National Basketball Association Player Future Performance Prediction Model By

TAN JIAN SEAN

TP046380

UC3F1911 CS(DA)

A project submitted in partial fulfillment of the requirements of Asia Pacific University of Technology and Innovation for the degree of

BSc (Hons) in Computer Science specialism in Data Analytics

Supervised by DR. POOLAN MARIKANNAN BOOMA

2nd Marker: DR. PREETHI

Aug-2020

Acknowledgement

I would like to thank those who plays a part in my academic accomplishments. First and foremost, my family who always have my back no matter the situation. I will be nowhere near where I am without you.

I will also like to thank my supervisor, Dr. Booma Poolan Marikannan, who makes time for me and guides me throughout the research. Last but not least, I want to thank the other lecturers and my classmates who had help me in different ways. Thanks for all the support.

Table of Contents

CHAPTER 1: INTRODUCTION TO THE STUDY

1.1 Background to the project

The sports world is an ever-expanding market. In 2014, the global sports industry has an estimated worth of 1.5 trillion in USD (Thabtah, Zhang and Abdelhamid, 2019). With the improvement of living standards around the globe, the wide spread of internet and mobile devices, more and more people are looking for an entertainment, a hobby to invest their time in. Sports is a huge beneficiary of the trend. With more and more fans around the world supporting their favourite players or teams, the sports market is more valuable and potential than ever. Just as the research of Golovko, Leonov and Pysarenko had pointed out, the global sports industry expected the figure of revenue to grow to 145 billion (Golovko, Leonov and Pysarenko, 2015). Sports brands who produces and sells sports-related products such as sports equipment, sports drinks, sports clothing finds potential in the increasing fanbase and globalized market. One of the popular ways of marketing is to sign brand ambassadors. However, with the market being dominated by big brands (Barroso Duarte, 2018), it is difficult for the upcoming company to sign established star players. The National Basketball Association, also known as the NBA, is another league of the case mentioned above.

1.2 Problem statements

As stated in 1.1 above, the global sports market is being dominated by well established companies like Nike, Adidas and Under Armour. The phenomenon extends to the signing of the players, where stars and superstars are ambassadors of the sports giants. For the small brands to compete, they can only resort to signing young, potential players that may turn into a star in the near future, just like Under Armour did in 2013. However, the question is: How many Stephen Curry are there to be found? What if Under Armour wasn't that lucky and Stephen Curry did not blossom into a superstar?

1.3 Rationale

Therefore, smaller companies are left with a choice of signing young emerging players that may turn out to be either a future all-star or a complete bust. With the proposed system, we look to predict the young, potential players future stardom and allows companies to sign them before their blossom into superstars, therefore owning a brand ambassador with a solid fanbase at a lower cost, but still will bring revenue to the investing companies.

1.4 Potential benefits

1.4.1 Tangible benefits

-cut down costs of signing a star player as brand ambassador

-lower chances of sunk costs by signing players that may not be good in the future

-allows start-up or smaller brands to advertise their brands by signing potential stars

1.4.2 Intangible benefits

-brings competition to the business field

-allows underrated professional players to be discovered

1.5 Target users

Emerging company who are looking to sign young NBA ambassador while they are young, less known and cheap.

1.6 Scope and objectives

1.6.1 Aim

The aim of this study is to deliver a predicting model that predicts future stardom of emerging player, thus delivering chance of competition for less established brands.

1.6.2 Objectives

In order to achieve the aim defined above, the following objectives are proposed:

- To collect and preprocess a dataset of historical rookie and sophomore players' (all-star and non-all-star) data, which includes traditional stats as well as advanced stats related to players performance;
- To build and train a prediction model that is capable of predicting if a rookie or sophomore player will be an all-star in 7 years' time
- To test and compare the accuracy of the final model with similar models

1.6.3 Deliverables - Functionality of the proposed system

A prediction model that can predict a future stardom of a young player in less than 7 years, based on the historical performance stats.

1.6.4 Nature of Challenges

The challenge here is mainly about the accuracy of the model, as in the real world, the signing of an ambassador can be costly, so signing an inaccurately predicted player will be a terrible decision for the company financially. The accuracy of the prediction model can be affected by a number of factors, namely the type of data to use, choices of different prediction models, also choosing the measurements for the player performance.

1.7 Overview of this report

The following **Chapter 2** will first touch on topics, from literature, like importance of ambassadors to a brand, sales distributions of brands in the sports market, how statistics is implemented in the NBA, technology utilized to capture rich data in the NBA, the general sports analytics framework and the various implementation of sports analytics. Then, three similar systems are studied and evaluated in terms of aim, methods and performance. **Chapter 3** discuss the language, IDE (Interactive Development Environment), libraries or tools as well as Operating System of this project which are all chosen after extensive research. **Chapter 4** dives into the chosen methodology of this study, which is CRISP-DM (CRoss-Industry Standard Processing for Data Mining), also how each phase relates to this study. Finally, **Chapter 5** is about conclusions and reflections upon completion of this Investigation Report.

1.8 Project Plan

TASK ID	TASK NAME	DURATION	START DATE	END DATE	STATUS
TSK-1	**Introduction to Study**	**5 DAYS**	**16/12/2019**	**20/12/2019**	**DONE**
TSK-1-1	Background to the project	1 DAY	16/12/2019	16/12/2019	DONE
TSK-1-2	Problem statement	1 DAY	17/12/2019	17/12/2019	DONE
TSK-1-3	Rationale	1 DAY	17/12/2019	17/12/2019	DONE
TSK-1-4	Potential Benefits	1 DAY	18/12/2019	18/12/2019	DONE
TSK-1-5	Target Users	1 DAY	18/12/2019	18/12/2019	DONE
TSK-1-6	Scope and Objectives	1 DAY	18/12/2019	18/12/2019	DONE
TSK-1-7	Overview of this project	1 DAY	19/12/2019	19/12/2019	DONE
TSK-1-8	Project Plan	1 DAY	20/12/2019	20/12/2019	DONE
TSK-2	**Literature Review**	**7 DAYS**	**23/12/2019**	**31/12/2019**	**DONE**
TSK-2-1	Introduction	1 DAY	23/12/2019	23/12/2019	DONE
TSK-2-2	Domain Research	2 DAYS	24/12/2019	25/12/2019	DONE
TSK-2-3	Similar Systems	3 DAYS	26/12/2019	30/12/2019	DONE
TSK-2-4	Summary	1 DAY	31/12/2019	31/12/2019	DONE
TSK-3	**Technical Research**	**12 DAYS**	**1/1/2020**	**16/1/2020**	**DONE**
TSK-3-1	Programming Language chosen	2 DAYS	1/1/2020	2/1/2020	DONE
TSK-3-2	IDE chosen	3 DAYS	3/1/2020	6/1/2020	DONE
TSK-3-3	Libraries chosen/ Tools chosen	3 DAYS	7/1/2020	9/1/2020	DONE
TSK-3-4	Operating System chosen	4 DAYS	10/1/2020	15/1/2020	DONE
TSK-3-5	Summary	1 DAY	16/1/2020	16/1/2020	DONE
TSK-4	**Methodology**	**6 DAYS**	**17/1/2020**	**23/1/2020**	**DONE**
TSK-4-1	Introduction	1 DAY	17/1/2020	17/1/2020	DONE
TSK-4-2	CRISP-DM	3 DAYS	20/1/2020	22/1/2020	DONE
TSK-4-3	Summary	1 DAY	23/1/2020	23/1/2020	DONE
TSK-5	**Conclusions and Reflections**	**1 DAY**	**24/1/2020**	**24/1/2020**	**DONE**

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

This project is driven by the curiosity of the feasibility and applicability of sports analytics towards the business aspect of the sports field. In the literature study part of this report, we dive into the domain research to understand the sports market field, the importance of sports brand ambassadors, the implementation of statistics in the National Basketball Association, the technologies used to capture the data as well as the general sports analytics framework that is implemented in the industry. The researcher also studied on Similar systems in order to produce a prediction model that fits the aim of the project best.

2.2 Domain research

2.2.1 Importance of Brand Ambassadors

The importance of a brand ambassadors speaks volume in Stephen Curry and Under Armour's partnership. In 2013, the sports equipment company signed the then young player to a 2-year shoe deal. By the end of 2015, the company's footwear segment made a 57% leap in growth, while revenues were increased by 95% (Team and Speculations, 2016). The athletic wear giant's success in recent years has a sure correlation with Stephen Curry's career finally going full bloom. Before the contract was signed, Curry is a promising talented player but with major concern in terms of his injury prone legs. Big brands like Nike was reluctant to sign him to any big deal despite his obvious upside and opted for other player with less history of injury. Fast forward to 2019, Curry is now one of the most successful individuals in the National Basketball Association. Awards like 3x NBA Champion, 2x Most Valuable Player, All-time playoff 3PM leader, single season 3PM record holder are just part of his impressive resume (NBA Stats, 2019).

2.2.2 Domination of Big Brands in the Market

It is no secret that the sports market, specifically in the footwear & apparel field, is being dominated by giants namely Nike, Adidas and Under Armour. As shown in Barroso

Duarte's research, even though Adidas, Under Armour and Puma are following Nike in the ranking, it isn't even close in real world market share (Barroso Duarte, 2018).

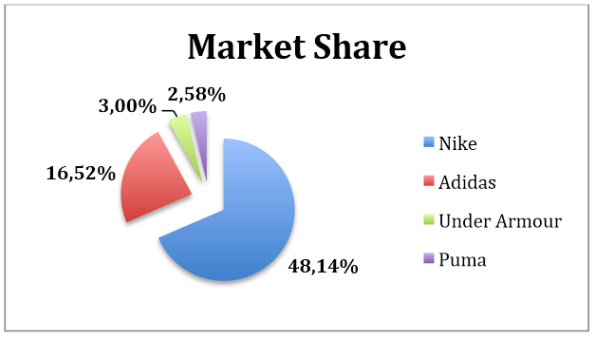


Figure 1:Market Share of Brands in Footwear & Apparel in 2017 (Barroso Duarte, 2018).

Moreover, Rovell also added that in major sports league like NBA and NFL, sports giant like Nike always manage to pull off official apparel deals (Rovell, 2015), resulting in all players in the league shown on TVs are in the Swoosh logo uniform (Garcia, 2018). To add to that, almost all of the biggest stars in their respective sports field are endorsers of Nike or Adidas. To name a few, Lebron James (Basketball), Serena Williams (Tennis), Tiger Woods (Golf) and Cristiano Ronaldo (Soccer) are all Nike ambassadors who frequently appears on advertisements.

2.2.3 Statistics in the NBA

The NBA is one of the most data-oriented sports leagues in the world, with statistics like points, assists, rebounds, steals and blocks being recorded since the 1973/1974 season (NBA.com/Stats, 2019). Twenty plus years later, advanced stats like Pace, Offensive and Defensive Efficiency are introduced, allowing analyst to look deeper into a team's performance, beyond the basic stats of a team that may be deceiving based on different context. To simplify, Pace is about measuring a team's (or player's) defensive and offensive stats by looking at them on a per-possession basis instead of number of points scored or allowed per game, since each team (or player) has a unique playing style and Pace. For

example, as stated by Ben Taylor, a comparison between advanced scoring stats of Oscar Robertson in 1962 and Kevin Durant in 2012 reveals the difference pace can make (Taylor, 2019).

Table 1: Scoring Efficiency comparison (Taylor, 2019)

Players (Season)	Points per game	Pos per 48 min	Pts per 75
Oscar Robertson (1962)	31	129	20.1
Kevin Durant (2012)	28	93	28.1

As shown in the table above, when looking at the points per game stats in the respective seasons, Robertson seems to be the superior scorer here. However, Robertson plays at a much faster pace compare to Durant. Upon adjusting the scoring average to the pace, Robertson's advantage seems to disappear as Durant turns out to be a far more effective scorer with an 8-point advantage per 75 possessions.

2.2.4 STATS SportVU Tracking System

Data analytics have changed the landscape of NBA, with the modern tools being implemented into the basketball court. In the year of 2009, the NBA started implementing the STATS SportVU tracking system and when on to be the first professional sports league to support player tracking in all games played (NBA, 2016). The system consists of six cameras, installed in a basketball arena, and is capable of tracking real-time position of all the players on court and the ball, 25 times per second. The system provides a rich data containing speed distance, player separation and ball possession. It creates an all new dimension of spatial data, which is a lot different than the conventional stat sheet data like points, rebounds and assists. One of the earliest teams to apply the SportVU system was Golden State Warriors, who have experienced great success for the past years.

Figure 2: Presentation of the SportVU system (STATS, 2019)

The rich combination of data gives the coaching staff, players and analyst the most comprehensive and sophisticated view ever of the game. The system can be used to prevent injuries, make better coaching decision and even maximize Player Performance (STATS, 2019). One of the products that comes from the ground-breaking system is an advanced statistic called Expected Possession Value (EPV) (Bornn et al., 2014), which as stated by Beardsley, is a huge key to the recent success for the Golden State Warriors (Beardsley, 2017).

2.2.5 The General Sports Analytics Framework

Sports Analytics, as defined by Alamar, is "the management of structured historical data, the application of predictive analytical models that utilize that data, and the use of information systems to inform decision makers and enable them to help their organizations in gaining a competitive edge on the field of play" (Morgulev, Azar and Lidor, 2018). The definition outlines the three main components which are data management, analytical models and information systems, and that the purpose of sports analytics is to aid the decision makers of the organization to gain an upper hand against the competition. The framework of sports analytics can be perceived as Figure 2.5.1 below:

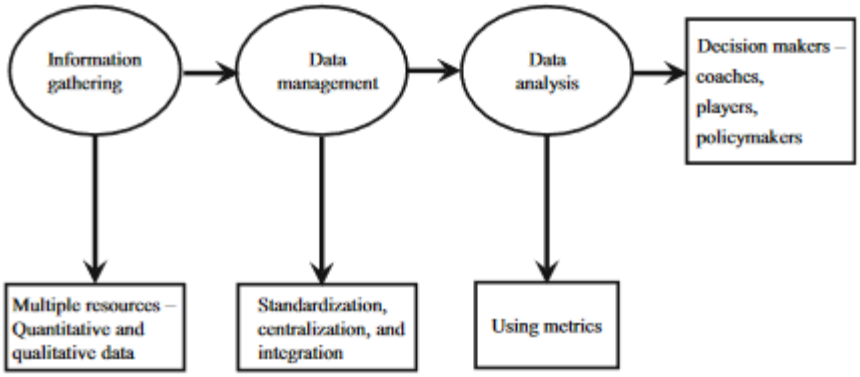


Figure 3: Sports Analytics Framework (Morgulev, Azar and Lidor, 2018)

The arrows in the figure demonstrates how the data flows through the components of the framework. All the data, coming from sources, has to be prepared at the data management stage. When the data is ready, it is sent to information systems or analytic models. Analytic models can act either as a provider of processed data to the information system or as an ad-hoc function to answer questions of the decision maker. The information system will present the resulting knowledge or information to the decision maker in an efficient and clear manner.

2.2.6 Implementation of Sports Analytics

With the emergence of sports analytics in the basketball field, a great number of research and projects had taken place with various approach and implementation in mind. The mentioned approach include study on prediction, correlation between different attributes, performance or strategy analysis as well as officiating assistance.

In the study by Varol Onunr Kayhan and Alison Watkins, they attempted to predict the outcome of the game real-time by comparing the snapshot of the game to snapshots from historical games. The snapshots taken usually includes point differential between the home and away team, it may also include other attributes in variations of models for the evaluation of the prediction models (Kayhan and Watkins, 2018). Another study (Ahmadalinezhad, Makrehchi and Seward, 2019) managed to predict the performance of different National

Basketball Association lineups from seasons throughout 2007-2016 by using combination of machine learning and network analysis. The proposed method, which uses graph theory and Inverse Squared Metric, was able to achieve average accuracy of 68%, which is a 10% improvement over the baseline result,

There are also studies that looks to find out correlations between certain attributes in the hoop game. For instance, R. Metulini, M.Manisera and P. Zuccolotto worked on a project that aims at discovering the relations between positions and movements of player in conjunction with the performance of the team collectively. Various techniques and methods are borrowed from different concepts such as network and complex systems, machine learning, statistics et cetera. They were able to have some interesting finding like players are more widespread on the offensive end compared to the defensive end (Metulini, Manisera and Zuccolotto, 2017). Another research by R. Metulini dives into the correlation between player position and team shooting performance using clustering technique (Metulini, 2018).

Another implementation of sports analytics on basketball is on the game play strategies aspect, where analyst like M. Manisera, R. Metulini and P. Zuccolotto uses spatial data to look into the players' position on court, then evaluate game plans by the performance or efficiency before looking to make some improvement based on their findings (Manisera, Metulini and Zuccolotto, 2019). On the other hand, J. Safir in his study *How Analytics, Big Data, and Technology Have Impacted Basketball's Quests to Maximize Efficiency and Optimization*, discussed how sport analytics had influenced the way the game is being played as the overall game shifts towards more of a pace-and-space style compared to the more stagnant style in the 90s. The main reason being analyst finds that the longer 3pt shot attempt is often efficient than long 2pt shot attempt as it is 1.5 times as valuable. Thus, with the number guys suggesting players to shoot for 3s, comes the dawn of the 3pt era (Safir, 2015).

With the game getting more and more fast paced, the officiating is also becoming increasingly difficult, especially on offensive foul calls. In order to improve the accuracy of the calls the officials make, K. Wang and R. Zemel ventured out to perform classification on the mentioned play calls by applying neural network models on data coming from the SportsVU system. Their study shows that the proposed models, especially the more refined

Recurrent Neural Network, is able to perform well in terms of classifying the play call while only learning from limited data (Wang and Zemel, 2016).

2.3 Similar Systems

There are a few projects with either similar intentions, context or implementations in comparison with this project, namely *Forecasting basketball players' performance using sparse functional data* (Vinué and Epifanio, 2019), *Predicting the performance of the players in NBA Players by divided regression analysis* (Goh et al., 2019) and *Predicting All Star Player in the National Basketball Association using Random Forest* (Soliman et al., 2017).

2.3.1 Forecasting basketball players' performance using sparse functional data

The first study aims to forecast the future performance of NBA player from the available sparse functional data. Since sparse data is very common in sports, the researchers think it will be reasonable to develop a methodology to perform prediction or forecast of player performance by using it. The two main metrics used are Box Plus-Minus (BPM) and Win Share (WS). Box Plus-Minus is chosen as it boils down a player's on-court contribution to a team, through points gained or lost, into a single value, and is available from the 1973-1974 season. WS calculates a player's contribution to the team's win and can counter the missing aspects from the BPM side of view, for example a player who puts up impressive stats but does not bring the team wins.

The team implemented two methodologies, namely Regularized optimization for prediction and estimation with sparse data (ROPES) and principle components analysis through conditional expectation (PACE). The ROPES approach is introduced by Alexander Dokumentov and Rob Hyndman and is intended to solve problems which involves the decomposition, smoothing and forecasting of two-dimensional sparse data (Vinué and Epifanio, 2019).

On the other hand, PACE relates to the concept of functional principle components analysis (FPCA), which is often used to reduce the dimension of data when observing random

curves. In order to counter the fact that usual computational methods are inefficient against irregularly spaced sparse functional data, a version of FCPA (Vinué and Epifanio, 2019), developed by Fang Yao et al, is implemented in this study as well. Another important concept that is brought up in the study is Archetypoid Analysis (ADA). To put it in a simplified manner, ADA is an analysis carried out to classify the data, in this case the players, into clusters of similar characteristics to retrieve more accurate and reliable results. In this study, ADA is performed in the form of Functional Archetypoid Analysis (FADA). Throughout ADA, coefficient values of the players were produced, and the product is used to classify the selected players into similar groups.

The results indicated that the ROPES methodology is capable of producing forecasting competitive relative to similar methods like Career-Arc Regression Model Estimator with Local Optimization (CARMELO). The ROPES and PACE methods had also been compared with simpler methods like average or naïve methods. PACE achieved the best results overall, with ROPES performing better than the simple methods.

2.3.2 Predicting the performance of the players in NBA Players by divided regression

In their study to attempt prediction of NBA players performance, Y.L. Goh, Y.H. Goh, R.L.L. Bin and W.H. Chee chose the predictive method of divided multiple linear regression model. The idea is to split the tremendous amount of data into *n* subsets of data. Multiple linear regression model is then installed into each of the sub data sets, calculating the coefficients of regression parameters. By going down the divide and conquer route, the computation load is eased by a significant amount.

The measurement that is selected to for the prediction is PTS, which is the amount of total points scored by a particular player in during the season. The 3 independent variables are FG (total field goals made), FT (total free throws made) and MP (total minutes played), with all being data dated from the 1997-1998 season till 2016-2017. Another concept mentioned in the paper is variance inflation factor (VIF), which is introduced to identify multicollinearity between variables. Multicollinearity is when two or more predictor variable is highly correlated with one another and should be avoided in a multiple linear regression

model as it causes serious instability of regression coefficients, ultimately leads to false or misleading results, as per Hall, Fienberg and Nardi (Goh, Goh, Raymond and Chee, 2019).

In the first implementation of the divided multiple linear regression models, the models are all safe from multicollinearity but suffers from the data not fitting a regression model, but more of a funnel shape as the response variable does not have a linear relationship with the regressors. To counter that, the Box-Cox method is used to transform the response variable (PTS). After that, the regression coefficient of the 3 variables (FG, MP, FT) are collected from the 5 sub data sets. Then, the mean of each regression coefficient is calculated before the regression model formula is produced.

2.3.3 Predicting All Star Player in National Basketball Association using Random Forest

This research tries to predict the selection of all-star players in the NBA through a data analytics rather than the traditional route of a voting system. The main activities of the study included identifying the key performance indicator (KPI) of a all-star player, then make prediction on the selection of all-star (all-star or no all-star) based on the KPIs.

The data utilized are divided into 3 sets of data, namely players' personal data, all-star players' data and non-all-star players' data. Soliman et al came up with two approach of prediction, each uses the KPI of player performance per minute (approach 1) or player performance per average of total minute (approach 2). The difference in the data will be that approach 1 essentially measures the performance of the player on a career basis, while approach 2 does it on a season basis.

The results from the Random Forest model appeared to be impressively accurate. Upon evaluation of the two approaches using the Area Under Receiver Operating Characteristic (ROC) Curve as well as Confusion Matrix, approach 2 came out on top in terms on accuracy (approx. 95% to approx. 93%). One of the main contributing factors can be that fact that KPI in approach 2 takes into consideration of more variable that create, protect or materialize a possession such as Offensive or Defensive Rebounds, Blocks, Turnover and more, through feature selection. On the other hand, approach 1 only looks at Field Gold attempts, 3-pt attempts, Free Throw attempts and other less relevant variables.

2.4 Summary

Through the paper study above, we found similarities and differences compared to this proposal of National Basketball Association Player Future Performance Prediction Model. The first paper aims to predict the career arc of a player using attributes like Box Plus-Minus and Win Share, and the results are competitive relative to other similar methods like Career-Arc Regression Model Estimator with Local Optimization (CARMELO). However, the prediction target is different from this study as the prediction of career arc is a lot different from the projection of the blossom of a player into a star player, as the length of a career is often longer than players' prime years.

The second study tried to predict the total points of a player will score in a season using divided multiple linear regression. The variables used are Field Goals Made, Free Throws Made and Minutes Played. However, the measurement of players performance through points scored in a season is somehow questionable, as the value or impact of a player is more often than not beyond just points.

Aside from prediction of NBA players performance, the last research aligned with this study in some other areas, including the methodology of CRISP-DM, and the interest in predicting the all-star validity of a player. To quantify the value of the players, the researchers looked through sets of data that measures players impact from various aspects of the games like Rebounds, Assists, Steals, Blocks, stats that is included in the second study and then some. The Approach 1 of the study look at statistics values that is measured on a career basis. Approach 2 utilized attributes that is measured on a season basis, similar to the second study, but included a more comprehensive sets of features such as Offensive Rebound, Turnovers, Assists and Steals. The end results confirm that Approach 2 can predict all-star more accurately, which matches the fact that all-stars players are mostly selected in seasons where they perform relatively better than their overall career. The slight difference in approach between said study and National Basketball Association Player Future Performance Prediction Model is that the latter aims to prediction future stardom of young players.

CHAPTER 3: TECHNICAL RESEARCH

3.1 Programming language chosen

3.1.1 Python

According to the Python Organization, “Python is powerful… and fast; plays well with others; runs everywhere; is friendly & easy to learn; is Open” (Python.org, 2020). The quote basically sums up all the benefits and strengths of the language. The language is not only free to use but is also one of the easiest language to understand or learn which can only help when a data scientist is working on a solution for tricky data problems. Another reason that the open-source language is beloved is because it features a rich set of libraries and tools for data science, as it even has dedicated libraries for different phases of data science projects (Kumar Bachheriya, 2019). Moreover, there is a huge community base where personnel who work on data science can ask answer questions about queries (Srivastava, 2019). All of the above makes for a top choice of language for any data science projects.

3.1.2 R

R is another open-source language that is often used in data related projects. That means there is over 8000 packages that is of contribution of the network readily available. Also, R comes in an environment system, that is, it features a “fully planned and coherent system” (R-project.org, 2020). Within the environment, one can find data handling and storage facility, data analysis tools, graphical facilities for data analysis and display methods and more.

One of the biggest features of R for data people is the graphical representation ability. R features some library that can put out aesthetically pleasing graphs or plots (Team, 2019). There are also packages that makes generating the reports for results of data analysis as simple as it can be.

However, there are areas that R shies in comparison with Python. For starter, R has a steeper learning curve than most of the languages, not to mention Python that is one of the

easiest languages to pick up. Moreover, R uses more memory than Python as it stores objects in physical memory. R also lacks basic security, which is an essential component for Python, which explains why the former cannot be embedded into a web application (Team, 2019).

Aside from above, both of the languages are highly compatible with other languages such as C, C++ or various database management system. Not to mention they are both platform independent and can run across Windows, Linux and Mac. With all the similarities and differences considered, Python will be the language of choice for this project for it is easy to understand and learn, better support of the community as well as the drawbacks of R mentioned above.

3.2 IDE (Interactive Development Environment) chosen

3.2.1 PyCharm

PyCharm is an IDE dedicated for Python. The IDE is one of the easiest to install and setup (Hooda, 2020). Aside from standard IDE features such as code editor, error highlighting, powerful debugger, UI customization and plugins, PyCharm also supports integration to tools and libraries like Numpy and Matplotlib and panda (Henrique Vasconcellos, 2018). PyCharm also comes pre-installed with an intelligent assistant that allows smart code completion, code inspection, on-the-fly error highlighting and quick-fixes, automated code refactoring as well as a variety of navigation capabilities (Sakuragi, 2019). Additional strengths of PyCharm includes active community support and easy to use. PyCharm is also ideal for scientific implementation of Python, and not a lot of IDE can be said so. However, like other products by Jetbrains, it requires payment to activate.

3.2.2 Spyder

Spyder was created mainly for the likes of data scientists or data engineers. Spyder features advanced level features for edit, debug and data exploration. The pros of Spyder includes syntax highlighting and auto code completion feature. It also works well with multi-language editor and supports extended plugins to enable new level of improvisation. The

downside of Spyder includes the inability to disable specific warnings and potential slowdown when too many plugins are invoked (Softwaretestinghelp.com, 2020).

3.2.3 Jupyter Notebook

Jupyter Notebook is a web server-client application which can be utilised both offline and online (Toomey, 2016), in the browser of the user's choice. It is, at the same time, more feature-pack than a text editor, but less complex than a full-blown IDE, making it relatively lightweight and easier to manipulate. This makes it ideal to work with dataset (data cleaning, data exploration or plotting) (Kazarinoff, 2020). Another feature of this IDE is that it allows HTML components like images and videos to be presented along with the code output. The output can then be exported into PDF, HTML or python format, making it perfect for data science projects (Henrique Vasconcellos, 2018).

One way of running Jupyter Notebook is through Anaconda Navigator (Nerds, 2018). The GUI allows users to access the Jupyter Notebook with a click of a button after installation.

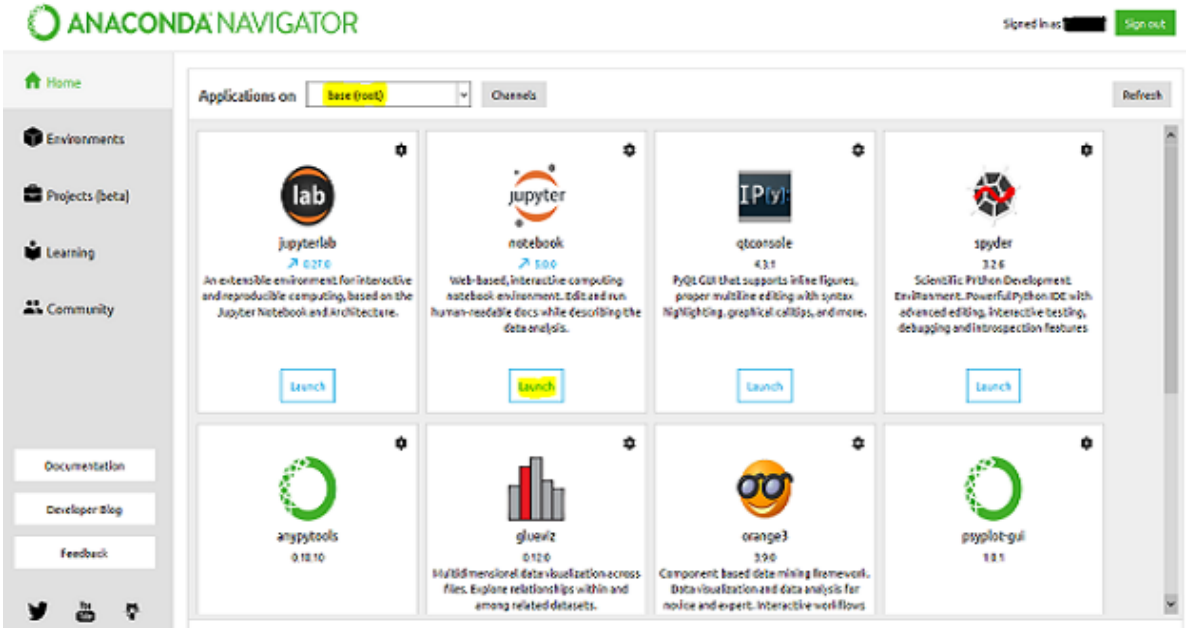


Figure 4: Anaconda Navigator GUI

Overall, Jupyter Notebook is selected as the IDE for this project for its easy installation and setup, easiness to manipulate and the fact that the researcher is more familiar with the platform.

3.3 Libraries chosen / Tools chosen

The list of libraries or packages that are required for this project:

1. **Pandas**. Pandas is a package that providing high-level, easy-to-use data structures and data analysis tools that enhance working experience with structured and time series data (PyPI, 2020).
2. **NumPy**. NumPy is a general-purpose array-processing package. It provides N-dimensional array objects and useful capabilities to deal with linear algebra, Fourier transform as well as random number capabilities (PyPI, 2020).
3. **SciPy**. SciPy depends on the NumPy and works with the latter's array. Scipy is built for mathematics, science and engineering (PyPI, 2020).
4. **Seaborn.** Seaborn is data visualization library based on Matplotlib that provides a high-level interface for drawing attractive and informative statistical graphics (Desai, 2019). **Matplotlib**, is included here despite the similarity in function, with Seaborn being more specialized in plotting of variable correlation. Matplotlib is a data graphing library in Python for 2D graphics. Its features include but is not limit to interactive graphing (PyPI, 2020).
5. **Sklearn.** Scikit learn is a library made for machine learning. It contains a series of helpful modules including classification, clustering, dimension reduction and preprocessing (scikit-learn, 2020).

3.4 Operating System chosen

Windows 10 is initially released on 29th July 2015 by Microsoft (Support.microsoft.com, 2020). Windows 10 gets regular updates as long as users allow automatic updates on their system, which means the system will get better on the promised 10 years update cycle (Bott, 2015). It is also the latest version of Windows to date. The features of Windows 10 can be referred at *https://www.microsoft.com/en-my/windows/features.*

3.5 Summary

As discussed above, the chosen tools for this project is as below:

Programming Language	Python
IDE	Jupyter Notebook
Libraries/ Tools	1. Pandas 2. Numpy 3. Scipy 4. Seaborn 5. Sklearn
Operating System	Windows 10

CHAPTER 4: METHODOLOGY

4.1 Introduction

The methodology chosen for this project is CRoss-Industry Standard Process for Data Mining (CRISP-DM). This methodology fits in nicely with the nature of this study which is a data analysis project. It will not only provide a structured approach for the researcher throughout the data mining process, but also will ensure that the output aligns with the business objective as well. CRISP-DM is widely adopted (Wiemer, Drowatzky and Ihlenfeldt, 2019), as the name suggest, due to the fact that it is less technology and problem sensitive than other data mining methodology such as SEMMA. The methodology is divided into 6 steps, but actually requires iterative movement between phases rather than being done with a phase regardless of the output (Dåderman. and Rosander, 2018).

4.2 CRISP-DM

The methodology chosen in this study is a modified version of general CRISP-DM. Usually, the CRISP-DM methodology is a 6-step framework where the steps are in order as below:

1) Business Understanding
2) Data Understanding
3) Data Preparation
4) Model Building
5) Testing and Evaluation
6) Deployment

With the modified version of the framework, we look to eliminate the Business Understanding step where usually, in business analytics, data analyst spend time trying to understand the business field knowledge. However, in this case, we have no need to understand the domain knowledge.

4.2.1 Data Understanding

At the beginning, knowledge of the data that will be processed is gathered. Users will get familiar with the data, which will help in identifying quality problems as well as executing basic exploration of the data. Interesting subsets may also be discovered in this stage. For the study, users will have to get familiar with the NBA statistics in this stage, have a general idea of the game, knows about simple stats like points, rebounds and steals, then advanced stats like Player Efficiency Rating (PER) and Net Rating before proceeding to attribute selection.

4.2.2 Data Preparation

This phase covers all activities that will turn the initial raw data into the final dataset which will be thrown into the modelling tool in the following stage. Data preparation usually include data cleaning, construction of new attributes, and transformation of data. For the study, this stage will create new advanced data of our target player. The attributes likely include the looks of Estimated Possession Value (EPV), plus/minus, Player Efficiency Rating and other advanced stats that measures the players efficiency as well as their impact on the court. Finally, a new binary attribute will be introduced as well, which is to measure if a player is a star in 3 years' time. The attribute is required as a target variable for the predicting model in the following stage in order to perform prediction.

4.2.3 Modelling

In the modelling stage, multiple modelling techniques are selected and applied. The parameters of the models will be fine tweak to produce optimal results. This phase typically has a strong connection with the Data Preparation phase, where one may go through the Data Preparation process again to get datasets of better fit for the model chosen. For this study, prediction models such as Decision Tree, Regression or Neural Network will be applied. The advanced stats mentioned in the previous stage will be fed into the models, with a purpose to train the models accuracy. The data will be partitioned into training, validation and test set. The training set will train the models accuracy, while the validation set will prevent situation

where the model overfits with the training set. The test set will be used to test the final accuracy of the model.

4.2.4 Evaluation

Before proceeding to this stage, the user will have a few models that performs great data analysis wise. Before deployment of the model, the model has to be evaluated carefully. Every step executed within the construction of the model will be reviewed, making sure that it will properly achieves the business objectives. For this study, the main objective is to produce a model that will give an accurate prediction on whether or not a player will turn into a superstar, thus aiding the business decision of a brand where they have to decide which player to invest on.

4.2.5 Deployment

When the model passes the evaluation, it is time for deployment. In this study, the deployment process is as simple as generating a report declaring the completion of the prediction model and applying the model into the latest player and statistic available. It is also suggested to create a feedback loop on the framework where real world results of the predictions made are feed back into the model, thus increasing the accuracy of the model for future use. This action can counteract the nature of NBA where factors that affect players success, like on-court rules, are constantly in change.

Success Criteria

To measure the success of the prediction model, we may refer back to the modelling stage of the CRISP-DM methodology. An amount of data will be partitioned and excluded from being used to build the prediction model. The test data will be used to test the accuracy of the model.

In this case, in order to define the success and future stardom of a young player looking at the players performance data, we have to set a bar for the player to achieve. The

terms decided here is for the player to achieve at least 17 points per game and +5 plus/minus per game in after playing for 3 seasons in the NBA.

4.3 Summary

Overall, CRISP-DM as a methodology matches this project well. The methodology enables a structured approach to plan a prediction model project in the sports domain, which will be utilized to meet the ultimate goal of generating business profit.

CHAPTER 5: DATA ANALYSIS

5.1 Introduction

This chapter will record how the data analysis is executed by the researcher in detail. It will cover phases including Data Exploration, Data Cleaning, Data Visualisation and Modelling.

There a number of assumptions made by the researcher in this study, as listed below:

1. The selection of all-star players is more based on statistical measured performance rather than fans' sentiment, like Zaza Pachulia was at top of vote pool due to fans from his home country Georgia despite posting unimpressive stat lines.
2. 2020 NBA data is not used, as the season is undergoing revamp process due to the global pandemic of COVID-19, so the data is incomplete and not available.
3. The collected data is enough to produce an accurate prediction model.
4. A time span of 7 years is chosen as that is typically the longest you can get a player to sign a contract.
5. The All-Star name list is used up to 2020 (the latest) despite the season being suspended and currently under revamp as it may show the results of young players in 2013.
6. Very few players have the same name to cause confusion, that the problem is neglectable.
7. Players' performance may be affected by factors like motivation, personalities, injury or other intangibles that is difficult to measure and predict, thus those possibilities are ignored.
8. It is hard to predict if a player will make a comeback after leaving the league for a few seasons, so the scenario where players have missing (non-consecutive) years are ignored.
9. The All-Star players mentioned in the data of following documentation are actually future All-Star players' data from their rookie season, for easier referencing.

5.2 Data Exploration

There is a total of 4 datasets. The first one, *Player_Data* contains individual statistics of each season of all the players who had ever played in the National Basketball Association from 1955 to 2019. The data is filled with traditional stats like Minutes Played and Field Goal Attended, Advanced Stats like Offensive Load and Passer Rating and also Proprietary stats by Ben Taylor of BackPicks like ScoreValue and BackPicks Box Plus Minus. The data is accessible if the user is a Patreon member of the Thinking Basketball community from the links below:

- *https://backpicks.com/metrics/player-seasons/*
- *https://backpicks.com/metrics/player-seasons/player-seasons-2/*

The metadata for this dataset is as per below:

- **Load =** offensive load, an estimate of the number of a player a player is "directly involved" in on offense every 100 possessions.
- **rTS% =** relative True Shooting percentage (true shooting compared to league average).
- **Box Creation =** An estimate of shots created for teammates per 100 possessions.
- **Passer Rating =** An estimate of a player's passing ability on (approximately) a 1-10 scale.
- **Spacing =** A basic estimate of player spacing using.
- **cTOV% =** creation-adjusted turnover rate, or turnovers committed as a percentage of offensive load.
- **ScoreVal =** Scoring value, an estimate of a player's points per 100 impact from scoring only.
- **PlayVal =** Playmaking value, an estimate of a player's points per 100 impact from playmaking only.
- **3p% Pro =** 3-point proficiency, a combination of 3-point volume and accuracy.
- **rORB% =** relative offensive rebounding percentage (rebounding compared to league average).
- **BPM =** Backpicks Box plus-minus model.
- **GPM Avg. =** Game-level plus minus average for a player's career (when available).

- **Scaled APM/g =** Scaled adjusted plus-minus value per game.
- **AuPM/g =** Augmented plus-minus, a plus-minus/box score hybrid that approximates adjusted plus-minus.
- **Scaled DAPM/g =** Scaled defensive adjusted plus-minus per game.
- **Team Ortg =** The relative offensive rating of a player's team in a given year.
- **IA =** Inflation Adjusted metrics. In the playoffs these are adjusted to opponent defensive rating.
- **PS =** postseason values.

The *AllStar2009* dataset is a subset of the Men's Professional Basketball dataset from kaggle.com. It contains all the All-Star players ever selected in the National Basketball Association from 1950 to 2009. It is posted by Sean Lahman in the link below:

- *https://www.kaggle.com/open-source-sports/mens-professionalbasketball/data?Select=basketball_series_post.csv*

The *AllStar2016* dataset contains lists of NBA players selected as All-Stars from 2000 to 2016. It is obtained from this link:

- *https://data.world/gmoney/nba-all-stars-2000-2016*

The *AllStar2020* dataset is manually extracted from basketball-reference.com, consisting of only the needed variables, name and year. The links are as below:

- *https://www.basketball-reference.com/allstar/NBA_2017.html*
- *https://www.basketball-reference.com/allstar/NBA_2018.html*
- *https://www.basketball-reference.com/allstar/NBA_2019.html*
- *https://www.basketball-reference.com/allstar/NBA_2020.html*

5.3 Data Preprocessing

The Data Preprocessing phase starts with the compiling of All-Star name list. The researcher first imports the 3 datasets containing All-Stars from different range of years.

```
import pandas as pd
#import data
allStar2009 = pd.read_csv('C:/Users/Dell/FYP/AllStar2009.csv')
allStar2016 = pd.read_csv('C:/Users/Dell/FYP/AllStar2016.csv')
allStar2020 = pd.read_csv('C:/Users/Dell/FYP/AllStar2020.csv')
```

Figure 5: Code snippet import csv files

The 3 dataframes are then showed for identification of the formats and attributes, so that the researcher can work with.

```
allStar2009.head(3)
```

	PlayerID	Lname	Fname	Year	Conference	League
0	abdulka01	Abdul-Jabbar	Kareem	1978	West	NBA
1	abdulka01	Abdul-Jabbar	Kareem	1969	East	NBA
2	abdulka01	Abdul-Jabbar	Kareem	1988	West	NBA

```
allStar2009.shape
```

```
(1406, 6)
```

```
allStar2016.head(3)
```

	Year	Player	Pos	HT	WT	Team	Selection Type	NBA Draft Status	Nationality
0	2016	Stephen Curry	G	6-Mar	190	Golden State Warriors	Western All-Star Fan Vote Selection	2009 Rnd 1 Pick 7	United States
1	2016	James Harden	SG	6-May	220	Houston Rockets	Western All-Star Fan Vote Selection	2009 Rnd 1 Pick 3	United States
2	2016	Kevin Durant	SF	6-Sep	240	Golden State Warriors	Western All-Star Fan Vote Selection	2007 Rnd 1 Pick 2	United States

```
allStar2016.shape
```

```
(439, 9)
```

```
allStar2020.head(3)
```

	Year	Player
0	2017	Anthony Davis
1	2017	James Harden
2	2017	Stephen Curry

```
allStar2020.shape
```

```
(97, 2)
```

Figure 6: Code Snippet showing All-Star dataframes

The researcher will only need the year and player's name for the compiled All-Star name list. Therefore, the redundant variables are removed. Also, the column players name are combined in the allStar2009 dataframe to make the variable labels uniform across 3 dataframes.

```
#2009, combine fname and lname into a new var 'Player'
allStar2009['Player'] = allStar2009['Fname']+' '+allStar2009['Lname']
# https://stackoverflow.com/questions/19377969/combine-two-columns-of-text-in-datafram

#removing useless columns from allStar2009
del allStar2009['PlayerID']
del allStar2009['Lname']
del allStar2009['Fname']
del allStar2009['Conference']
del allStar2009['League']

#increment the year by 1, as the all star game is the
#second half of the season, next year
allStar2009.Year += 1

allStar2009.head()
```

	Year	Player
0	1979	Kareem Abdul-Jabbar
1	1970	Kareem Abdul-Jabbar
2	1989	Kareem Abdul-Jabbar
3	1988	Kareem Abdul-Jabbar
4	1987	Kareem Abdul-Jabbar

```
#2016, remove useless columns
del allStar2016['Pos']
del allStar2016['HT']
del allStar2016['WT']
del allStar2016['Team']
del allStar2016['Selection Type']
del allStar2016['NBA Draft Status']
del allStar2016['Nationality']

#increment the year by 1, as the all star game is the
#second half of the season, next year
allStar2016.Year += 1

allStar2016.head()
```

	Year	Player
0	2017	Stephen Curry
1	2017	James Harden
2	2017	Kevin Durant
3	2017	Kawhi Leonard
4	2017	Anthony Davis

Figure 7: Code Snippet remove variables

The three dataframes are then ready to be combined into a single dataframe of All-Star names and years. The list is then sorted based on years.

```
#combining the 3 allstar lists into a single dataframe
combine = [allStar2009,allStar2016,allStar2020]
allStar = pd.concat(combine)
# https://pandas.pydata.org/pandas-docs/stable/user_guide/merging.html

#sort the players according to ascending years, to avoid introducing noise
#when removing duplicates and keeping first entry
allStar = allStar.sort_values('Year',ascending = True)
allStar
```

	Year	Player
48	1951	Paul Arizin
142	1951	Vince Boryla
750	1951	Ed Macauley
379	1951	Larry Foust
272	1951	Bob Davies
...	...	...
75	2020	LeBron James
74	2020	Anthony Davis
73	2020	Kawhi Leonard
83	2020	Jayson Tatum
96	2020	Brandon Ingram

1942 rows × 2 columns

Figure 8: Code Snippet combine dataframes

Then the entries are drop with the exception of the first entry for each player. The *Year* column is renamed to *1stAS_yr*. That way, the researcher has a list of All-Star players' name and the first year they were selected as All-Star. The list is then saved into .csv file for further use.

```
#remove duplicate names, keep first allStar entry
allStar = allStar.drop_duplicates(subset=['Player'],keep='first')

allStar.shape
```

```
(443, 2)
```

```
allStar = allStar.rename(columns={'Year':'1stAS_yr'})
```

Save unique players name & first time all star year to *AllStar.csv*

```
#save unique allStar name list to .csv
allStar.to_csv(r'C:/Users/Dell/FYP/AllStar.csv')
```

Figure 9: Code Snippet filter All-Star entries

Next up is to import the *Player_Data.csv* file into a dataframe, just like the *allStar* files shown above. Then it is the removal of unnecessary data. Attributes like *GPM avg.*, *AUpm/g*, Post Season values and 3yr average values are all removed as they contain too much missing values and is difficult to impute. Team Rating values are also removed, as the team performance is affected by too many outside factors like coaching, other players or organization strategy rather than the player himself.

```
#drop less relevant column
#most of these contains a lot of null values that cant be replaced with 0
#players dont get to play in the PS, data not available, etc.
#replace with 0 will create inbalance
drop_col = ['GPM avg.', 'Scaled APM/g', 'AuPM/g', 'Scaled DAPM/g',
            'PS MP', 'PS BPM', 'RS+PS BPM/g','3yr PS IA Pts 75', '3yr PS rTS',
            ' 3yr PS Box OC', '3yr PS Passer Rating', '3yrPs ScoreVal',
            '3yr PlayVal', '3yr AuPM/g', 'RS BPM', '3yr PS BPM',
            '3yr RS+PS  BPM/g', 'Team SRS', 'Team Ortg', 'Team Drtg']
playersData = playersData.drop(drop_col, axis = 1)
'''
Author: Harshit Tyagi
Date: 2020
Availablilty: https://towardsdatascience.com/hitchhikers-guide-to-exploratory-data-ana
'''
```

Figure 10: Code Snippet of attributes removal

Then, the *Yr/Team* column is renamed as *Year*, and the value inside is derived into only the second year of a season, as the All-Star games are held then.

```
#extract just the year from the yr/team column
playersData['Yr/Team']=playersData['Yr/Team'].str[:2]+playersData['Yr/Team'].str[5:7]
'''
Author: Jono
Date: 2020
Availablilty: https://stackoverflow.com/questions/63036156/manipulate-string-in-python
'''
#rename column to year
playersData = playersData.rename(columns={'Yr/Team':'Year'})

#replace noise of year 1900 to 2000
playersData['Year'] = playersData['Year'].replace('1900','2000')
playersData.head()
```

	Player	Year	Age	MP	G	FGA	Load	Pts/75	rTS%	Box Creation	...	FT%	3p Pro	rORB%	rDREB%	OBPM	BPM	IA Pts/75
0	A.C. Green	1986	22.0	1542	82	11.8	18.9	11.9	0.023	0.6	...	0.61	0.02	0.0	0.0	-0.4	0.4	12.2
1	A.C. Green	1987	23.0	2240	79	12.4	19.3	13.5	0.061	0.5	...	0.78	0.00	4.5	5.5	1.1	2.0	13.7
2	A.C. Green	1988	24.0	2636	82	11.8	19.1	12.9	0.043	0.4	...	0.77	0.00	4.5	5.7	0.9	1.6	13.2
3	A.C. Green	1989	25.0	2510	82	14.5	22.4	15.6	0.057	0.6	...	0.79	0.03	5.7	6.6	1.5	2.3	15.9
4	A.C. Green	1990	26.0	2709	82	14.8	22.0	14.6	0.011	0.4	...	0.75	0.11	5.1	4.8	1.1	2.1	14.9

5 rows × 26 columns

Figure 11: Code Snippet of modifying column 'Yr/Team'

The researcher then filtered out the players who had less than 7 seasons entries from the data, as per assumption in the introduction.

```
#remove players who plays less than 7 years (7 seasons included)
n = playersData[['Player']]
playersData = playersData[n.replace(n.apply(pd.Series.value_counts)).gt(6).all(1)]

playersData.shape
# https://stackoverflow.com/questions/48513886/how-can-i-remove-rows-where-frequency-o
```

```
(12200, 26)
```

Figure 12: Code Snippet of filtering out players with less than 7 seasons

After that, columns like *Year*, *ScoreVal*, *PlayVal*, *rORB%* and *rDREB%* are converted from type object into type float64 so that they can be calculated.

```
#convert playersData Year to to int64 so
#that it is comparable with allStar Year
playersData['Year'] = playersData['Year'].astype(str).astype(int)
playersData = playersData.astype({'Year':'int64'})
#convert other columns into workable dtypes
playersData['ScoreVal'] = playersData['ScoreVal'].astype(float)
playersData['PlayVal'] = playersData['PlayVal'].astype(float)
playersData['rORB%'] = playersData['rORB%'].astype(float)
playersData['rDREB%'] = playersData['rDREB%'].astype(float)

playersData.dtypes
```

```
rTS%              float64
Box Creation      float64
Passer Rating     float64
Spacing           float64
cTOV%             float64
ScoreVal          float64
PlayVal           float64
FTA/100           float64
FT%               float64
3p Pro            float64
rORB%             float64
rDREB%            float64
OBPM              float64
BPM               float64
IA Pts/75         float64
IA PPG            float64
IA APG            float64
IA RPG            float64
dtype: object
```

Figure 13: Code Snippet of datatype conversion

Then the researcher checks for the missing data using the code below:

```
#to check for missing value
to_add = playersData[(playersData['MP']>=0)]
col_count = to_add.count()
print(col_count)
```

```
Player           12200
Year             12200
Age              12200
MP               12200
G                12200
FGA              12200
Load             12200
Pts/75           12200
rTS%             12183
Box Creation     12200
Passer Rating    12199
Spacing          12199
cTOV%            12199
ScoreVal         12199
PlayVal          12199
FTA/100          12200
FT%              12200
3p Pro           10273
rORB%            12040
rDREB%           12040
OBPM             12183
BPM              12183
IA Pts/75        12200
IA PPG           12048
IA APG           12048
IA RPG           12048
dtype: int64
```

Figure 14: Code Snippet check for missing value

As the missing value problem is not profound, they are imputed with mean value from their respective column, then checked again for missing value.

```
#replace missing values
playersData['ScoreVal'] = playersData['ScoreVal'].fillna(playersData['ScoreVal'].mean(
playersData['PlayVal'] = playersData['PlayVal'].fillna(playersData['PlayVal'].mean())
playersData['rORB%'] = playersData['rORB%'].fillna(playersData['rORB%'].mean())
playersData['rDREB%'] = playersData['rDREB%'].fillna(playersData['rDREB%'].mean())
playersData['rTS%'] = playersData['rTS%'].fillna(playersData['rTS%'].mean())
playersData['Passer Rating'] = playersData['Passer Rating'].fillna(playersData['Passer
playersData['Spacing'] = playersData['Spacing'].fillna(playersData['Spacing'].mean())
playersData['cTOV%'] = playersData['cTOV%'].fillna(playersData['cTOV%'].mean())
playersData['OBPM'] = playersData['OBPM'].fillna(playersData['OBPM'].mean())
playersData['BPM'] = playersData['BPM'].fillna(playersData['BPM'].mean())
playersData['IA PPG'] = playersData['IA PPG'].fillna(playersData['IA PPG'].mean())
playersData['IA APG'] = playersData['IA APG'].fillna(playersData['IA APG'].mean())
playersData['IA RPG'] = playersData['IA RPG'].fillna(playersData['IA RPG'].mean())
'''
Author: bmu & Nae
Date: 2020
Availablilty: https://stackoverflow.com/questions/18689823/pandas-dataframe-replace-na
'''
#if fillna parameter inplace is true, it will return nothing
#to check for missing value
to_add = playersData[(playersData['MP']>=0)]
col_count = to_add.count()
print(col_count)
```

```
Player           12200
Year             12200
Age              12200
MP               12200
G                12200
FGA              12200
Load             12200
Pts/75           12200
rTS%             12200
Box Creation     12200
Passer Rating    12200
Spacing          12200
cTOV%            12200
ScoreVal         12200
PlayVal          12200
FTA/100          12200
FT%              12200
3p Pro           10273
rORB%            12200
rDREB%           12200
OBPM             12200
BPM              12200
IA Pts/75        12200
IA PPG           12200
IA APG           12200
IA RPG           12200
```

Figure 15: Code Snippet impute missing value

Each of the data entry is then marked with a new column *id*, for cross-referencing purpose later on. A second dataframe *playersDataX* is created, and the original dataframe *playersData* is then filtered to keep only the first entry of each player, so that the researcher has only the players' rookie data.

```
#create 'pid' to indentify specific player from specific season
id = list(range(12200))
playersData['pid'] = id
```

```
#take only rookie data
#remove duplicate names, keep first season only
playersDataX = playersData.drop_duplicates(subset=['Player'],keep='first')

playersDataX.shape
```

```
(1123, 27)
```

Figure 16: Code Snippet filtering player entries

The two dataframe is then cross-referenced and the rookie year entries are removed from the original dataset. A third dataframe is then generated by again, keeping the first entry of each player on the original dataframe. The third dataframe will contain all the players' second year, also known as sophomore year's data.

```
#delete the players' rookie season from the data by cross ref with
#the list, identifying the 'pid' column
cond = playersData['pid'].isin(playersDataX['pid'])
playersData.drop(playersData[cond].index, inplace = True)
playersData.shape
```

```
(11077, 27)
```

```
#keep only the 'first' year row to acquire sophomore yr data
#take only rookie data
#remove duplicate names, keep first season only
playersDataY = playersData.drop_duplicates(subset=['Player'],keep='first')
playersDataY.shape
```

```
(1123, 27)
```

Figure 17: Code Snippet retrieving sophomore year data

Then rookie year dataframe is then combined with the sophomore year dataframe into a single dataframe to worked on.

```
#combine 1st yr data and 2nd year data to get the data we need
playersData =  playersDataX.append(playersDataY)
playersData = playersData.drop(['pid'], axis = 1)
playersData
```

Figure 18: Code Snippet combining dataframes

By cross-referencing playersData dataframe with the All-Star, two new columns are added into the former dataframe: allStar (binary to show if the player is future All-Star, will be used as target variable) and 1stAS_yr (the year the player is selected as All-Star for the first time).

```
#by cross referencing with allStar list
allStar = pd.read_csv('C:/Users/Dell/FYP/AllStar.csv')
allStar.rename(columns={'Unnamed: 0':'index'},inplace=True)
allStar = allStar.drop(['index'], axis=1)

#create binary var of allStar status
playersData = playersData.assign(allStar=playersData.Player.isin(allStar.Player).astyp

#create int var ASYear by also cross referencing allStar list
#merge allStar with playersData by 'Player'
playersAS = playersData.merge(allStar, on='Player', how = 'left')
'''
Author: Blazina
Date: 2020
Availablilty: https://stackoverflow.com/questions/52943166/get-corresponding-column-va
'''
#rename year_x to Year and year_y into 1stAS_yr
playersAS = playersAS.rename(columns={'Year_x':'Year'})
playersAS = playersAS.rename(columns={'Year_y':'1stAS_yr'})

playersAS.head()
```

	Player	Year	Age	MP	G	FGA	Load	Pts/75	rTS%	Box Creation	...	rORB%	rDREB%	OBPM	BPM	IA Pts/75	IA PPG	IA APG
0	A.C. Green	1986	22.0	1542	82	11.8	18.9	11.9	0.023	0.6	...	0.0	0.0	-0.4	0.4	12.2	6.1	0.6
1	Aaron Brooks	2008	23.0	608	51	19.7	35.4	17.3	-0.005	5.4	...	-2.8	-6.8	-0.2	-2.0	17.8	5.3	1.8
2	Aaron McKie	1995	22.0	827	45	16.1	26.7	13.6	-0.043	2.1	...	-1.7	-0.2	0.2	1.0	13.8	6.4	1.9
3	Aaron Williams	1997	25.0	553	32	14.0	20.8	14.5	0.063	0.3	...	6.4	3.1	-0.6	-0.3	15.2	6.4	0.4
4	Adam Keefe	1993	22.0	1549	82	11.9	21.2	12.9	0.028	0.5	...	5.4	5.9	-0.4	-0.1	13.1	6.3	0.9

5 rows × 28 columns

Figure 19: Code Snippet creating new variables

For baselining purpose, variable *yr2AS* is created by subtracting the Year from *1stAS_yr*. It indicates the number of years one takes to get selected as All-Star.

```
playersAS['yr2AS']=playersAS['1stAS_yr']- playersAS['Year']
playersAS.head()
```

	Player	Year	Age	MP	G	FGA	Load	Pts/75	rTS%	Box Creation	...	rDREB%	OBPM	BPM	IA Pts/75	IA PPG	IA APG	IA RPG	a
0	A.C. Green	1986	22.0	1542	82	11.8	18.9	11.9	0.023	0.6	...	0.0	-0.4	0.4	12.2	6.1	0.6	4.6	
1	Aaron Brooks	2008	23.0	608	51	19.7	35.4	17.3	-0.005	5.4	...	-6.8	-0.2	-2.0	17.8	5.3	1.8	1.1	
2	Aaron McKie	1995	22.0	827	45	16.1	26.7	13.6	-0.043	2.1	...	-0.2	0.2	1.0	13.8	6.4	1.9	2.8	
3	Aaron Williams	1997	25.0	553	32	14.0	20.8	14.5	0.063	0.3	...	3.1	-0.6	-0.3	15.2	6.4	0.4	4.3	
4	Adam Keefe	1993	22.0	1549	82	11.9	21.2	12.9	0.028	0.5	...	5.9	-0.4	-0.1	13.1	6.3	0.9	4.9	

5 rows × 29 columns

Figure 20: Code Snippet create new variable yr2AS

The researcher then filters out the players that took more than 7 years to become All-Star, as most of them are either noise or not relevant for the purpose of this study.

```
#remove players who took more than 7 years to become all Star
playersAS.drop(playersAS.loc[playersAS['yr2AS']> 7].index, inplace=True)

#remove players who took negative yrs to become AS, most likely dont have
#their rookie yr(missing) in Player_Data
playersAS.drop(playersAS.loc[playersAS['yr2AS']< 0].index, inplace=True)

playersAS.shape
```

```
(2088, 29)
```

Figure 21: Code Snippet filter players from yr2AS

The outliers are determined in the plotting phase below, and the researcher used All-Star group for baseline for each of the independent variables to avoid removing the already relatively low numbered All-Star entries.

```
def percentile_based_outlier(data, threshold):
    diff = (100 - threshold) / 2
    minval, maxval = np.percentile(data, [diff, 100 - diff])
    return (data < minval) | (data > maxval)
'''
Author: Vigneshwer Dhinakaran
Date: 2020
Availablilty: https://www.kite.com/blog/python/data-analysis-visualization-python/
```

```
# 99.5
col_names = ['Pts/75', 'PlayVal', 'FTA/100', 'rDREB%', 'OBPM']
threshold = 98

fig, ax = plt.subplots(len(col_names), figsize=(8,40))

for i, col_val in enumerate(col_names):
    x = Data[col_val][:1346]
    sns.distplot(x, ax=ax[i], rug=True, hist=False)
    outliers = x[percentile_based_outlier(x, threshold)]
    ax[i].plot(outliers, np.zeros_like(outliers), 'ro', clip_on=False)

    ax[i].set_title('Outlier detection - {}'.format(col_val), fontsize=15)
    ax[i].set_xlabel(col_val, fontsize=10)

    print(outliers)

plt.show()
```

```
63      23.7
74      24.0
76       7.1
116      7.1
155     25.4
261     24.3
285      4.8
301      7.4
328      7.5
417      6.5
472     24.1
523      6.8
620     23.9
671      5.2
682     23.8
685      7.7
703      7.5
721     26.6
753      6.7
830      6.4
1022    24.5
1071    23.6
1090    23.9
1096    23.1
1144     7.0
1200    23.1
1240    23.1
1330     6.8
Name: Pts/75, dtype: float64
```

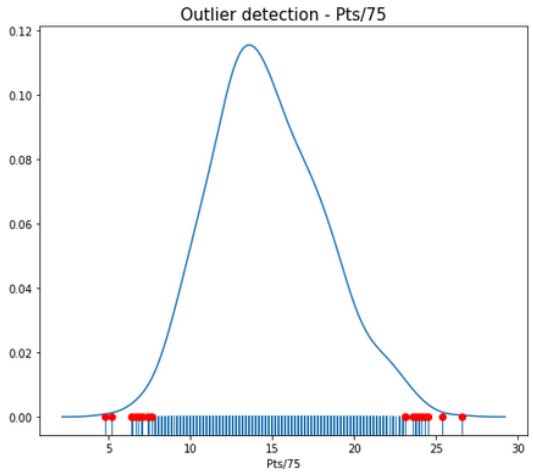


Figure 22: Code Snippet outlier detection

The outliers range are then recorded and removed as shown below:

```
Data = pd.read_csv('C:/Users/Dell/FYP/OmniX.csv')
Data.rename(columns={'Unnamed: 0':'index'},inplace=True)
Data = Data.drop(['index'], axis=1)
Data = Data.drop('3p Pro', axis = 1)
```

```
#remove rows with outliers by value
Data = Data[Data.iloc[:, 6].between(14.1, 43.4, inclusive=True)]#Load
Data = Data[Data.iloc[:, 7].between(6.5, 24.3, inclusive=True)]#Pts/75
Data = Data[Data.iloc[:, 8].between(-0.101, 0.09, inclusive=True)]#rTS%
Data = Data[Data.iloc[:, 9].between(0.2, 6.9, inclusive=True)]#Box Creation
Data = Data[Data.iloc[:, 10].between(1.4, 7.6, inclusive=True)]#Passer Rating
Data = Data[Data.iloc[:, 11].between(54, 128, inclusive=True)]#Spacing
Data = Data[Data.iloc[:, 12].between(0.072, 0.23, inclusive=True)]#cTOV%
Data = Data[Data.iloc[:, 13].between(-1.7, 1.4, inclusive=True)]#ScoreVal
Data = Data[Data.iloc[:, 14].between(-1.7, 1.2, inclusive=True)]#PlayVal
Data = Data[Data.iloc[:, 15].between(1.4, 12.3, inclusive=True)]#FTA/100
Data = Data[Data.iloc[:, 16].between(0.48, 0.9, inclusive=True)]#FT%
Data = Data[Data.iloc[:, 17].between(-5.2, 10.5, inclusive=True)]#rORB%
Data = Data[Data.iloc[:, 18].between(-9.5, 16.4, inclusive=True)]#rDREB% 30
Data = Data[Data.iloc[:, 19].between(-3.0, 2.8, inclusive=True)]#OBPM
Data = Data[Data.iloc[:, 20].between(-3.5, 4.0, inclusive=True)]#BPM
Data = Data[Data.iloc[:, 21].between(7.3, 24.6, inclusive=True)]#IA Pts/75
Data = Data[Data.iloc[:, 22].between(0, 25.2, inclusive=True)]#IA PPG
Data = Data[Data.iloc[:, 23].between(0, 6.9, inclusive=True)]#IA APG LLL
Data = Data[Data.iloc[:, 24].between(0, 13.1, inclusive=True)]#IA RPG

Data.shape
```

```
(1737, 28)
```

Figure 23: Code Snippet outlier removal

Finally, the dataframe is saved as .csv file, ready for visualisation and data exploratory.

```
#save playersAS to .csv
playersAS.to_csv(r'C:/Users/Dell/FYP/OmniX.csv')
```

Figure 24: Code Snippet save dataframe into .csv

5.4 Data Visualization

5.4.1 Univariate Analysis

Histogram

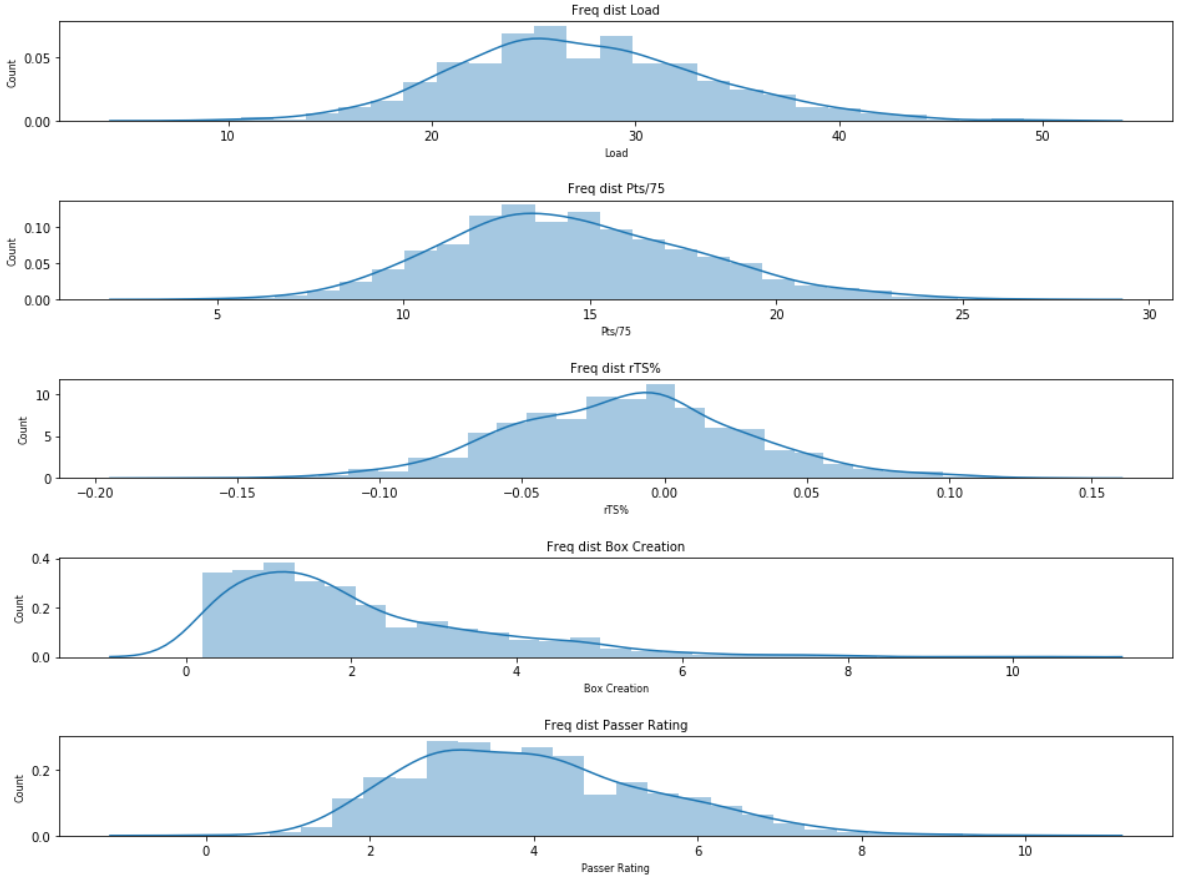


Figure 25: Histogram for Load, Pts/75, rTS%, Box Creation and Passer Rating

From the figure above, the researcher observed that *Load*, *Pts/75*, *rTS%* and *Passer Rating* all demonstrate traits of normal distribution, with *rTS%* slightly skewed to the right while the rest of the 3 slightly skewed to the left. *Box Creation*, however, is heavily skewed to the left, with most of the players posting values ranging from 0.2 to just below 2. This suggests that *Box Creation* will have significantly more outliers than the rest of the observations here.

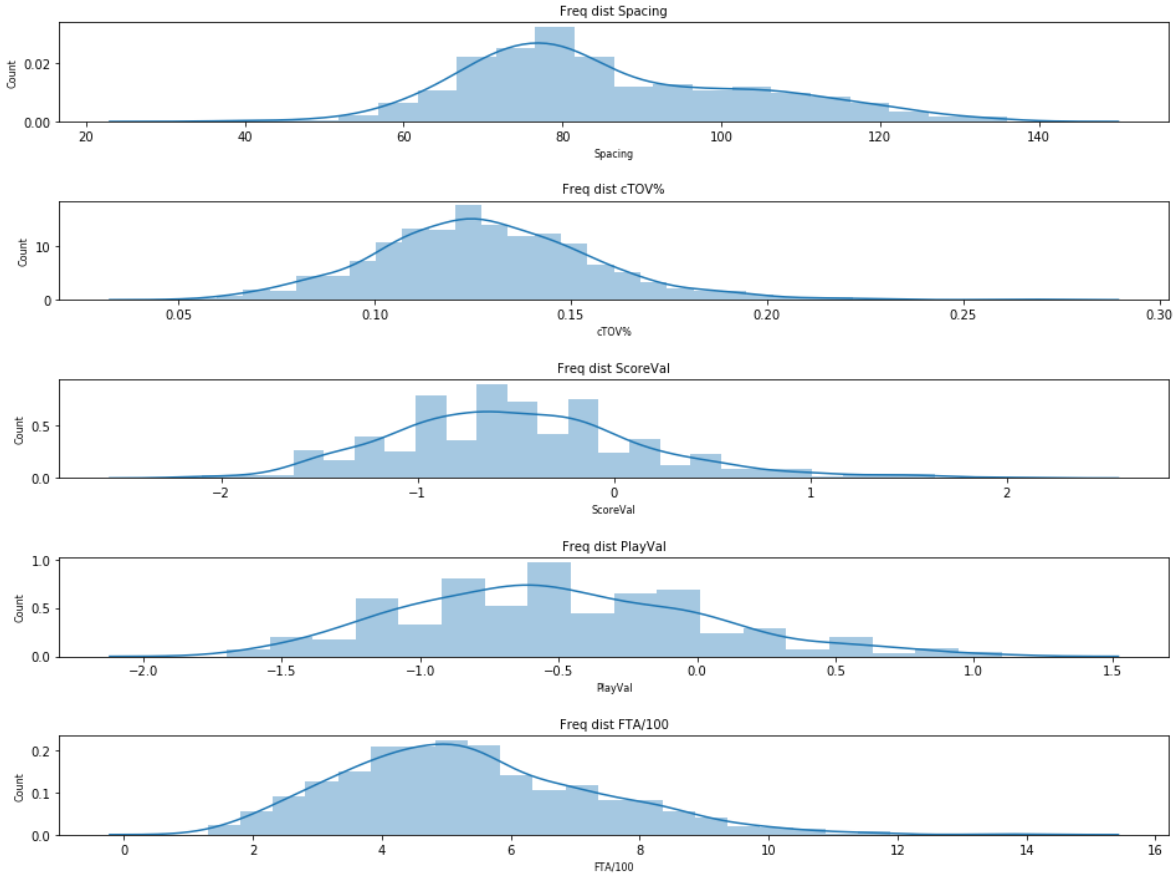


Figure 26: Histogram for Spacing, cTOV%, ScoreVal, PlayVal and FTA/100

Figure 6 shows that Spacing, *cTOV%*, *ScoreVal*, *PlayVal* and *FTA/100* are all normally distributed for all historical and present players in the league. Spacing and *PlayVal* is spread relatively wider than other variables (maybe due to the modern league shifting towards more of pace and space playing style compared to era before 90s). *Spacing* and *FTA/100* are left skewed while the rest does not show trait of skewing.

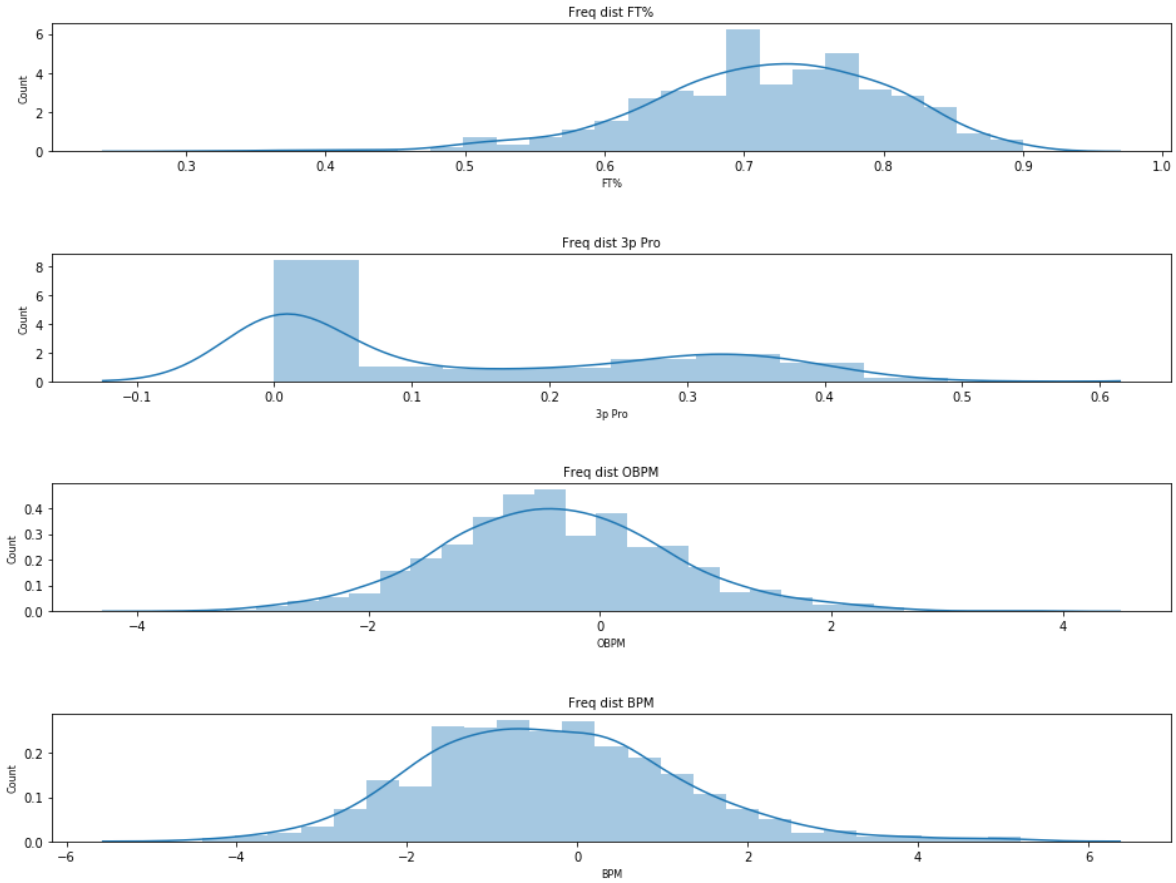


Figure 27: FT%, 3p Pro, OBPM and BPM

In Figure 7 above, *3pt pro* is the only observation that is not normally distributed. A huge portion of players posts a value of 0. The data is likely not complete and is highly likely to contain noise. The measurement should be removed from the dataset to avoid affecting the accuracy of prediction model.

FT% is right-skewed, with a significant portion of players averaging around 70%. *OBPM* and *BPM* both have most players at between -2 to just over 1.

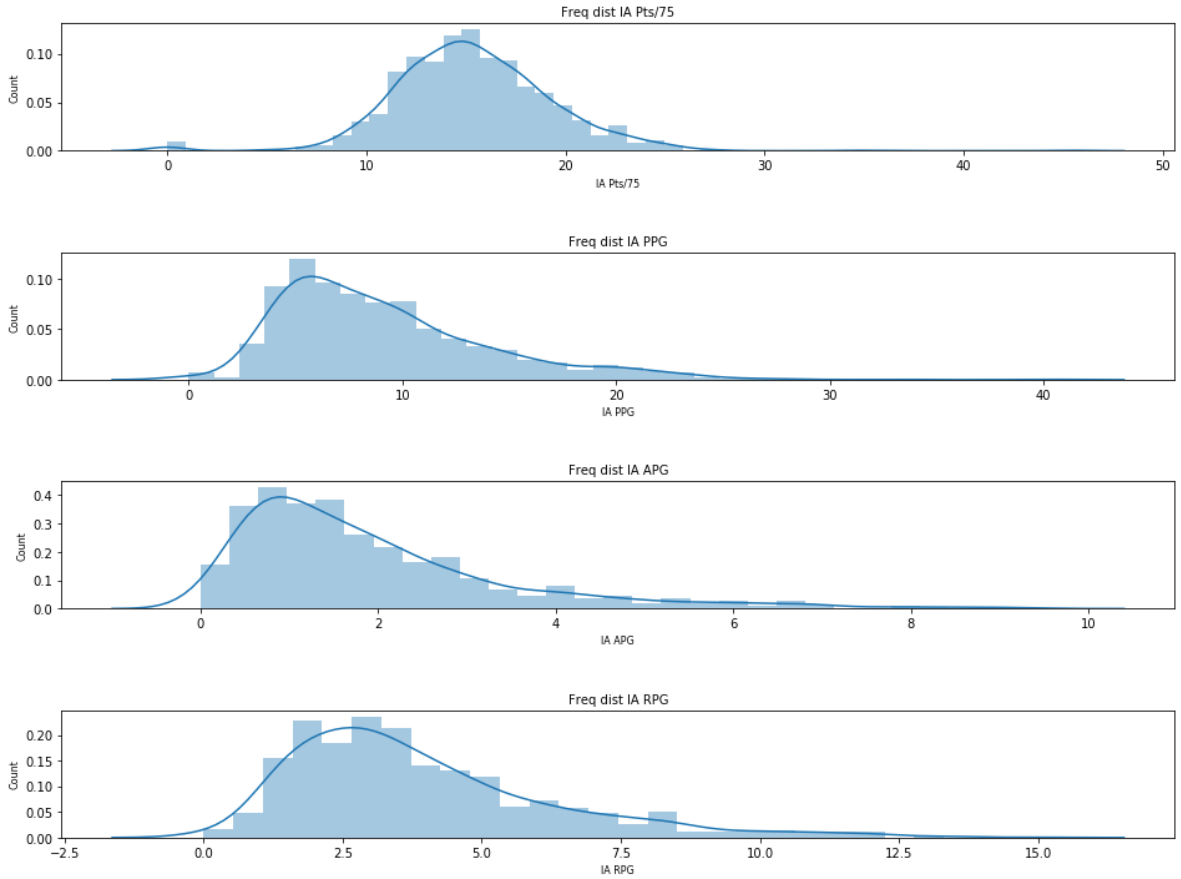


Figure 28: Histogram for IA Pts/75, IA PPG, IA APG and IA RPG

All the Inflation Adjusted values, as demonstrated in Figure 8 above, are left skewed. Most of the players average lower numbers, while only a selected class of elite players posts shockingly higher values when compared to the majority. Most of the players, upon adjustments, average 10-20 *Pts/75*, 4-11 *PPG*, 0.6-2.2 *APG*, 1.2-4.9 *RPG*.

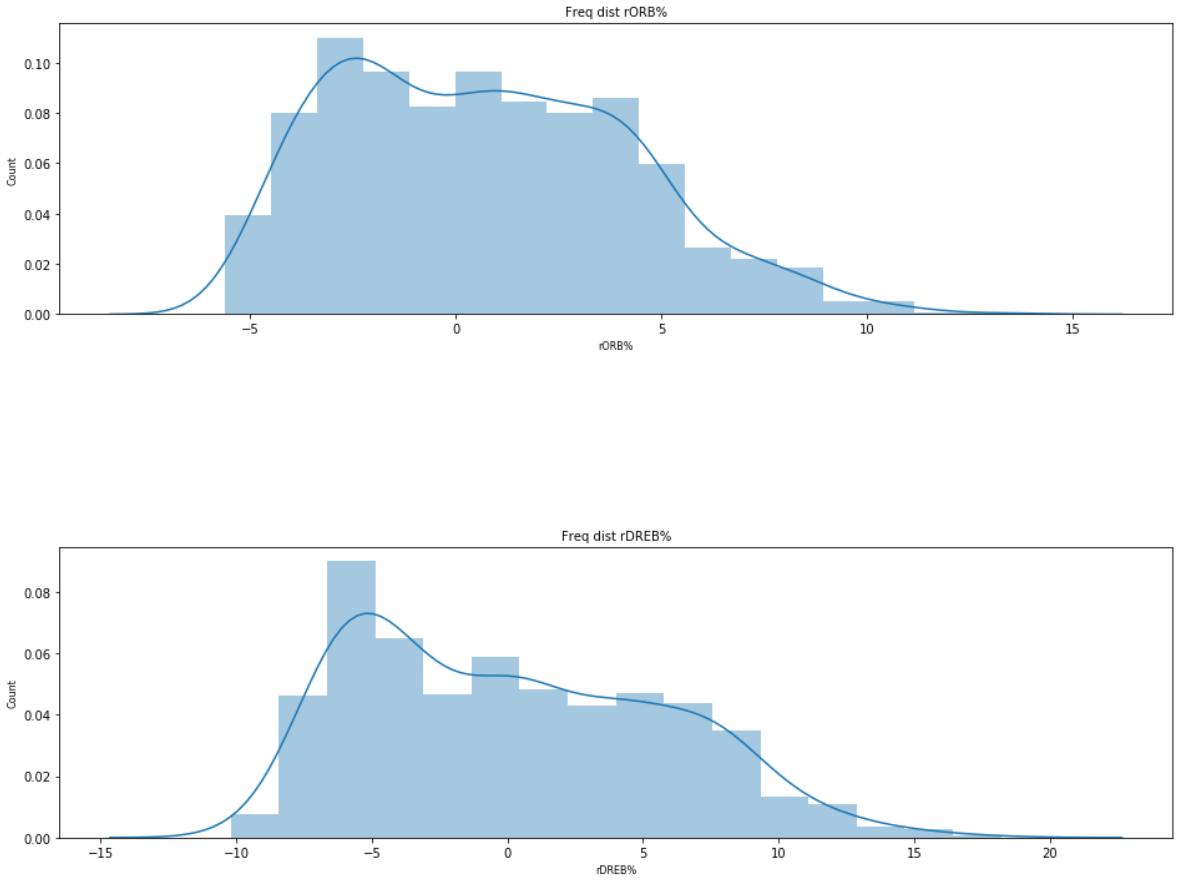


Figure 29: Histogram for rORB% and rDREB%

Figure 9 shows that majority of the players posts a *rORB%* from just over -5 to 5, and a *rDREB%* ranging from just over -7.5 to 7.5. Just like most of the observations above, both statistics are left-skewed, indicating that only a relatively low number of outstanding players produce higher than average numbers.

Box Plots

Box Plots are useful way of identifying the range of dataset, as well as other attributes such as Interquartile Range (IQR), median and most importantly the outliers. Below showcases the box plots of the variables: *Load*, *Pts/75*, *rTS%*, *Box Creation*, *Passer Rating*, *Spacing*, *cTOV%*, *ScoreVal*, *PlayVal*, *FTA/100*, *FT%*, *3p Pro*, *OBPM*, *BPM*, *IA Pts/75*, *IA PPG*, *IA APG*, *IA RPG*. The blue one represents values from all of the players while the red one indicates values of only All-Star players. As All-Star players are typically of higher calibre and posts better than average numbers, the All-Star only plots are made to establish a baseline

for outliers. This is to avoid too many All-Star entries being removed and introduce class imbalance problem.

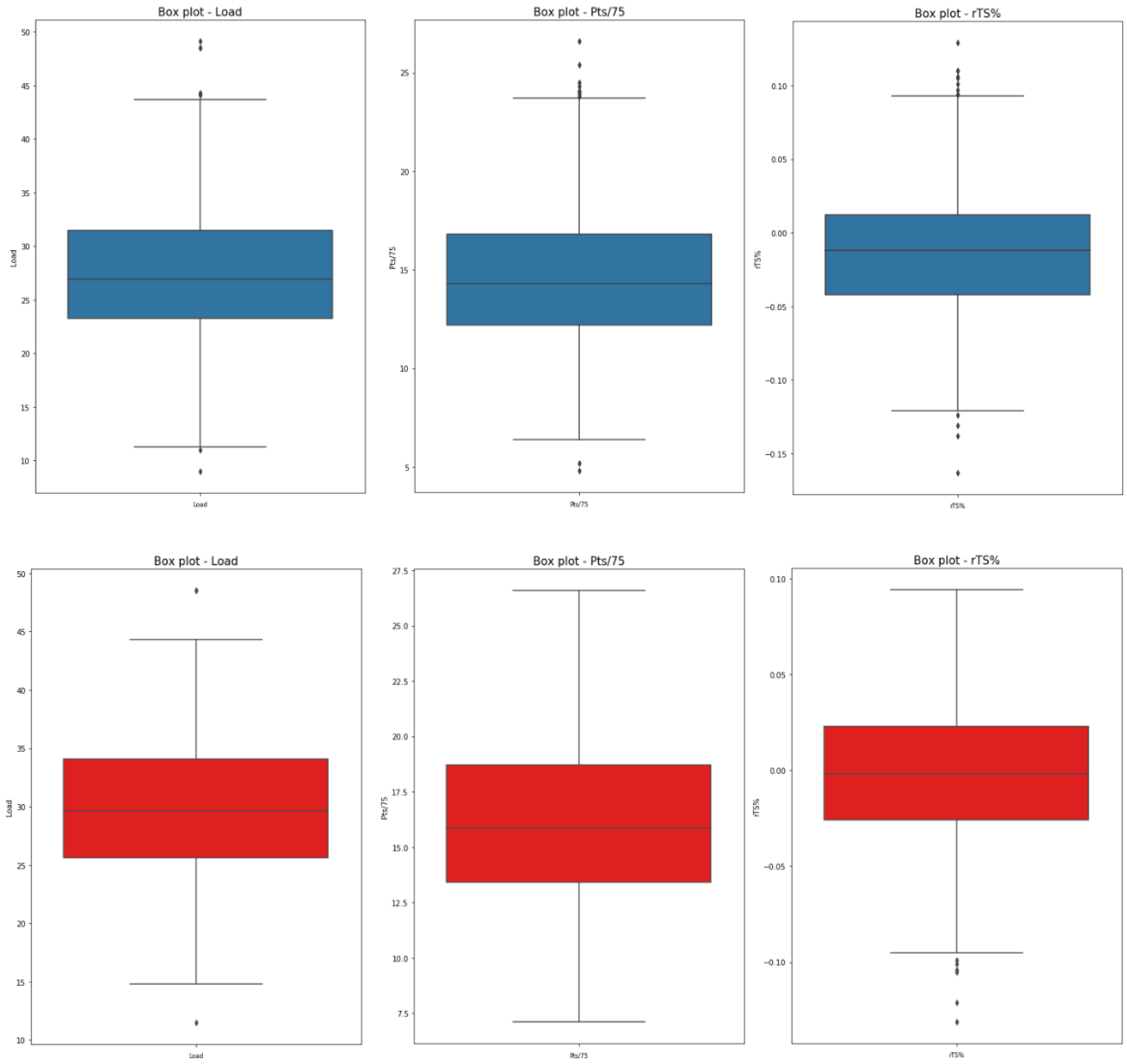


Figure 30: Load, Pts/75 and rTS% Box Plot of All Players(Blue) and AllStar(Red)

As shown in pairs of box plots above, All-Stars typically carries heavier *offensive load*, thus only the values closer to 50 is considered as outliers compared to the average counterpart where values over 45 is considered as outliers. In terms of *Pts/75*, the All-Stars averages higher numbers as well, the outliers of upper bound in the blue plot are likely all All-Star players. *rTS%* carries on the trend with All-Star players posting better than average numbers, which are represented as outliers in the blue

plot. There are outliers of *rTS%* lower than about -10% from both graphs, which should be taken care of.

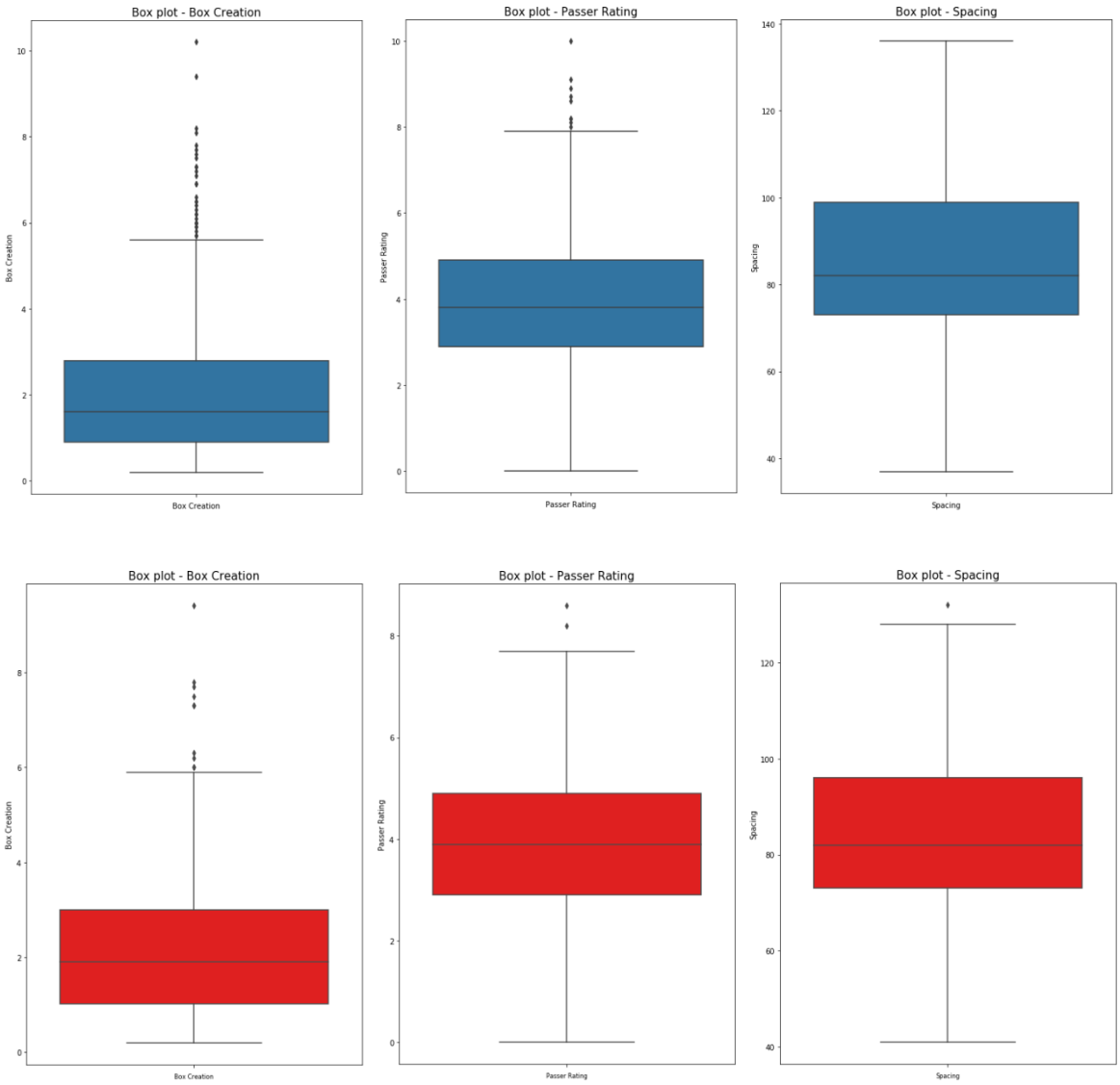


Figure 31: Box Creation, Passer Rating and Spacing Box Plot of All Players(Blue) and AllStar(Red)

The *Box Creation* and *Passer Rating* value from both groups possess about the same range, with a significant number of outliers from 6 and 8 upwards respectively. Spacing is different than the variables discussed above, with the average players group having no outliers compared to the All-Star group, which has some outliers around the 130 mark. The range of the average players are also wider

than the All-Stars, which can be contributed to the better players having more gravity. The All-Stars typically attracts more defender, leaving more space for their teammates, the more average players.

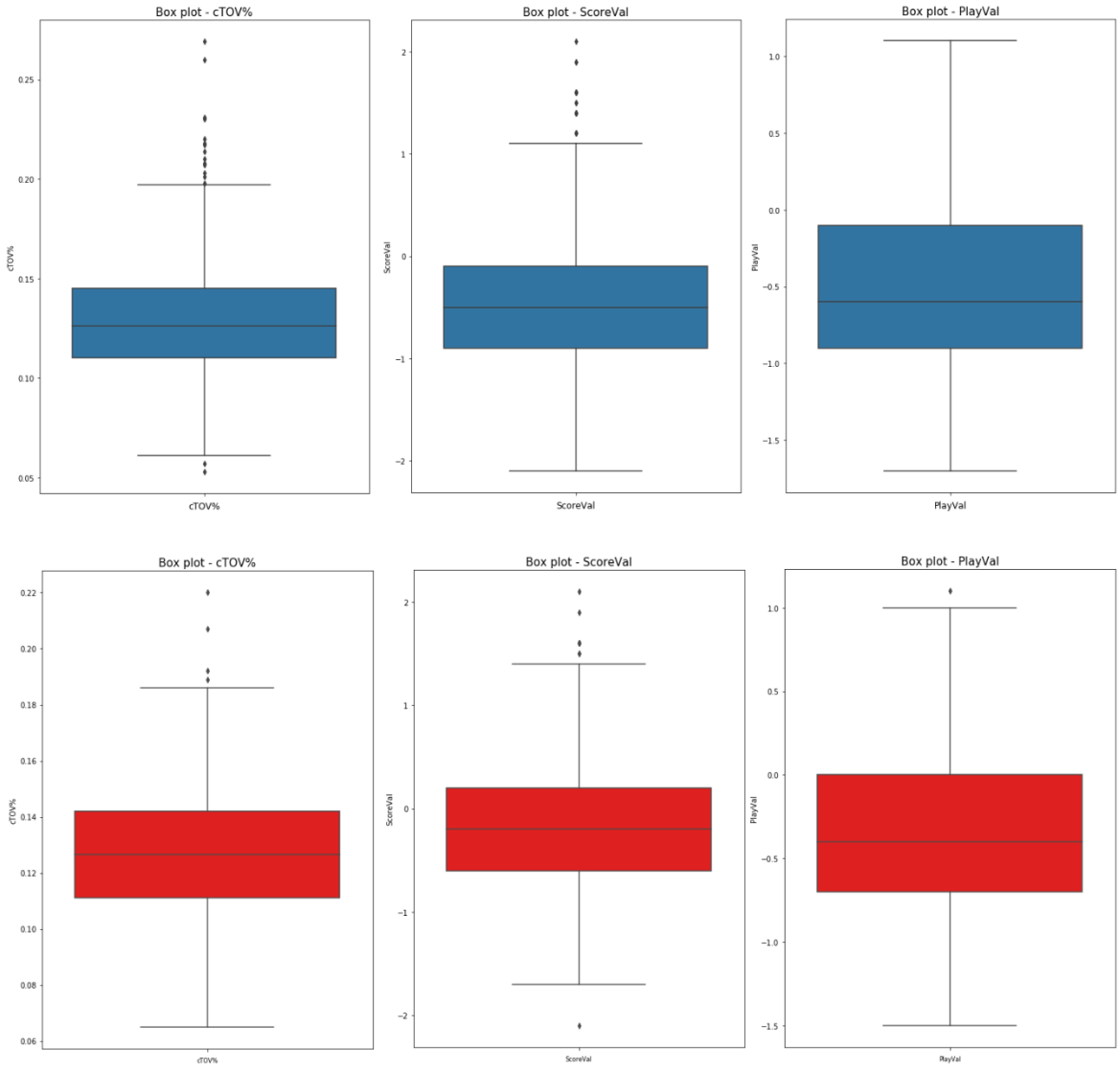


Figure 32: cTOV%, ScoreVal and PlayVal Box Plot of All Players(Blue) and AllStar(Red)

The All-Stars posts *cTOV%* of shorter range compared to regular players. That means they have lower creation-adjusted turnover rate at the top due to better ability, but still posts higher lower bound number because they typically carry more load, thus more space for error. All-Star players also provided better ScoreVal overall, thus representing some of the outliers over the value of about 1 in the blue plots. In terms of PlayVal, they average similar numbers to the league average, while having slightly higher lower bound.

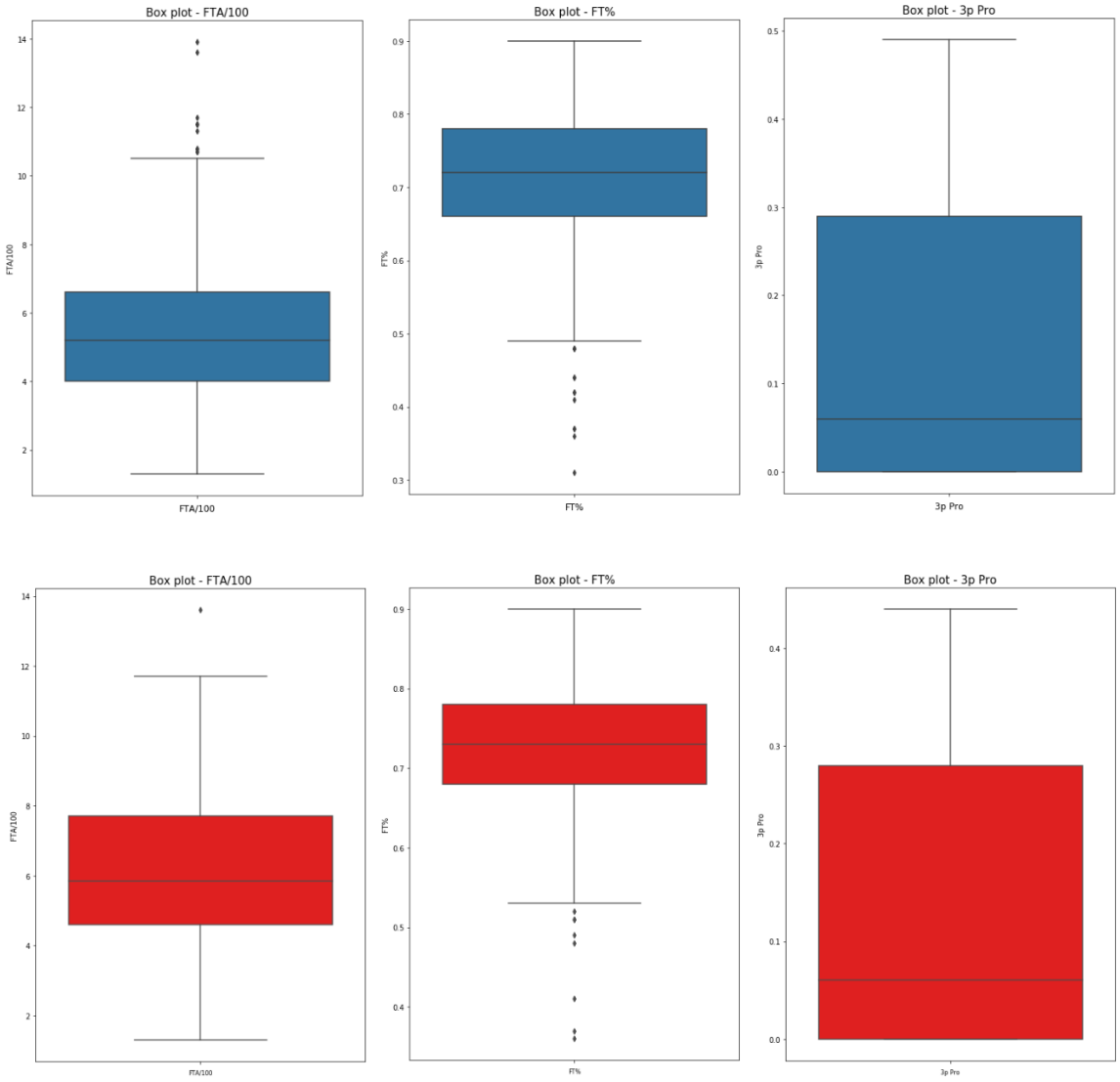


Figure 33: FTA/100, FT% and 3p Pro Box Plot of All Players(Blue) and AllStar(Red)

The box plots show that All-Stars players averages more *FTA/100* (up to 12) than average, thus a lot of the outliers on the blue plot represents them. *FT%* wise, there are little difference between the two box plots, with the league average posting outliers with value down to 30%. 3p Pro is of the same story, with the league average having a higher maximum value (nearly 50%) than the All-Stars (about 45%).

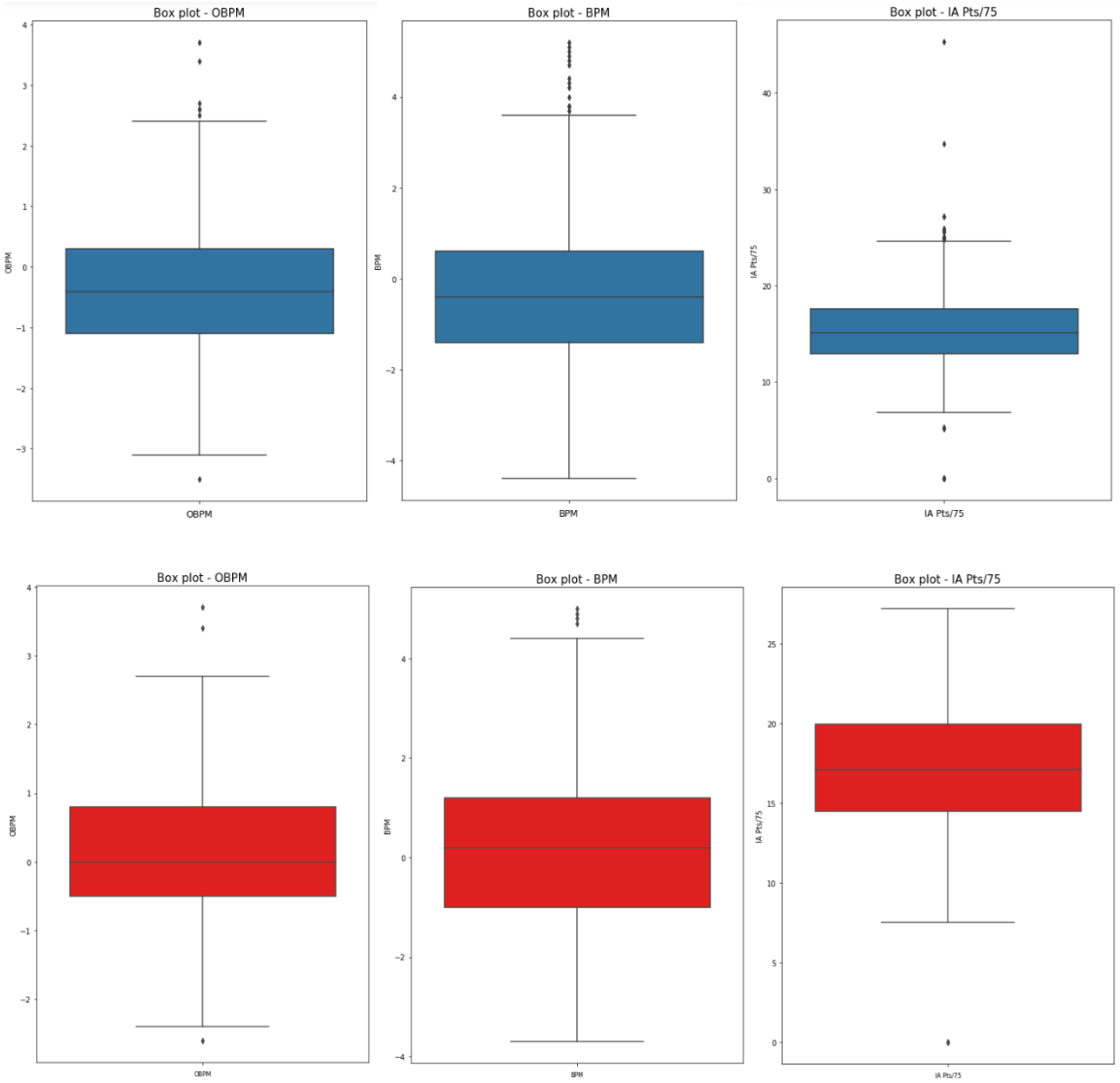


Figure 34: OBPM, BPM and IA Pts/75 Box Plot of All Players(Blue) and AllStar(Red)

As per the graphic shown above, All-Star players posts better *OBPM* and *BPM* numbers, ranging from about -2.5 to just under 3 and just above -4 to over 4 respectively. The league average is around -3 to 2.5 for *OBPM* and under -4 to under 4 for *BPM*. *IA Pts/75* is another variable that shows the All-Star set as superior, with it ranging from 7.5 to over 25 and the league average ranging from 7.5 to about 25. With the All-Star stats taken as baseline, *OBPM* shows little outlier while the rest of the two shows a considerable number of outliers.

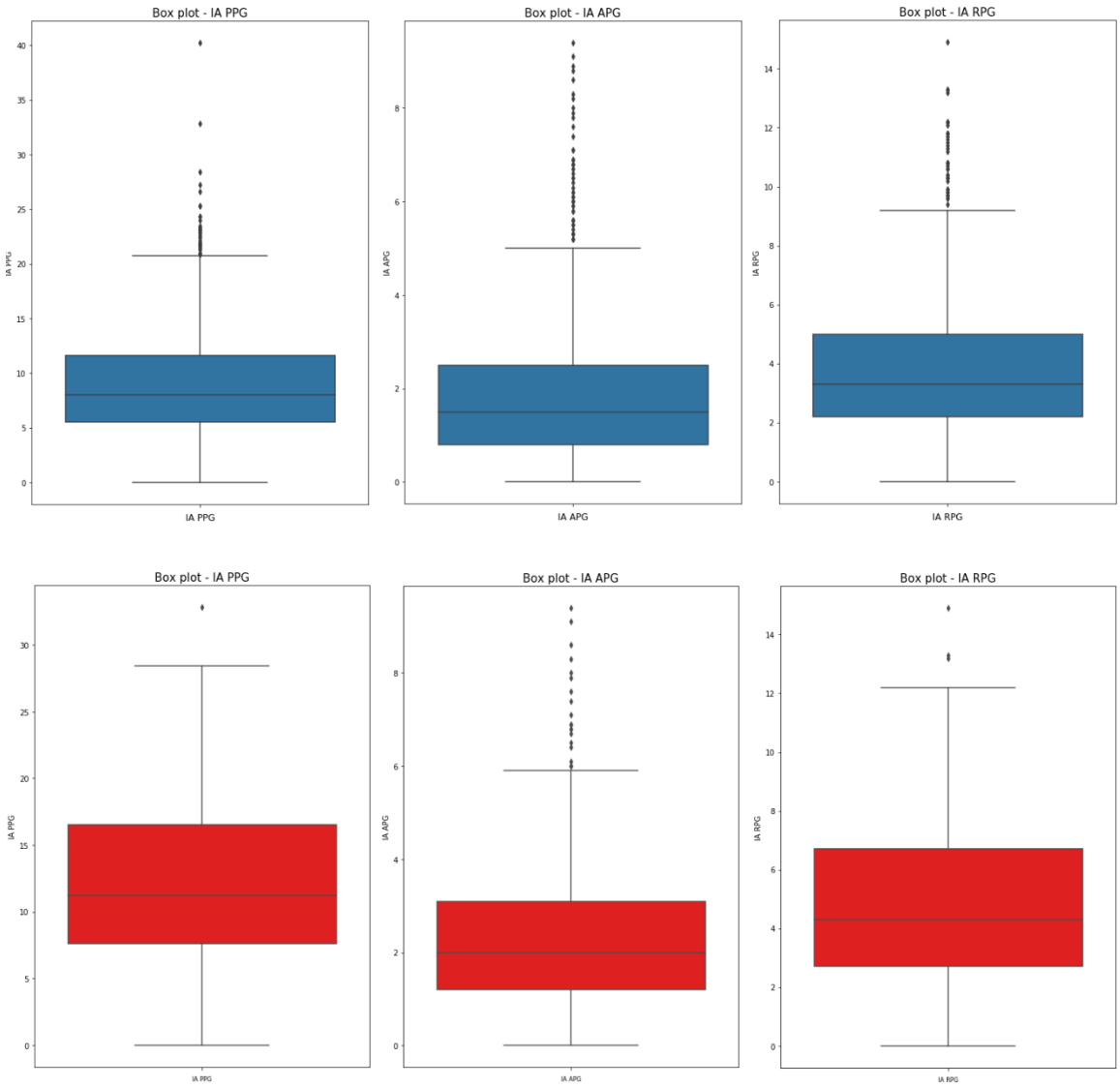


Figure 35: IA PPG, IA APG and IA RPG Box Plot of All Players(Blue) and AllStar(Red)

With the *IA PPG*, the All-Stars posts numbers from 0 to over 27.5, while the league average ranges from 0 up to just over 20. Therefore, the outlier for *IA PPG* in blue plot are mostly All-Star players. As for *IA APG*, the difference is not as significance. The All-Star numbers can go up to around 6, while the blue plot shows maximum of around 5. There is a significant number of outliers showing up in this variable. The *IA RPG* has the red group posting numbers up to 12, outclassing the blue group's number at about 9. Most of the outliers in the blue plot represents the outstanding performers in the All-Star group in this case.

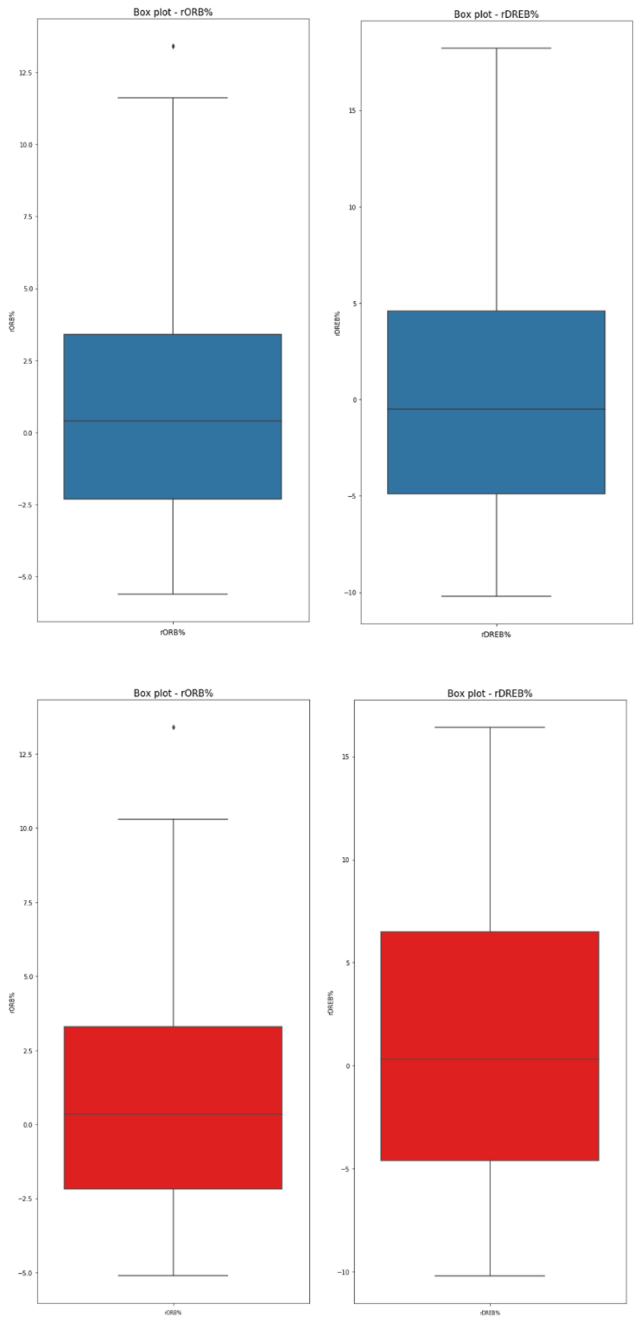


Figure 36: rORB% and rDREB% Box Plot of All Players(Blue) and AllStar(Red)

rORB% and *rDREB%* are two numbers where the league average has wider range compared to the All-Star group. There are little to no outlier here and no need for referencing the All-Star numbers as baseline. The distribution of these statistic numbers are quite similar.

5.4.2 Bivariate and Multivariate analysis

Below is the bivariate analysis of the independent variables in the form of scatter plots. The combinations that shows a potential linear correlation are circled out in red, and are marked with numbers, to be discussed below.

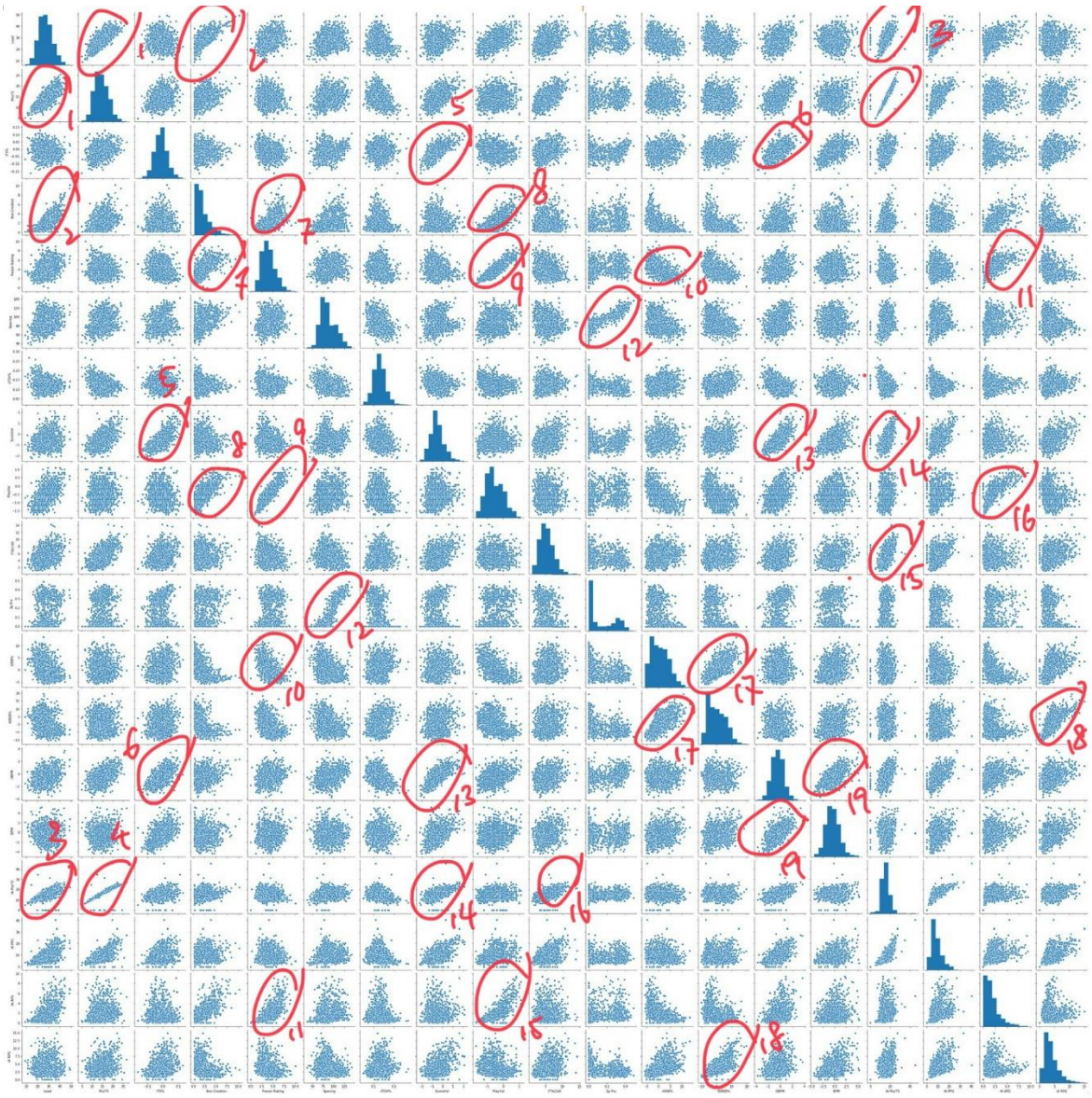


Figure 37: Bivariate Analysis Plot

1. *Load & Pts/75*, more offensive load leads to more productivity (pts), **relatively strong** positive linear relationship
2. *Load & Box Creation*: The heavier offensive load a player carries, the more shots he creates for his teammates. **Reasonably strong** positive relationship

3. *Load & IA PPG*: same as 1, weak positive linear relationship
4. *Pts/75 & IA PPG*: same as 1, **very strong** positive linear relationship, as both are adjusted value
5. *rTS% & ScoreVal*: *ScoreVal* measures players impact from scoring, rTS% measures players scoring efficiency. Weak positive linear relationship
6. *rTS% & OBPM*: The more efficient the player is at shooting the ball, the higher *Offensive Box Plus-Minus* he posts. Weak positive linear relationship.
7. *Box Creation &Passer Rating*: The better the player's passing ability is, the more shots he creates for his teammates. Weak positive linear relationship.
8. *Box Creation & PlayVal*: The more shots one creates for his teammates, the higher playmaking value he provides. Weak positive linear relationship.
9. *Passer Rating & PlayVal*: Passer Rating measures players passing ability, which correlates with his playmaking value (measured by PlayVal), **strong** linear positive relationship
10. *Passer Rating & rORB%*: Players who are good at offensive rebounding generally has lesser playmaking ability. Weak negative linear relationship.
11. *Passer Rating & IA APG*: The better the passing ability, the more assist one records per game after inflation adjustments. Weak positive linear relationship.
12. *Spacing & 3p pro*: *Spacing* measures the amount of spacing players provide and use (based on team scheme and teammates), thus correlates with 3pt proficiency. 3pt shooting contributes to spacing! **Reasonably strong** positive linear relationship. Not super linear maybe due to 3pt pro stat not very 'tidy'.
13. *ScoreVal & OBPM*: Offensive Box Plus Minus. Both associates with scoring efficiently & impact. Weak linear positive relationship
14. *ScoreVal & IA Pts/75*: more adjusted PPG one average, the higher the *ScoreVal*. Weak positive linear relationship
15. *PlayVal & IA APG*: The higher playmaking impact the player has, the more IA APG he produce. Weak positive linear relationship
16. *FTA/100 & IA Pts/75*: the more FT one attempts, the more pts he averages (adjusted). Weak positive linear relationship

17. *rORB% & rDREB%*: Both indicates the ability of a player to rebound the ball, the former is on the offensive end, while the latter is on the defensive end. These measurements have a weak positive linear relationship.
18. *rDREB% & IA RPG*: The better the player is at defensive rebounding, the higher number of Rebounds per game will he grab, weak linear positive relationship.
19. *OBPM & BPM*: The higher plus-minus value one provides on the offensive end, the more it translates to his overall plus-minus value. Weak positive linear relationship.

5.5 Hypothesis

The hypothesis section is to find out which attributes are highly correlated with the target variable and make a hypothesis that these attributes will have higher impact in the following classification performed.

5.4.3 Pearson Correlation Analysis

This method is chosen here as the data shows characteristics of normally distributed and the attributes are linearly related, while also being continuous numeric variables. The mentioned characteristics matches the criteria of Pearson Correlation Analysis well. The *pearsonr* function from *scipy.stats* will return two values: *Pearson correlation coefficient* which represents the strength of linear relationships between two variables (ranging from -1 as perfect negative relationship to 1 as perfect positive relationship) and the *p-value* which represents the significance of the relationship (if p value <= 0.5, a significant relationship exists).

The tables below show the Pearson Correlation Coefficient between the independent variables, the p-values of the coefficients, as well as the two metrics of each independent variable against the dependent variable.

Table 2: Correlation Coefficients between independent variables

	MP	Load	rTS%	Box Creation	Passer Rating	Spacing	cTOV %	Score Val	PlayVa l	FTA/1 00	FT%	rORB %	rDREB %	OBPM	BPM	IA Pts/75	IA PPG	IA APG	IA RPG
MP	1.000																		
Load	0.336	1.000																	
rTS%	0.200	-0.085	1.000																
Box Creation	0.201	0.749	-0.073	1.000															
Passer Rating	0.094	0.383	-0.269	0.640	1.000														
Spacing	0.083	0.322	0.103	0.428	0.288	1.000													
cTOV%	-0.110	-0.229	0.045	-0.189	-0.162	-0.395	1.000												
ScoreVal	0.452	0.142	0.719	-0.102	-0.305	0.003	-0.212	1.000											
PlayVal	0.292	0.543	-0.144	0.702	0.883	0.210	-0.171	0.018	1.000										
FTA/100	0.235	0.388	0.361	0.068	-0.289	-0.148	0.112	0.407	-0.142	1.000									
FT%	0.149	0.384	0.147	0.391	0.299	0.690	-0.286	0.147	0.339	-0.033	1.000								
rORB%	-0.063	-0.414	0.234	-0.628	-0.604	-0.386	0.199	0.198	-0.648	0.294	-0.459	1.000							
rDREB%	0.085	-0.343	0.205	-0.527	-0.527	-0.326	0.312	0.264	-0.481	0.262	-0.427	0.762	1.000						
OBPM	0.478	0.352	0.590	0.260	0.212	0.332	-0.432	0.707	0.364	0.268	0.300	-0.011	-0.052	1.000					
BPM	0.396	0.020	0.513	0.044	0.102	0.029	0.000	0.504	0.170	0.251	-0.037	0.254	0.293	0.633	1.000				
IA Pts/75	0.331	0.646	0.314	0.265	-0.138	0.204	-0.303	0.553	0.063	0.531	0.251	-0.010	0.022	0.490	0.165	1.000			
IA PPG	0.771	0.569	0.291	0.286	0.000	0.159	-0.222	0.580	0.226	0.444	0.220	-0.058	0.090	0.576	0.367	0.766	1.000		
IA APG	0.493	0.591	-0.049	0.729	0.718	0.236	-0.073	-0.005	0.796	0.002	0.318	-0.527	-0.386	0.398	0.239	0.194	0.471	1.000	
IA RPG	0.581	-0.029	0.308	-0.276	-0.336	-0.175	0.095	0.485	-0.219	0.373	-0.211	0.561	0.735	0.305	0.479	0.346	0.616	-0.002	1.000

Table 3: P-value of correlation coefficients between independent variables

	MP	Load	rTS%	Box Creation	Passer Rating	Spacing	cTOV%	ScoreVal	PlayVal	FTA/100	FT%	rORB%	rDREB%	OBPM	BPM	IA Pts/75	IA PPG	IA APG	IA RPG
MP	0.00E+00																		
Load	3.55E-29	0.00E+00																	
rTS%	6.90E-11	5.65E-03	0.00E+00																
Box Creation	4.88E-11	2.38E-189	1.75E-02	0.00E+00															
Passer Rating	2.38E-03	5.10E-38	6.76E-19	8.59E-122	0.00E+00														
Spacing	7.19E-03	9.24E-27	8.39E-04	7.05E-48	1.66E-21	0.00E+00													
cTOV%	3.67E-04	5.98E-14	1.50E-01	6.52E-10	1.24E-07	1.49E-40	0.00E+00												
ScoreVal	4.48E-54	3.88E-06	6.79E-168	9.81E-04	4.35E-24	9.31E-01	3.89E-12	0.00E+00											
PlayVal	3.88E-22	2.47E-81	2.70E-06	3.25E-156	0.00E+00	7.05E-12	2.59E-08	5.56E-01	0.00E+00										
FTA/100	1.14E-14	5.31E-39	9.97E-34	2.73E-02	1.24E-21	1.56E-06	2.92E-04	3.20E-43	4.06E-06	0.00E+00									
FT%	1.33E-06	2.78E-38	1.81E-06	1.10E-39	4.06E-23	2.97E-149	3.86E-21	1.68E-06	1.33E-29	2.92E-01	0.00E+00								
rORB%	4.00E-02	1.01E-44	1.79E-14	4.35E-116	1.89E-105	1.37E-38	7.20E-11	1.09E-10	9.48E-126	2.28E-22	7.95E-56	0.00E+00							
rDREB%	5.96E-03	2.81E-30	1.92E-11	6.16E-76	5.73E-76	2.42E-27	4.47E-25	3.51E-18	5.53E-62	6.02E-18	1.07E-47	5.34E-200	0.00E+00						
OBPM	6.25E-61	5.94E-32	2.53E-99	1.18E-17	4.33E-12	2.06E-28	5.88E-49	7.31E-160	3.44E-34	1.08E-18	3.03E-23	7.14E-01	9.21E-02	0.00E+00					
BPM	1.04E-40	5.15E-01	2.04E-71	1.57E-01	9.29E-04	3.44E-01	9.95E-01	8.85E-69	3.32E-08	1.59E-16	2.37E-01	6.45E-17	3.74E-22	1.04E-118	0.00E+00				
IA Pts/75	3.01E-28	5.39E-125	1.72E-25	2.33E-18	7.60E-06	2.80E-11	1.20E-23	5.45E-85	3.98E-02	2.02E-77	1.53E-16	7.52E-01	4.72E-01	1.62E-64	7.95E-08	0.00E+00			
IA PPG	2.64E-207	7.07E-91	6.48E-22	3.57E-21	9.95E-01	2.29E-07	3.45E-13	3.63E-95	1.19E-13	6.62E-52	6.29E-13	5.89E-02	3.59E-03	6.76E-94	9.58E-35	2.75E-203	0.00E+00		
IA APG	2.98E-65	6.24E-100	1.16E-01	2.17E-174	3.18E-167	1.07E-14	1.86E-02	8.67E-01	3.29E-230	9.59E-01	3.72E-26	5.17E-76	1.26E-38	4.35E-41	4.64E-15	2.40E-10	6.70E-59	0.00E+00	
IA RPG	1.04E-95	3.46E-01	1.53E-24	7.56E-20	3.94E-29	1.15E-08	2.01E-03	6.72E-63	7.59E-13	6.25E-36	5.69E-12	4.52E-88	6.42E-179	5.07E-24	3.39E-61	8.53E-31	1.03E-110	9.42E-01	0.00E+00

The highlighted values in Table 2&3 shows that there are strong correlations within the independent variables, and all of the correlations (-0.5<rho>0.5) are statistically significant (pval<0.05).

Table 4: Correlation coefficients and p-values of independent variabales against target variable (allStar)

	Load	rTS%	Box Creatio	Passer Rating	Spacing	cTOV%	ScoreVal	PlayVal	FTA/100	FT%	rORB%	rDREB%	OBPM	BPM	IA Pts/75	IA PPG	IA APG	IA RPG
corr_coef	0.27134	0.16207	0.0924	0.00651	-0.0335	-0.0383	0.343841	0.16925	0.26893	0.06035	0.01423	0.08549	0.33399	0.24532	0.28626	0.40722	0.2167	0.28903
pval	3.69E-19	1.31E-07	2.74E-03	8.33E-01	2.79E-01	2.16E-01	1.77E-30	3.49E-08	7.76E-19	5.07E-02	6.45E-01	5.60E-03	9.43E-29	7.66E-16	3.10E-21	3.61E-43	1.30E-12	1.24E-21

Table 4 gives the researcher an idea of how strong the correlation between the independent variables and the dependent variable is, and whether they are statistically significant. This way, the researcher can perform feature selection based on the strength of correlation, while also avoiding the selection of heavily correlated independent values by using Table 2&3.

5.4.4 Feature Importance Analysis

The researcher also performed feature importance analysis using two models, *DecisionTreeClassifier* and *RandomForestClassifier* and obtained two different results:

Table 5 (left): Feature Importance ranking from DecisionTreeClassifier

Colu	Featur	Feature
0	IA PPG	0.235173
1	MP	0.075105
2	BPM	0.074621
3	rORB%	0.069683
4	Load	0.065157
5	ScoreVal	0.061941
6	FTA/100	0.057608
7	OBPM	0.049488
8	Spacing	0.048014
9	Box Creation	0.041267
10	IA RPG	0.038826
11	rTS%	0.036652
12	cTOV%	0.032974
13	rDREB%	0.029898
14	PlayVal	0.027570
15	IA APG	0.018452
16	IA Pts/75	0.017954
17	Passer Rating	0.011284
18	FT%	0.008334

Colum n1	Feature	new fi
0	IA PPG	0.102072
1	MP	0.088535
2	ScoreVal	0.063973
3	OBPM	0.061755
4	IA Pts/75	0.060011
5	FTA/100	0.053146
6	IA RPG	0.051502
7	BPM	0.051335
8	Load	0.051214
9	Spacing	0.049125
10	rTS%	0.047557
11	rDREB%	0.045703
12	cTOV%	0.045488
13	rORB%	0.043422
14	IA APG	0.041495
15	Box Creation	0.039116
16	Passer Rating	0.036963
17	FT%	0.035188
18	PlayVal	0.032401

Table 6 (right): Feature Importance Ranking from RandomForestClassifier

The reason of avoiding highly correlated independent values as input in a model is that they can cause unnecessary complexity as there may be redundancy within the said attributes (Tavory, 2017). This is especially profound for models like logistic regression or Random Forest, both selected for this study.

As a conclusion for the hypothesis, the three approach of correlation analysis indicates three different rankings of features to experiment on in the modelling stage, as shown in the table below:

Index	1	2	3	4	5	6	7	8	9
Random Forest	IA PPG	MP	ScoreVal	OBPM	IA Pts/75	FTA/100	IA RPG	BPM	Load
Decision Tree	IA PPG	MP	BPM	rORB%	Load	ScoreVal	FTA/100	OBPM	Spacing
Pearson Analysis	IA PPG	ScoreVal	MP	OBPM	FTA/100	IA Pts/75	Load	BPM	IA RPG

10	11	12	13	14	15	16	17	18	19
Spacing	rTS%	rDREB%	cTOV%	rORB%	IA APG	Box Creation	Passer Rat	FT%	PlayVal
Box Creation	IA RPG	rTS%	cTOV%	rDREB%	PlayVal	IA APG	IA Pts/75	Passer Rat	FT%
IA APG	rTS%	PlayVal	Box Creation	rDREB%	FT%	rORB%	Passer Rat	Spacing	cTOV%

Figure 38: Feature Selection of 3 approaches

5.6 Modelling

The first step to modelling is to deal with class imbalance problem. The data contains more than 1200 regular players and around 480 All-Star players. The difference is due to the nature of the game as only less than 30 from hundreds of players are selected as All-Star each year. The approach chosen is to undersample the non-All-Star entries.

```
Data = Data.drop(Data.query('allStar < 1').sample(frac=.6).index)
'''
Author: piRSquared
Date: 2020
Availablilty: https://stackoverflow.com/questions/41912204/randomly-remove-rows-from-d
'''
#
Data['allStar'].value_counts().plot.bar(title='number of future All-Stars', color=('bl
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x2b4d8541f88>
```

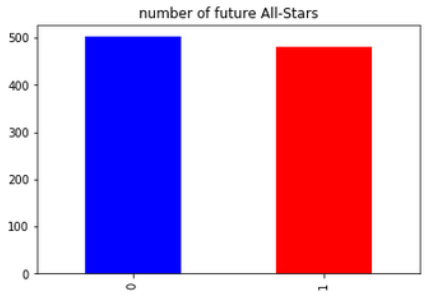


Figure 39: Code Snippet undersample data

The researcher tried out multiple approach for the feature selection for this study. The first approach is by referring to the Pearson analysis results from the hypothesis mentioned above, by choosing the independent variables with strong correlation to the dependent variable but removing them if they are heavily correlated with a higher ranked independent variable.

Another approach is by measuring the feature importance using classifiers like Decision Tree and Random Forest. The two models produced slightly different results, as shown in hypothesis section earlier.

```
from sklearn.ensemble import RandomForestClassifier

my_importance_model = RandomForestClassifier(n_estimators=10000, random_state=0, n_job
my_importance_model.fit(x, y)
print(my_importance_model.feature_importances_)
'''
Author: Farhad Malik
Date: 2020
Availablilty: https://medium.com/fintechexplained/processing-data-to-improve-machine-l
'''
```

Figure 40: Code Snippet measuring feature importance

These approaches of feature selection are implemented together with the Stepwise selection method where features are added in one by one according to their correlation strength and checking the accuracy change. The feature selection together with the actual modelling phase is an iterative process where the researcher will choose to add in new features to the mix or drop existing features depending on the observed accuracy score of executed models. This approach is a reference to the study by Kumar and Chong as shown in the figure below:

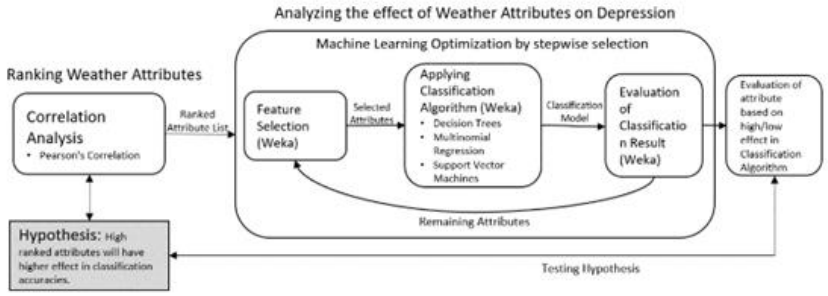


Figure 41: Feature Selection Methodology (Kumar and Chong, 2018)

The data is then split into training and testing set with a ratio of 8 to 2. After that, the data is standardized so that independent variables of larger scales like *Spacing* or *Load* will not end up outweighing those of smaller scales like *rTS%* or *cTOV%*, as the process standardize the range of all the independent variables.

```
# #set variables as x, target as y

x_col = ['Load', 'ScoreVal', 'FTA/100', 'BPM', 'IA PPG', 'rORB%']
x = Data[x_col]
y = Data['allStar']

#split the data into train test set
x_train, x_test, y_train, y_test = train_test_split(x,y, test_size = 0.2)
#https://stackoverflow.com/questions/13610074/is-there-a-rule-of-thumb-for-how-to-divi

#standardize the input data using Standard Scaler, get better results
scaler = StandardScaler()
x_train = scaler.fit_transform(x_train)
x_test = scaler.transform(x_test)
```

Figure 42: Code Snippet train test split & standardization

The training data is then ready to be fitted into different types of classifier models. The models chosen here include: Logistic Regression, Neural Network, Support Vector Machine and Random Forest. The researcher used the *GridSearchCV* function which allows the user to try out different combination of parameters for the specific model, also known as hyper-parameter tuning. The *RandomizedCV* function is also put to practice here on more resource

intensive model to cut down on time spent as it can perform randomly sampled grid search and obtain similar results. These models are executed multiple times and average score are taken to counter the slight variation of score between runs due to the randomness.

Model 1: Logistic Regression(baseline model)

```
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import GridSearchCV
logModel = LogisticRegression()
param_grid = [
    {'penalty' : ['l1', 'l2', 'elasticnet', 'none'],
    'C': np.logspace(-4,1,4,20),
     'solver': ['lbfgs', 'newton-cg', 'liblinear', 'sag', 'saga'],
     'max_iter': [100, 1000, 2500, 5000]
    }
]

clf1 = GridSearchCV(logModel, param_grid = param_grid, cv = 10,
                    verbose=True, scoring = 'accuracy', n_jobs=-1)

best_clf1 = clf1.fit(x_train, y_train)
best_clf1.score(x_test, y_test)
```

```
Fitting 10 folds for each of 320 candidates, totalling 3200 fits
```

```
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 4 concurrent workers.
[Parallel(n_jobs=-1)]: Done 312 tasks      | elapsed:    0.2s
[Parallel(n_jobs=-1)]: Done 3200 out of 3200 | elapsed:    2.3s finished
```

```
0.6395939086294417
```

Figure 43: Code Snippet GridSearchCV with Logistic Regression

```
from sklearn.neural_network import MLPClassifier
from sklearn.model_selection import RandomizedSearchCV
#use classifier because target is binary (category)
nnModel = MLPClassifier()

parameters = {'solver': ['lbfgs'], 'max_iter':
              [1000,1100,1200,1300,1400,1500,1600,1700,1800,1900,2000 ],
              'alpha': 10.0 ** -np.arange(1, 10),
              'hidden_layer_sizes':np.arange(10, 15),
              'random_state':[0,1,2,3,4,5,6,7,8,9]}

clf2 = RandomizedSearchCV(nnModel, parameters, cv=10, verbose=True,
                    scoring = 'accuracy', n_jobs=-1)

results = []
total = 0
for counter in range(1, 6):
    best_clf2  = clf2.fit(x_train, y_train)
    total = total + best_clf2.score(x_test, y_test)
    results.append(best_clf2.score(x_test, y_test))

print(results)
print(total/5)

best_clf2 = clf2.fit(x_train, y_train)
best_clf2.score(x_test, y_test)
```

Figure 44: Code Snippet RandomizedCV with MLPClassifier (Neural Network)

Another advantage of these two functions is that they have Cross Validation function built in. With the *cv* parameter, the users can determine how many sets to split the training set into, and the modelling will be ran using different split of the data fitted in, giving a score that represents the model's accuracy on unseen dataset (Brownlee, 2018).

Finally, the trained model will be tested with the test data to acquire the accuracy score which will be the ultimate measurement for the accuracy of this classification study.

5.7 Summary

In short, the researcher approaches the selection of feature with three importance rankings along with stepwise selection method, before splitting the data into train and test set. The training set will be fitted into GridSearchCV or Randomized functions of different models before being each of the selected best models are tested with the test set to determine all the models' accuracy score.

CHAPTER 6: RESULTS AND DISCUSSION

6.1 Introduction

There are 4 possible outcomes for each classification made, as shown in the table below:

Table 7: Possible outcomes for the classification

	Predicted non-All-Star	Predicted All-Star
Actual non-All-Star	True Negative, TN	False Positive, FP
Actual All-Star	False Negative, FN	True Positive, TP

The accuracy metric used here for the classification is the true outcomes divided by the total number of classifications made, as in:

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

The reason of choosing this metric is to avoid bias on accuracy of either the All-Star or non-All-Star classification due to the use case of this research. The investment made on signing a professional athlete for years is no small amount. Therefore, this model should make sure it can identify True Positives and True Negative at the same time.

6.2 Results and discussion

The results of discussion will be split into 3 approaches, each showing results of different models from the different approach of feature selection. Then the best model from each of the approach will be compared and the one with the best score will be deemed as the best feature selection and model configuration type for this study.

6.2.1 Approach 1 (Pearson Correlation Coefficient Analysis)

Model	Accuracy	Parameters
Logistic Regression	**0.745**	C: 0.0001, max_iter: 100, penalty: l2, solver: lbfgs
Neural Network	0.671	solver: lbfgs, random_state: 5, max_iter: 1000, hidden_layer_sizes: 12, alpha: 0.01
Support Vector Machine	0.744	kernel: rbf, gamma: 0.3, C: 1
Random Forest	0.742	n_estimators: 1000, min_samples_leaf: 5, max_leaf_nodes: 10, max_depth: 25

6.2.2 Approach 2 (Decision Tree Classifier Feature Importance)

Model	Accuracy	Parameters
Logistic Regression	**0.745**	C: 0.0001, max_iter: 100, penalty: l2, solver: liblinear
Neural Network	0.682	solver: lbfgs, random_state: 4, max_iter: 1100, hidden_layer_sizes: 12, alpha: 0.0001
Support Vector Machine	0.724	kernel: linear, C: 10
Random Forest	0.722	n_estimators: 1000, min_samples_leaf: 10, max_leaf_nodes: 2, max_depth: 30

6.2.3 Approach 3((Random Forest Classifier Feature Importance)

Model	Accuracy	Parameters
Logistic Regression	**0.750**	C: 0.0001, max_iter: 100, penalty: l2, solver: liblinear
Neural Network	0.692	solver: lbfgs, random_state: 9, max_iter: 1200, hidden_layer_sizes: 11, alpha: 1e-06
Support Vector Machine	0.720	kernel: rbf, gamma: 0.3, C: 1
Random Forest	0.718	n_estimators: 10, min_samples_leaf: 2, max_leaf_nodes: 10, max_depth: None

Across the board, logistic regression is best performing later regardless of the feature selection method, while Neural Network is the worst performing of the bunch. Support Vector Machine and Random Forest performs similarly. Of the 3 approaches, the one using Pearson

Correlation Analysis for feature selection yields the best overall result. The researcher believes that is because of the features selected. Approach 2 & 3 selected features *Load* and *Pts/75* respectively and both of them are heavily correlated with the most important features of all, *IA PPG*. To simplify, the more Load one carries, the more he is likely to produce, leading to high IA PPG. On the other hand, *IA Pts/75* is just pace adjusted value of *IA PPG*. This had led to the researcher's firm believe of the results difference is significant.

The top models from each of the approach is shown as below:

Table 8: Models comparison

App.	Model	Accuracy	Parameters
1	Logistic Regression	0.745	C: 0.0001, max_iter: 100, penalty: l2, solver: lbfgs
2	Logistic Regression	0.745	C: 0.0001, max_iter: 100, penalty: l2, solver: liblinear
3	Logistic Regression	**0.750**	C: 0.0001, max_iter: 100, penalty: l2, solver: liblinear

The Logistic regression model with the best score is the last one, with parameters of *C=0.0001*, *max_iter = 100*, *penalty=l2* and *solver=liblinear*.

As a conclusion, the approach that yields the best result is to implement the Pearson Correlation Analysis for feature selection and Logistic Regression with the parameters mentioned above.

CHAPTER 7: CONCLUSIONS AND REFLECTIONS

At the end of this project, the researcher is able to build a classification model that is able to predict if an NBA player will become an All-Star player in 7 years' time based on his rookie or sophomore year stat line.

The researcher believed he had done enough research and was able to achieve what he had set out to accomplish at the start of the project. However, there is no denial of the fact that there is always more research to do to make the final product more refined. For example, despite trying out with three feature selection method in this study, there are more feature selection methods that may prove to be better for this study as feature selection is a book length topic and is crucial to the accuracy of any prediction model.

Along the process of completing the research, multiple ideas for future improvement on this study had come up. First of all is the **data** being used can see big upgrades in the future. The data being used in this study are often heavily correlated as the attributes measures very similar aspect of the game from different perspective. For example, *IA PPG* is adjusted scoring average and is a reflection of a player's scoring ability, which is measured by *ScoreVal*, while *OBPM* is also a reflection of a players' impact on the offensive end. *Load* is another attribute that correlates with all the offensive metrics as the more *Load* one has, the more likely he will produce. The attributes used in this study, although are advanced statistics, is less comprehensive of the aspects of basketball compared to the study by Soliman et al, therefore the classification is less accurate. As the sports analytics market grows exponentially, more investments will be made into the sensors to acquire more contextualize data as well as the research of interpreting the data, which will greatly benefit a research like this. The second improvement to be made in the future is the **way to handle the outliers**. This research mixes the All-Star players, which often posts literally outstanding numbers, together with the average players, making their entries often sticks out as outliers. For the purpose of this study, the researcher chose to remove the minimum number of outliers to avoid losing out too many All-Star entries. The improvement here can be other methods to handling the outliers like cluster analysis or even when the data grows more in amount, the outlier will not be so significant. The other improvement is about **feature selection**, as there

may be a better combination of features for this study that requires extensive research on its own as mentioned earlier.

All in all, although the classification produced here is not of the highest accuracy, it is a business decision aid at the end of the day. Business intelligence models like this aims to provide intelligence to the high management of a business so that they can make conscious and well-informed decision.

REFERENCES

1) Thabtah, F., Zhang, L. and Abdelhamid, N. (2019). NBA Game Result Prediction Using Feature Analysis and Machine Learning. *Annals of Data Science*, [online] 6(1), pp.103-116. Available at: https://link.springer.com/article/10.1007/s40745-018-00189-x [Accessed 27 Apr. 2019].

2) Golovko, E., Leonov, Y. and Pysarenko, L. (2015). MODERN TREND IN SPORTS INDUSTRY. *Innovative methods in teaching english in higher and secondary education*, [online] p.90. Available at: https://core.ac.uk/reader/33758220 [Accessed 26 Apr. 2019].

3) Barroso Duarte, G. (2018). Nike Inc. Company Report. [online] pp.3-26. Available at: https://run.unl.pt/bitstream/10362/35352/1/Duarte_2018.pdf [Accessed 26 Apr. 2019].

4) Team, T. and Speculations, G. (2016). *Under Armor: How Stephen Curry Helped Sell Shoes*. [online] Forbes.com. Available at: https://www.forbes.com/sites/greatspeculations/2016/02/18/under-armor-how-stephen-curry-helped-sell-shoes/#c044e6765772 [Accessed 26 Apr. 2019].

5) NBA Stats. (2019). *All Time Leaders*. [online] Available at: https://stats.nba.com/alltime-leaders/?SeasonType=Playoffs&StatCategory=FG3M [Accessed 26 Apr. 2019].

6) Rovell, D. (2015). NBA, Nike have near-$1B apparel deal. [online] ESPN.com. Available at: http://www.espn.com/nba/story/_/id/13053413/nba-signs-8-year-apparel-deal-nike [Accessed 26 Apr. 2019].

7) Garcia, A. (2018). *NFL and Nike extend uniform deal through 2028*. [online] CNNMoney. Available at: https://money.cnn.com/2018/03/27/news/companies/nike-nfl-gear-contract/index.html [Accessed 26 Apr. 2019].

8) NBA.com/Stats (2019). FAQ. [online] NBA Stats. Available at: https://stats.nba.com/help/faq/ [Accessed 27 Apr. 2019].

9) Taylor, B., 2019. *The "Advanced" Box Score Explained | Pace & Four Factors (NBA Stats 101 Part 3)*. [online] Youtube. Available at: <https://www.youtube.com/watch?v=gJAoM-eF_f8> [Accessed 4 July 2020].

10) NBA, 2016. *Stats LLC And NBA To Make STATS Sportvu Player Tracking Data Available To More Fans Than Ever Before - NBA.Com: NBA Communications*. [online] NBA.com: NBA Communications. Available at: <https://pr.nba.com/stats-llc-nba-sportvu-player-tracking-data/> [Accessed 4 July 2020].

11) STATS. (2019). Basketball Player Tracking for Pro Teams | SportVU | STATS. [online] Available at: https://www.stats.com/sportvu-basketball/ [Accessed 27 Apr. 2019]

12) Baek, H., 2019. *Animating Expected Possession Value In The NBA | Inside The Tidyverse*. [online] Inside the Tidyverse. Available at: <http://insidethetv.rbind.io/post/animating-expected-possession-value-in-the-nba/> [Accessed 4 July 2020].

13) Beardsley, B. (2017). *Winning with Data Science, Golden State Warriors Style - Dataconomy*. [online] Dataconomy. Available at:

https://dataconomy.com/2017/07/golden-state-warriors-data-science/ [Accessed 27 Apr. 2019]

14) Morgulev, E., Azar, O. and Lidor, R., 2018. Sports analytics and the big-data era. *International Journal of Data Science and Analytics*, 5(4), pp.213-222.

15) Kayhan, V. and Watkins, A. (2018). A Data Snapshot Approach for Making Real-Time Predictions in Basketball. *Big Data*, 6(2), pp.96-112.

16) Ahmadalinezhad, M., Makrehchi, M. and Seward, N. (2019). Basketball Lineup Performance Prediction Using Network Analysis. *2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*.

17) Metulini, R., Manisera, M. and Zuccolotto, P. (2017). Sensor Analytics in Basketball. *Proceedings of the 6th International Conference on Mathematics in Sport*, ISBN 978-88-6938-058-7.

18) Metulini, R. (2018). Players Movement and Team Shooting Performance: A Data Mining approach for Basketball. *49th Scientific meeting of the Italian Statistical Society" SIS2018 proceeding.*

19) Manisera, M., Metulini, R. and Zuccolotto, P. (2019). Basketball Analytics Using Spatial Tracking Data. *New Statistical Developments in Data Science*, SIS 2017, Florence, Italy, June 28-30, pp.305-318.

20) Safir, Jonathan, "How Analytics, Big Data, and Technology Have Impacted Basketball's Quest to Maximize Efficiency and Optimization" (2015). *Senior Capstone Projects.* Paper 390.

21) Wang, K. and Zemel, R. (2016). Classifying NBA Offensive Plays Using Neural Networks. *MIT Sloan Sports Analytics Conference 2016 Research Papers Competition*.

22) Vinué, G. and Epifanio, I. (2019). Forecasting basketball players' performance using sparse functional data*. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 12(6), pp.534-547.

23) Goh, Y., Goh, Y., Ling, R. and Chee, W. (2019). Predicting the performance of the players in NBA Players by divided regression analysis. *Malaysian Journal of Fundamental and Applied Sciences*, 15(3), pp.441-446.

24) Soliman, G., Misbah, A., El-Nabawy, A. and Eldawlatly, S. (2017). Predicting All Star Player in the National Basketball Association using Random Forest. *Intelligent Systems Conference 2017*, pp.706-713.

25) Vinué, G. and Epifanio, I., 2019. Forecasting basketball players' performance using sparse functional data. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, *12*(6), pp.534-547.

26) Python.org. (2020). *Welcome to Python.org*. [online] Available at: https://www.python.org/about/ [Accessed 6 Feb. 2020].

27) Srivastava, S. (2019). *Top 10 Data Science Programming Languages for 2020*. [online] Analytics Insight. Available at: https://www.analyticsinsight.net/top-10-data-science-programming-languages-for-2020/ [Accessed 6 Feb. 2020].

28) Kumar Bachheriya, A. (2019). *Top 6 Data Science Programming Languages for 2019*. [online] Medium. Available at: https://medium.com/datadriveninvestor/top-6-data-science-programming-languages-for-2019-39ba1b6819a8 [Accessed 6 Feb. 2020].

29) R-project.org. (2020). *R: What is R?*. [online] Available at: https://www.r-project.org/about.html [Accessed 6 Feb. 2020].

30) Team, D. (2019). *Pros and Cons of R Programming Language - Unveil the Essential Aspects! - DataFlair*. [online] DataFlair. Available at: https://data-flair.training/blogs/pros-and-cons-of-r-programming-language/ [Accessed 6 Feb. 2020].

31) Hooda, S. (2020). *What is the Best Python IDE for Data Science? - KDnuggets*. [online] KDnuggets. Available at: https://www.kdnuggets.com/2018/11/best-python-ide-data-science.html [Accessed 6 Feb. 2020].

32) Henrique Vasconcellos, P. (2018). *Top 5 Python IDEs For Data Science*. [online] DataCamp Community. Available at: https://www.datacamp.com/community/tutorials/data-science-python-ide [Accessed 6 Feb. 2020].

33) Sakuragi, H. (2019). *Morioh - Connecting with Programmers and Developers all over the World*. [online] Morioh.com. Available at: https://morioh.com/p/a6236d3c9539 [Accessed 6 Feb. 2020].

34) PyPI. (2020). *pandas*. [online] Available at: https://pypi.org/project/pandas/ [Accessed 10 Feb. 2020].

35) PyPI. (2020). *numpy*. [online] Available at: https://pypi.org/project/numpy/ [Accessed 10 Feb. 2020].

36) PyPI. (2020). *scipy*. [online] Available at: https://pypi.org/project/scipy/ [Accessed 10 Feb. 2020].

37) PyPI. (2020). *matplotlib*. [online] Available at: https://pypi.org/project/matplotlib/ [Accessed 10 Feb. 2020].

38) scikit-learn, 2020. *Scikit-Learn: Machine Learning In Python — Scikit-Learn 0.23.2 Documentation*. [online] Scikit-learn.org. Available at: <https://scikit-learn.org/stable/> [Accessed 19 August 2020].

39) Support.microsoft.com. (2020). [online] Available at: https://support.microsoft.com/en-us/help/4000823 [Accessed 10 Feb. 2020].

40) Bott, E. (2015). *Microsoft commits to 10-year support lifecycle for Windows 10 | ZDNet*. [online] ZDNet. Available at: https://www.zdnet.com/article/microsoft-commits-to-10-year-support-lifecycle-for-windows-10/ [Accessed 10 Feb. 2020].

41) Wiemer, H., Drowatzky, L. and Ihlenfeldt, S., 2019. Data Mining Methodology for Engineering Applications (DMME)—A Holistic Extension to the CRISP-DM Model. *Applied Sciences*, 9(12), p.2407.

42) Dåderman, A. and Rosander, S., 2018. Evaluating frameworks for implementing machine learning in signal processing: A comparative study of CRISP-DM, semma and kdd.

43) Desai, R. (2019). *Top 10 Python Libraries for Data Science*. [online] Medium. Available at: https://towardsdatascience.com/top-10-python-libraries-for-data-science-cd82294ec266 [Accessed 11 Feb. 2020].

44) Softwaretestinghelp.com. (2020). *12 Best Python IDEs and Code Editors in 2020*. [online] Available at: https://www.softwaretestinghelp.com/python-ide-code-editors/ [Accessed 11 Feb. 2020].

45) Toomey, D., 2016. *Learning Jupyter*. Packt Publishing Ltd.

46) Kazarinoff, P., 2020. *Why Jupyter Notebooks? - Problem Solving With Python*. [online] Problemsolvingwithpython.com. Available at: <https://problemsolvingwithpython.com/02-Jupyter-Notebooks/02.02-Why-Jupyter-Notebooks/> [Accessed 13 July 2020].

47) Nerds, R., 2018. *A Beginner'S Guide To Installing Jupyter Notebook Using Anaconda Distribution*. [online] Medium. Available at: <https://medium.com/@neuralnets/beginners-quick-guide-for-handling-issues-launching-jupyter-notebook-for-python-using-anaconda-8be3d57a209b> [Accessed 13 July 2020].

48) Tavory, A., 2017. *In Supervised Learning, Why Is It Bad To Have Correlated Features?*. [online] Data Science Stack Exchange. Available at: <https://datascience.stackexchange.com/questions/24452/in-supervised-learning-why-is-it-bad-to-have-correlated-features> [Accessed 18 August 2020].

49) Kumar, S. and Chong, I., 2018. Correlation Analysis to Identify the Effective Data in Machine Learning: Prediction of Depressive Disorder and Emotion States. *International Journal of Environmental Research and Public Health*, 15(12), p.2907.

50) Brownlee, J., 2018. *A Gentle Introduction To K-Fold Cross-Validation*. [online] Machine Learning Mastery. Available at: <https://machinelearningmastery.com/k-fold-cross-validation/> [Accessed 19 August 2020].

APPENDICES

FYP POSTER

APU : 87275
PLS V1.0

Project Log Sheet – Supervisory Session

Notes on use of the project log sheet:

1. This log sheet is designed for meetings of more than 15 minutes duration, of which there must be at minimum SIX (6) during the course of the project (SIX mandatory supervisory sessions).
2. The student should prepare for the supervisory sessions by deciding which question(s) he or she needs to ask the supervisor and what progress has been made (if any) since the last session, and noting these in the relevant sections of the form, effectively forming an agenda for the session.
3. A log sheet is to be brought by the STUDENT to each supervisory session.
4. The actions by the student (and, perhaps the supervisor), which should be carried out before the next session should be noted briefly in the relevant section of the form.
5. The student should leave a copy (after the session) of the Project Log Sheet with the supervisor and to the administrator at the academic counter. A copy is retained by the student to be filed in the project file.
6. It is recommended that students bring along log sheets of previous meetings together with the project file during each supervisory session.
7. The log sheet is an important deliverable for the project and an important record of a student's organisation and learning experience. The student **must** hand in the log sheets as an appendix of the final year documentation, with sheets dated and numbered consecutively.

Student's name: TAN JIAN SEAN **Date:** 27/11/2019 **Meeting No:** 1

Project title: ~~Mrs Hema Latha Krishna Nair~~ FYP **Intake:** UC3F1911CS(DA)

☐ **Entry logged into PAGOL**

Supervisor's name: Mrs. HEMA LATHA KRISHNA NAIR **Supervisor's signature:**

Items for discussion (noted by student before mandatory supervisory meeting):

1. FYP proposal
2. overall feasibility
3. alternative plan
4.

Record of discussion (noted by student during mandatory supervisory meeting):

1. Proposal doesn't contribute to the program Learning outcome for Computer Science
2. Scope is too wide, need to narrow down
3. Redo aim, objectives and deliverables
4.

Action List (to be attempted or completed by student by the next mandatory supervisory meeting):

1. Objective & Scope
2. Technical implementation
3.

Note: A student should make an appointment to meet his or her supervisor (via the consultation system) at least ONE (1) week prior to a mandatory supervisor session – please see document on project timelines. In the event a supervisor could not be booked for consultation, the project manager should be informed ONE (1) week prior to the session so that a meeting can be subsequently arranged.

Project Log Sheet

Student Copy

APU : 87276

PLS V1.0

Project Log Sheet – Supervisory Session

Notes on use of the project log sheet:

1. This log sheet is designed for meetings of more than 15 minutes duration, of which there must be at minimum SIX (6) during the course of the project (SIX mandatory supervisory sessions).
2. The student should prepare for the supervisory sessions by deciding which question(s) he or she needs to ask the supervisor and what progress has been made (if any) since the last session, and noting these in the relevant sections of the form, effectively forming an agenda for the session.
3. A log sheet is to be brought by the STUDENT to each supervisory session.
4. The actions by the student (and, perhaps the supervisor), which should be carried out before the next session should be noted briefly in the relevant section of the form.
5. The student should leave a copy (after the session) of the Project Log Sheet with the supervisor and to the administrator at the academic counter. A copy is retained by the student to be filed in the project file.
6. It is recommended that students bring along log sheets of previous meetings together with the project file during each supervisory session.
7. The log sheet is an important deliverable for the project and an important record of a student's organisation and learning experience. The student **must** hand in the log sheets as an appendix of the final year documentation, with sheets dated and numbered consecutively.

Student's name: TAN JIAN SEAN **Date:** 5/12/2019 **Meeting No:** 2

Project title: FYP **Intake:** UC3F1911 CS(DA)

☐ **Entry logged into PAGOL**

Supervisor's name: Mrs. HEMA LATHA KRISHNA NAIR **Supervisor's signature:**

Items for discussion (noted by student before mandatory supervisory meeting):

1. New Proposal
2. PSF
3. Aim, Objective, Scope
4. Methodology

Record of discussion (noted by student during mandatory supervisory meeting):

1.
2.
3.
4.

Action List (to be attempted or completed by student by the next mandatory supervisory meeting):

1.
2.
3.

Note: A student should make an appointment to meet his or her supervisor (via the consultation system) at least ONE (1) week prior to a mandatory supervisor session – please see document on project timelines. In the event a supervisor could not be booked for consultation, the project manager should be informed ONE (1) week prior to the session so that a meeting can be subsequently arranged.

Project Log Sheet

Student Copy

APU : 87238

PLS V1.0

Project Log Sheet – Supervisory Session

Notes on use of the project log sheet:

1. This log sheet is designed for meetings of more than 15 minutes duration, of which there must be at minimum SIX (6) during the course of the project (SIX mandatory supervisory sessions).
2. The student should prepare for the supervisory sessions by deciding which question(s) he or she needs to ask the supervisor and what progress has been made (if any) since the last session, and noting these in the relevant sections of the form, effectively forming an agenda for the session.
3. A log sheet is to be brought by the STUDENT to each supervisory session.
4. The actions by the student (and, perhaps the supervisor), which should be carried out before the next session should be noted briefly in the relevant section of the form.
5. The student should leave a copy (after the session) of the Project Log Sheet with the supervisor and to the administrator at the academic counter. A copy is retained by the student to be filed in the project file.
6. It is recommended that students bring along log sheets of previous meetings together with the project file during each supervisory session.
7. The log sheet is an important deliverable for the project and an important record of a student's organisation and learning experience. The student **must** hand in the log sheets as an appendix of the final year documentation, with sheets dated and numbered consecutively.

Student's name: TAN JIAN SEAN **Date:** 13/1/19 **Meeting No:** 3

Project title: National Basketball Association Player Future Performance Prediction Model **Intake:** UC3F1911 CS(DA)

☐ Entry logged into PAGOL

Supervisor's name: Ms. HEMA LATHA KRISHNA NAIR **Supervisor's signature:**

Items for discussion (noted by student before mandatory supervisory meeting):

1. Ethics form
2.
3.
4.

Record of discussion (noted by student during mandatory supervisory meeting):

1. Sources of research
2.
3.
4.

Action List (to be attempted or completed by student by the next mandatory supervisory meeting):

1. list out and attached the discussed sources
2.
3.

Note: A student should make an appointment to meet his or her supervisor (via the consultation system) at least ONE (1) week prior to a mandatory supervisor session – please see document on project timelines. In the event a supervisor could not be booked for consultation, the project manager should be informed ONE (1) week prior to the session so that a meeting can be subsequently arranged.

Project Log Sheet

Student Copy

APU : 87277

PLS V1.0

Project Log Sheet – Supervisory Session

Notes on use of the project log sheet:

1. This log sheet is designed for meetings of more than 15 minutes duration, of which there must be at minimum SIX (6) during the course of the project (SIX mandatory supervisory sessions).
2. The student should prepare for the supervisory sessions by deciding which question(s) he or she needs to ask the supervisor and what progress has been made (if any) since the last session, and noting these in the relevant sections of the form, effectively forming an agenda for the session.
3. A log sheet is to be brought by the STUDENT to each supervisory session.
4. The actions by the student (and, perhaps the supervisor), which should be carried out before the next session should be noted briefly in the relevant section of the form.
5. The student should leave a copy (after the session) of the Project Log Sheet with the supervisor and to the administrator at the academic counter. A copy is retained by the student to be filed in the project file.
6. It is recommended that students bring along log sheets of previous meetings together with the project file during each supervisory session.
7. The log sheet is an important deliverable for the project and an important record of a student's organisation and learning experience. The student **must** hand in the log sheets as an appendix of the final year documentation, with sheets dated and numbered consecutively.

Student's name: TAN JIAN SEAN **Date:** 21/1/2020 **Meeting No:** 4

Project title: National Basketball Association Player Future Performance Prediction Model **Intake:** UC3F1911CS(DA)

☐ Entry logged into PAGOL

Supervisor's name: Ms. HEMA LATHA KRISHNA NAIR **Supervisor's signature:**

Items for discussion (noted by student before mandatory supervisory meeting):

1. IR Chapter 2 : Literature Review
2. Chapter 3 : Technical Research
3.
4.

Record of discussion (noted by student during mandatory supervisory meeting):

1. Literature review must be in-depth
2.
3.
4.

Action List (to be attempted or completed by student by the next mandatory supervisory meeting):

1. ~~Re~~ Finish by this week for review
2.
3.

Note: A student should make an appointment to meet his or her supervisor (via the consultation system) at least ONE (1) week prior to a mandatory supervisor session – please see document on project timelines. In the event a supervisor could not be booked for consultation, the project manager should be informed ONE (1) week prior to the session so that a meeting can be subsequently arranged.

Project Log Sheet

Student Copy

(APIIT: Serial Number)

PLS V1.0

Project Log Sheet – Supervisory Session

Notes on use of the project log sheet:

1. This log sheet is designed for meetings of more than 15 minutes duration, of which there must be at minimum SIX (6) during the course of the project (SIX mandatory supervisory sessions).
2. The student should prepare for the supervisory sessions by deciding which question(s) he or she needs to ask the supervisor and what progress has been made (if any) since the last session, and noting these in the relevant sections of the form, effectively forming an agenda for the session.
3. A log sheet is to be brought by the STUDENT to each supervisory session.
4. The actions by the student (and, perhaps the supervisor), which should be carried out before the next session should be noted briefly in the relevant section of the form.
5. The student should leave a copy (after the session) of the Project Log Sheet with the supervisor and to the administrator at the academic counter. A copy is retained by the student to be filed in the project file.
6. It is recommended that students bring along log sheets of previous meetings together with the project file during each supervisory session.
7. The log sheet is an important deliverable for the project and an important record of a student's organisation and learning experience. The student **must** hand in the log sheets as an appendix of the final year documentation, with sheets dated and numbered consecutively.

Student's name: …Tan Jian Sean… …….. **Date:** …30/6/2020 … **Meeting No:** …5…..

Project title: National Basketball Association Player Future Performance Prediction Model **Intake:** UC3F1911CS(DA)

Supervisor's name: …Ms. Hema Latha Krishna Nair **Supervisor's signature:** ……..…

Items for discussion (noted by student before mandatory supervisory meeting):

1. IR area of improvement
2. method of data gathering
3.
4.

Record of discussion (noted by student during mandatory supervisory meeting):

1. data can be manual download (csv file), no need for api stuff
2.
3.
4.

Action List (to be attempted or completed by student by the next mandatory supervisory meeting):

1. do a table of problem statement, objective, all scope (must match), model (measurement/ clustering, classification) (8th of July)
2. **goal and objective rewrite (for ir), methodology, literature reviews(ref within 5 years)**
3.

Note: A student should make an appointment to meet his or her supervisor (via the consultation system) at least ONE (1) week prior to a mandatory supervisor session – please see document on project timelines. In the event a supervisor could not be booked for consultation, the project manager should be informed ONE (1) week prior to the session so that a meeting can be subsequently arranged.

Project Log Sheet

(APIIT: Serial Number)

PLS V1.0

Project Log Sheet – Supervisory Session

Notes on use of the project log sheet:

1. This log sheet is designed for meetings of more than 15 minutes duration, of which there must be at minimum SIX (6) during the course of the project (SIX mandatory supervisory sessions).
2. The student should prepare for the supervisory sessions by deciding which question(s) he or she needs to ask the supervisor and what progress has been made (if any) since the last session, and noting these in the relevant sections of the form, effectively forming an agenda for the session.
3. A log sheet is to be brought by the STUDENT to each supervisory session.
4. The actions by the student (and, perhaps the supervisor), which should be carried out before the next session should be noted briefly in the relevant section of the form.
5. The student should leave a copy (after the session) of the Project Log Sheet with the supervisor and to the administrator at the academic counter. A copy is retained by the student to be filed in the project file.
6. It is recommended that students bring along log sheets of previous meetings together with the project file during each supervisory session.
7. The log sheet is an important deliverable for the project and an important record of a student's organisation and learning experience. The student **must** hand in the log sheets as an appendix of the final year documentation, with sheets dated and numbered consecutively.

Student's name: …Tan Jian Sean… …….. **Date:** …7/7/2020 … **Meeting No:** …6…..

Project title: National Basketball Association Player Future Performance Prediction Model **Intake:** UC3F1911CS(DA)

Supervisor's name: …Ms. Hema Latha Krishna Nair **Supervisor's signature:** ……..…

Items for discussion (noted by student before mandatory supervisory meeting):

1. Mapping of scope & objectives
2. method of data gathering
3.
4.

Record of discussion (noted by student during mandatory supervisory meeting):

1. refinement on scope & objectives
2. methods to achieve the 3 objectives
3.
4.

Action List (to be attempted or completed by student by the next mandatory supervisory meeting):

1. collect data set
2. OLAP reporting on dataset
3. star schema (design) for how to data

Note: A student should make an appointment to meet his or her supervisor (via the consultation system) at least ONE (1) week prior to a mandatory supervisor session – please see document on project timelines. In the event a supervisor could not be booked for consultation, the project manager should be informed ONE (1) week prior to the session so that a meeting can be subsequently arranged.

Project Log Sheet

(APIIT: Serial Number)

PLS V1.0

Project Log Sheet – Supervisory Session

Notes on use of the project log sheet:

1. This log sheet is designed for meetings of more than 15 minutes duration, of which there must be at minimum SIX (6) during the course of the project (SIX mandatory supervisory sessions).
2. The student should prepare for the supervisory sessions by deciding which question(s) he or she needs to ask the supervisor and what progress has been made (if any) since the last session, and noting these in the relevant sections of the form, effectively forming an agenda for the session.
3. A log sheet is to be brought by the STUDENT to each supervisory session.
4. The actions by the student (and, perhaps the supervisor), which should be carried out before the next session should be noted briefly in the relevant section of the form.
5. The student should leave a copy (after the session) of the Project Log Sheet with the supervisor and to the administrator at the academic counter. A copy is retained by the student to be filed in the project file.
6. It is recommended that students bring along log sheets of previous meetings together with the project file during each supervisory session.
7. The log sheet is an important deliverable for the project and an important record of a student's organisation and learning experience. The student **must** hand in the log sheets as an appendix of the final year documentation, with sheets dated and numbered consecutively.

Student's name: …Tan Jian Sean… …….. **Date:** …30/6/2020 ... **Meeting No:** …7…...

Project title: National Basketball Association Player Future Performance Prediction Model **Intake:** UC3F1911CS(DA)

Supervisor's name: …Dr. Booma P M……..….. **Supervisor's signature: ……..…**

Items for discussion (noted by student before mandatory supervisory meeting):

1. Briefing on FYP approach, aim and objective

2. reason for change of supervisor

3. current progress

4. change of IDE (to Jupyter Notebook)

Record of discussion (noted by student during mandatory supervisory meeting):

1. final product should include interactive interface for user to input player's name and get binary output of prediction

2.

3.

4.

Action List (to be attempted or completed by student by the next mandatory supervisory meeting):

1. perform data exploration & data cleaning

2. book next meeting ASAP

3.

Note: A student should make an appointment to meet his or her supervisor (via the consultation system) at least ONE (1) week prior to a mandatory supervisor session – please see document on project timelines. In the event a supervisor could not be booked for consultation, the project manager should be informed ONE (1) week prior to the session so that a meeting can be subsequently arranged.

(APIIT: Serial Number)

PLS V1.0

Project Log Sheet – Supervisory Session

Notes on use of the project log sheet:

1. This log sheet is designed for meetings of more than 15 minutes duration, of which there must be at minimum SIX (6) during the course of the project (SIX mandatory supervisory sessions).
2. The student should prepare for the supervisory sessions by deciding which question(s) he or she needs to ask the supervisor and what progress has been made (if any) since the last session, and noting these in the relevant sections of the form, effectively forming an agenda for the session.
3. A log sheet is to be brought by the STUDENT to each supervisory session.
4. The actions by the student (and, perhaps the supervisor), which should be carried out before the next session should be noted briefly in the relevant section of the form.
5. The student should leave a copy (after the session) of the Project Log Sheet with the supervisor and to the administrator at the academic counter. A copy is retained by the student to be filed in the project file.
6. It is recommended that students bring along log sheets of previous meetings together with the project file during each supervisory session.
7. The log sheet is an important deliverable for the project and an important record of a student's organisation and learning experience. The student **must** hand in the log sheets as an appendix of the final year documentation, with sheets dated and numbered consecutively.

Student's name: …Tan Jian Sean… …….. **Date:** …12/8/2020 ... **Meeting No:** …6…...

Project title: National Basketball Association Player Future Performance Prediction Model **Intake:** UC3F1911CS(DA)

Supervisor's name: …Dr. Booma P M……..….. **Supervisor's signature:** ……..…

Items for discussion (noted by student before mandatory supervisory meeting):

1. **Assumptions for the project**
2. **Need of referencing the code**
3. **Outliers removal**
4.

Record of discussion (noted by student during mandatory supervisory meeting):

1. **Actions taken are acceptable**
2.
3.
4.

Action List (to be attempted or completed by student by the next mandatory supervisory meeting):

1. **proceed on to modelling stage**
2. **study on and perform multivariate analysis**
3.

Note: A student should make an appointment to meet his or her supervisor (via the consultation system) at least ONE (1) week prior to a mandatory supervisor session – please see document on project timelines. In the event a supervisor could not be booked for consultation, the project manager should be informed ONE (1) week prior to the session so that a meeting can be subsequently arranged.

Project Log Sheet

PROJECT PROPOSAL FORM

FYP Title:

National Basketball Association Player Future Performance Prediction Model

Introduction:

The sports world is an ever-expanding market. In 2014, the global sports industry has an estimated worth of 1.5 trillion in USD (Thabtah, Zhang and Abdelhamid, 2019). With the improvement of living standards around the globe, the wide spread of internet and mobile devices, more and more people are looking for an entertainment, a hobby to invest their time in. Sports is a huge beneficiary of the trend. With more and more fans around the world supporting their favourite players or teams, the sports market is more valuable and potential than ever. Just as Golovko, Leonov and Pysarenko had pointed out, the global sports industry expected the figure of revenue to grow to 145 billion (Golovko, Leonov and Pysarenko, 2015). Sports brands who produces and sells sports-related products such as sports equipment, sports drinks, sports clothing finds potential in the increasing fanbase and globalized market. One of the popular ways of marketing is to sign brand ambassadors. However, with the market being dominated by big brands (Barroso Duarte, 2018), it is difficult for the upcoming company to sign established star players. The National Basketball Association, also known as the NBA, is another league of the case mentioned above. Therefore, smaller companies are left with a choice of signing young emerging players that may turn out to be either a future all-star or a complete bust. With the proposed system, we look to predict the young, potential players' future stardom and allows companies to sign them before their blossom into superstars, therefore owning a brand ambassador with a solid fanbase that will bring revenue to the investing companies.

Problem Statement:

The global sports market is being dominated by well established companies like Nike, Adidas and Under Armour. The phenomenon extends to the signing of the players, where stars and superstars are ambassadors of the sports giants. For the small brands to compete, they can only resort to signing young, potential players that may turn into a star in the near future, just like Under Armour did in 2013. However, the question is: How many Stephen Curry are there

to be found? What if Under Armour wasn't that lucky and Stephen Curry did not blossom into a superstar?

Project Aim:

The aim of this study is to deliver a predicting model that predicts future stardom of emerging player, thus delivering chance of competition for less established brands.

Project Objectives:

To develop a predicting model, a few objectives have to be achieved. First of all, sufficient data have to be acquired since the prediction result takes a few years to come out. After that, a suitable prediction model should be chosen for implementation. The data will have to be partitioned into training set and validation set. The training set will be used to fine tune the model, making sure the accuracy is up to requirement. The validation set is used to make sure the model doesn't overfit to the training set, which will affect the accuracy of the model.

Literature Review:

Importance of Brand Ambassadors

The importance of a brand ambassadors speaks volume in Stephen Curry and Under Armour's partnership. In 2013, the sports equipment company signed the then young player to a 2-year shoe deal. By the end of 2015, the company's footwear segment made a 57% leap in growth, while revenues were increased by 95% (Team and Speculations, 2016). The athletic wear giant's success in recent years has a sure correlation with Stephen Curry's career finally going full bloom. Before the contract was signed, Curry is a promising talented player but with major concern in terms of his injury prone legs. Big brands like Nike was reluctant to sign him to any big deal despite his obvious upside and opted for other player with less history of injury. Fast forward to 2019, Curry is now one of the most successful individuals in the National Basketball Association. Awards like 3x NBA Champion, 2x Most Valuable Player, All-time playoff 3PM leader, single season 3PM record holder are just part of his impressive resume (NBA Stats, 2019).

Domination of Big Brands in the Market

It is no secret that the sports market, specifically in the footwear & apparel field, is being dominated by giants namely Nike, Adidas and Under Armour. As shown in Barroso Duarte's research, even though Adidas, Under Armour and Puma are following Nike in the ranking, it isn't even close in real world market share (BARROSO DUARTE, 2018).

Statistics in the NBA

The NBA is one of the most data-oriented sports league in the world, with statistics like points, assists, rebounds, steals and blocks being recorded since the 1973/1974 season (NBA.com/Stats, 2019). Twenty plus years later, advanced stats like Pace, Offensive and Defensive Efficiency are introduced, allowing analyst to look deeper into a team's performance, beyond the basic stats of a team that may be deceiving based on different context. To simplify, Pace is about measuring a team's defensive and offensive stats by looking at them on a per-possession basis instead of number of points scored or allowed per game, since each team has a unique playing style and Pace. For example, as stated by Ben Taylor, a comparison between advanced scoring stats of Oscar Robertson in 1962 and Kevin Durant in 2012 reveals the difference pace can make (Taylor, 2019).

Table 9: Scoring Efficiency comparison (Taylor, 2019)

Players (Season)	Points per game	Pos per 48 min	Pts per 75
Oscar Robertson (1962)	31	129	20.1
Kevin Durant (2012)	28	93	28.1

As shown in the table above, when looking at the points per game stats in the respective seasons, Robertson seems to be the superior scorer here. However, Robertson plays at a much faster pace compare to Durant. Upon adjusting the scoring average to the pace, Robertson's advantage seems to disappear as Durant turns out to be a far more effective scorer with an 8-point advantage per 75 possessions.

STATS SportVU Tracking System

Data analytics have changed the landscape of NBA, with the modern tools being implemented into the basketball court. In the year of 2009, the NBA started implementing the STATS SportVU tracking system and when on to be the first professional sports league to support player tracking in all games played (NBA, 2016). The system consists of six cameras, installed in a basketball arena, and is capable of tracking real-time position of all the players on court and the ball, 25 times per second. The system provides a rich data containing speed distance, player separation and ball possession. It creates an all new dimension of spatial data, which is a lot different than the conventional stat sheet data like points, rebounds and assists. One of the earliest teams to apply the SportVU system was Golden State Warriors, who have experienced great success for the past years.

The rich combination of data gives the coaching staff, players and analyst the most comprehensive and sophisticated view ever of the game. The system can be used to prevent injuries, make better coaching decision and even maximize Player Performance (STATS, 2019). One of the products that comes from the ground-breaking system is an advanced statistic called Expected Possession Value (EPV) developed by Cervone, D'Amour, Bornn, and Kirk Goldsberry (Baek, 2019), which as stated by Beardsley, is a huge key to the recent success for the Golden State Warriors (Beardsley, 2017).

The General Sports Analytics Framework

Sports Analytics, as defined by Alamar, is "the management of structured historical data, the application of predictive analytical models that utilize that data, and the use of information systems to inform decision makers and enable them to help their organizations in gaining a competitive edge on the field of play" (Morgulev, Azar and Lidor, 2018). The definition outlines the three main components which are data management, analytical models and information systems, and that the purpose of sports analytics is to aid the decision makers of the organization to gain an upper hand against the competition.

All the data, coming from sources, has to be prepared at the data management stage. When the data is ready, it is sent to information systems or analytic models. Analytic models can act either as a provider of processed data to the information system or as an ad-hoc function to answer questions of the decision maker. The information system will present the resulting knowledge or information to the decision maker in an efficient and clear manner.

Deliverables:

A prediction model that will be able to predict future stardom of young NBA player based on the players performance data.

PROJECT SPECIFICATION FORM

FYP Title:

National Basketball Association Player Future Performance Prediction Model

Brief Description on project Background:

Problem Context

The sports world is an ever-expanding market. In 2014, the global sports industry has an estimated worth of 1.5 trillion in USD. With the improvement of living standards around the globe, the wide spread of internet and mobile devices, more and more people are looking for an entertainment, a hobby to invest their time in. Sports is a huge beneficiary of the trend. With more and more fans around the world supporting their favorite players or teams, the sports market is more valuable and potential than ever. Just as the research of Golovko, Leonov and Pysarenko had pointed out, the global sports industry expected the figure of revenue to grow to 145 billion. Sports brands who produces and sells sports-related products such as sports equipment, sports drinks, sports clothing finds potential in the increasing fanbase and globalized market. One of the popular ways of marketing is to sign brand ambassadors. However, with the market being dominated by big brands, it is difficult for the upcoming company to sign established star players. The National Basketball Association, also known as the NBA, is another league of the case mentioned above.

Rationale

Therefore, smaller companies are left with a choice of signing young emerging players that may turn out to be either a future all-star or a complete bust. With the proposed system, we look to predict the young, potential players future stardom and allows companies to sign them before their blossom into superstars, therefore owning a brand ambassador with a solid fanbase at a lower cost, but still will bring revenue to the investing companies.

Tangible Benefits

-cut down costs of signing a star player as brand ambassador

-lower chances of signing a player that may not be good in the future

-allows start-up or smaller brands to advertise their brands by signing potential stars

Intangible benefits

-brings competition to the business field

-allows underrated professional players to be discovered

Nature of Challenge

The challenge here is mainly about the accuracy of the model, as in the real world, the signing of a ambassador can be costly, so signing an inaccurately predicted player will be a terrible decision for the company financially. The accuracy of the prediction model can be affected by a number of factors, namely the type of data to use, choices of different prediction models, also choosing the measurements for the player performance.

Brief Description of project Objectives:

Deliverable

A prediction model that can predict a future stardom of a young player in less than 3 years, based on the historical performance stats.

Brief Description of resources needed by the Proposal:

Hardware

A computer with a 64-bit system, has internet connection, at least 4GB of RAM,

Software

- Java Development Kits (JDKs)/Java Runtime Environments (JREs)
- Java Application Servers
- Web Browsers
- Adobe Flash Player
- SAS® Enterprise Miner™ 14.3

Access to information or Expertise

This system will need consistent data from nba official statistics website, stats.nba.com, which is d daily. Also, in order to understand the advanced statistics better, one may need to refer to papers produced by the researcher that came up with those specific measurements.

Academic research being carried out and other information, techniques being learned:

Online Resources

- Home, S. (2019). Season Leaders. [online] NBA Stats. Available at: https:stats.nba.com/players/ [Accessed 16 Dec. 2019].

Brief description of the development plan for the proposed project:

CRISP-DM

The methodology chosen in this study is a modified version of general CRISP-DM. CRISP-DM stands for Cross-Industry Standard Process for Data Mining. Usually, the CRISP-DM methodology is a 6-step framework where the steps are in order as below: 1) Business Understanding, 2) Data Understanding, 3) Data Preparation, 4) Model Building, 5) Testing and Evaluation, 6) Deployment. With the modified version of the framework, we look to eliminate the Business Understanding step where usually, in business analytics, data analyst spend time trying to understand the business field knowledge. However, in this case, we have no need to understand the domain knowledge.

Data Understanding

At the beginning, knowledge of the data that will be processed is gathered. Users will get familiar with the data, which will help in identifying quality problems as well as executing basic exploration of the data. Interesting subsets may also be discovered in this stage. For the study, users will have to get familiar with the NBA statistics in this stage, have a general idea of the game, knows about simple stats like points, rebounds and steals before they move on to advanced stats like Player Efficiency Rating (PER) and Net Rating.

Data Preparation

This phase covers all activities that will turn the initial raw data into the final dataset which will be thrown into the modelling tool in the following stage. Data preparation usually include attribute ion, data cleaning, construction of new attributes, and transformation of data. For the study, this stage will introduce new advanced data of our target player. The attributes likely include the looks of Estimated Possession Value (EPV), plus/minus, Player Efficiency Rating and other advanced stats that measures the players efficiency as well as their impact on the court. Finally, a new binary attribute will be introduced as well, which is to measure if a player is a star in 3 years' time. The attribute is required as a target variable for the predicting model in the following stage in order to perform prediction.

Modeling

In the modeling stage, multiple modeling techniques are ed and applied. The parameters of the models will be fine tweak to produce optimal results. This phase typically has a strong connection with the Data Preparation phase, where one may go through the Data Preparation process again to get datasets of better fit for the model chosen. For this study, prediction models such as Decision Tree, Regression or Neural Network will be applied. The advanced stats mentioned in the previous stage will be fed into the models, with a purpose to train the models accuracy. The data will be partitioned into training, validation and test set. The training set will train the models accuracy, while the validation set will prevent situation where the model overfits with the training set. The test set will be used to test the final accuracy of the model.

Evaluation

Before proceeding to this stage, the user will have a few models that performs great data analysis wise. Before deployment of the model, the model has to be evaluated carefully. Every step executed within the construction of the model will be reviewed, making sure that it will properly achieves the business objectives. For this study, the main objective is to produce a model that will give an accurate prediction on whether or not a player will turn into a superstar, thus aiding the business decision of a brand where they have to decide which player to invest on.

Deployment

When the model passes the evaluation, it is time for deployment. In this study, the deployment process is as simple as generating a report declaring the completion of the prediction model and applying the model into the latest player and statistic available. It is also suggested to a feedback loop on the framework where real world results of the predictions made are feed back into the model, thus increasing the accuracy of the model for future use. This action can counteract the nature of NBA where factors that affect players success, like on-court rules, are constantly in change.

Proposed Prediction Model

In order to build the model, featured subset from the initial dataset has to be ed. Then, the ed dataset for testing and training will be partitioned into testing and training dataset. After that, prediction models like decision trees, neural network or regression will be implemented. After analysing the performance of the different models, the best performing model will be ed as the final model.

Brief description of the evaluation and test plan for the proposed project:

Success Criteria

To measure the success of the prediction model, we may refer back to the modelling stage of the CRISP-DM methodology. An amount of data will be partitioned and excluded from being used to build the prediction model. The test data will be used to test the accuracy of the model.

Office Record	Receipt – Fast-Track Ethical Approval
Date Received:	Student name: Student number:
Received by whom:	Received by: Date:

APU
FAST-TRACK ETHICAL APPROVAL FORM (STUDENTS)

Tick one box: ☐ TAUGHT POSTGRADUATE project ☑ UNDERGRADUATE project
☐ TAUGHT POSTGRADUATE MODULE assignment
☐ TAUGHT UNDERGRADUATE MODULE assignment

Title of Specialism on which enrolled ... BSc (Hons) in Computer Science with specialism in Data Analytics

Tick one box: Full-Time Study ☑ or Part-Time Study ☐

Title of project ... National Basketball Association Player Future Performance Prediction Model

Name of student researcher ... TAN JIAN SEAN

Name of supervisor ... Mrs. HEMA LATHA KRISHNA NAIR

Student Researchers- please note that certain professional organisations have ethical guidelines that you may need to consult when completing this form.

Supervisors/Module Tutors - please seek guidance from the Chair of the APU Research Ethics Committee if you are uncertain about any ethical issue arising from this application.

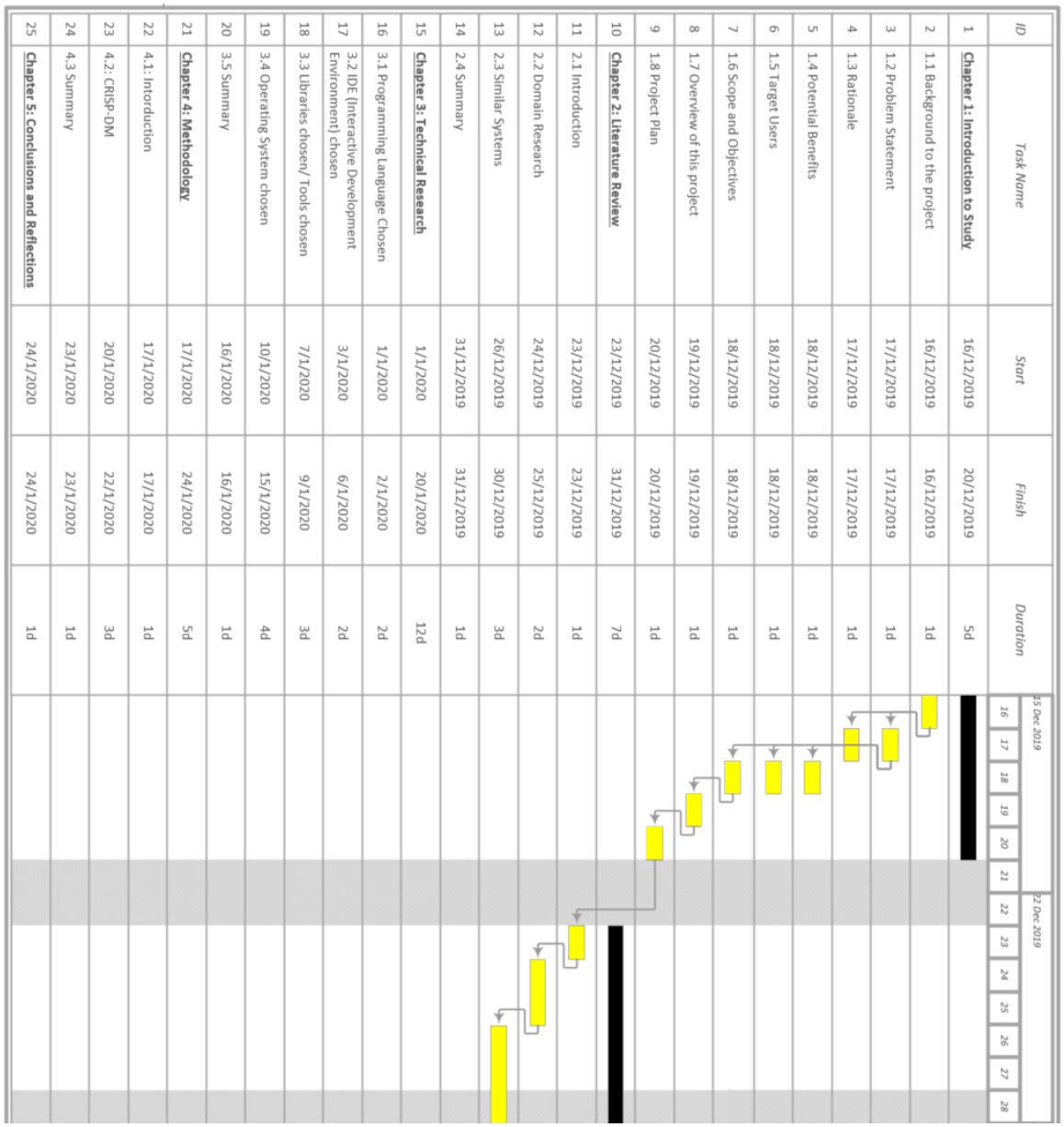
		YES	NO	N/A
1	Will you describe the main procedures to participants in advance, so that they are informed about what to expect?	✓		
2	Will you tell participants that their participation is voluntary?	✓		
3	Will you obtain written consent for participation?	✓		
4	If the research is observational, will you ask participants for their consent to being observed?	✓		
5	Will you tell participants that they may withdraw from the research at any time and for any reason?	✓		
6	With questionnaires and interviews will you give participants the option of omitting questions they do not want to answer?	✓		
7	Will you tell participants that their data will be treated with full confidentiality and that, if published, it will not be identifiable as theirs?	✓		
8	Will you give participants the opportunity to be debriefed i.e. to find out more about the study and its results?	✓		

If you have ticked **No** to any of Q1-8 you should complete the full Ethics Approval Form.

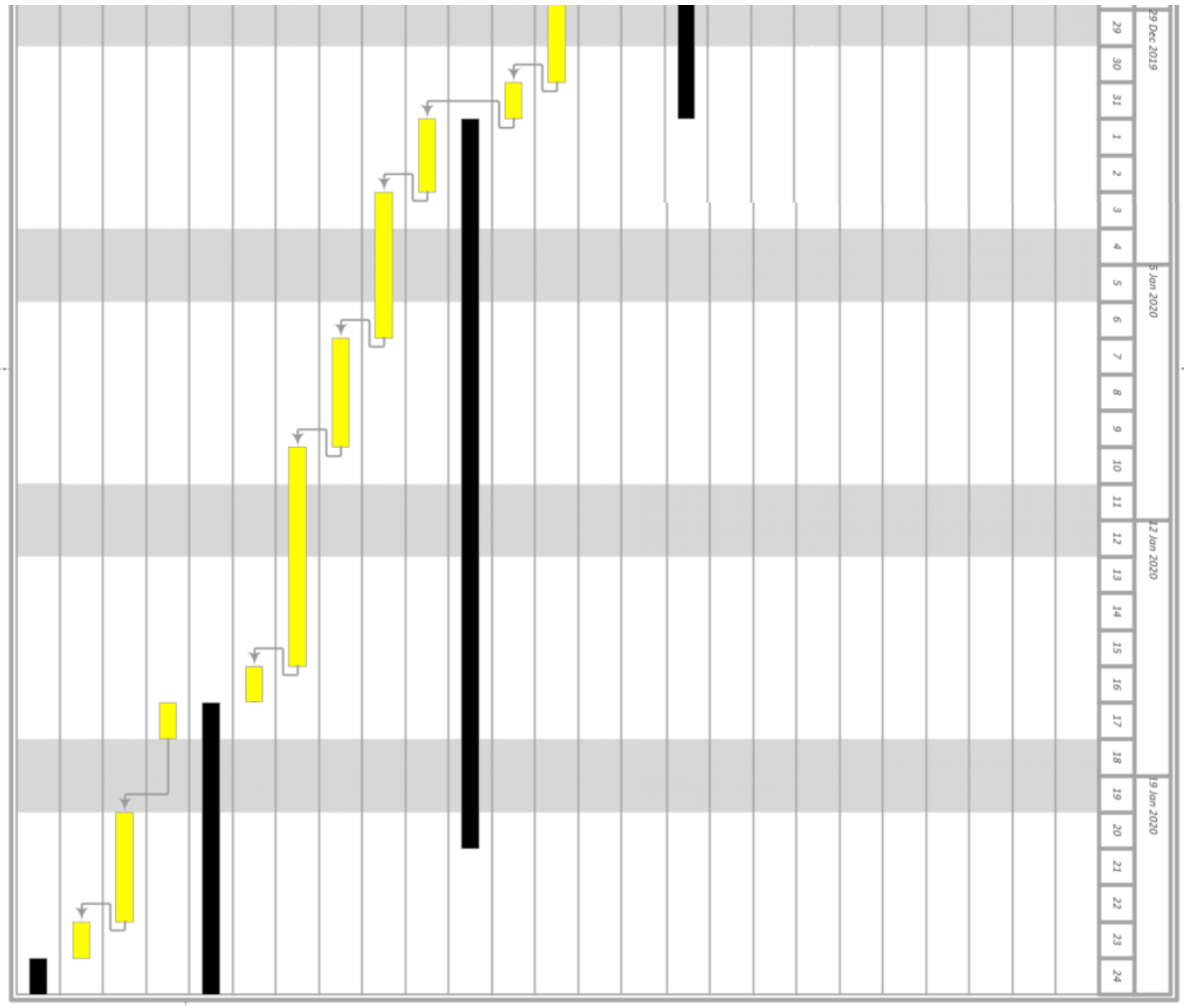
		YES	NO	N/A
9	Will your project deliberately mislead participants in any way?		✓	
10	Is there any realistic risk of any participants experiencing either physical or psychological distress or discomfort?		✓	
11	Is the nature of the research such that contentious or sensitive issues might be involved?		✓	

If you have ticked **Yes** to 9, 10 or 11 you should complete the full Ethics Approval Form. In relation to question 10 this should include details of what you will tell participants to do if they should experience any problems (e.g. who they can contact for help). You may also need to consider risk assessment issues.

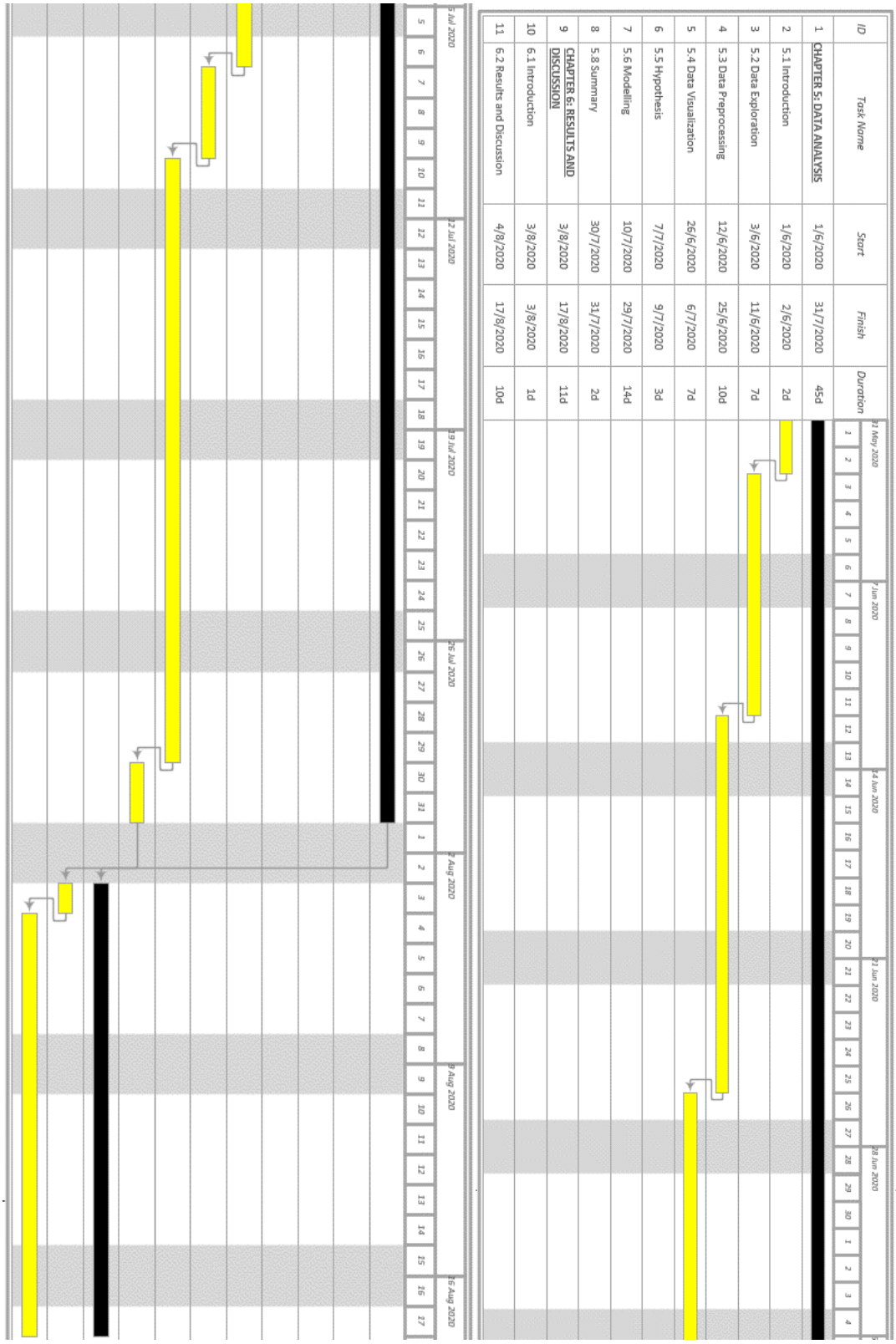
GANTT CHART FOR FYP SEMESTER 1



ID	Task Name	Start	Finish	Duration
1	**Chapter 1: Introduction to Study**	16/12/2019	20/12/2019	5d
2	1.1 Background to the project	16/12/2019	16/12/2019	1d
3	1.2 Problem Statement	17/12/2019	17/12/2019	1d
4	1.3 Rationale	17/12/2019	17/12/2019	1d
5	1.4 Potential Benefits	18/12/2019	18/12/2019	1d
6	1.5 Target Users	18/12/2019	18/12/2019	1d
7	1.6 Scope and Objectives	18/12/2019	18/12/2019	1d
8	1.7 Overview of this project	19/12/2019	19/12/2019	1d
9	1.8 Project Plan	20/12/2019	20/12/2019	1d
10	**Chapter 2: Literature Review**	23/12/2019	31/12/2019	7d
11	2.1 Introduction	23/12/2019	23/12/2019	1d
12	2.2 Domain Research	24/12/2019	25/12/2019	2d
13	2.3 Similar Systems	26/12/2019	30/12/2019	3d
14	2.4 Summary	31/12/2019	31/12/2019	1d
15	**Chapter 3: Technical Research**	1/1/2020	20/1/2020	12d
16	3.1 Programming Language Chosen	1/1/2020	2/1/2020	2d
17	3.2 IDE (Interactive Development Environment) chosen	3/1/2020	6/1/2020	2d
18	3.3 Libraries chosen/ Tools chosen	7/1/2020	9/1/2020	3d
19	3.4 Operating System chosen	10/1/2020	15/1/2020	4d
20	3.5 Summary	16/1/2020	16/1/2020	1d
21	**Chapter 4: Methodology**	17/1/2020	24/1/2020	5d
22	4.1: Intorduction	17/1/2020	17/1/2020	1d
23	4.2: CRISP-DM	20/1/2020	22/1/2020	3d
24	4.3 Summary	23/1/2020	23/1/2020	1d
25	**Chapter 5: Conclusions and Reflections**	24/1/2020	24/1/2020	1d


29 Dec 2019
29
30
31
1
2
3
4
5 Jan 2020
5
6
7
8
9
10
11
12 Jan 2020
12
13
14
15
16
17
18
19 Jan 2020
19
20
21
22
23
24

GANTT CHART FOR FYP SEMESTER 2



ID	Task Name	Start	Finish	Duration
1	CHAPTER 5: DATA ANALYSIS	1/6/2020	31/7/2020	45d
2	5.1 Introduction	1/6/2020	2/6/2020	2d
3	5.2 Data Exploration	3/6/2020	11/6/2020	7d
4	5.3 Data Preprocessing	12/6/2020	25/6/2020	10d
5	5.4 Data Visualization	26/6/2020	6/7/2020	7d
6	5.5 Hypothesis	7/7/2020	9/7/2020	3d
7	5.6 Modelling	10/7/2020	29/7/2020	14d
8	5.8 Summary	30/7/2020	31/7/2020	2d
9	CHAPTER 6: RESULTS AND DISCUSSION	3/8/2020	17/8/2020	11d
10	6.1 Introduction	3/8/2020	3/8/2020	1d
11	6.2 Results and Discussion	4/8/2020	17/8/2020	10d