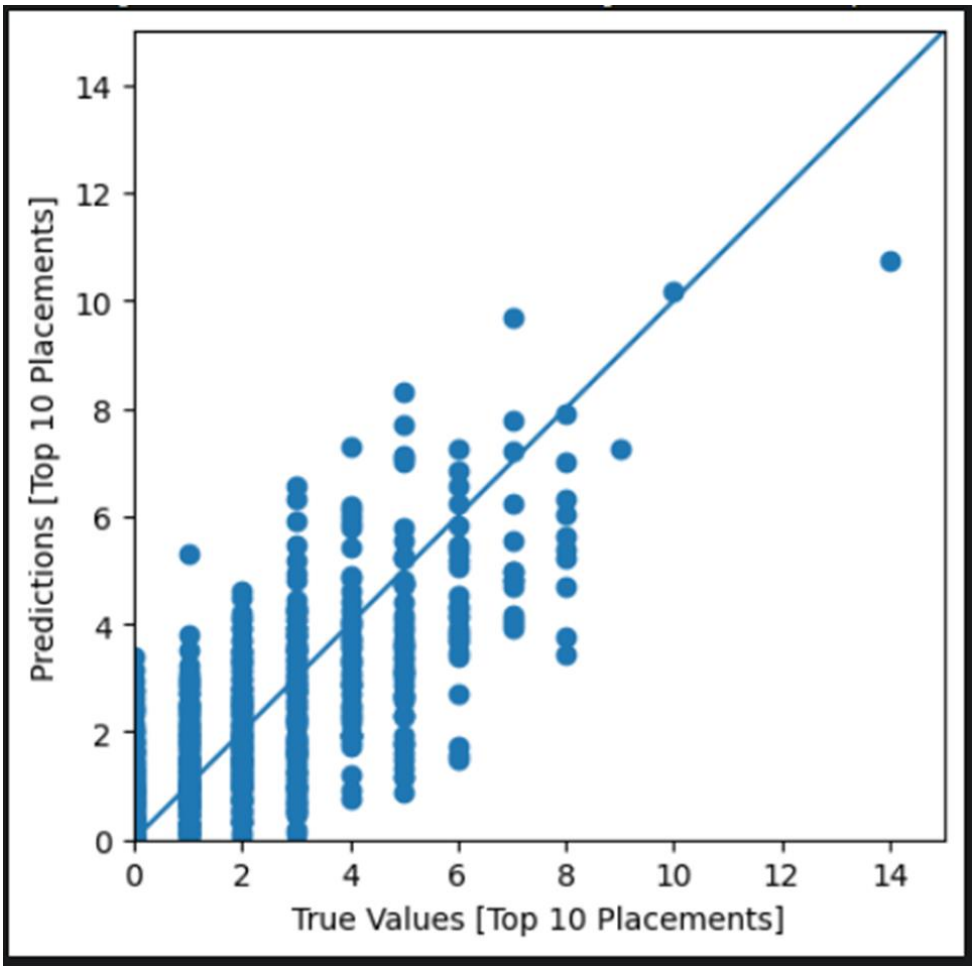


Fig 32.

[4-F]

As we can see from the graph (Fig 32.) above, the final version of our model is a much-improved version to our original. The ability for the neural network to distinguish the top players has drastically increased and the overall spread of plot points has shrunk to being closer to the optimal y=x line. The final model still has the tendency to undervalue the performance metrics of players; however, the previous accuracy and error graphs suggest that further training will help to rectify these inequalities. Furthermore, the vast number of players who fall within the 2-6 top 10 placements in a year section of the graph seem to now have greater densities closer to the ideal points marked by the straight line. Overall, this graph supplemented by the information gained from the previous two graphs have implied the need for further training across more epochs and are showing a steady improvement across all evaluative statistics with time. This is an ideal result and supports the requirements set during the project objectives.

5.5: Tuned Neural Network Model Testing:

The following results are comprised of the custom test data entered into our completed final model to see how our model would fair in predicting Top 10 values for players in more recent years outside of the testing data's scope. A custom player derived from the mean values in each field has been used for one test and the other is based off the stats produced by the 2022 world number 1 ranked player 'John Rahm'.

Fig 33. [Coding Block 27]

```
#Made an array of values for a made up player with average PGA Tour Pro statistics just to see whether my model can give me a reasonable prediction.
Prediction_data = np.array([[2020, 80, 65, 315, 62, 30, 55, 72, 0.2, 0.41,-0.1,0.1,0.4]])
#Make predictions on the new data using the trained model
predictions = dnn_model3.predict(Prediction_data)
#Print the predicted values for the number of times a player is predicted to place in the top 10 for tournaments across the year.
print(predictions)
```

```
1/1 [==============================] - 0s 19ms/step
[[1.4040424]]
```

The mean player statistics for each feature generated a prediction of 1.4 Top 10 placements in a year. This value created is currently underappreciating the aptitude of the average professional players performance as we would expect this number to be around 2.1. That being said the model is not far off and has been able to interpret the playing statistics somewhat successfully into predicting a number in the generally acceptable range that falls in line with eth accuracy and error results provided in graphs () and ().

Fig 34. [Coding Block 28]

```
#I thought it might be interesting to expermient with the model to see how it would interprate newer data from the current world No.1 player Jon Rah
#Bearing in mind the actual number of top 10 Placements he had in 2022 was 8.
Prediction_dataJonRahm = np.array([[2022, 70, 65, 321, 72, 29, 57, 69.7, 0.367, 1.66,1.025,0.363,-0.8]])
predictionsJonRahm = dnn_model3.predict(Prediction_dataJonRahm)
print(predictionsJonRahm)
#This result seems to be underestimating the performance, further analysis of this will be studied in full research paper.
```

```
1/1 [==============================] - 0s 17ms/step
[[3.9107678]]
```

The outstanding year performance produced by John Rahm in 2022 created a data entry for a player performing to a high standard, thus resulting in the model's prediction of 3.9 Top 10 Placements in a year. Given that John Rahm's actual number for that year was 8 follows the pattern of underestimating player performance previously discussed throughout my results. The value 3.9 still suggests that this player is performing at double the performance of the average and therefore showing the models ability to comprehend the difference in placement consistency for the top end of players compared to the rest of the field.

Overall, the results produced have been insightful and fall within the expectations of our project objectives, I will look to evaluate the extent of success towards meeting the project requirements in the next section.

Chapter 6: Conclusions and Evaluations:

6.1: Project Objectives Review

This section looks at reviewing the projects objectives against the results produced and discussing the extent to which they were fulfilled and the reasoning behind any shortcomings. My objectives remained relatively consistent throughout the project and the planning phase as I was able to develop the necessary models and documents within the time constraints. All objectives were previously mentioned in the introductory chapter and PDD which can be found in the Appendices A.

Objective	Review
1.) This project shall develop and AI model using Neural Networks that will be able to predict an effective metric for player performance valuation.	This objective has been met as the project results and code package file show the creation of several neural network models to predict the number of top 10 placements a PGA Tour players will achieve based on their yearly stats. The direct outcomes contributing to the success of this objective can be found in Appendices [4-A to 4-F] and the code in the Appendices coding blocks [13 to 26].
1.a) The performance valuation will be able to generate predictions when tested to have an accuracy of at least 0.5 based on the professional player test data.	This objective has been partially met as our final model was able to reach a peak accuracy of 0.36. This was after an initial run of hyper parameter tuning for optimisation. The trend from the graphs (see Appendices [4-E] or the results analysis) in the results suggest that with further epochs (iterations of the training data run through) we would see this number reach the desired 0.5 given more time and processing power. The limitations of my personal computer played a large factor in the shortcomings for this objective.
1.b) The predictive model will be able to be used as insight into the potential performance of players outside of the training datasets scope and could apply to players from other tours or levels of the sport (College Golf, Olympic Golf)	This objective was completed to a desirable degree as the final model built offered the ability to project predictions for unseen data from various custom sources. The data trialled (See Appendices [Coding Block 27 and 28 and results chapter]) on our model came from a custom created player comprised of the mean statistics and another was tested by using the stats of the 2022 world number one ranked player. Both results fell within the expected accuracies our evaluative models indicated and were in line with the skill sets of the players they were replicating.
1.c) The model developed will be subject to evaluation and improvement through comparison and optimisation.	This objective was met to a high standard as the Grid search hyper parameter tuning and evaluation of several models as mentioned in the methodology and shown in results and Appendices (see in Appendices [5-A and 5-B]) created a drastically improved model as we saw the Absolute Means Squared error and accuracy metrics improve with each epoch, and the future trend suggested this improvement to continue.
2.) This project shall research into the effective measures of value a player is deemed to have and to what effect features will determine the extent of the success of players.	This objective has been successfully met through the large amount of research conducted through the literature review and the analysis of several golf related papers and articles. The features determining the success of players has been studied greatly in the results analysis and methodology sections of the paper.
2.a) derive an effective understanding for performance value based on statistical research and	The thorough analysis of the PGA Tour Data set conducted through the results section of the paper and the creation of many graphs (can be found in results and Appendices [1-A to 3-B]) to

data analysis of the PGA Tour golfer data collection. Analysis should extensively cover a wide variety of performance statistics.	support this has led to tremendous insight regarding the nature of statistics in golf. Several performance statistics have been looked at and scrutinised in depth thus supporting the requirements of this objective being met completely.
3.) This project shall streamline the indecision process of selecting golfers for sponsorships and tournament planning by giving insight into the factors that make a good golfer and how this can be measured.	This objective has been met to varying degrees of success dependant on the projects usage to the various beneficiaries mentioned in the introduction. I can confidently say that the research and analysis into professional golf data has provided a deep exploration into the use case for statistics in the sport and the importance of the many variables covered. To golfers and coaches, this analysis would be invaluable and hopefully shed light on areas of the game that make major impacts on consistent overall performance. Furthermore, the neural network models developed can be of great use to sponsors in assisting in critical decision making. The extent to this utility in its current state would need to further improve to be considered a reliable indicator for sponsors to make investments based upon the numbers my model generates. Further training and optimisation could help with this in the future.
3.a) This measure of effectiveness of this projects goals towards helping indecision can be gauged by how well the models perform and the time saved in using this new method.	The model most definitely streamlines the process of evaluating player performance using yearly statistics. The project as a whole supports the users decision making process and will hopefully provide cutting edge insight to those looking to value golfers.

6.2: Project impact and effect on the field's literature

I believe that my project has provided benefits to the field of data science in sports, with particular emphasis on golf and the need for exploratory data analysis and machine learning. The project took a great deal of inspiration from several studies regarding the evaluation of player performances across a variety of sports as shown in my literature review. I believe that I have successfully built upon the works (Prater 2018) and (Jong 2019) started and provided a necessary contribution to the study of data in golf through the complex analysis of PGA data and the development of several neural network models to predict the success of PGA Tour golfers.

The realm of possibilities with the growth in artificial intelligence and rapid advancements in technology has made the work produced in this field to be in higher demand and the study of data and machine learning that I found in my literature review to be of great significance. I believe my project offers a good use case for the necessity of machine learning in modern sports.

6.3: Project Evaluations (Areas to improve upon in future developments and potential missteps to avoid if project is ever replicated)

There are several areas of my project, with the benefit of hindsight, that I would seek to improve upon when developing new projects or replicating this one. A better sense of planning with regard to the formation of my end dataset would have provided the project with more variety in player data and more opportunity for my models to learn. The dataset I used was limited to a single tour and I believe that my results would have been more successful if data from LIVgolf Tour or the DP World Tour had been included. Furthermore, the use of better hardware and time management to train my models through hyper parameter tuning would have been of significant benefit. I underestimated the processing times required for my tuning and was therefore unable to exhaustively test out all the parameters I wanted. I believe with this in mind, future projects and this one will be able to reach their true potential. Finally, the need to build a concrete set of objectives early on with tangible values for checking progress will help to consistently keep track of development and ensure that requirements are not being forgotten.

6.4: Final Conclusions (Personal Lessons Learned and Challenges Faced)

Overall, this project has been a massive learning experience and has deeply affected the way I look at and approach computational problems. I have made great personal development in several key skills such as project management, research, and professionalism. The project has forced me to carefully manage deadlines and plan/structure work well ahead of time. From a more technical perspective the project has opened my eye to learning more about machine learning and coding in Python as this field and all of its use cases and capabilities has captured my attention and I now look to conduct further studies along this path. I look forward to seeing how this field develops and how I can continue to contribute to it.

Chapter 7: Glossary:

I recognise the wide variety of golfing terminology that this essay will require me to use and have therefore created this glossary to assist any readers in minimising confusion and building a better understanding for the meaning behind the more technical terms used within the sport.

7.1: Golf Terms Definitions:

Par- The appropriate number of strokes taken on a hole to be deemed an even score, consisting of par 3, 4 and 5 depending on the length and difficulty of hole. Secondary definition of par is taking the equivalent number of strokes to the par of the hole, colloquially “making a par”. (Even)

Birdie- A recorded score on a hole one fewer than the stated par (-1)

Eagle- Recorded score on a hole of two strokes fewer than the stated par (-2)

Bogey- Recorded score on a hole of one stroke more than the stated par (+1) Further higher scores are referred to as their multiple over a standard bogey e.g., Double Bogey (+2) is two strokes above par, triple bogey, quadruple bogey etc.

Tee- The designated area to make your opening stroke on any particular hole. Also known as the teeing area. Secondary definition is a wooden peg upon which the ball sits that can be employed only when striking from the teeing area.

Fairway- The area of short grass between the tee and the green

Green- The designated putting zone where the hole is situated. Also referred to as” putting surface”.

Fringe- The slightly longer length of grass about the periphery of the green, generally of the same length or slightly shorter than the fairway.

Rough- The penalising longer grasses beyond the limits of the fairway line.

Handicap- The average score relative to par of a golfer.

Scrambling- The ability to play a shot and hole a putt from the immediate surround of the green.

Score- The cumulative number of shots taken in a full round of golf. This can also be expressed as a value relative to par. For example, ‘+5’ if the total number of strokes taken is 5 above the summation all the par scores on every hole.

Driving- The act of hitting a driver from the tee.

Putting- The act of rolling the ball on the green.

Round- Refers to the singular completion of all hole son a particular course. Can also be used to indicate which attempt around the golf course the player is currently or was playing i.e., “Player X is currently completed his third round”

Strokes Gained (SG)- The number of shots fewer a player scored than the average of the field.

Strokes Gained Approach (SG: APR)- The number of shots fewer a player scored than the average of the field from only their approach play.

Strokes Gained OF THE TEE (SG: OTT)- The number of shots fewer a player scored than the average of the field from only their tee shots.

Strokes Gained Around Green (SG: ARG)- The number of shots fewer a player scored than the average of the field with shots played within close proximity to the green.

Strokes Gained Putts (SG: P)- The number of shots fewer a player scored than the average of the field from their putting.

Strokes Gained Total (SG: Total)- The number of shots fewer a player scored than the average of the field over the entire round of golf.

Green In Regulation (GIR)- The quantification of the expected number of strokes to hit the green, equivalent to the par of the hole subtract two.

PGA- Professional Golf Association.

R&A- Royal and Ancient.

FedEx Points- The season long point system for ranking payer performance on the PGA Tour, eponymous of its main sponsor.

Earnings- The total financial gain of a player. (More often referring only to prize money).

Major- Phrase referring to the four tournaments of the greatest import in a season. These are The Masters, British Open, US Open and PGA Championship.

7.2: Coding/ Machine Learning Definitions:

Machine Learning- Computer systems that can learn without following strict algorithms.

Neural Network- Computer system modelled on the human brain.

Hyper Parameter- Variables effecting the learning process of a model.

Model- An algorithm based on a data set.

Feature- Variables the neural network will learn from.

Label- The correct description for sample that usually appears in the test data.

Chapter 8: Reference List (Harvard Style):

8.1: Write-up References:

The following references are from various works ranging from research papers, articles, and published books. All of the works below supported the development of my project and have been cited below using Harvard referencing style.

Hebb, D.O. (2002) *The organization of behaviour: A neuropsychological theory*. Mahwah, NJ: Erlbaum, Lawrence, Associates.

Pineau, J. et all (2021) *Improving reproducibility in Machine Learning Research (a report from the neurips 2019 reproducibility program)*, *The Journal of Machine Learning Research*. Available at: https://dl.acm.org/doi/10.5555/3546258.3546422 (Accessed: February 20, 2023).

McCulloch, W.S. (1990) *A logical calculus of the ideas immanent in nervous activity*, *A LOGICAL CALCULUS OF THE IDEAS IMMANENT IN NERVOUS ACTIVITY*. Available at: https://www.cs.cmu.edu/~./epxing/Class/10715/reading/McCulloch.and.Pitts.pdf (Accessed: February 26, 2023).

Baker, M. (2016) *IS THERE A REPRODUCIBILITY CRISIS?*, *A Nature survey lifts the lid on how researchers view the 'crisis' rocking science and what they think will help*. Available at: https://www.icts.uci.edu/education/ffast1/nature.pdf (Accessed: February 26, 2023).

Nassif , A.B. (2019) *Speech recognition using Deep Neural Networks: A systematic review* . Available at: https://ieeexplore.ieee.org/document/8632885 (Accessed: March 2, 2023).

Nasteski, V. (2017) *An overview of the supervised machine learning methods* . ResearchGate. Available at: https://www.researchgate.net/profile/Vladimir-Nasteski/publication/328146111_An_overview_of_the_supervised_machine_learning_methods/links/5c1025194585157ac1bba147/An-overview-of-the-supervised-machine-learning-methods.pdf (Accessed: March 2, 2023).

Bishop, C.M. (2009) *Pattern recognition and machine learning*. New York, NY: Springer Verlag.

Cook, E. (1966) *Percentage baseball*. Cambridge: M.I.T. Press.

Brooks, J. and Kerr, M. et all (2016) *Developing a data-driven player ranking in soccer using predictive Modelling*. Available at: https://dl.acm.org/doi/pdf/10.1145/2939672.2939695 (Accessed: March 10, 2023).

Pappalardo, L.P. and Cintia et all, P. (2019) *PlayeRank: Data-Driven Performance Evaluation and player ranking in soccer via a machine learning approach: ACM Transactions on Intelligent Systems and Technology: Vol 10, no 5*, *ACM Transactions on Intelligent Systems and Technology*. Available at: https://dl.acm.org/doi/10.1145/3343172 (Accessed: March 10, 2023).

Grycmann, P. and Maszczyk, A. et all (2015) *Modelling analysis and prediction of women javelin throw results in the years 1946 - 2013*, *Biology of sport*. U.S. National Library of Medicine. Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5394849/ (Accessed: March 11, 2023).

Park, J. (2019) *Can We Predict If a PGA Tour Player Won a Tournament in That Year and Their Earnings?*, *PGA Tour machine learning project*. Kaggle. Available at:

https://www.kaggle.com/code/jmpark746/pga-tour-machine-learning-project/notebook (Accessed: March 14, 2023).

Prater, D (2018) *PGA-tour-data-science-Project*, *PGA Tour machine learning project*. GitHub. Available at: https://github.com/daronprater/PGA-Tour-Data-Science-Project/blob/master/PGA%20Tour%20Machine%20Learning%20Project%20-%20Classification.ipynb (Accessed: March 14, 2023).

Corcoran, M. (2019) *Wise guys: Data golf is taking analytics to a whole new level (pay attention, gamblers)*, *Golf*. Available at: https://golf.com/news/features/data-golf-analytics-new-level-pay-attention-gamblers/ (Accessed: March 15, 2023).

Arastey, G.M. (2020) *The increasing presence of data analytics in golf*, *Sport Performance Analysis*. Available at: https://www.sportperformanceanalysis.com/article/increasing-presence-of-data-analytics-in-golf (Accessed: March 15, 2023).

Alhamdan, W. and Howe, J.M. (2021) Classification of date fruits in a controlled environment using convolutional neural networks, Springer. Available at: https://openaccess.city.ac.uk/id/eprint/25158/ (Accessed: April 1, 2023).

8.2: Code references:

Code has either been used directly from the following places or adapted/learnt from to support my coding needs. The following are the references for all snippets of code that supported the creation of my end products.

(Prater, 2018) Prater, D. (2018) 'daronprater/PGA-Tour-Data-Science-Project'. Available at: https://github.com/daronprater/PGA-Tour-Data-Science-Project/blob/85a58ca806bf2ba2deb74b7f96471f5fcb36a551/PGA%20Tour%20Machine%20Learning%20Project%20-%20Classification.ipynb (Accessed: 10th March 2023).

(Jong, 2019) Jong (2019) PGA Tour Machine Learning Project (2019). Available at: https://kaggle.com/code/jmpark746/pga-tour-machine-learning-project (Accessed: 18th March 2023).

(Stewart PhD, 2023) Mathew Stewart PhD, M.S. (2023) Simple Guide to Hyperparameter Tuning in Neural Networks, Medium. Available at: https://towardsdatascience.com/simple-guide-to-hyperparameter-tuning-in-neural-networks-3fe03dad8594 (Accessed: 10th April 2023).

(Basic regression | TensorFlow Core, 2023) Basic regression: Predict fuel efficiency | TensorFlow Core (2023) TensorFlow. Available at: https://www.tensorflow.org/tutorials/keras/regression (Accessed: 29th March 2023).

(Harvpan, 2018) Harvpan (2018) 'Answer to "Change the regression line of Seaborn's pairplot"', Stack Overflow. Available at: https://stackoverflow.com/a/50724511 (Accessed: 20th March 2023).

(Moffitt, 2017) Moffitt, C. (2017) Guide to Encoding Categorical Values in Python - Practical Business Python. Available at: https://pbpython.com/categorical-encoding.html (Accessed: 15th April 2023).

Chapter 9: Appendices

Contents

Appendices A……46

A. Project Definition Document……46

Appendices B……57

B. Distribution Plots……57

B. Line Plots……65

B. Pair Plots……71

B. Models……73

B. Data……77

Raw Source Code……79

ReadMe……89

9.1: Appendices A: Project Definition Document

IN3007/INM450 Individual Project

AI modelling using Neural Networks to predict rookie golfer performance valuations for sponsors/advertisers/tournament planners based on professional data.

Adam Khanzada, adam.khanzada@city.ac.uk

Consultant Name: Michael Garcia Ortiz

Clients: N/A

Project Definition Document: Proposal

Problem to be solved:

I am looking to solve the problem that is faced by many organisations involved in professional golf pertaining to the correct evaluation of a rookie/amateur players performance valuation. This would be a useful statistic for advertisers, sponsors, and tournament managers to be able to effectively plan and decide on rookie/amateur players worthiness of sponsorship.

It is very hard for sponsors to predict the success and popularity of newcomers to the sport so I seek to alleviate the burden of waiting on tournament results by offering a predictive model to help derive valuations for these new players. This will allow for more informed decisions to be made on the sponsorship investments on players companies need to make.

Overall, this is solving the problems of indecision and time-loss created by the uncertainty in a rookie players prospects for several beneficiaries in the golfing world.

Project Objectives: (prone to change after discussions with supervisor)

1.This project shall develop and AI model that will be able to predict player performance valuations to a significant degree of precision and accuracy.

1.a) The performance valuations will be able to generate predictions when tested to have an accuracy of at least 85% based on the professional player test data.

1.b) The predictive model will be able to be used as insight into the potential performance of amateur players, the success of this can be tested by using amateur friendly metrics for success such as handicap against our performance valuation.

1.c) The model will allow for the generation of a performance value for a rookie season player based on player stats to be decided.

2.This project shall research into the effective measures of value a player is deemed to have and what effects this value to sponsors and other benefactors

2.a) derive an effective algorithm for performance value based on marketing and statistical research. The success of this can be tested in a more personally evaluative process and can also receive input from testers.

3. This project shall streamline the indecision process of selecting golfers for sponsorships and tournament planning

3.a) This measure of effectiveness can be gauged by feedback and calculating the time saved between deciding with and without the model present for test players.

Project Beneficiaries: (may need to prioritise focus on catering to the needs to one)

I believe that my project will offer great benefit and insight to several parties.

Golf Sponsors

Sponsors will greatly benefit from being able to successfully model the potential success of up-and-coming players. The predictive model I hope to build will greatly assist them in deciding which players are worthwhile time and money investments for sponsorship deals. For a sponsor there is great uncertainty and indecision created by the lack of tournament wins or popularity metrics for new players. My model seeks to offer insight into the value these new players can offer sponsors from an exposure/money generation standpoint. Sponsors are always on the lookout for standout players in performance and audience reception and I believe the model I am developing will stream-line the currently conjectural ineffective system that wastes time and opportunity.

Golf Advertisers

Golf advertisers for events will be able to benefit from my research and modelling as they can come to better conclusions on which players are able to sell the tournaments and appropriately use them in advertisement campaigns. These campaigns typically feature the top players consisting of seasoned veterans of the sport but where the issue arises is deciding on which of the new generation of players to feature in these ad campaigns. This problem is especially prevalent in the golfing scene as the demographic for golfers is ageing and advertisers are always trying to tap into a new demographic by catering adverts to younger generations. The model I wish to create will seek to assist in deciding on the players to feature thus hopefully propelling increased viewership and therefore sales for advertisers.

Tournament Planners

Tournament planners work in a similar strain to the advertisers as they seek to promote tournaments in several different ways. A tournament planner is continually looking to select the best players to fulfil the role of creating the ideal sports viewing for an audience. To effectively do this the planner will need to understand how to pick and choose the playing field for a given tournament and how to best match up players in golf groups to create fierce competition. This can be a struggle at times, especially when trying to gauge where to slot in newer players. My AI model will help with this as the valuation derived by my research should offer insight into how a planner should manage the flow of new coming players and where to seed them in future tournaments to receive optimal returns (revenue)

Golf Players

Finally, the players themselves should be able to gain valuable insight from the model I hope to create as I believe it is important for an induvial, especially in the sports scene, to be aware of the value they bring to the sport. The players should be able to know what is affecting their future prospects when it comes to getting sponsorship deals, advertisements and rankings in tournaments. I believe that the research and modelling I hope to do will be able to shed light on how a player can better their performance and individual value they have to offer.

Work Plan (deadlines may vary and milestones will most likely change)

Output	Resources required	Start Date	End Date
Research/Literature review into the statistics important to golfer success	Internet, books, statistical data, time	After PDD accepted, already begun	1 weeks after proposal accepted
Data collection of PGA golfer tournament data and earnings…etc	PGA tour data (easily accessible)	Already sourced out some useful data	Within 1 week of proposal accepted
Cleansing and filtration of data to create coherent and understandable data set	Any data manipulation software (Jupyter notebook) using Python, Excel, time	Mid- February	20th February
Witten introduction of project report (draft)	Time, research	Mid-February	20th February
Initial steps taken towards building AI model, testing several methods and approaches	Python AI libraries, research into neural networks, GitHub, time	End of February	Start of March
Methodology draft and thought process behind decisions written	Time, research	End of February	March 10th
Coding aspect completed for AI modelling	Python AI libraries, research into neural networks, GitHub, time	Ongoing throughout	Middle of March
Generation of useful graphics and statistics based on results of AI model	Python libraries such as matplotlib and excel, research into other metrics	ongoing	End of March
Predictive testing based on sample rookie/amateur data conducted	Rookie test data, dummy data, volunteer golf data, could use own data	Start of April	April 10th
Written reports on coding process and testing process	Research, time	Start of April	April 15th
Evaluative write up to see areas that could improve (draft)	Research, time	Start of April	April 20th
Goals lined up against success in meeting requirements	Research, time	End of April	End of April
Conclusion of report written	Research, time	Start of May	Start of May

Project Risks

Risks to my project:

The requirement for me to have profound understanding of the significance of golfing statistics and which statistics bare greater relevance to different parties presents the risk of shifting objectives and requirements that may be hard to keep up with. I will have to plan ahead effectively and potentially cater my research/findings to a select beneficiary as opposed to having such broad target audience.

Another risk to my project comes from the uncertainty in the data set I plan on using and whether my ability to decide on what should be filtered, cleaned and considered relevant data are skewed by my lack of subject awareness. I hope to mitigate the risk of developing poor insights from my data selection by having done a great deal of research into the data and coming to informed decisions on what data is important. I also have a deep passion for golf and have avidly followed the sport for many years, so my base understanding is to a competent degree.

Finally, I believe a great risk to my project will be the learning curve involved with the development of an AI model using neural networks. This will have me stepping outside of my comfort zone when it comes to AI, and I hope that I am able to effectively manage and learns the ins and outs of this field within a timely manner to be able to produce findings of value to my potential clients.

Risks arising from my project:

A morally ethical risk arising from my project would be the potential enabling of my research and modelling to be used to effectively gamble as the predictive modelling of a players value may directly correlate to success (tournament wins). This unintended use for my work could be utilised to exploit gambling companies or promote the act of gambling itself.

Another risk stemming from my project would be the potential to halt/stagger a golfer's professional progress as my statistics and modelling could potentially result in a poor performance valuation for certain players thus hindering prospective advertising or sponsorship deals. This risk can me mitigated by carefully researching the field extensively and developing a thorough model that accurately predicts potential.

Risks to people involved in my project:

The only risk I could see to people involved in the project would be during the testing phase where I may use amateur golfing data to gauge the general validity of my research and whether my findings would line up with the playing level of the test data entered. This could result in the demoralisation of amateur golfers if desired results are not generated.

Research Ethics Review Form: BSc, MSc and MA Projects

Computer Science Research Ethics Committee (CSREC)

http://www.city.ac.uk/department-computer-science/research-ethics

Undergraduate and postgraduate students undertaking their final project in the Department of Computer Science are required to consider the ethics of their project work and to ensure that it complies with research ethics guidelines. In some cases, a project will need approval from an ethics committee before it can proceed. Usually, but not always, this will be because the student is involving other people ("participants") in the project.

In order to ensure that appropriate consideration is given to ethical issues, all students must complete this form and attach it to their project proposal document. There are two parts:

PART A: Ethics Checklist. All students must complete this part.
The checklist identifies whether the project requires ethical approval and, if so, where to apply for approval.

PART B: Ethics Proportionate Review Form. Students who have answered "no" to all questions in A1, A2 and A3 and "yes" to question 4 in A4 in the ethics checklist must complete this part. The project supervisor has delegated authority to provide approval in such cases that are considered to involve MINIMAL risk.
The approval may be ***provisional*** – *identifying the planned research as* likely to involve MINIMAL RISK.
In such cases you must additionally seek ***full approval*** from the supervisor as the project progresses and details are established. ***Full approval*** must be acquired in writing, before beginning the planned research.

A.1 If you answer YES to any of the questions in this block, you must apply to an appropriate external ethics committee for approval and log this approval as an External Application through Research Ethics Online - https://ethics.city.ac.uk/		*Delete as appropriate*
1.1	Does your research require approval from the National Research Ethics Service (NRES)? *e.g. because you are recruiting current NHS patients or staff?* *If you are unsure try - https://www.hra.nhs.uk/approvals-amendments/what-approvals-do-i-need/*	**NO**
1.2	Will you recruit participants who fall under the auspices of the Mental Capacity Act? *Such research needs to be approved by an external ethics committee such as NRES or the Social Care Research Ethics Committee - http://www.scie.org.uk/research/ethics-committee/*	**NO**
1.3	Will you recruit any participants who are currently under the auspices of the Criminal Justice System, for example, but not limited to, people on remand, prisoners and those on probation? *Such research needs to be authorised by the ethics approval system of the National Offender Management Service.*	**NO**
A.2 If you answer YES to any of the questions in this block, then unless you are applying to an external ethics committee, you must apply for approval from the Senate Research Ethics Committee (SREC) through Research Ethics Online - https://ethics.city.ac.uk/		*Delete as appropriate*
2.1	Does your research involve participants who are unable to give informed consent? *For example, but not limited to, people who may have a degree of learning disability or mental health problem, that means they are unable to make an informed decision on their own behalf.*	**NO**

2.2	Is there a risk that your research might lead to disclosures from participants concerning their involvement in illegal activities?	**NO**
2.3	Is there a risk that obscene and or illegal material may need to be accessed for your research study (including online content and other material)?	**NO**
2.4	Does your project involve participants disclosing information about special category or sensitive subjects? *For example, but not limited to: racial or ethnic origin; political opinions; religious beliefs; trade union membership; physical or mental health; sexual life; criminal offences and proceedings*	**NO**
2.5	Does your research involve you travelling to another country outside of the UK, where the Foreign & Commonwealth Office has issued a travel warning that affects the area in which you will study? *Please check the latest guidance from the FCO - http://www.fco.gov.uk/en/*	**NO**
2.6	Does your research involve invasive or intrusive procedures? *These may include, but are not limited to, electrical stimulation, heat, cold or bruising.*	**NO**
2.7	Does your research involve animals?	**NO**
2.8	Does your research involve the administration of drugs, placebos or other substances to study participants?	**NO**
A.3 If you answer YES to any of the questions in this block, then unless you are applying to an external ethics committee or the SREC, you must apply for approval from the Computer Science Research Ethics Committee (CSREC) through Research Ethics Online - https://ethics.city.ac.uk/ **Depending on the level of risk associated with your application, it may be referred to the Senate Research Ethics Committee.**		*Delete as appropriate*
3.1	Does your research involve participants who are under the age of 18?	**NO**
3.2	Does your research involve adults who are vulnerable because of their social, psychological or medical circumstances (vulnerable adults)? *This includes adults with cognitive and / or learning disabilities, adults with physical disabilities and older people.*	**NO**
3.3	Are participants recruited because they are staff or students of City, University of London? *For example, students studying on a particular course or module.* *If yes, then approval is also required from the Head of Department or Programme Director.*	**NO**
3.4	Does your research involve intentional deception of participants?	**NO**
3.5	Does your research involve participants taking part without their informed consent?	**NO**
3.5	Is the risk posed to participants greater than that in normal working life?	**NO**
3.7	Is the risk posed to you, the researcher(s), greater than that in normal working life?	**NO**

A.4 If you answer YES to the following question and your answers to all other questions in sections A1, A2 and A3 are NO, then your project is deemed to be of MINIMAL RISK. **If this is the case, then you can apply for approval through your supervisor under PROPORTIONATE REVIEW. You do so by completing PART B of this form.** **If you have answered NO to all questions on this form, then your project does not require ethical approval. You should submit and retain this form as evidence of this.**		*Delete as appropriate*
4	Does your project involve human participants or their identifiable personal data? *For example, as interviewees, respondents to a survey or participants in testing.*	**NO**

PART B: Ethics Proportionate Review Form

If you answered YES to question 4 and NO to all other questions in sections A1, A2 and A3 in PART A of this form, then you may use PART B of this form to submit an application for a proportionate ethics review of your project. Your project supervisor has delegated authority to review and approve this application under proportionate review. You must receive final approval from your supervisor in writing before beginning the planned research.

However, if you cannot provide all the required attachments (see B.3) with your project proposal (e.g. because you have not yet written the consent forms, interview schedules etc), the approval from your supervisor will be ***provisional***. You **must** submit the missing items to your supervisor for approval prior to commencing these parts of your project. Once again, you must receive written confirmation from your supervisor that any provisional approval has been superseded by with ***full approval*** of the planned activity as detailed in the full documents. **Failure to follow this procedure and demonstrate that final approval has been achieved may result in you failing the project module.**

Your supervisor may ask you to submit a full ethics application through Research Ethics Online, for instance if they are unable to approve your application, if the level of risks associated with your project change, or if you need an approval letter from the CSREC for an external organisation.

B.1 The following questions must be answered fully. All grey instructions must be removed.		*Delete as appropriate*
1.1.	Will you ensure that participants taking part in your project are fully informed about the purpose of the research?	**YES / NO**
1.2	Will you ensure that participants taking part in your project are fully informed about the procedures affecting them or affecting any information collected about them, including information about how the data will be used, to whom it will be disclosed, and how long it will be kept?	**YES / NO**
1.3	When people agree to participate in your project, will it be made clear to them that they may withdraw (i.e. not participate) at any time without any penalty?	**YES / NO**
1.4	Will consent be obtained from the participants in your project? Consent from participants will be necessary if you plan to involve them in your project or if you plan to use identifiable personal data from existing records. "Identifiable personal data" means data relating to a living person who might be identifiable if the record includes their name, username, student id, DNA, fingerprint, address, etc. *If YES, you must attach drafts of the participant information sheet(s) and consent form(s) that you will use in section B.3 or, in the case of an existing dataset, provide details of how consent has been obtained.* *You must also retain the completed forms for subsequent inspection. Failure to provide the completed consent request forms will result in withdrawal of any earlier ethical approval of your project.*	**YES / NO**
1.5	Have you made arrangements to ensure that material and/or private information obtained from or about the participating individuals will remain confidential?	**YES / NO**

B.2 If the answer to the following question (B2) is YES, you must provide details		*Delete as appropriate*
2	Will the research be conducted in the participant's home or other non-University location? *If **YES**, you must provide details of how your safety will be ensured.*	**YES / NO**

B.3 Attachments **ALL of the following documents MUST be provided to supervisors if applicable.** **All must be considered prior to final approval by supervisors.** **A written record of final approval must be provided and retained.**	***YES***	***NO***	***Not Applicable***
Details on how safety will be assured in any non-University location, including risk assessment if required (see B2)			
Details of arrangements to ensure that material and/or private information obtained from or about the participating individuals will remain confidential (see B1.5) *Any personal data must be acquired, stored and made accessible in ways that are GDPR compliant.*			
Full protocol for any workshops or interviews**			
Participant information sheet(s)**			
Consent form(s)**			
Questionnaire(s)** *sharing a Qualtrics survey with your supervisor is recommended.*			
Topic guide(s) for interviews and focus groups**			
Permission from external organisations or Head of Department** *e.g. for recruitment of participants*			

***If these items are not available at the time of submitting your project proposal, then **provisional approval** can still be given, under the condition that you must submit the final versions of all items to your supervisor for approval at a later date. **All** such items **must** be seen and approved by your supervisor before the activity for which they are needed begins. Written evidence of **final approval** of your planned activity must be acquired from your supervisor before you commence.*

Changes

If your plans change and any aspects of your research that are documented in the approval process change as a consequence, then any approval acquired is invalid. If issues addressed in Part A (the checklist) are affected, then you must complete the approval process again and establish the kind of approval that is required. If issues addressed in Part B are affected, then you must forward updated documentation to your supervisor and have received written confirmation of approval of the revised activity before proceeding.

Templates for Consent and Information

You must use the templates provided by the University as the basis for your participant information sheets and consent forms. You **must** adapt them according to the needs of your project before you submit them for consideration.

Participant Information Sheets, Consent Forms and Protocols must be consistent. Please ensure that this is the case prior to seeking approval. Failure to do so will slow down the approval process.

We strongly recommend using Qualtrics to produce digital information sheets and consent forms.

Further Information

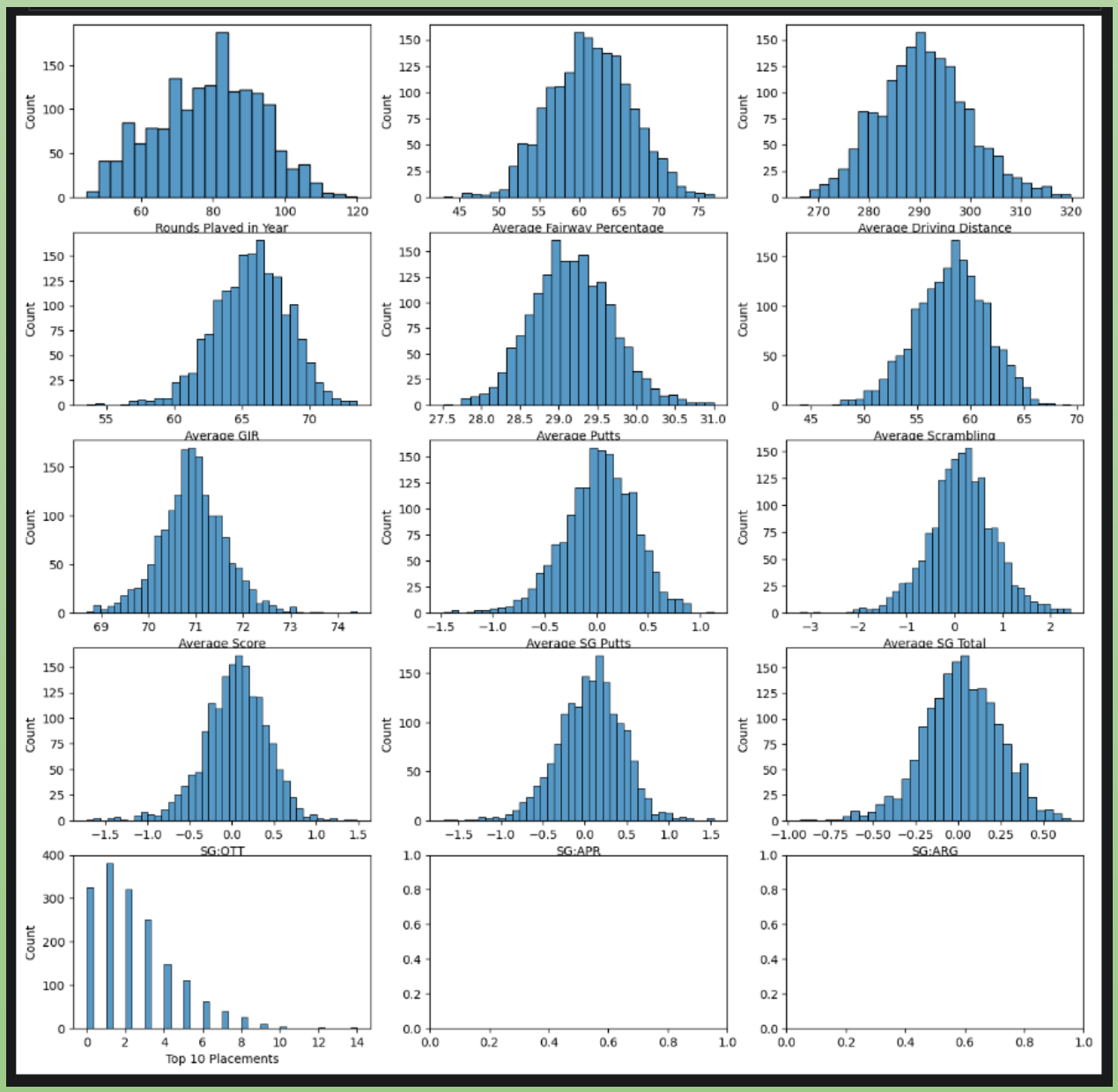
http://www.city.ac.uk/department-computer-science/research-ethics

https://www.city.ac.uk/research/ethics/how-to-apply/participant-recruitment

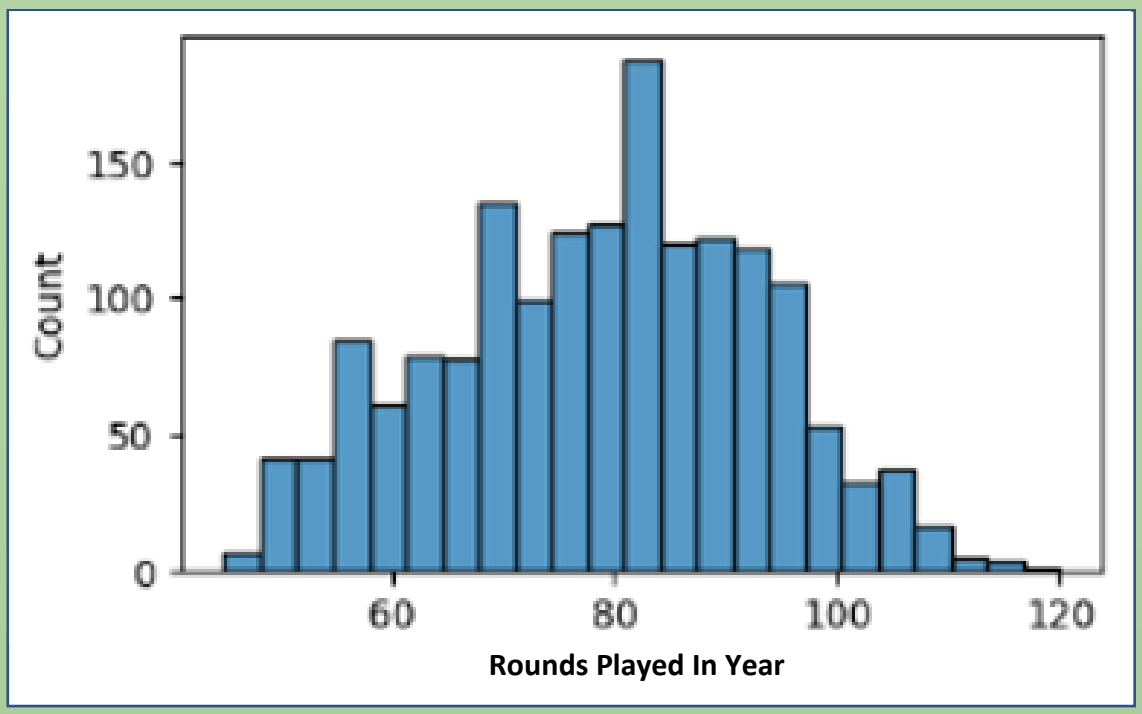
https://www.city.ac.uk/research/ethics

9.2: Appendices B: Reuse Summary

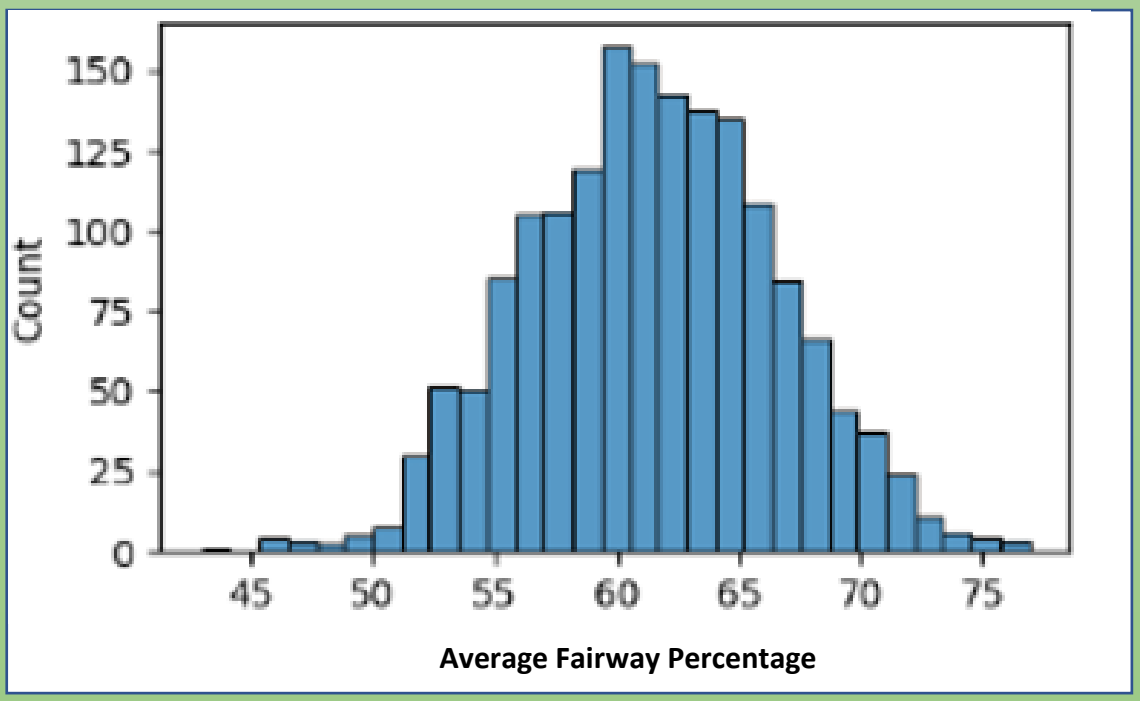
Distribution Plots [1-A]



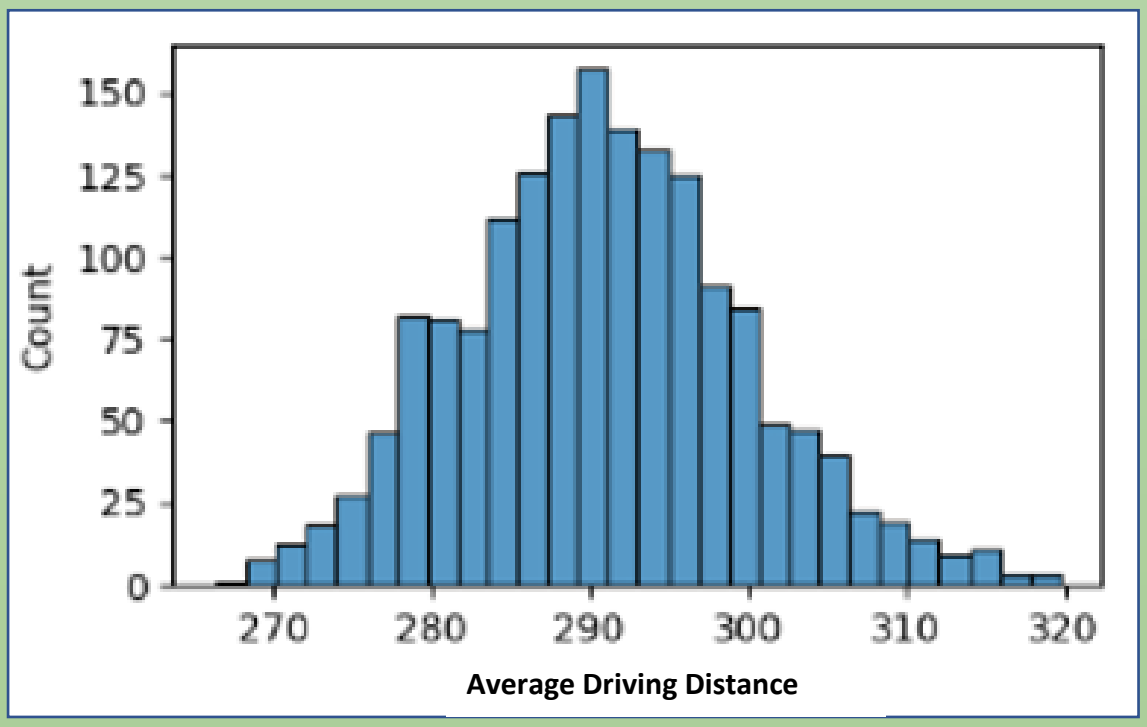
Distribution Plot [1-B]



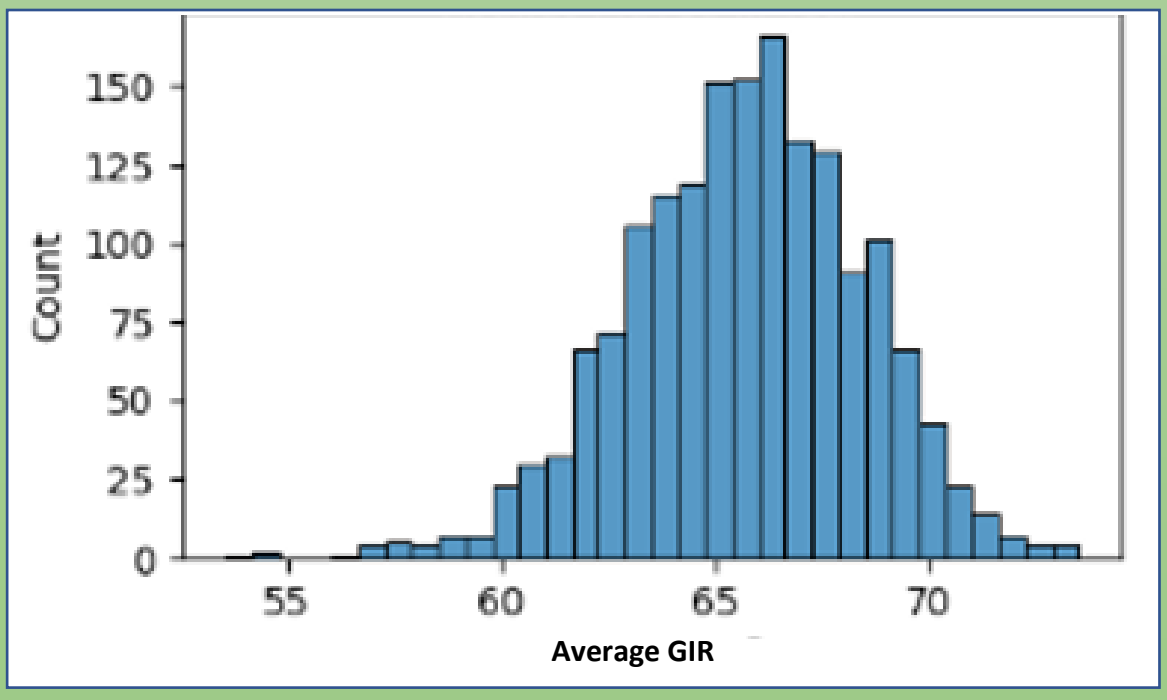
Distribution Plot [1-C]



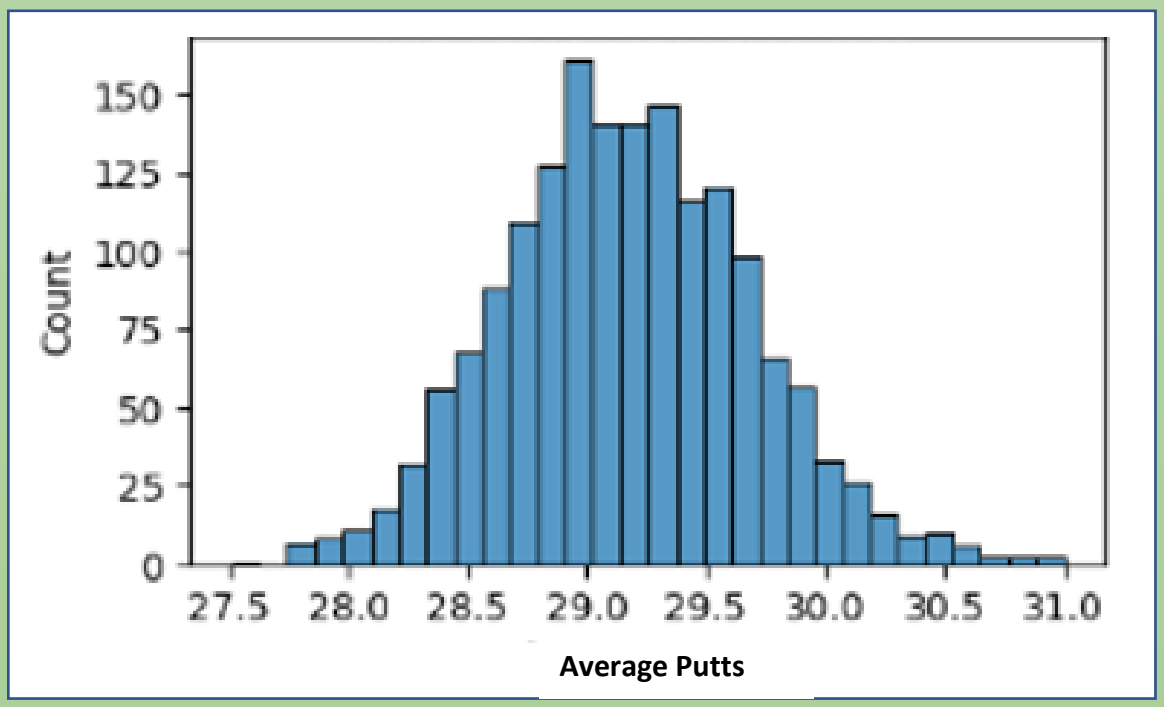
Distribution Plot [1-C]



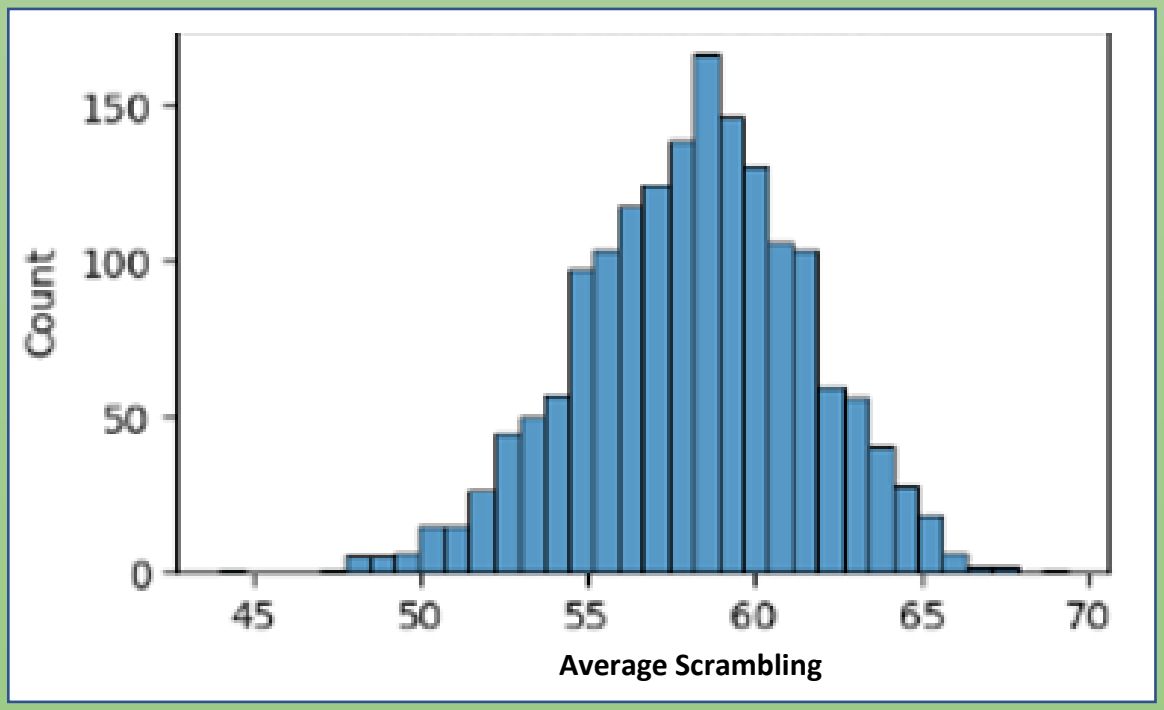
Distribution Plot [1-D]



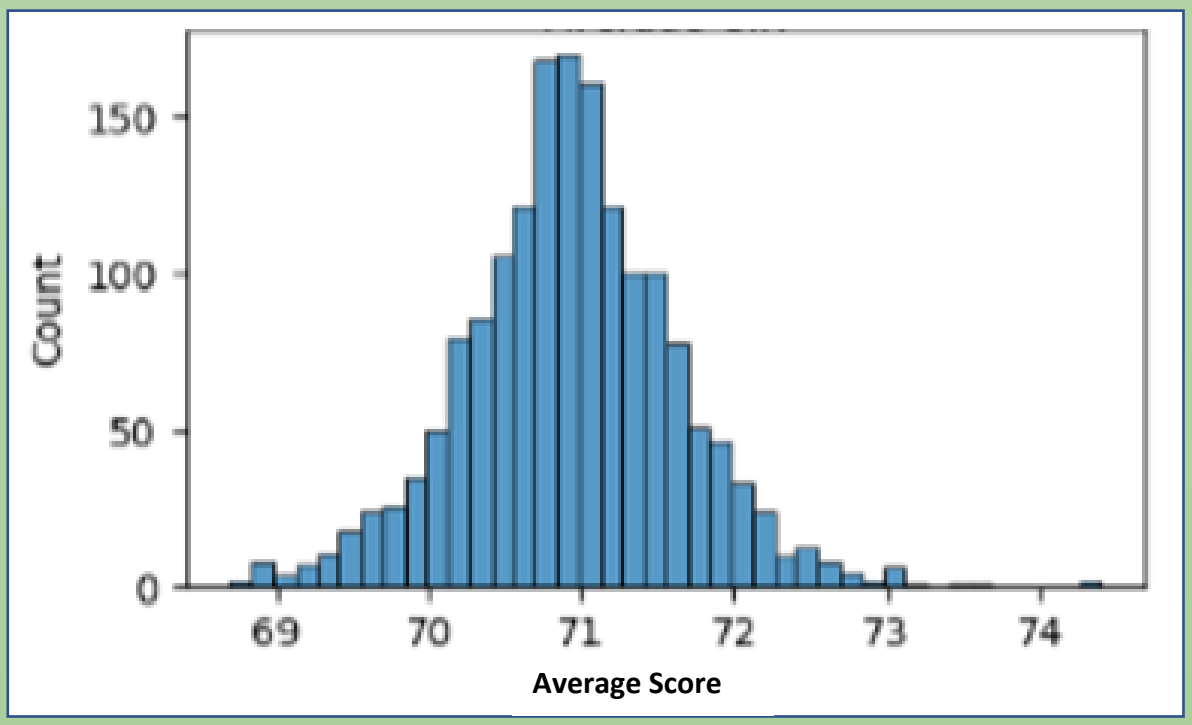
Distribution Plot [1-E]



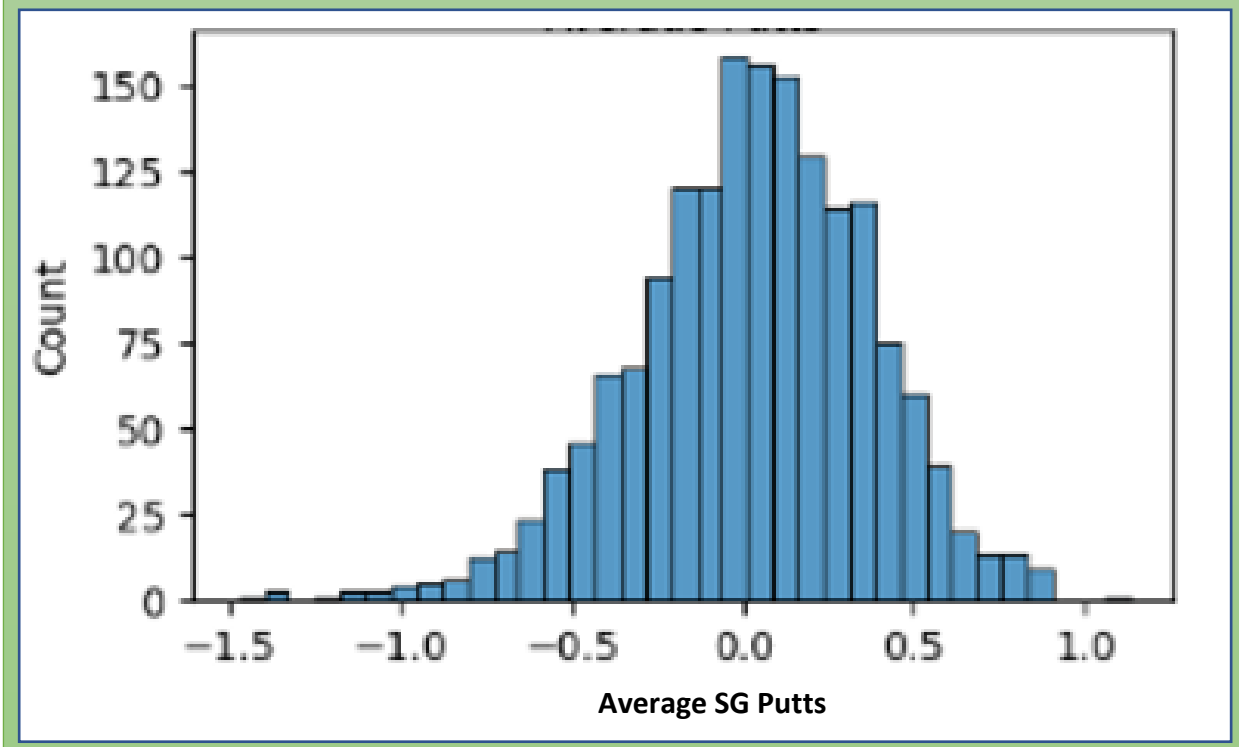
Distribution Plot [1-F]



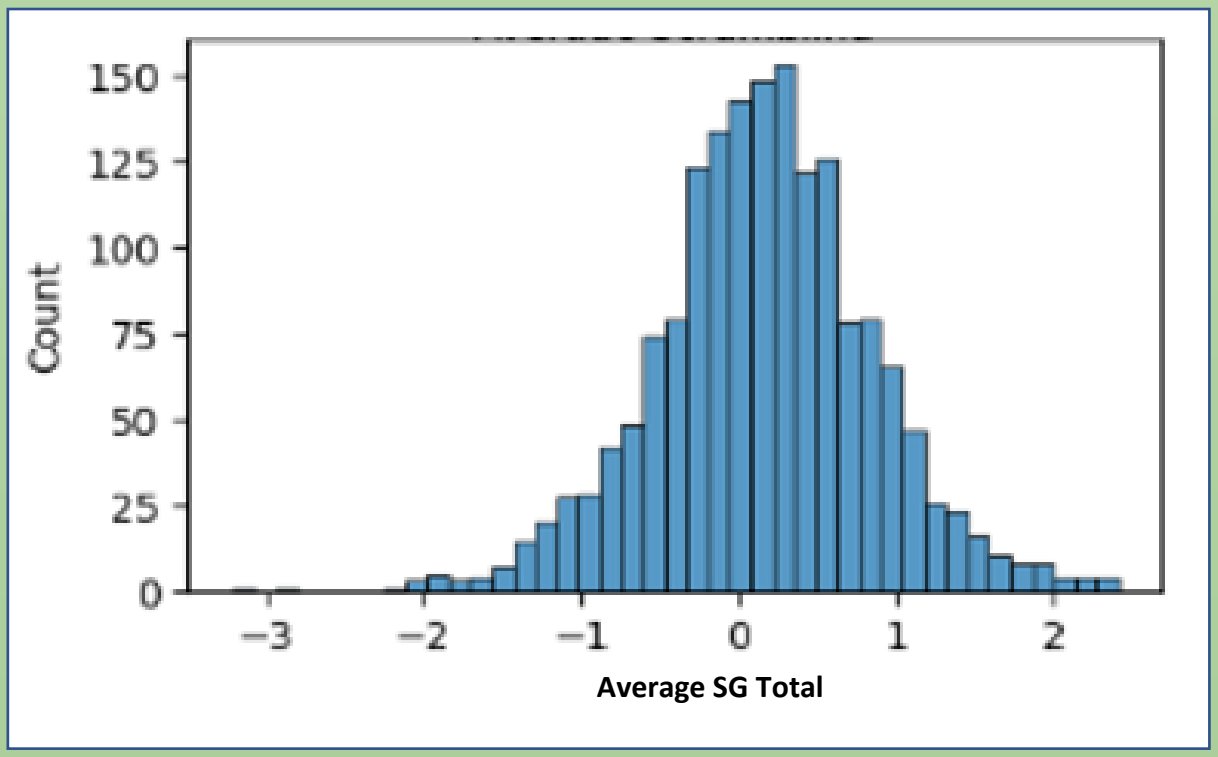
Distribution Plot [1-G]



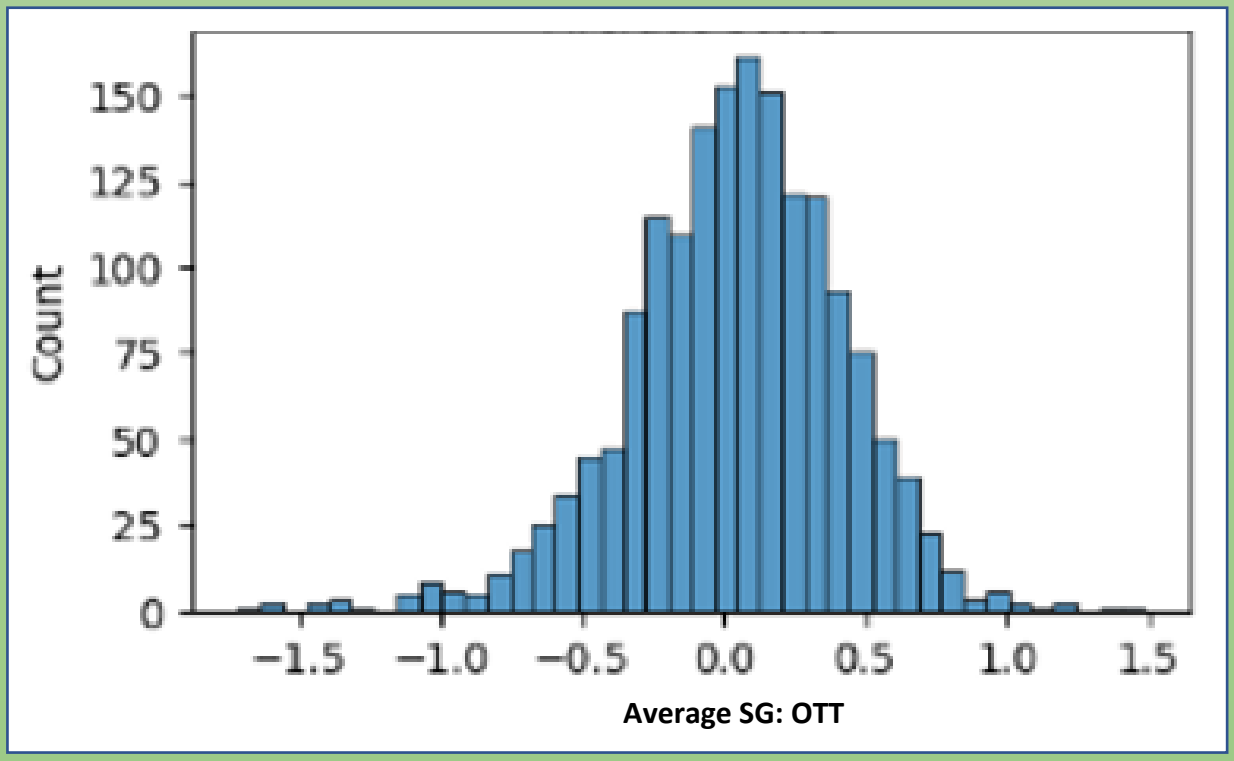
Distribution Plot [1-H]



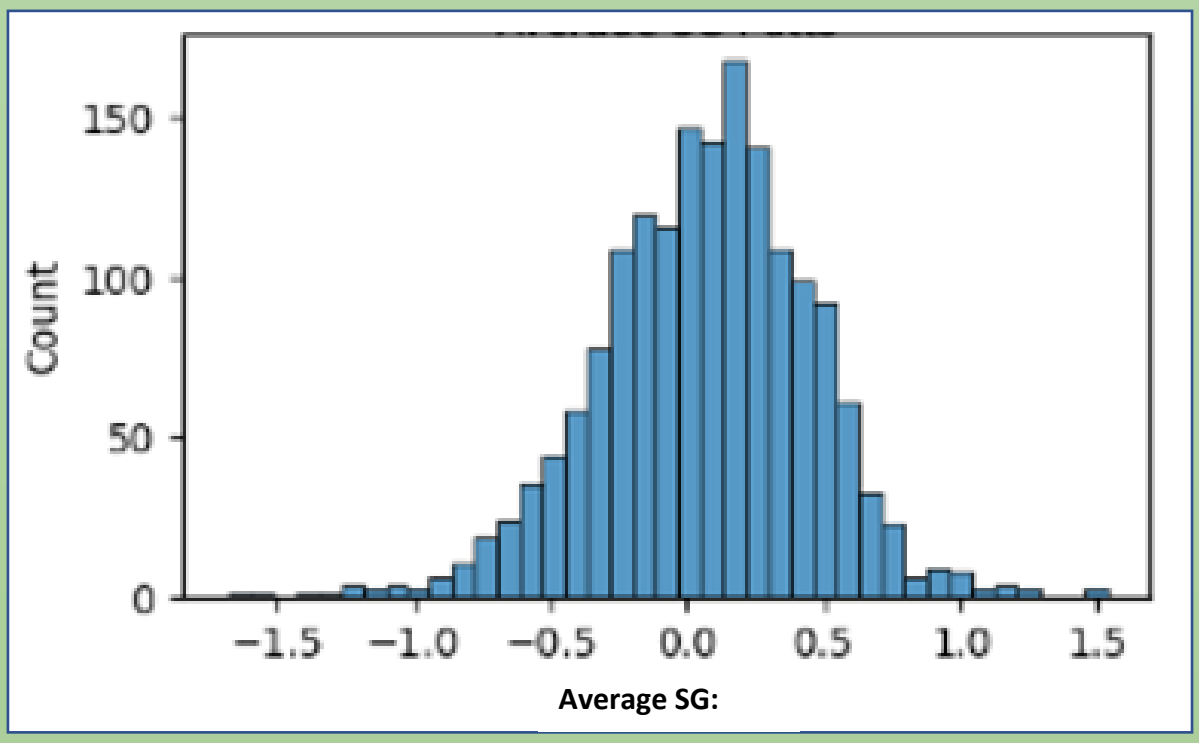
Distribution Plot [1-I]



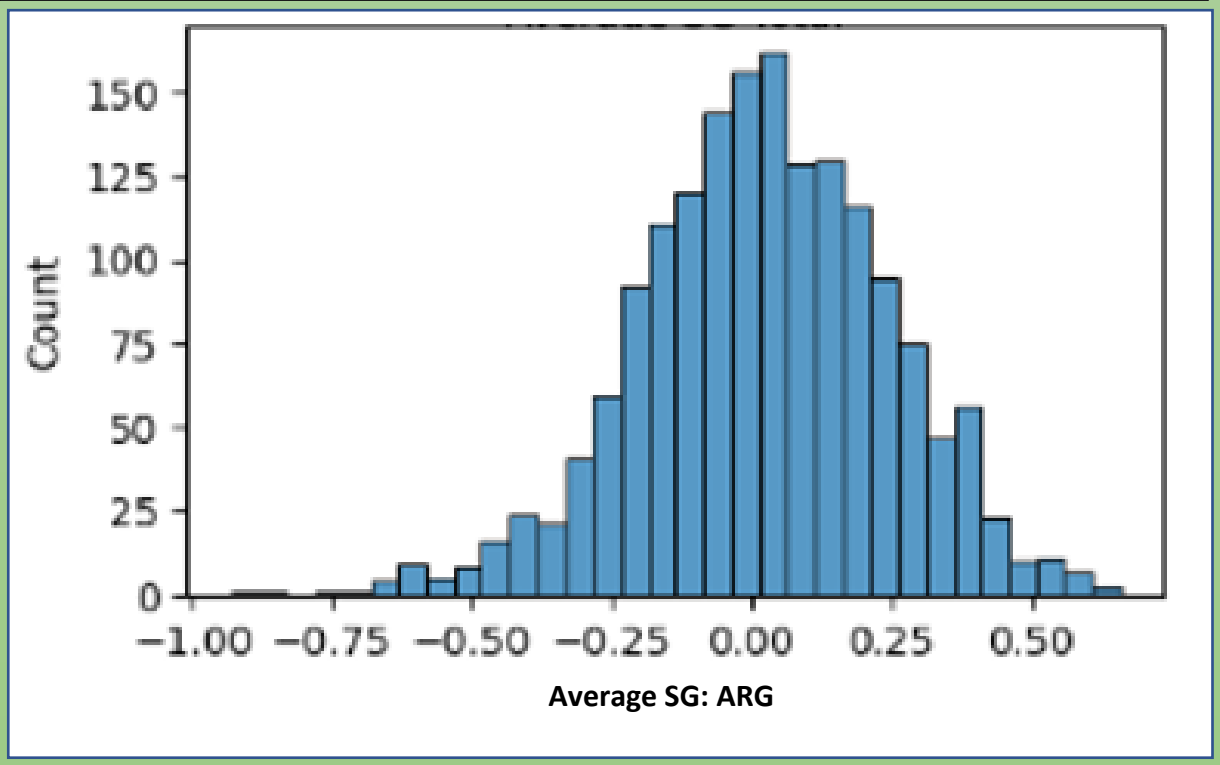
Distribution Plot [1-J]



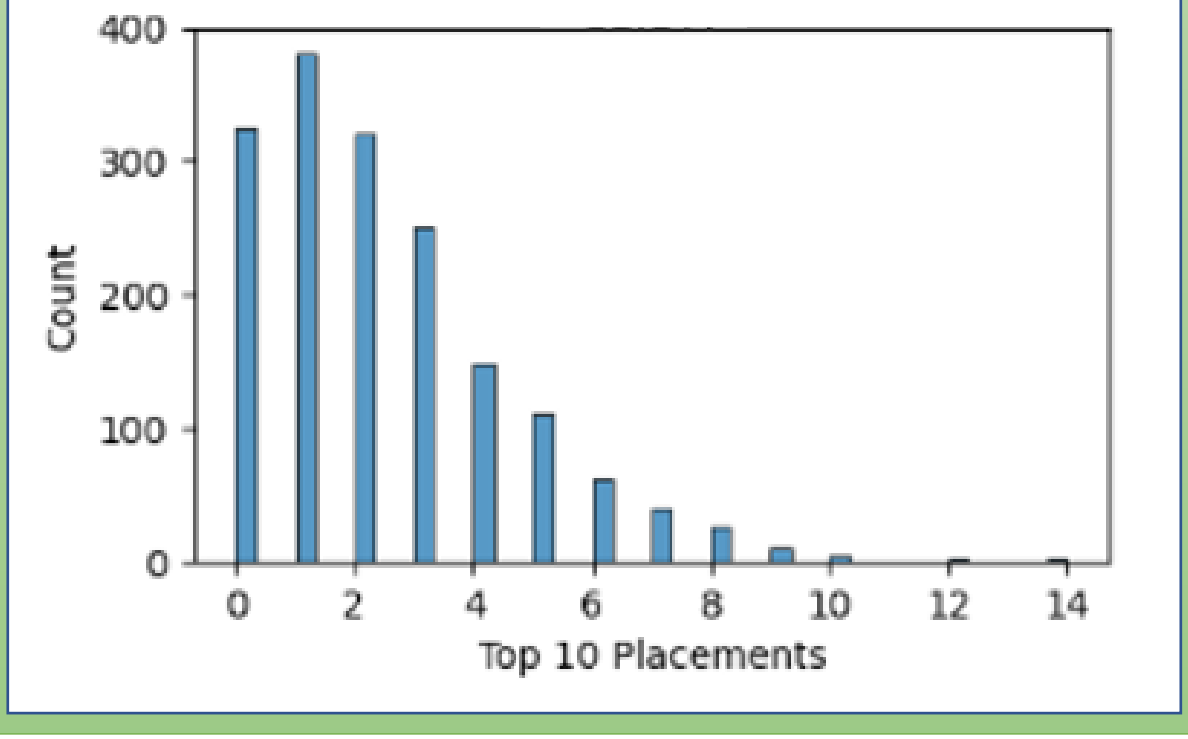
Distribution Plot [1-K]



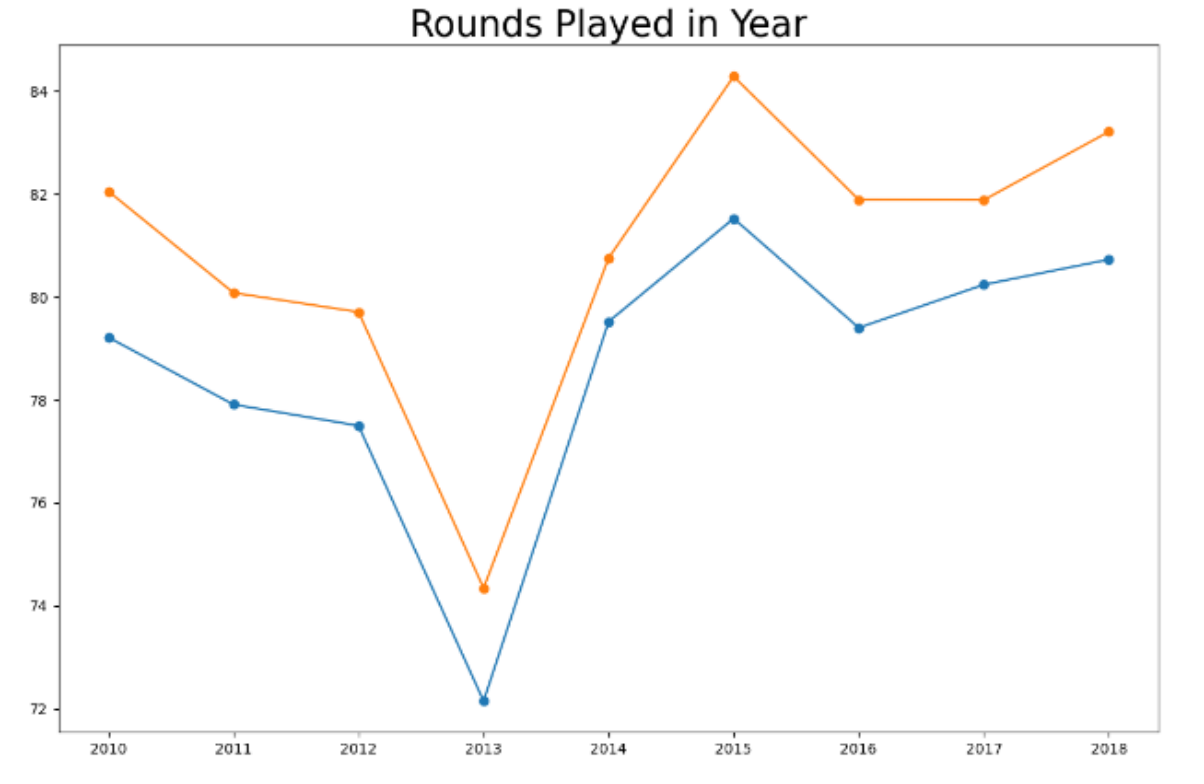
Distribution Plot [1-L]



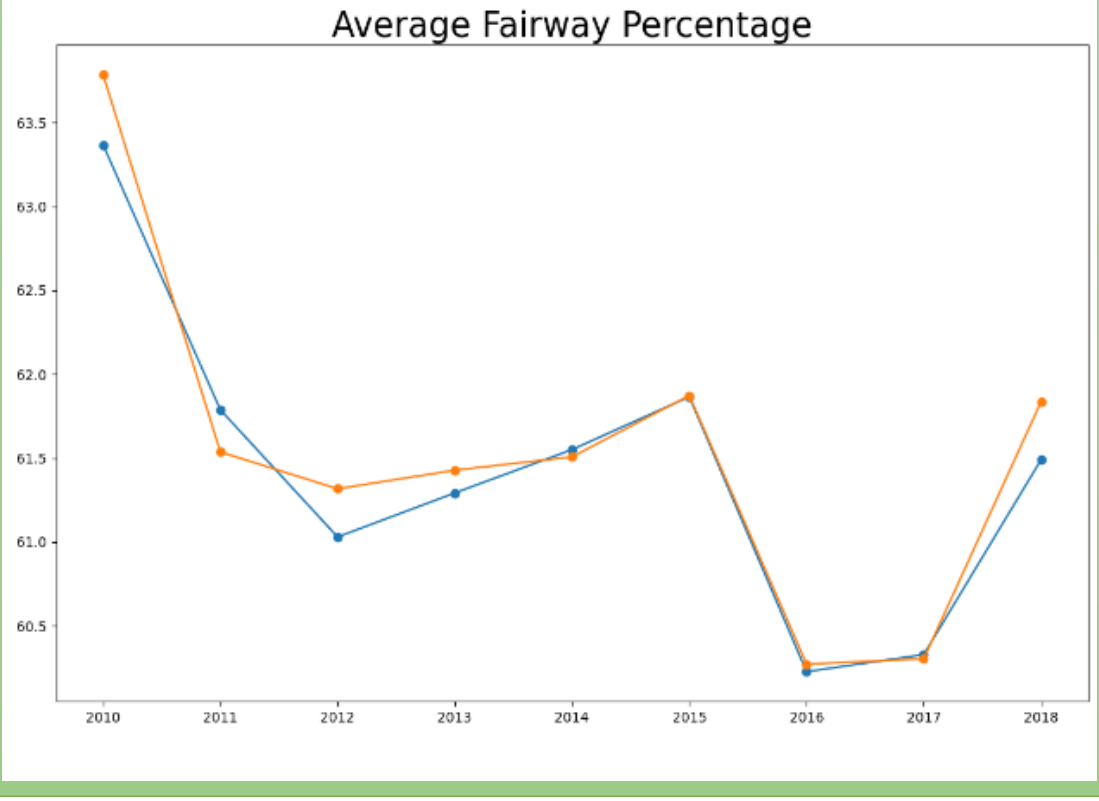
Distribution Plot [1-M]



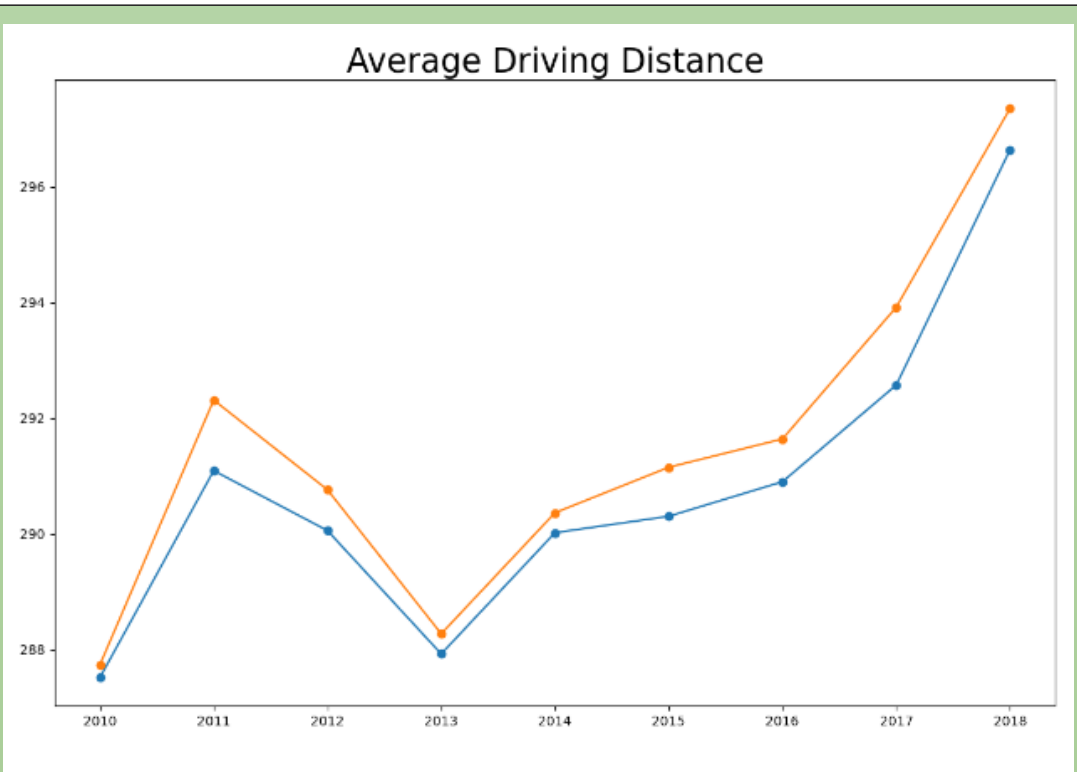
Comparative Line Chart [2-A]



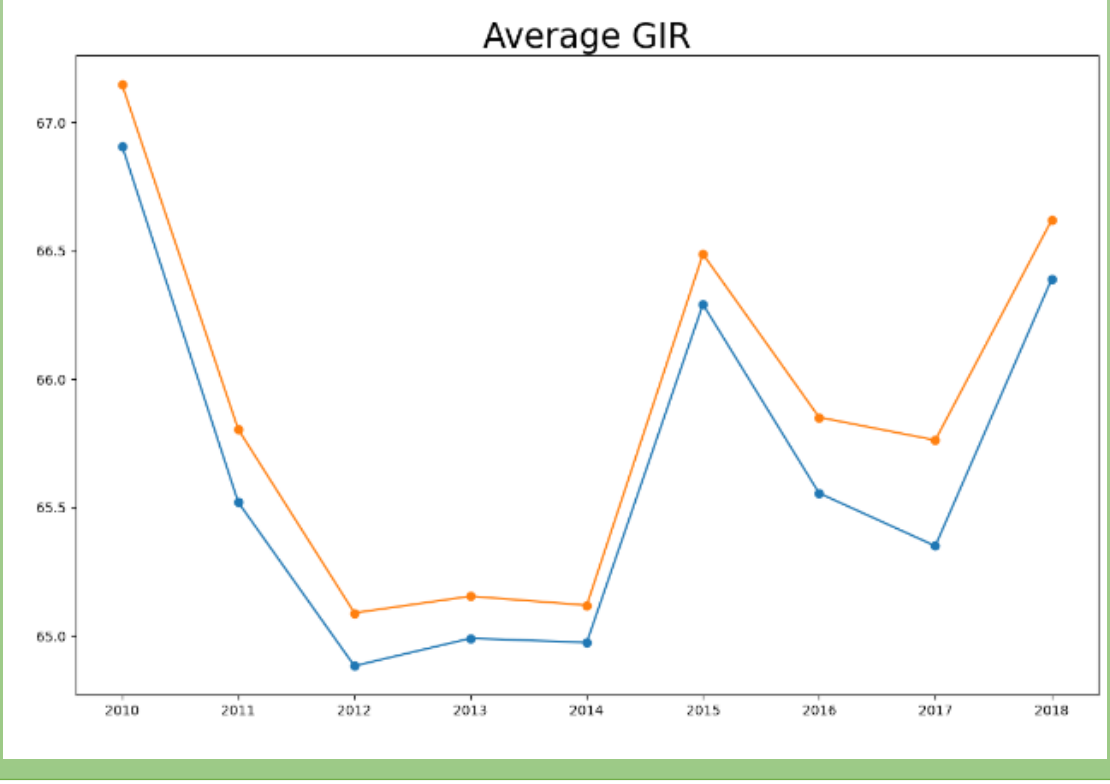
Comparative Line Chart [2-B]



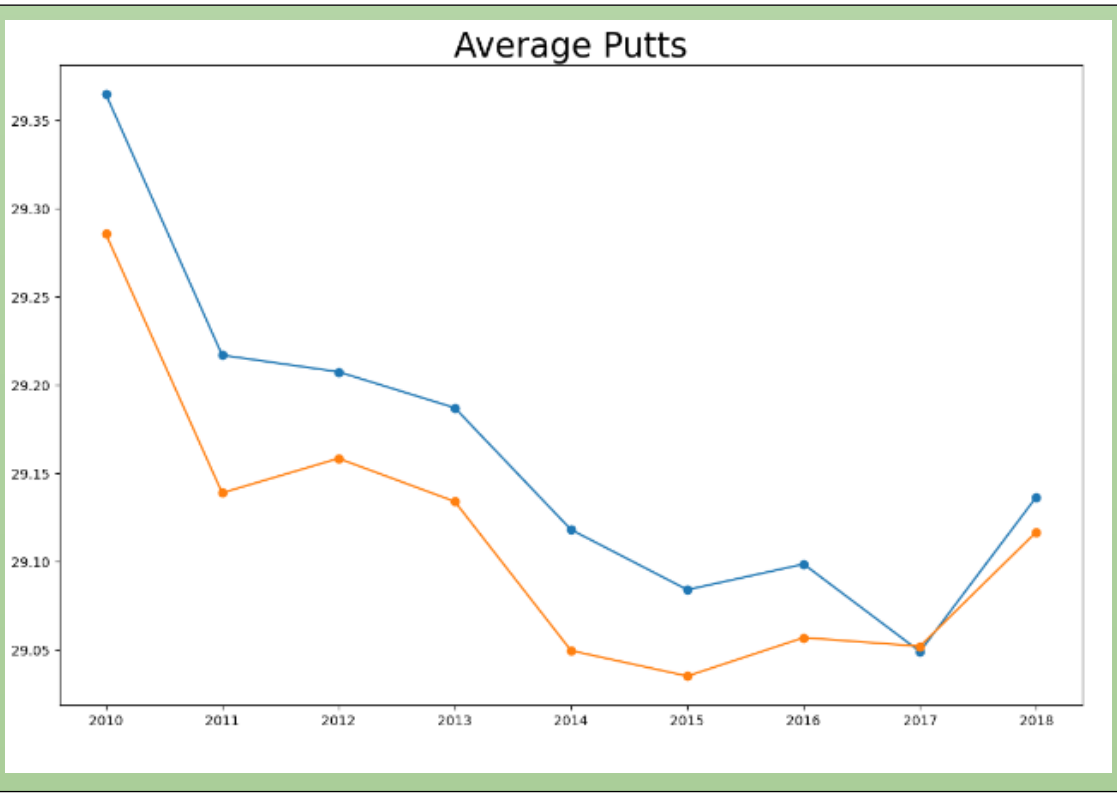
Comparative Line Chart [2-C]



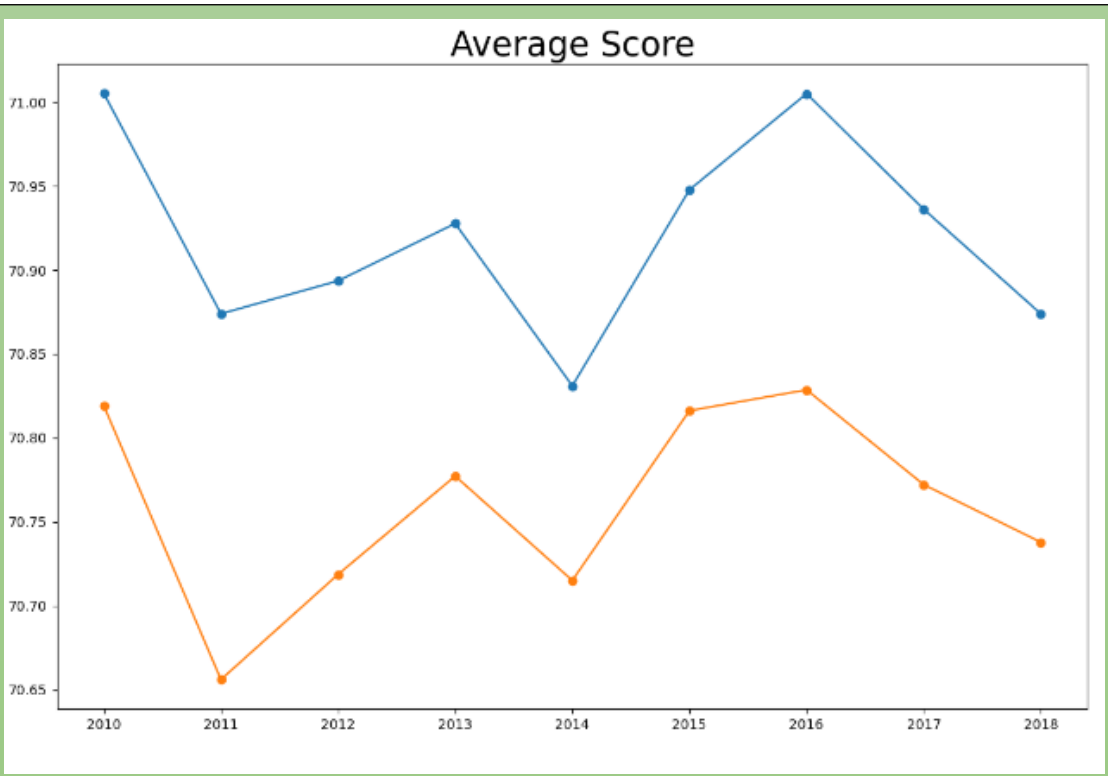
Comparative Line Chart [2-D]



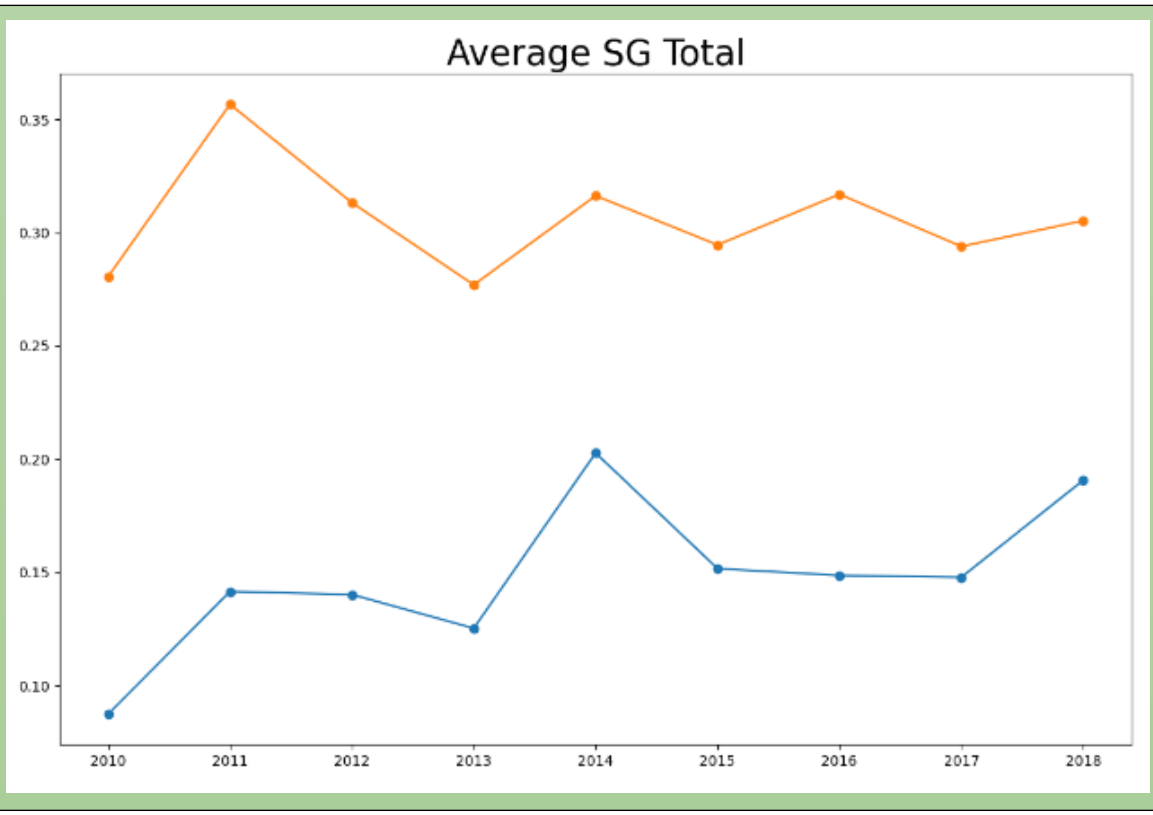
Comparative Line Chart [2-E]



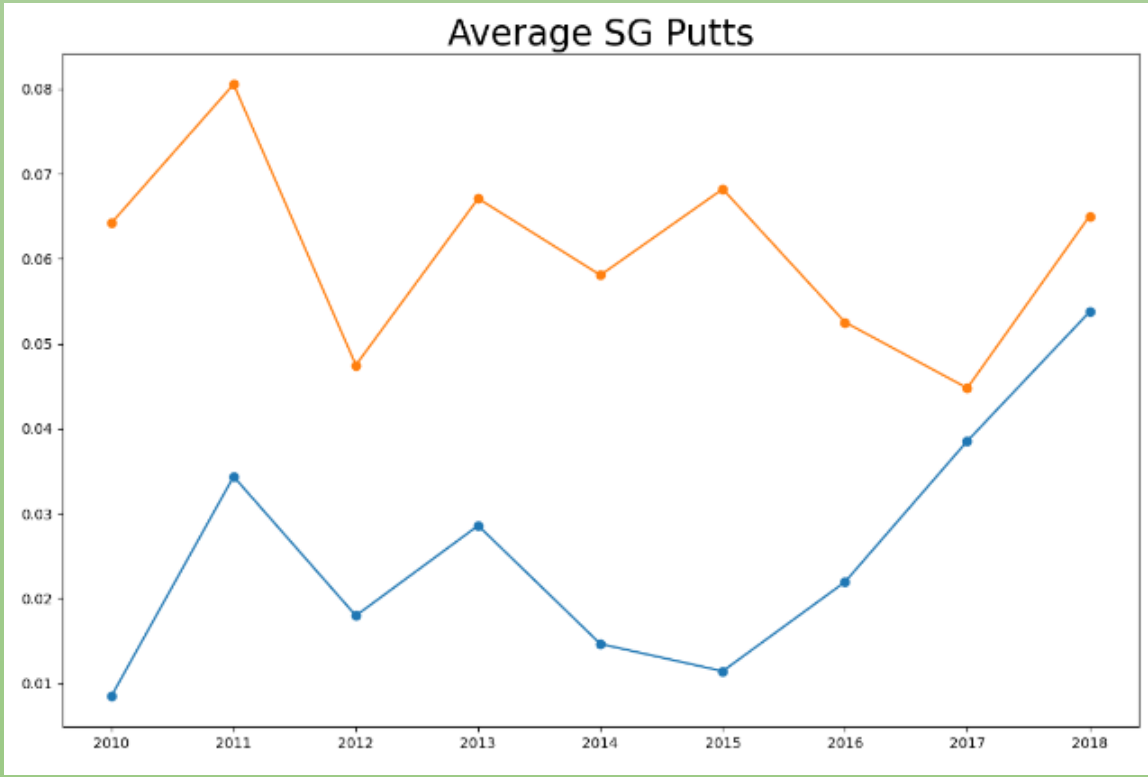
Comparative Line Chart [2-F]



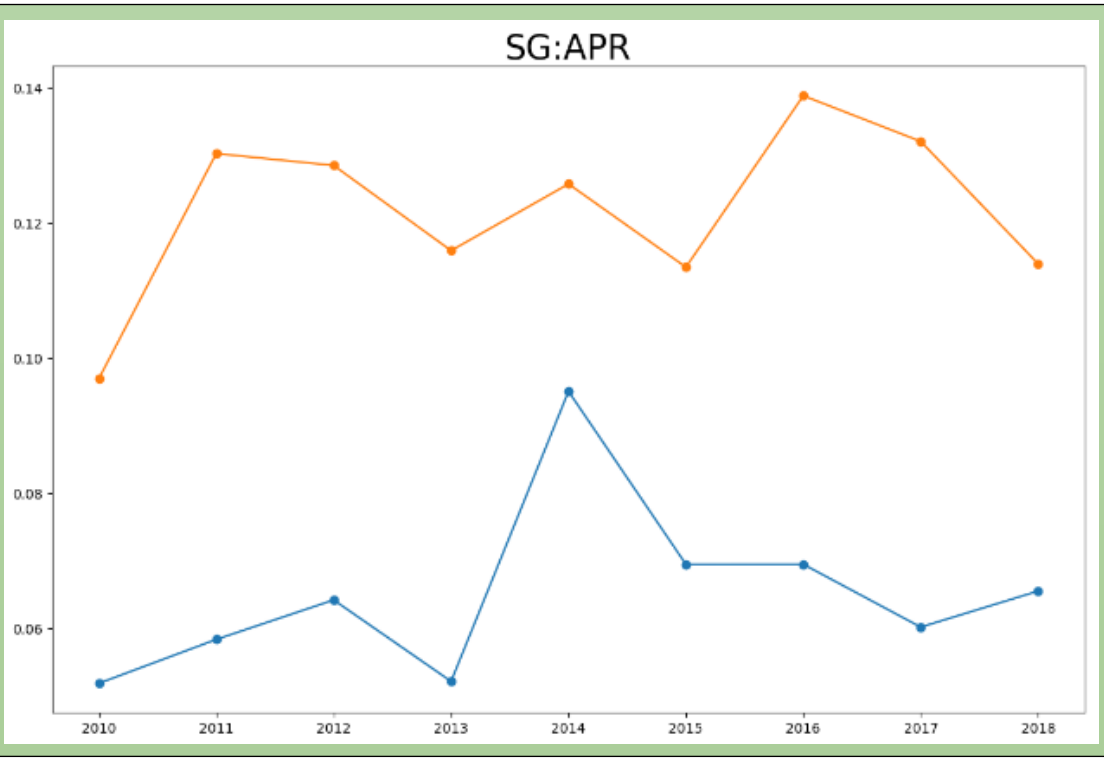
Comparative Line Chart [2-G]



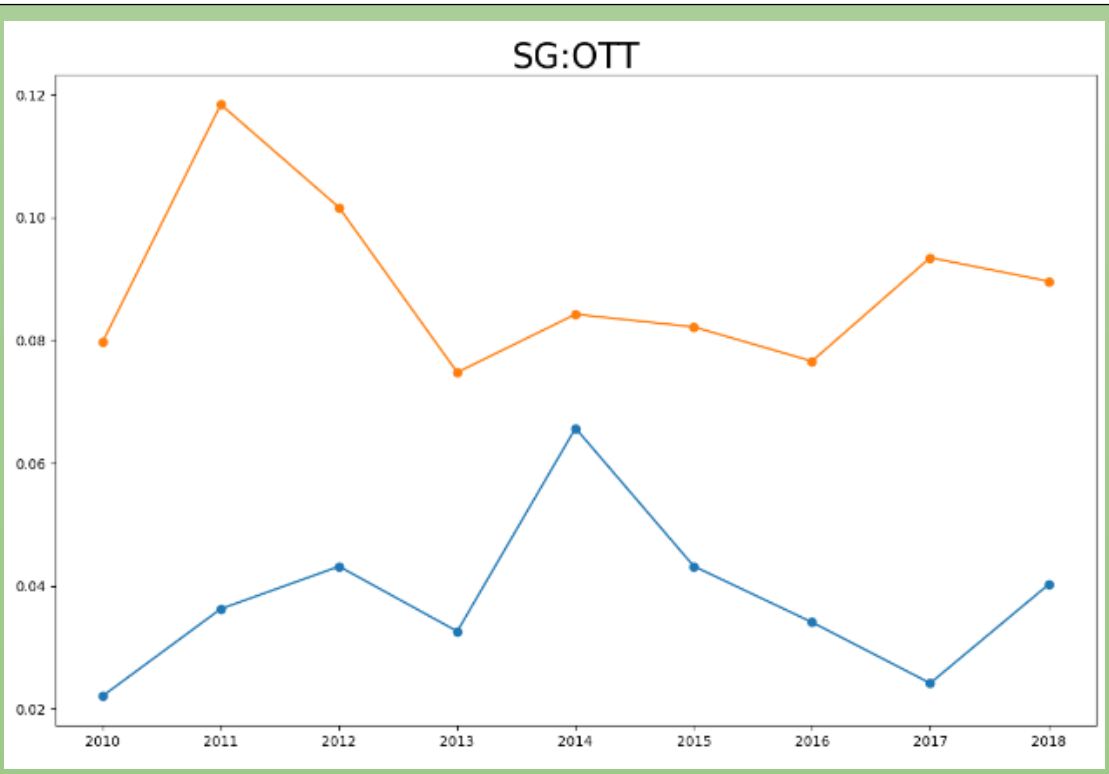
Comparative Line Chart [2-H]



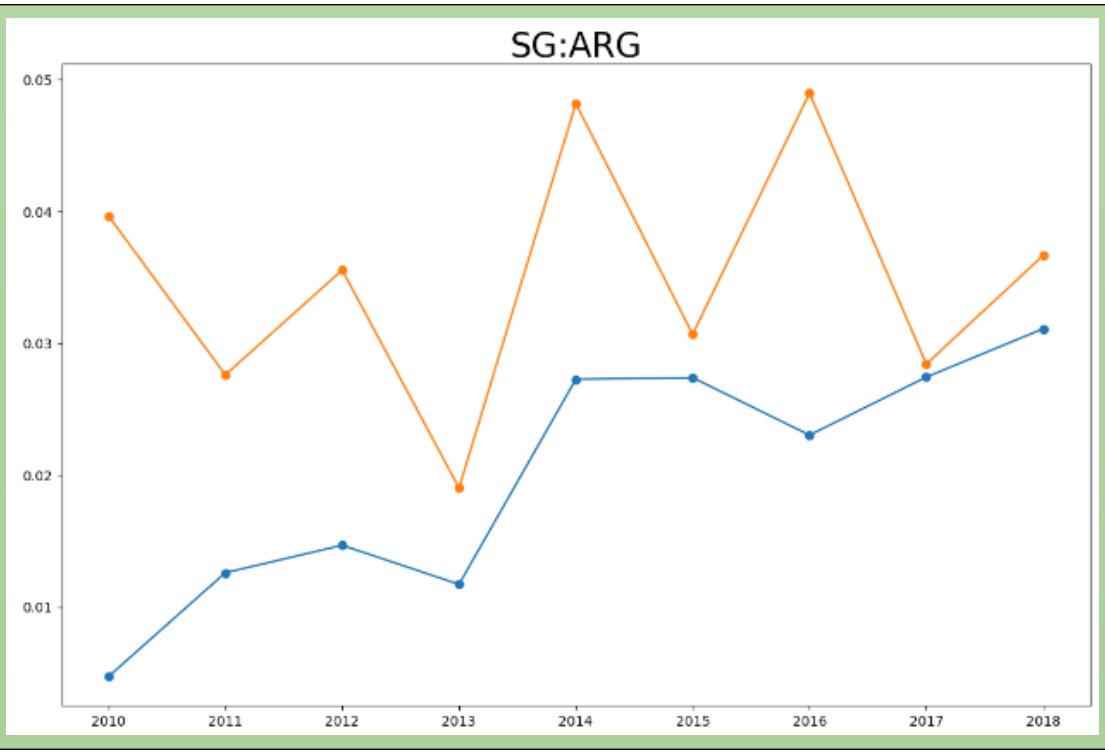
Comparative Line Chart [2-I]



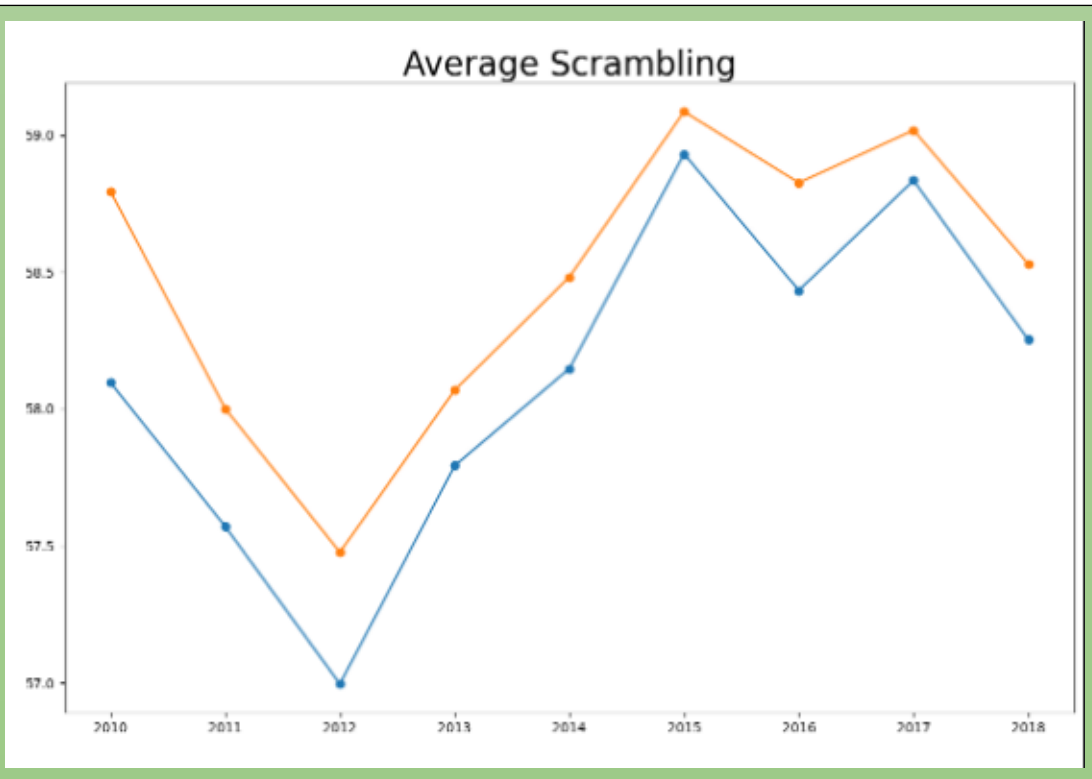
Comparative Line Chart [2-J]



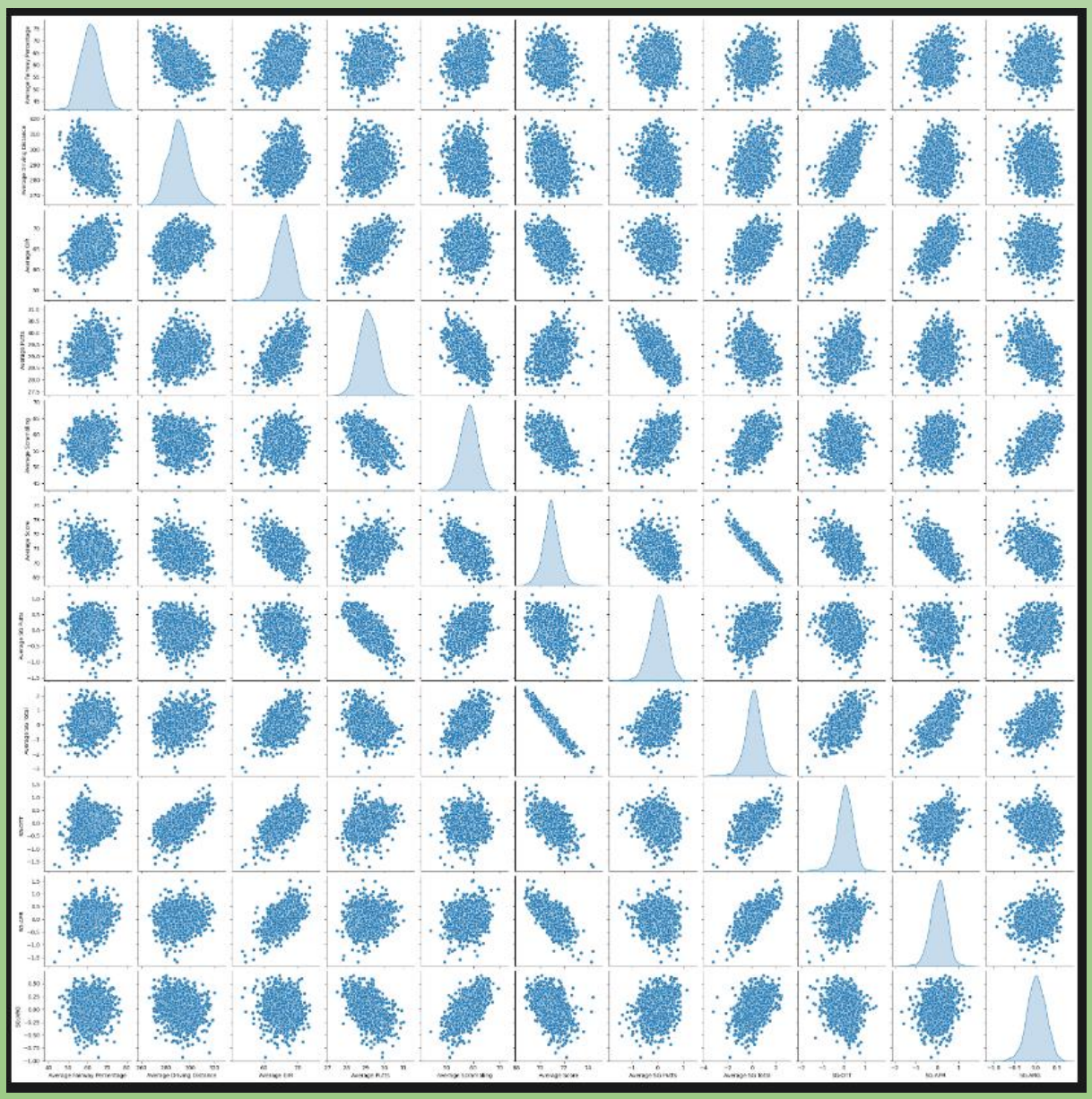
Comparative Line Chart [2-K]



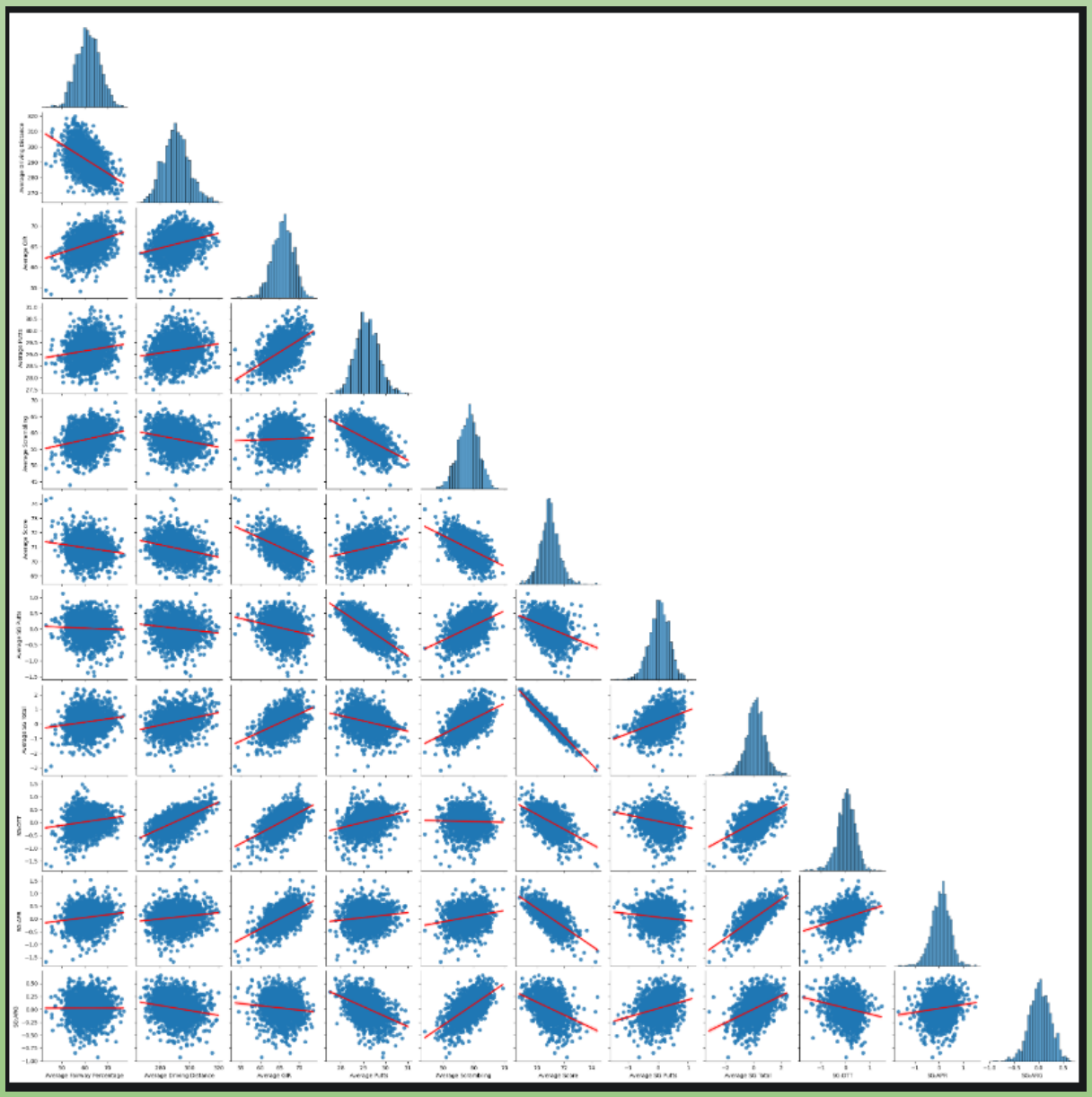
Comparative Line Chart [2-L]



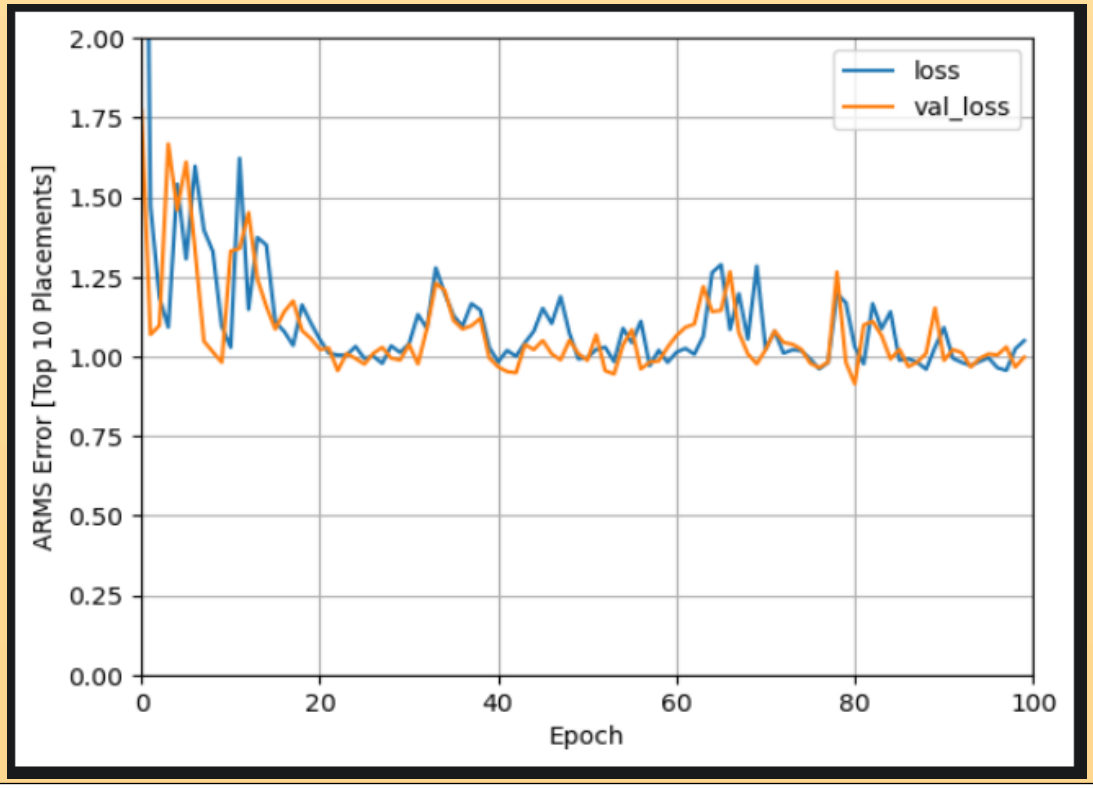
Correlation Pair Plot Scatter [3-A]



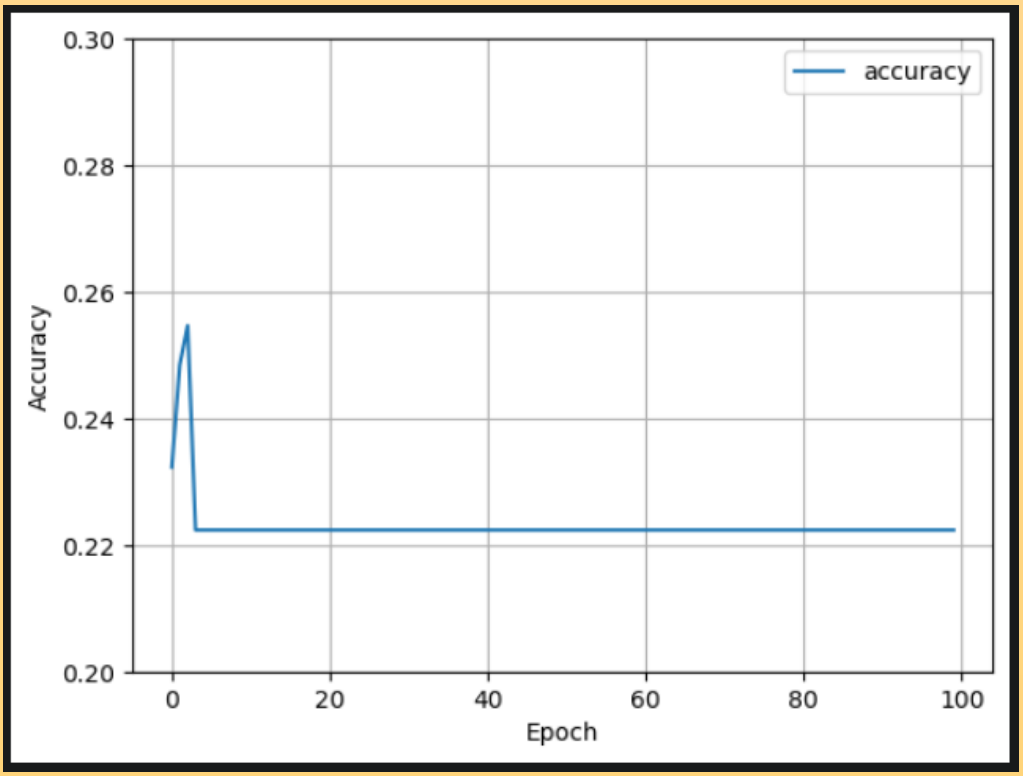
Correlation Pair Plot Scatter [3-B]



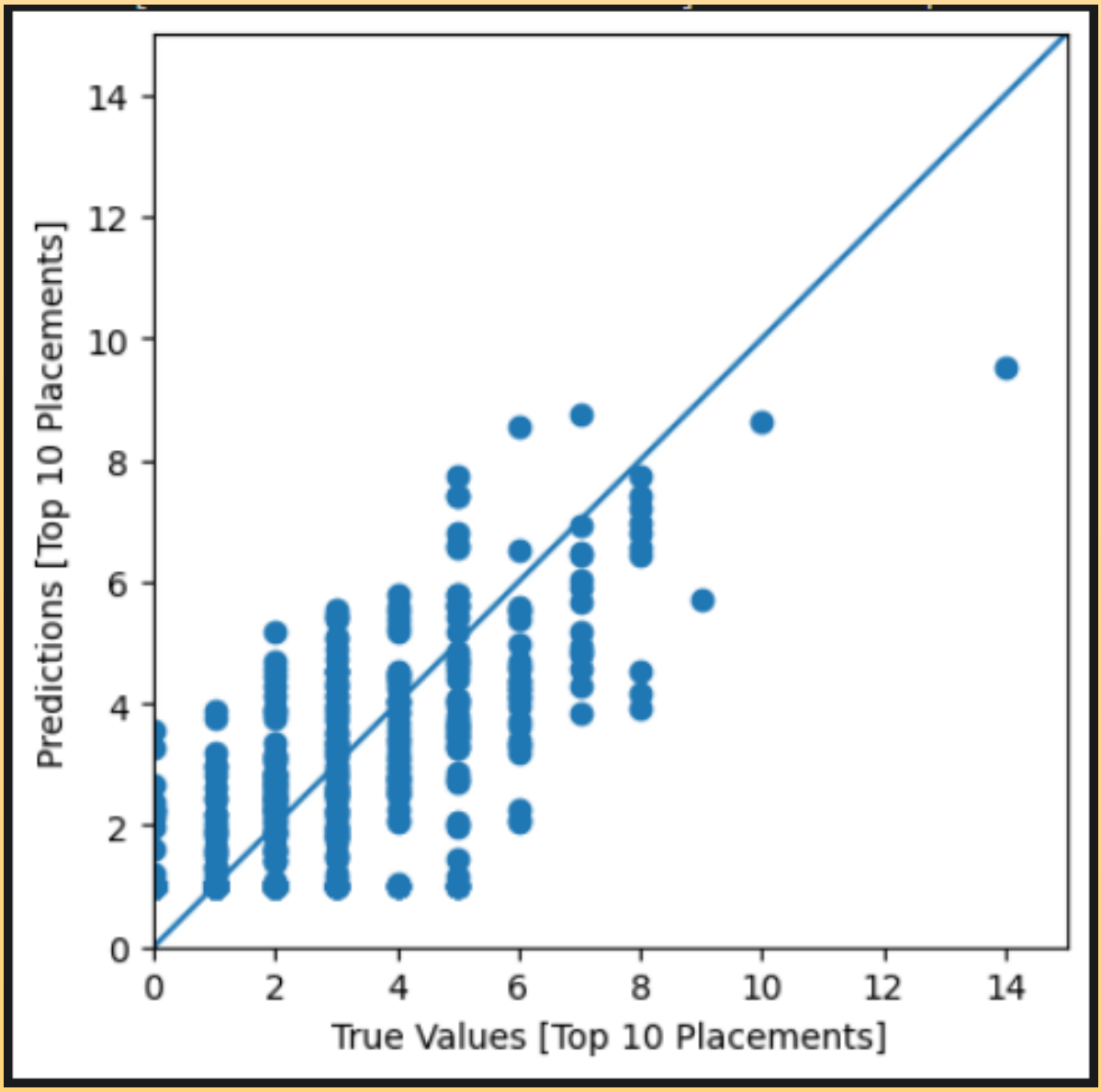
Error Model Version 1 [4-A]



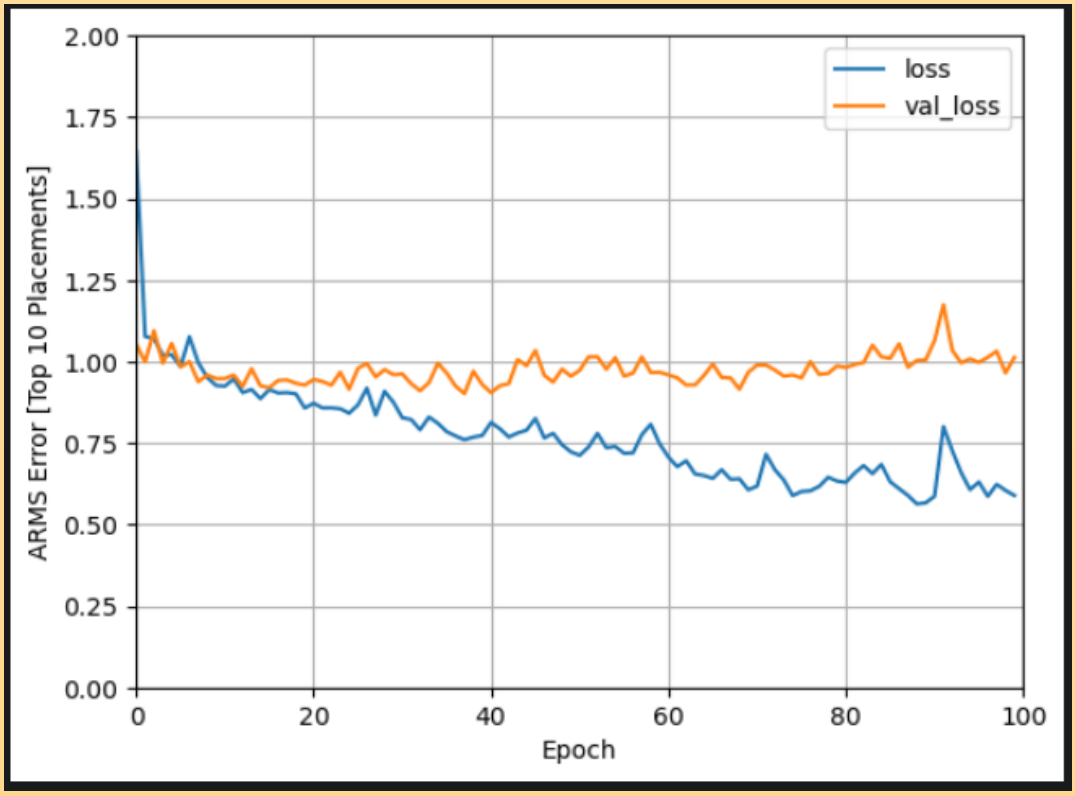
Accuracy Model Version 1 [4-B]



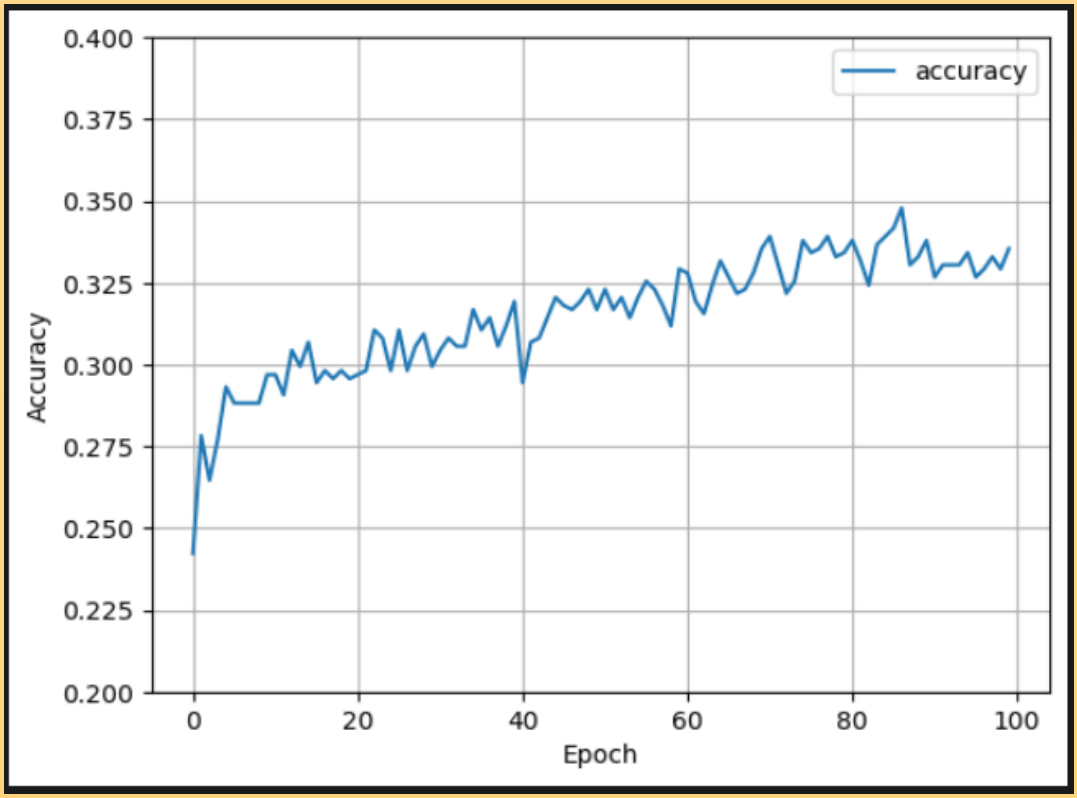
Prediction Model Version 1 [4-C]



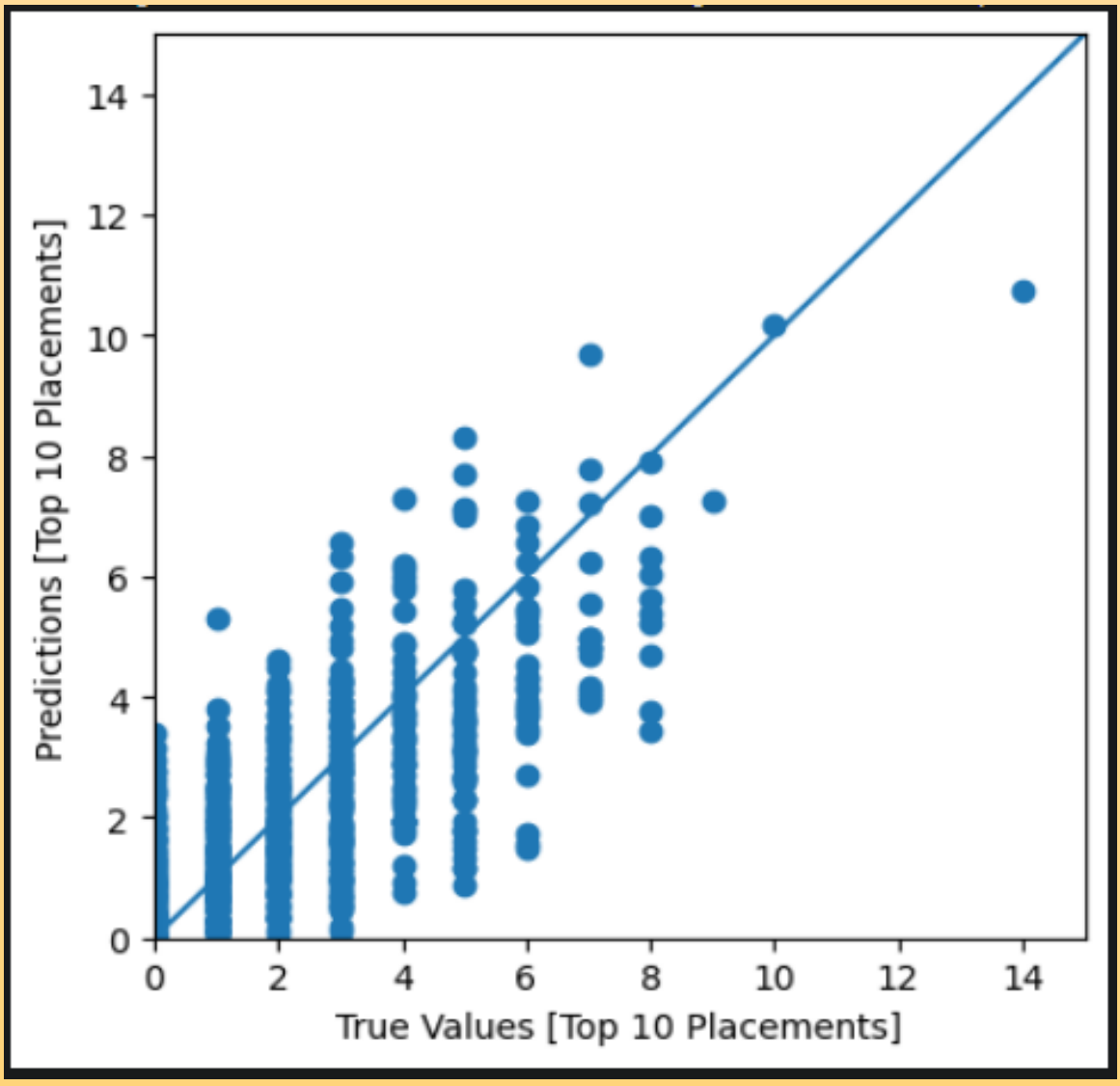
Error Model Version 2 [4-D]



Accuracy Model Version 2 [4-E]



Prediction Model Version 2 [4-F]



Grid Search Tuning V1 [5-A]

```
Epoch 1/50
6/6 [==============================] - 1s 45ms/step - loss: 70.1423 - accuracy: 0.2127 - val_loss: 72.8773 - val_accuracy: 0.2176
Epoch 2/50
6/6 [==============================] - 0s 8ms/step - loss: 36.6170 - accuracy: 0.2306 - val_loss: 31.3746 - val_accuracy: 0.1833
Epoch 3/50
6/6 [==============================] - 0s 7ms/step - loss: 19.7162 - accuracy: 0.2306 - val_loss: 22.7884 - val_accuracy: 0.2176
Epoch 4/50
6/6 [==============================] - 0s 7ms/step - loss: 13.3111 - accuracy: 0.2187 - val_loss: 2.8829 - val_accuracy: 0.2385
Epoch 5/50
6/6 [==============================] - 0s 8ms/step - loss: 6.4780 - accuracy: 0.2386 - val_loss: 5.0555 - val_accuracy: 0.1997
Epoch 6/50
6/6 [==============================] - 0s 7ms/step - loss: 4.7796 - accuracy: 0.2664 - val_loss: 2.7963 - val_accuracy: 0.2429
Epoch 7/50
6/6 [==============================] - 0s 7ms/step - loss: 3.4860 - accuracy: 0.2644 - val_loss: 4.5015 - val_accuracy: 0.2101
Epoch 8/50
6/6 [==============================] - 0s 7ms/step - loss: 3.7858 - accuracy: 0.2565 - val_loss: 3.1482 - val_accuracy: 0.2325
Epoch 9/50
6/6 [==============================] - 0s 7ms/step - loss: 2.9622 - accuracy: 0.2624 - val_loss: 3.6709 - val_accuracy: 0.2191
Epoch 10/50
6/6 [==============================] - 0s 7ms/step - loss: 3.3023 - accuracy: 0.2744 - val_loss: 3.1022 - val_accuracy: 0.2295
Epoch 11/50
6/6 [==============================] - 0s 7ms/step - loss: 3.3008 - accuracy: 0.2584 - val_loss: 3.4528 - val_accuracy: 0.2280
Epoch 12/50
6/6 [==============================] - 0s 7ms/step - loss: 2.8130 - accuracy: 0.2863 - val_loss: 2.9806 - val_accuracy: 0.2265
```

Grid Search Tuning V2 [5-B]

```
Epoch 81/100
11/11 [==============================] - 0s 4ms/step - loss: 1.2723 - accuracy: 0.2642 - val_loss: 2.4515 - val_accuracy: 0.1863
Epoch 82/100
11/11 [==============================] - 0s 4ms/step - loss: 1.7962 - accuracy: 0.2592 - val_loss: 1.7206 - val_accuracy: 0.2161
Epoch 83/100
11/11 [==============================] - 0s 4ms/step - loss: 1.7880 - accuracy: 0.2552 - val_loss: 1.2025 - val_accuracy: 0.2340
Epoch 84/100
11/11 [==============================] - 0s 4ms/step - loss: 2.2219 - accuracy: 0.2493 - val_loss: 2.0085 - val_accuracy: 0.2206
Epoch 85/100
11/11 [==============================] - 0s 4ms/step - loss: 1.5075 - accuracy: 0.2512 - val_loss: 1.3117 - val_accuracy: 0.2280
Epoch 86/100
11/11 [==============================] - 0s 4ms/step - loss: 1.2264 - accuracy: 0.2681 - val_loss: 1.1717 - val_accuracy: 0.2444
Epoch 87/100
11/11 [==============================] - 0s 4ms/step - loss: 1.1663 - accuracy: 0.2790 - val_loss: 1.6974 - val_accuracy: 0.2191
Epoch 88/100
11/11 [==============================] - 0s 4ms/step - loss: 1.4025 - accuracy: 0.2681 - val_loss: 1.7733 - val_accuracy: 0.2116
Epoch 89/100
11/11 [==============================] - 0s 4ms/step - loss: 1.4593 - accuracy: 0.2671 - val_loss: 1.1296 - val_accuracy: 0.2459
Epoch 90/100
11/11 [==============================] - 0s 4ms/step - loss: 1.1969 - accuracy: 0.2711 - val_loss: 1.2584 - val_accuracy: 0.2295
Epoch 91/100
11/11 [==============================] - 0s 4ms/step - loss: 1.3158 - accuracy: 0.2731 - val_loss: 1.2693 - val_accuracy: 0.2295
Epoch 92/100
11/11 [==============================] - 0s 4ms/step - loss: 1.1946 - accuracy: 0.2771 - val_loss: 1.4496 - val_accuracy: 0.2206
Epoch 93/100
11/11 [==============================] - 0s 4ms/step - loss: 1.3119 - accuracy: 0.2810 - val_loss: 1.1010 - val_accuracy: 0.2444
Epoch 94/100
11/11 [==============================] - 0s 4ms/step - loss: 1.2365 - accuracy: 0.2771 - val_loss: 1.4893 - val_accuracy: 0.2206
Epoch 95/100
11/11 [==============================] - 0s 4ms/step - loss: 1.3816 - accuracy: 0.2671 - val_loss: 1.7673 - val_accuracy: 0.2116
Epoch 96/100
11/11 [==============================] - 0s 4ms/step - loss: 1.4122 - accuracy: 0.2622 - val_loss: 1.5342 - val_accuracy: 0.2221
Epoch 97/100
11/11 [==============================] - 0s 5ms/step - loss: 1.3139 - accuracy: 0.2681 - val_loss: 1.5670 - val_accuracy: 0.2206
Epoch 98/100
11/11 [==============================] - 0s 5ms/step - loss: 1.3485 - accuracy: 0.2552 - val_loss: 1.1875 - val_accuracy: 0.2444
Epoch 99/100
11/11 [==============================] - 0s 4ms/step - loss: 1.2180 - accuracy: 0.2771 - val_loss: 1.2708 - val_accuracy: 0.2265
Epoch 100/100
11/11 [==============================] - 0s 4ms/step - loss: 1.2035 - accuracy: 0.2800 - val_loss: 1.0938 - val_accuracy: 0.2444
Best hyperparameters:  {'batch_size': 100, 'epochs': 100, 'hidden_layer_size': 64, 'learning_rate': 0.01, 'optimizer': 'Adam'}
Best mean validation score:  -1.3847172260284424
```

Data Analysis Table 1 [5-C]

	Year	Rounds Played in Year	Average Fairway Percentage	Average Driving Distance	Average GIR	Average Putts	Average Scrambling	Average Score	Average SG Putts	Average SG Total	SG:OTT	SG:APR	SG:ARG
0	2018	60	75.19	291.5	73.51	29.93	60.67	69.617	-0.207	1.153	0.427	0.960	-0.027
3	2018	78	71.94	289.2	68.80	29.17	64.16	70.015	-0.271	0.941	0.406	0.532	0.273
6	2018	93	71.29	295.7	71.09	29.89	54.80	70.404	0.037	0.686	0.378	0.298	-0.027
7	2018	94	70.16	295.2	68.84	29.04	61.05	70.152	0.546	1.133	0.364	0.345	-0.122
11	2018	94	69.11	295.1	71.56	29.67	60.93	70.436	-0.250	0.619	0.439	0.415	0.014
...	...	...	...	...	...	...	...	...	...	...	...	...	...
1658	2010	88	56.03	296.8	67.74	29.11	54.79	71.196	0.215	-0.004	-0.013	-0.065	-0.142
1662	2010	89	55.44	297.3	66.86	29.04	56.80	70.497	0.409	0.630	0.232	0.064	-0.074
1671	2010	79	53.49	291.7	66.51	29.36	58.86	71.171	0.171	-0.227	-0.233	-0.211	0.044
1672	2010	70	52.80	287.6	61.72	28.97	54.20	71.751	0.144	-0.520	-1.027	0.228	0.130
1675	2010	82	51.29	292.9	65.88	29.14	58.46	70.953	0.252	0.093	-0.538	0.336	0.047

Data Analysis Table 2 [5-D]

	count	mean	std	min	25%	50%	75%	max
Year	1007.0	2014.010924	2.611886	2010.000	2012.0000	2014.000	2016.0000	2018.000
Rounds Played in Year	1007.0	78.872890	14.371171	45.000	68.5000	80.000	89.0000	116.000
Average Fairway Percentage	1007.0	61.336634	5.141426	43.020	57.7050	61.490	64.9050	76.880
Average Driving Distance	1007.0	291.011718	9.097686	268.900	285.0000	290.700	296.7000	319.700
Average GIR	1007.0	65.650149	2.750847	53.540	63.8200	65.810	67.6100	73.520
Average Putts	1007.0	29.172324	0.522625	27.510	28.8100	29.150	29.5200	31.000
Average Scrambling	1007.0	58.071072	3.383710	44.010	55.8350	58.270	60.4700	67.160
Average Score	1007.0	70.937447	0.715134	68.698	70.4840	70.920	71.3680	74.400
Average SG Putts	1007.0	0.020973	0.346637	-1.475	-0.1865	0.043	0.2565	1.130
Average SG Total	1007.0	0.141721	0.706060	-3.209	-0.2650	0.149	0.5805	2.406
SG:OTT	1007.0	0.042942	0.382708	-1.717	-0.1655	0.059	0.2890	1.367
SG:APR	1007.0	0.058404	0.385300	-1.680	-0.1885	0.077	0.3090	1.533
SG:ARG	1007.0	0.019934	0.219986	-0.845	-0.1260	0.026	0.1695	0.660
Top 10 Placements	1007.0	2.327706	2.094644	0.000	1.0000	2.000	3.0000	12.000

9.3: Raw Source Code with Comments from Product Package File:

Coding Block 1

```
#The following code shows the various python libraries and packages
utilised when conducting my data analysis and neural network model
development.
#Numpy and Pandas will be used to drop my data into data frames and numpy
arrays for cleansing, filtering and general pre-modelling analysis.
#I will be using seaborn and matplotlib for graphical representations of my
data and models.
#As we can see below, my Machine Learning models would make use of
TensorFlow, SKLearn and Keras.
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder, StandardScaler
from imblearn.over_sampling import RandomOverSampler
from imblearn.under_sampling import RandomUnderSampler
from keras.models import Sequential, Model
from keras.layers import Dense, LSTM, Dropout, Input, Conv1D, MaxPooling1D,
Flatten
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import KFold
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import OrdinalEncoder

#This import will be important for the future optimization of my models
using a grid search method for hyper parameter tuning
from sklearn.model_selection import GridSearchCV

#The following code is to retain continuity in my models as reloading
models and re-doing calculations can yield varying results.
#Setting seeds for TensorFlow and Numpy will prevent these changes.
tf.random.set_seed(50)
np.random.seed(50)
tf.keras.utils.set_random_seed(50)
```

Coding Block 2

```
#loading data from my CSV file collected into data frame.
#Its worth mentioning that the CSV address is likely to change when the
final package is submitted for marking. You may need to change this for
when you run the code alongside the data.
df = pd.read_csv('FinalProjectPGATourData.csv')

# Examining the first 5 rows of data to see if all my columns are being
pulled correctly from the csv.
print(df.head())
```

Coding Block 3

```
# Changing all empty values to 0's
# Changing some incorrect variable typing's to keep consistency
df['Top 10 Placements'].fillna(0, inplace=True)
df['Top 10 Placements'] = df['Top 10 Placements'].astype(int)
df.dropna(axis = 0, inplace=True)
df['Rounds Played in Year'] = df['Rounds Played in Year'].astype(int)

#wanted to test some code to convert player names to numbers as Tensors can
only accept variables of the number class.
#I don't think this was used as I ended up dropping player names and player
code as they were not useful features for my models.
ord_enc = OrdinalEncoder()
df["player_code"] = ord_enc.fit_transform(df[["Player Name"]])
df = df.drop(columns=['Player Name'])
#player conversion code loosely based on the worked presented by (Moffitt,
2017).
```

Coding Block 4

```
#I created this initial set of graphs to get a gauge of the distribution of
each of my features.
#Knowing the shape of my data and how well the features vary from player to
player would help me to rationalise feature choices and the effectiveness
they would have on determining player performance.
#This is also very helpful in seeing that my label choice of number of 'Top
10 Placements' is highly concentrated at the lower values of 0, 1 and 2.
#Fruther explanation and analysis will be discussed in my research paper.
function, ax = plt.subplots(nrows = 5, ncols = 3, figsize=(15,15))
#No need for distribution of names or years
```

```
spread = df.loc[:,df.columns!='Player Name'].columns
spread = df.loc[:,df.columns!='Year'].columns
x = 0
y = 0
#Looping through the various columns of data and drawing histograms to
represent the distributions.
#3x5 grid design to cover all the features.
for i, column in enumerate(spread):
    p = sns.histplot(df[column], ax=ax[x][y])
    y += 1
    if y == 3:
        y = 0
        x += 1
#This type of graphing was inspired by the work of similar golf stats
analysis found in (Prater, 2018).
#Snippets of his code assisted the development of these graphs.
#Code takes a few seconds to run.
```

Coding Block 5

```
#Code written to display visually the difference in statistics between the
player who have at least place in the top 10 once and the rest of the
field.
function, cord = plt.subplots(nrows = 6, ncols = 2, figsize=(30,60))
spread = df.loc[:,df.columns!='Player Name'].columns
spread = spread[spread != 'Year']
spread = spread[spread != 'Top 10 Placements']

x = 0
y = 0
#Certain stats in gold are better the lower they are, such as number of
putts and average score.
lower_better = ['Average Putts', 'Average Score']
for i, column in enumerate(spread):
#data is being grouped by year to show the differences in stats on a year
to year basis.
    avg = df.groupby('Year')[column].mean()
    best = upperEchelonPlayers.groupby('Year')[column].mean()
    cord[x,y].plot(avg, 'o-',)
    cord[x,y].plot(best, 'o-',)
    cord[x,y].set_title(column, fontsize = 25)

    y += 1
    if y == 2:
```

```
            y = 0
            x += 1
#As we can see from the graphs generated below, Players that have had at
least a single top 10 placement are overall achieving better statistics for
every feature compared to the rest of the field.
#This is pretty consistent throughout the years and supports my feature
choices as they seem to be contributing factors to the number of top 10
placements.
#Further explanations and analysis of graphs below will be done in my
research paper.
#Comparative idea inspired by code produced in (Jong, 2019), snippets of
his code used here.
```

Coding Block 6

```
#Pairplot graphs to see correlation between features listed below.
sns.pairplot(df[['Average Fairway Percentage','Average Driving
Distance','Average GIR','Average Putts','Average Scrambling','Average
Score','Average SG Putts','Average SG Total','SG:OTT','SG:APR','SG:ARG']],
diag_kind='kde')
#Good to understand how my chosen features are connected and contribute to
success stats.
#Further analysis in full research paper.
#Its also worth noting that pair plots take a while to run. (around 40
seconds)
```

Coding Block 7

```
#Decided it was worth recreating without the excess tables and having
regression lines to see positive and negative correlations.
sns.pairplot(df[['Average Fairway Percentage','Average Driving
Distance','Average GIR','Average Putts','Average Scrambling','Average
Score','Average SG Putts','Average SG
Total','SG:OTT','SG:APR','SG:ARG']],kind='reg', corner=True,
plot_kws={'line_kws':{'color':'red'}})
#Further analysis in full research paper.
#Its also worth noting that pair plots take a while to run. (around 40
seconds).
#Some code was utilised from (Harvpan, 2018) when it came to changing
colours and layout of pair plot.
```

Coding Block 8

```
#Splitting my data fractionally between testing and training data.
#I went for a 60/40 fata split. Subject to change.
train_dataset = df.sample(frac=0.6, random_state=0)
test_dataset = df.drop(train_dataset.index)
```

Coding Block 9

```
#Separating all feature columns and target label columns.
```

```
train_features = train_dataset.copy()
test_features = test_dataset.copy()
#test and train data and test and train labels.
#labels will be number of top 10 Placements as this is a good measure of
golfer success.
train_labels = train_features.pop('Top 10 Placements')
test_labels = test_features.pop('Top 10 Placements')
#Further elaboration as to why top 10 Placements is a good metric and why I
believe its better than wins in Full Research Paper.
#split code was helped using code written by (Basic regression | TensorFlow
Core, 2023).
```

Coding Block 10

```
#Creating data type cohesion, making everything a float64.
train_features['Year'] = train_features['Year'].astype('float64')
train_features['Rounds Played in Year'] = train_features['Rounds Played in
Year'].astype('float64')
```

Coding Block 11

```
#Using a TensorFlow/Keras function to normalize continuous data.
normalizer = tf.keras.layers.Normalization(axis=-1)
```

Coding Block 12

```
#Apllying the normalizer to the training data.
normalizer.adapt(train_features)
```

Coding Block 13

```
#Construction of first deep neural network model using Adam optimizer and a
learning rate of 0.1.
#arbrtrary recommend values chosen for first model creation, will be
optimised later.
def build_and_compile_model1(norm):
  model1 = keras.Sequential([
      norm,
      layers.Dense(64, activation='relu'),
      layers.Dense(64, activation='relu'),
      layers.Dense(1)
  ])
#Calculate Absolute mean square error and Accuracy as evaluative metrics
for model.
  model1.compile(loss='mean_absolute_error',metrics = 'accuracy',
                 optimizer=tf.keras.optimizers.Adam(0.1))
  return model1
```

```
#Code has been used and changed from (Basic regression | TensorFlow Core,
2023), helpful when learning setup and basic optimization.
```

Coding Block 14

```
#Building and compiling model.
dnn_model1 = build_and_compile_model1(normalizer)
#Model summary.
dnn_model1.summary()
```

Coding Block 15

```
%%time
#The code %%time prints the wall time for this chunk of code.

#The model is being developed to see how well it fits to similar data.

history = dnn_model1.fit(
    train_features,
    train_labels,
#The following parameters are standard ones chosen for initial results
(unoptimized).
    validation_split=0.2,
#The 100 epochs refers to the 100 iterations of the complete training data
set run through.
#Batch size refers to the number of samples processed before the model is
updated.
    verbose=0, epochs=100, batch_size=200)
```

Coding Block 16

```
#A graphical line plot has been created using the history above to gauge
the Absolute Root Mean Square Error as more epochs are iterated through.
def plot_loss(history):
#Plotting the losses and val_losses.
  plt.plot(history.history['loss'], label='loss')
  plt.plot(history.history['val_loss'], label='val_loss')
#Graph parameters for visuals
  plt.xlim([0, 100])
  plt.ylim([0, 2])
  plt.xlabel('Epoch')
  plt.ylabel('ARMS Error [Top 10 Placements]')
  plt.legend()
  plt.grid(True)
```

```
#Analysis of graph will be conducted in full research Paper.
#Code has been used and changed from (Basic regression | TensorFlow Core,
2023).
```

Coding Block 17

```
#New plot to show how model accuracy changes with more iterations of the
training data being run through.
def plot_accuracy(history):
  plt.plot(history.history['accuracy'], label='accuracy')
  plt.ylim([0.2, 0.3])
  plt.xlabel('Epoch')
  plt.ylabel('Accuracy')
  plt.legend()
  plt.grid(True)
#Analysis of graph will be conducted in full research Paper.
#Code has been used and changed from (Basic regression | TensorFlow Core,
2023).
```

Coding Block 18

```
#New regression plot to show how models precited values vary from the true
values of player top 10 placements.
test_predictions = dnn_model1.predict(test_features).flatten()

a = plt.axes(aspect='equal')
plt.scatter(test_labels, test_predictions)
#True value versus predicted values.
plt.xlabel('True Values [Top 10 Placements]')
plt.ylabel('Predictions [Top 10 Placements]')
#Highest number of top 10 placemnets in the test data is 14.
lims = [0, 15]
plt.xlim(lims)
plt.ylim(lims)
_ = plt.plot(lims, lims)
#Analysis of graph in Research Paper.
#Code has been used and changed from (Basic regression | TensorFlow Core,
2023).
```

Coding Block 19

```
# Set up the hyperparameter grid for hyper parameter optimization using a
grid search technique.
param_grid = {
#Decided to iterate through the following parameters using a dictionary
array system.
    'learning_rate': [0.001, 0.01, 0.1],
    'batch_size': [100, 200],
    'epochs': [50, 100],
    'optimizer': ['Adam','rmsprop'],
```

```
    'hidden_layer_size':[64]
}
#I will explain my parameter choices in the research paper.
```

Coding Block 20

```
#Rebuilding a model for the hyper parameter tuning
#Could be made more efficient but this allows me to easily see when each of
my models is being made
#A big difference in this model is that it will be taking its parameters
from the param_grid made above and iterating through them.
def build_and_compile_model2(learning_rate, batch_size, epochs,
optimizer,hidden_layer_size ):
    model2 = keras.Sequential([
        layers.Dense(hidden_layer_size, activation='relu'),
        layers.Dense(hidden_layer_size, activation='relu'),
        layers.Dense(1)
    ])

    optimizer = tf.keras.optimizers.get(optimizer)
    optimizer.learning_rate= learning_rate
    model2.compile(loss='mean_absolute_error', metrics='accuracy',
optimizer=optimizer)


return model2
#This model similar to the initially built one was built with the aid of
(Basic regression | TensorFlow Core, 2023).
```

Coding Block 21

```
#This is where the Grid Search takes place.
grid_search = GridSearchCV(

keras.wrappers.scikit_learn.KerasRegressor(build_fn=build_and_compile_model
2),
    param_grid=param_grid,
    cv=2,
    verbose=0
)

#Fit the grid search to the training data
grid_search.fit(train_features, train_labels,
validation_data=(test_features, test_labels))

# Print the best hyperparameters and the corresponding mean validation
score to get an idea of the best parameters
print("Best hyperparameters: ", grid_search.best_params_)
print("Best mean validation score: ", grid_search.best_score_)
#The parameter settings that provide the best overall results will be used
to construct the final models.
```

```
#It is worth noting that due to hardware capabilities I have had to limit
the amount of epochs and different parameters I iterate through as this
process takes a great deal of time
#I do not recommend running this code as it can take my processor over 30
minutes to complete all the iterations of potential models.
#Or if you do decide to run this code. Pause it after you are satisfied
with its output.

#The usage of a grid search method was researched by me and helped with the
tutorials provided in (Stewart PhD, 2023).
#More code after results set produced from grid search below.
```

Coding Block 22

```
#Now that my grid search is finished running it has given me the optimal
parameter tuning to give the best results
#I will now create a new final model using these refined settings and see
how my evaluative metrics hopefully improve.
def build_and_compile_model3(norm):
  model3 = keras.Sequential([
      norm,
      layers.Dense(64, activation='relu'),
      layers.Dense(64, activation='relu'),
      layers.Dense(1)
  ])

  model3.compile(loss='mean_absolute_error',metrics = 'accuracy',
                 optimizer=tf.keras.optimizers.Adam(learning_rate= 0.01)
                 )
#Using new learning rate
  return model3
#This model similar to the initially built one was built with the aid of
(Basic regression | TensorFlow Core, 2023).
```

Coding Block 23

```
#Building and compiling model.
dnn_model3 = build_and_compile_model3(normalizer)
#Model summary.
dnn_model3.summary()
```

Coding Block 24

```
#A graphical line plot has been created using the history above to gauge
the Absolute Root Mean Square Error as more epochs are iterated through.
def plot_loss2(history2):
  plt.plot(history2.history['loss'], label='loss')
  plt.plot(history2.history['val_loss'], label='val_loss')
  plt.ylim([0, 2])
  plt.xlim([0, 100])
  plt.xlabel('Epoch')
```

```
  plt.ylabel('ARMS Error [Top 10 Placements]')
  plt.legend()
  plt.grid(True)
#Analysis of graph will be conducted in full research Paper.
#Code has been used and changed from (Basic regression | TensorFlow Core, 2023).
```

Coding Block 25

```
#Graphical plot has been created to see how my model's accuracy will change through each iteration of the training data. (epochs)
def plot_accuracy2(history2):
  plt.plot(history2.history['accuracy'], label='accuracy')
  plt.ylim([0.2, 0.4])
  plt.xlabel('Epoch')
  plt.ylabel('Accuracy')
  plt.legend()
  plt.grid(True)
#Analysis of graph will be conducted in full research Paper.
#Code has been used and changed from (Basic regression | TensorFlow Core, 2023).
```

Coding Block 26

```
#Predictions are being plotted against the actual values to see how well my model is performing.
test_predictions2 = dnn_model3.predict(test_features).flatten()

b = plt.axes(aspect='equal')
plt.scatter(test_labels, test_predictions2)
plt.xlabel('True Values [Top 10 Placements]')
plt.ylabel('Predictions [Top 10 Placements]')
lims = [0, 15]
plt.xlim(lims)
plt.ylim(lims)
_ = plt.plot(lims, lims)
#Analysis of graph will be conducted in full research Paper.
#Code has been used and changed from (Basic regression | TensorFlow Core, 2023).
```

Coding Block 27

```
#Made an array of values for a made up player with average PGA Tour Pro statistics just to see whether my model can give me a reasonable prediction.
Prediction_data = np.array([[2020, 80, 65, 315, 62, 30, 55, 72, 0.2, 0.41,-0.1,0.1,0.4]])
#Make predictions on the new data using the trained model
predictions = dnn_model3.predict(Prediction_data)
#Print the predicted values for the number of times a player is predicted to place in the top 10 for tournaments across the year.
print(predictions)
```

Coding Block 28

```
#I thought it might be interesting to experiment with the model to see how
it would interpret newer data from the current world No.1 player Jon Rahm
in his previous season.
#Bearing in mind the actual number of top 10 Placements he had in 2022 was
8.
Prediction_dataJonRahm = np.array([[2022, 70, 65, 321, 72, 29, 57, 69.7,
0.367, 1.66,1.025,0.363,-0.8]])
predictionsJonRahm = dnn_model3.predict(Prediction_dataJonRahm)
print(predictionsJonRahm)
#This result seems to be underestimating the performance, further analysis
of this will be studied in full research paper.
```

9.4 ReadMe File Code Running Instructions

Neural Network based project researching how to predict PGA Tour player performance based on a variety of playing statistics from 2012 to 2018.

I have chosen to predict the number of top 10 placements a player gets in tournaments throughout the year.

The data collected has been credited to the PGA Tour ('https://www.pgatour.com/stats') and all referencing has been completed in the python coding file using Harvard referencing.

All the code is commented in detail and has specific instructions for sections of code that have run requirements or take a long time.

It is worth noting that to run the code, I used a Jupyter Notebook and several imports of various python libraries such as TensorFlow, Keras, Numpy and Pandas.

Imports have been listed at the start of the coding section in the python file.

Furthermore, it is also worth mentioning that you may need to change the address of the csv data file in the code document depending on where you choose to run it.

This will make sure that the data is used and imported correctly when you run the code.

I will also link the GitHub repository I used to keep version control of my work:
https://github.com/AdamKhanzada/FinalProject

Data File can be located using the GitHub repository link and from the final product package submitted on Moodle.