2021

Identifying significant features for Player Evaluation in NFL comparing ANNs and Traditional Models

Ronan Walsh
Technological University Dublin

Recommended Citation
Walsh, R. (2021). Identifying significant features for Player Evaluation in NFL comparing ANNs and Traditional Models. Technological University Dublin. DOI: 10.21427/EAC6-4R95

Identifying significant features for Player Evaluation in NFL comparing ANNs and Traditional Models

Ronan Walsh

A dissertation submitted in partial fulfilment of the requirements of
Technological University of Dublin for the degree of
M.Sc. in Computing (Data Science)

12/06/2021

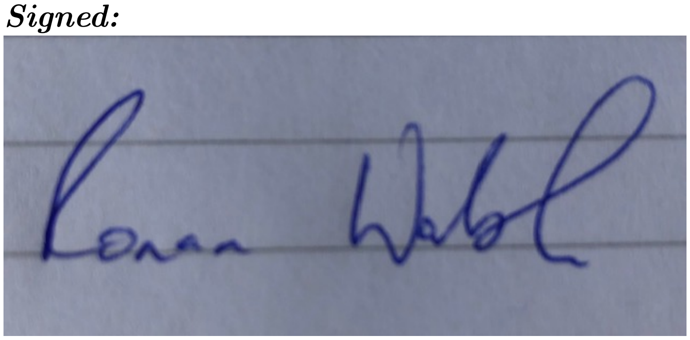
Declaration

I certify that this dissertation which I now submit for examination for the award of MSc in Computing (Data Science), is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

This dissertation was prepared according to the regulations for postgraduate study of the Technological University of Dublin and has not been submitted in whole or part for an award in any other Institute or University.

The work reported on in this dissertation conforms to the principles and requirements of the Institute's guidelines for ethics in research.

Signed:



Date: 12/06/2021

Abstract

The evaluation of player performance in sports is popular and important in modern sports, enabling teams to use real data in the construction of their rosters. This dissertation proposes to apply machine learning algorithms to predicting the player evaluations from a leading NFL analytics company who use a combination of statistics and expert evaluation. In addition, it will investigate what features are significant in the evaluation of a position. Data for the dissertation is obtained from multiple online sources - Pro Football Reference and Pro Football Focus (the the NFL analytics company). These data sets are combined and analysed before applying six different approaches to the problem. The use of Neural Networks (both Single and Multi Layer) as an approach is evaluated against the other approaches of Support Vector Regression (SVR), Linear Regression, Decision Trees and XGBoost. They will be evaluated using accuracy, root mean squared error and the p-value from a t-test. Wrapper methods of Sequential Feature Selection and Permutation Importance are both used to discover relevant features. SVR was the best performing approach with 74% accuracy for QB, 76% accuracy for WR and 59% for RB. Both XGBoost and the Neural Network implementations performed well in comparison. The relevant features that were uncovered fell into two distinct categories. First is a measure of the ability of the player to make an impact on the game when they are involved and receive the ball. The second is a highlight of the importance of solid foundations and basics.

Keywords: NFL, Neural Networks, SVR, Linear Regression, XGBoost, Decision Trees, Deep Neural Networks, Wrapper Methods

Acknowledgments

First of all I would like to express my sincere thanks to my dissertation advisor Patricia O'Byrne. Her guidance and support throughout the whole dissertation process was of immense help and kept me constantly on track. Without her patience and support the process would have been a lot more chaotic.

I would like to thank the staff of TU Dublin that assisted and enlightened me throughout my time in the University.

A big thanks goes out to my colleagues at work, especially my managers who provided the platform for me to undertake this and had to listen to me talk about it constantly.

To my family and friends, a big thank you for all the support and encouragement throughout. For all of the times that I missed I will try to catch up now.

And finally love and a massive and very profound thanks to my wonderful mother, who not alone supported me and provided me with encouragement throughout, even when I was down, but also had to put up with me throughout the various COVID lockdowns. Without your support and love I wouldn't have been able to complete this.

Contents

List of Figures

List of Tables

List of Acronyms

QB	Quarterback
RB	Running Back
WR	Wide Receiver
TE	Tight End
CB	Corner Back
SS	Safety
LB	Linebacker
DI	Defender Interior
ED	Edge Defender
SVM	Support Vector Machines
ANN	Artificial Neural Network
PFF	Pro Football Focus
PFR	Pro Football Reference
RPO	Run Pass Option
TD	Touchdown
MAE	Mean Absolute Error
MSE	MeanSquared Error
SVR	Support Vector Machine Regression
SVM	Support Vector Machines
CNN	Convolutional Neural Networks
RNN	Recurrent Neural Networks
sgd	stochastic gradient decent
SFS	Sequential Feature Selector
ANY/A	Adjusted Net Yards Per Attempt
AY/A	Adjusted Yards Per Attempt

Chapter 1

Introduction

Player evaluation in the NFL is of paramount importance to teams, scouts and fantasy football players alike. The ability to evaluate the performance of a player can be of significant use in contract negotiations and renewals (Byanna & Klabjan, 2016) as well as for fantasy football performance (Landers & Duperrouzel, 2019). Although traditionally football teams employed teams of scouts to observe and evaluate player performance across all teams in the league as well as at the collegiate level.

1.1 Background

With the release of the movie "Moneyball" (Chang & Zenilman, 2013) more modern analytical methods and statistics are being used throughout the evaluation process by NFL teams, but these are often kept in house unavailable to public eyes. Companies such as Pro Football Focus have risen to aid the NFL teams with their analytical capabilities whilst also providing analytics and evaluations/ratings available to third parties and the public for a price.

The works of Devarakonda (2019) and Yurko(2019) have previously looked into the prediction of performance at specific positions with Yurko (2019) looking to implement the famed Win Above Ratio (WAR) metric for offensive players using statistical models. The use of machine learning in prediction and evaluation has been used to evaluate performance across features collated from match data in multiple sports (Oytun, Tinazci, Sekeroglu, Acikada, & Yavuz, 2020) as well as in the NFL (Devarakonda & Colson, 2019). Research in soccer has also investigated methods of identifying the significant features arising from the application of machine learning to the evaluation of player performance (Nsolo, Lambrix, & Carlsson, 2019).

1.2 Research Project/problem

Much of the published research into the evaluation of players in the NFL has focused on the use of raw data obtained from game data, either in game or summarised data. While the evaluation by NFL clubs and analytics companies have made extensive use of expert opinion and analysis earned over decades of studying the game and players, most of the research to date has focused on using raw data to reduce the need for the subjective expert analysts currently used. Working with subjective data could provide added insight and depth for models to learn from.

Many of the machine learning models used to date in evaluating NFL players performance have focused on traditional learning models such as SVM, Regression, Decision Trees, etc. Neural Networks and deep learning models have successfully been used in other sports to evaluate performance (Liu & Schulte, 2018; Liu, Zhu, & Schulte, 2019; Oytun et al., 2020). With a large set of features available, neural networks can help learn non-linear relationships that may exist within the data and learn from feedback from new data.

While a lot of analysis has been performed on the evaluation on player performance in the NFL, little has been published in terms of identifying the significant features that feed into that evaluation and if so, then the research focuses on one position group mainly on the offensive side of the teams.

1.2.1 Research Question

Do artificial neural network techniques applied to the evaluation of NFL players achieve a similar or better regression evaluation (R-squared) to that of traditional machine learning techniques using domain expert evaluations as target features while helping to identify the most important and impactful features on a position by position basis?

1.2.2 Hypothesis

Alternate Hypothesis

Neural Networks offer comparable regression performance to traditional approaches when evaluating player performance in NFL against industry evaluation and can help identify the most important and impactful metrics using wrapper method for player evaluation.

Null Hypothesis

Neural Networks do not offer comparable regression performance to traditional approaches when evaluating player performance in NFL against industry evaluation using R2 and can help identify the most important and impactful metrics using wrapper method for player evaluation.

1.3 Research Objectives

The objective of this study is twofold: to compare the performance of neural network models against other machine learning models for the use of player performance evaluation, as well as discovering the significant features that contribute to the evaluation of the players on a position by position basis.

This study will be carried out in 5 parts. The first part will involve the collation of the data sets from internet resources and getting the data into a form that is ready for the subsequent modelling phases. The following two phases will involve modelling of the data using the machine learning models discussed. An evaluation phase of the machine learning models will follow to find the best model for the evaluation of player performance as well as comparing the results of artificial neural networks to other models. The final phase will involve looking at discovering the most significant features from the most performant model.

As mentioned above the first part of the study will involve retrieving the data from the online resources. There are two main online resources from which to obtain the data. The first resource contains grades from PFF[1] which are compiled from analytics as well as expert analysts in the field of NFL player performance evaluation. There are a number of grades per position group, but for all offensive players there is an offensive grade and for all defensive players there is a defensive grade. These grades will be used as a target variable for our models. Some of the other grades may also be used as features in the model. The second resource is from Pro Football Reference[2] which collates statistical information for players for seasons or games. For the purposes of this study we will use the season data as that is what the grades above are based upon. There are 4 major groupings of data that we will look at: Passing, Rushing, Receiving and Defense with each having both standard and advanced features which will all be included. Not all players will be in each grouping - for example Quarterback won't be in Defense. As a result each position group (Quarterback, Wide Receiver, Running Back, Tight End, and Defensive units linebackers, cornerbacks, defensive line) will need to be constructed with this in mind. Once the position group data sets have been created, we will then look at statistically analysing each feature, choosing the most relevant and accounting for missing values (if numeric most probably with zero). Another resource that was considered is the DVOA grade from Football Outsiders[3] which rates players taking into account quality of opposition, amongst other variables against a league average. However, outside of the grade it didn't provide any more data than the other two.

Once the data sets have been created and cleaned for use by the models, we will continue to the second part of the study which involves training models using traditional (non-neural network) models. These models will be used to predict the evaluation of a players performance over the course of a season. We will look to implement four different models: Linear Regression, Decision Tree, SVM and Gradient Boosting (XG-Boost). These models encompass differing approaches predicting the evaluation, with

[1] `https://www.pff.com/grades`
[2] `https://www.pro-football-reference.com/`
[3] `https://www.footballoutsiders.com/`

Linear looking for a linear relationship while SVM allows for a model that looks for non-linear relationships to help predict the target, while Gradient Descent allows us to look at the best predictions across an ensemble of models. All models will use K-fold cross validation to test the effectiveness of the models - with the value of K set to 10 to decrease bias.

The third part of the study will focus on the artificial neural network models that are to be evaluated. In this case we will use a standard Multilayer Perceptron (MLP) and a single layer neural net. The MLP is a standard neural network that is feedforward and use back propagation for training. It can have multiple layers with non-linear activation functions which can allow it to look for non-linear relationships. These will also use K-fold cross validation for the same reason with the value of K also set to 10.

The fourth part of the study involves the evaluation of the models and this will be covered in the following section. The final part of the study is to obtain significant features for the players that contribute towards the players performance evaluation for their position. It is proposed to use the wrapper method similar to Nsolo ((Nsolo et al., 2019) to obtain these significant features. The Wrapper method uses a greedy algorithm to evaluate all combinations against r-squared and returns a combination of features that returns the optimal results.

All work in constructing the data set will involve Python using common libraries and frameworks such as Numpy and Pandas. When modelling the traditional machine learning models we will also use Python as the main programming language, again using Pandas as well as SciPy and SciKit Learn. Modelling the neural network models will also use these technologies as well as TensorFlow for some more neural network specific implementations.

1.4 Research Methodologies

The data used is a combination from online data sets which use both quantitative data such as height, weight, etc. as well as expert industry metrics (Primary mixed data). The research will use an inductive approach, where we have a theory that neural networks techniques are as good as traditional techniques when evaluating players and we will observe the results and confirm the hypothesis empirically through experiment and metrics.

1.5 Scope and Limitations

The scope of this dissertation is to build a number of models, based upon historical NFL seasons (10 previous seasons), to predict an evaluation grade for a player as well as discovering some of the key features that contribute to the evaluation of the player. Data sets will be compiled from two main online data sources - Pro Football Reference[4] and Pro Football Focus[5]. Both compile data for each player over the course of a season. While they overlap in a number of metrics, they also have individual features that are unique to that data set. The prediction of the evaluation grade will be completed by 6 different approaches - Linear Regression, Decision Tree, Support Vector Regression, XGBoost, Neural Network and Multi Layer Perceptron. This will allow for the evaluation of Neural Networks against other methodologies as an approach to this problem. In addition, the most relevant features will be deduced from the most accurate model. The models are applied to three positions on the team (Quarterback, Wide Receiver and Running Back) as representatives of the application of the approaches.

Pro Football Focus also employs analysts and NFL experts such as former players and coaches to evaluate players on a play-by-play basis. This overall evaluation factors in a number of factors that may not be represented in the data. For example as American Football is a team game, raw statistics may not paint the full picture. For example, a Quarterback may throw a pass to a Wide Receiver on his team, but the Wide Receiver may not catch the ball and it may ricochet into an opponents hands which may result in a score for the other team. In this instance the Quarterback may have executed the play perfectly but in the raw data it appears as an interception and touchdown for the opposing team. This highlights that there may be features that are

[4] https://www.pro-football-reference.com/
[5] https://www.pff.com/

either not provided online or not yet compiled that would provide more insight into the performance of the players.

As mentioned, Pro Football Focus has analysts evaluate players on a play-by-play basis. The data sets that are being used span an entire season/year. This means that some of the nuances of the individual plays are lost when all the data is summarized for the year. It also means that each record is for one player for that position per year. Whereas play-by-play data may result in thousands of records on which to train, year-by-year data results in a lot less with just 311 Quarterback records over the 10 years.

1.6 Document Outline

The structure for the rest of this thesis is outlined below. It is broken down into the steps taken to address the thesis - initial research performed via Literature Review, data analysis and approach definition in Experiment Design and Methodology, Result and Evaluation in Chapter 4 and finally conclusion in the final chapter where the results will be summarised.

1.6.1 Literature Review

This chapter looks at previous works that have looked at the application of machine learning in sport - in particular in the application of player evaluation. This ranges from previous studies within the sport of American Football such as in sports betting or the evaluation of Offensive Linemen, but also in other sports such as football and ice hockey. These works use a variety of approaches to evaluating player performance including many of the machine learning approaches used in this thesis such as wrapper method and the various machine learning approaches such as Neural Networks, SVRs, etc. They also work over different time domains such as year-by-year, match-by-match and play-by-play. Many of these concepts are used as inspiration within the thesis. Also discussion of the approaches to be taken is included.

1.6.2 Experiment Design and Methodology

This chapter focuses on how the experiment was conducted. It looks at the data - how it was obtained and prepared. It also investigates some of the insights discovered from that data. It outlines the various approaches used through the process to generate an evaluation. It also outlines the approaches used to determine the most significant features in the evaluation of the players.

1.6.3 Results, Evaluation and Discussion

Results of the experiments are presented here. Comparisons between the different approaches are displayed and discussed here for each position. Also investigation into the significant features found for each position are presented and discussed here.

1.6.4 Conclusion

In the final chapter the results, observations and findings are summarized with additional work that could be done discussed. Further recommendations will be proposed.

Chapter 2

Literature Review

In this chapter, the background and research performed are presented. Some context of the NFL, the sport and the positions analysed are presented in section 2.1. Once the scene has been set in terms of the NFL, the research into some evaluation and prediction in other sports is presented in section 2.2. Some approaches to solve similar problems in the NFL are presented next in section 2.3. Gaps in the research follow in section 2.4. Finally, discussion of the methods and approaches to be used in the experiment and detailed in section 2.5.

2.1 NFL Background

The National Football League (NFL) is the professional league for the sport of American Football. It consists of 32 teams divided equally across two conferences - the National Football Conference (NFC) and American Football Conference (AFC). The league was formed in 1920, but it's modern incarnation was formed after a merger with another professional league in 1966[1]. From that merger the final game Super Bowl arrived. This is played between the conference winners and determines the winner of the league at the end of the season. Conference winners are determined through a series of play-off matches at the end of the season where the top teams in the conference

[1] `https://www.profootballhof.com/football-history/chronology-of-professional-football/`

play-off after a regular season of 16 league matches.

A match in the sport is played between two teams, where each team has three different positional groups - offensive, defensive and special teams. The Offensive team group tries to progress down the field to get the ball (which is shaped similar to a rugby ball) into a zone at the other end of the pitch called the end zone. They can do this by either throwing the ball down the field and catching the ball, or by running the ball down the field into the end zone. It is the Defensive team groups job to stop the Offensive team by tackling them either when they run the ball or after they have caught a catch. They can however, catch the ball if it is thrown and turn the ball over. The Offensive team have 4 opportunities to move the ball 10 yards. As soon as they manage to move the ball 10 yards (achieving a First Down) then they will have another 4 attempts to move it 10 yards closer to the end zone (or into the end zone). However, if they don't make 10 yards, then they have to kick the ball back to the other team using the special teams group.

Each position group contains multiple positions and for this thesis three are being used as representatives. Teams acquire players in one of three ways. The first is by trading for a player from another team, giving compensation in return. The next is by acquiring free agents - players who are free of contracts with other teams and clubs don't have to pay compensation. The last is the draft, where teams use a draft process to acquire the best talent from college teams. This draft enables the worst performing teams to acquire the best talent from college.

The first position that will be looked at is the Quarterback position[2]. This is often the most pivotal position on the team. This player receives the ball at the start of an offensive play and either throws the ball to other offensive players or hands the ball off to the running back. They will try to avoid defensive players before throwing and if a defensive player tackles them to the ground it is called a sack. However, if they throw a ball and a defensive player catches the ball it is called an interception. Both of these are negative plays for a Quarterback.

[2] `https://howtheyplay.com/team-sports/Offensive-and-Defensive-Football-Positions-Explained`

The second position is that of wide receiver. This player runs down the field looking to catch the ball from a pass thrown by the Quarterback. They can score touchdowns from throws, and often use a combination of their speed and elusiveness to score or get more yards after a catch. They also need to be effective in running routes and in catching the ball.

The final position is that of the running back. This player normally is handed the ball by the Quarterback and has to run towards the end zone - this is often referred to as a rushing play. They get further down the pitch by either evading defensive players with the help of other offensive players, or by using their power to burst through tackles made by defensive players. Recently they have been used more and more to also catch the ball.

2.2 Evaluation and Prediction in Sports

The use of neural networks within sports is gaining in popularity and is used in a variety of different sports and use cases.

As sports betting is a big industry it is not unusual to see it put to use to try to solve the problem of sports prediction (Purucker, 1996; McCabe & Trevathan, 2008; David, Pasteur, Ahmad, & Janning, 2011; Maszczyk et al., 2014; Bunker & Thabtah, 2019). Given the difficulty in predicting accurate scores, Artificial Neural Networks (ANN) have been used primarily as a classification problem, to determine winners of games.

ANNs have also been used to identify talent and mine data in the Australian Football League (Mccullagh, 2010). Once again ANNs in this instance were used to classify talent as good or bad and then compared to recruitment managers to see if they could be used to assist the recruitment managers.

ANNs have also been applied to the prediction of sports injuries before they happen (McCullagh & Whitfort, 2013). While still further investigation is needed, the initial performance correctly predicted over 80 percent correct classification that an injury would occur. Again the ANN was used as a classifier to predict whether a high or low injury risk would occur. ANNs have also been studied to analyse and optimization of sports training (Perl, 2001).

Indeed when it comes to other sports, ANNs have been used in the evaluation of player performance and for quantifying the impact that a player has on a game and the result that it entails. Ljung and Liu (Liu & Schulte, 2018; Ljung, Carlsson, & Lambrix, 2019; Liu et al., 2019; Liu, Luo, Schulte, & Kharrat, 2020) each used deep learning methodologies to determine a Q function and value from in game play-by-play events across both ice hockey and soccer. They used reinforcement learning techniques to attach value to each of the players actions throughout the course of a match.

ANNs have also been used across multiple sports to evaluate the performance of players using data after the fact. Oytun (Oytun et al., 2020) investigated the performance of olympic handball players while comparing the results against more traditional machine learning models. Using athletes measurements such as BMI as well as their performance in handball specific tests, they were able to construct a number of experiments that, through using r-squared, mean standard error and mean error, that they were able to establish a non-linear relationship between the factors using a radial basis function neural network. ANNs have been investigated as an evaluation tool and predictor in other sports such as NBA (Ji & Li, 2013; Hore & Bhattacharya, 2018), Cricket (Iyer & Sharda, 2009; Saikia, Bhattacharjee, & Lemmer, 2012), Cycling (Kataoka & Gray, 2019) , Swimming (Silva et al., 2007), Archery (Muazu Musa et al., 2019) and general sports performance (Namatevs, Aleksejeva, & Polaka, 2016). Aalbers and Nsolo (Aalbers & Van Haaren, 2019; Nsolo et al., 2019) each investigated identifying roles, top performers and attributes that are shown by the top performers. Aalbers (Aalbers & Van Haaren, 2019) used industry expert data with help from industry experts to define the roles they wished to analyse, while Nsolo (Nsolo et al., 2019) used the wrapper and filter methods to identify the most relevant attributes based on the position in the soccer team.

2.3 Approaches to solve the problem

Approaches observed in relation to NFL data mimic many of the use cases mentioned above, but not the use of ANNs. For example, in terms of Fantasy Football betting (Landers & Duperrouzel, 2019), machine learning algorithms such as boosted decision trees have been used to predict the fantasy points that a player may earn in any given week of games. Pelechrinis and Yurko (Pelechrinis, Winston, Sagarin, & Cabot, 2019; Yurko, Ventura, & Horowitz, 2019) each mainly using statistical methods or traditional machine learning methods. Yurko (Yurko et al., 2019) took the approach as outlined in “Moneyball” (Chang & Zenilman, 2013) - looking at WAR. Other evaluations of player performance have also used traditional methods such as regression, decision trees, SVM, etc. (Byanna & Klabjan, 2016; Porter, 2018; Devarakonda & Colson, 2019).

2.4 Gaps in Research

In terms of gaps in the current literature in relation to NFL and player performance evaluation, the majority of the papers discovered focused primarily on traditional machine learning models that either rely on linear relationships or needs the model to be retrained after a period of new data arriving. Neural Networks allow us to look for non-linear relationships in data which can help us with our predictive power. Some papers related to soccer (Aalbers & Van Haaren, 2019; Nsolo et al., 2019) have looked at identifying features/roles that are shown by the top performers, something which has not been focused on in the NFL literature that was read. Many look at player evaluation in relation to a specific position or position group - specifically offensive players (Byanna & Klabjan, 2016; Yurko et al., 2019; Devarakonda & Colson, 2019). However, given that many of the top earners in 2020 are defensive players, it also makes sense to evaluate defensive players and identify the features that identify the top defensive performers also.

2.5 Methods

In the following sections, the methods used in this thesis to implement the models as well as to extract the most relevant features (Regularization, Early Stopping, Drop Out, Cross Validation, Grid Search, Decision Tree, Linear Regression, SVR, XGBoost, Neural Network and MLP) are outlined.

2.5.1 Model Creation

The following sections outline the machine learning algorithms used in this thesis.

Regularization

A major problem that can affect the performance of machine learning models is over-fitting. This occurs where the trained model is well suited to the data on which it was trained but doesn't perform as well when applied to additional data such as testing data. The model maps closely to the training data including noise/outliers as opposed to the general pattern or trend.

One solution to mitigate such an issue is to use Regularization. Regularization discourages learning a more complex model by penalising extreme values for the parameters in the model. Regularization makes use of a penalty that penalises strong parameters unless they are actually required. It does this by introducing a penalization term to the cost function of the algorithm in question. (Cortes, Mohri, & Rostamizadeh, 2009)

There are two forms of regularization - L1 (Lasso) and L2 (Ridge Regression) - each of which differ by the regularization term used. Each form multiplies a penalty known as a regularization parameter,λ, by a sum. For L1, this sum is the sum of the weights in the model. For L2, it is the square of the sum of the weights in the model.

L1 Regularization Term = $\lambda \sum_{i=1}^{n} \theta_i$

L2 Regularization Term = $\lambda \sum_{i=1}^{n} \theta_i^2$

The difference in summing the square the weights (L2) as opposed to summing the weights (L1) is that L2 will minimize the impact of irrelevant features rather than remove them from the model. On the other hand, L1 can set the weights to zero if they are not relevant - feature reduction.

Early Stopping

Another method to mitigate for over-fitting is called Early Stopping. It is also a form of regularization which operates by limiting the number of iterations of learning based upon a loss metric. As neural network models learn, they begin to generalise and this leads to a decrease in a generalization error/loss metric. However, at some point, this will begin to rise as the model beings to over-fit. Early Stopping attempts to detect this point and to prevent any further over-fitting.

To do this we determine the metric (loss/prediction error) and then monitor the metric at the end of every epoch. The loss error is computed in each epoch by running a validation dataset against the model trained by a training dataset. The validation dataset is different to the test dataset which is used to test the accuracy of the final model. The validation dataset is used to validate while training.

As mentioned previously, the loss error is monitored for each epoch. Initially the loss error should decline. At some point however it may begin to overfit and the loss error will increase. However, one increase may not indicate overfitting. To alleviate for these local minima, the loss is monitored to only stop the training early if a certain number of consecutive epochs produce higher loss errors (Prechelt, 1998).

Drop Out

Another method to avoid over-fitting for neural networks is called "Dropout". With Dropout, a neural network is trying to learn a sparse network with independent neurons. By learning more independent neurons, it will mean that less of them are dependent on other neurons and are more useful. If neurons rely on too many other neurons then the network can become more fragile and there are more dependencies based upon the training data - less general.

To implement Dropout, random neurons are dropped out each epoch. This is achieved by supplying a mask to the hidden layer, which results in none of the signal being propagated through those neurons in the network, but also that the back-propagation of the error will also not be applied to those neurons (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). For each epoch, a different random set of neurons are dropped. This should result in neurons that are more independent of other features. The number of neurons dropped is controlled by a dropout parameter that sets the percentage of neurons to be dropped in each epoch.

Cross Validation

We have seen previously the use of a validation dataset, to help validate the learning of a model. As previously mentioned, this can be achieved by splitting the training dataset into a training dataset and a validation dataset. This validation set is held out of the learning and used in each iteration/epoch to validate the performance of the model. This method is known as the holdout method. However, depending on the size of the training dataset in the holdout, there may not be some under-fitting if the dataset is small. This can occur as the model may not see enough patterns in the data.

However, there is also another method known as K-Fold Cross Validation which can also be used. In K-Fold Cross Validation the dataset is split into K subsets of the same dataset. The holdout described previously is then ran K times, once for each subset of data, and the loss is averaged across the K runs (Refaeilzadeh, Tang, & Liu, 2009). Every data point is therefore considered in a validation set exactly once and ran in a training set k-1 times, helping us to reduce overfitting and use as much of the data as possible for training the model. Generally a value of 10 is used for the number of K.

Grid Search

Grid Search allows us to tune our desired hyper-parameters for our models. It allows us to provide a set of values that we are interested in exploring for their effectiveness in learning the best model. By running the model against all the provided hyper-parameters it can provide us with the optimal hyper-parameters from those supplied.

It runs for all combination of the hyper-parameters. For each value for a hyper-parameter i.e. tolerance, it learns a model for every different combination of the other supplied hyper-parameters. This can be computationally expensive, especially with a large number of hyper-parameters. However, for the supplied hyper parameters it retrieves the best performing set (Hsu, Chang, & Lin, 2003).

Decision Tree

Decision Trees are a supervised learning approach to predict values based upon a set of input features. The output can be either discrete values as classification or they can be continuous for regression. The goal is to predict a target variable based upon a simple set of decision rules that have been learned from the data. A Decision Tree is constructed by recursively splitting the original dataset into sub sets which identify with a particular sub node or leaf node. Decision Trees are often done in this top down fashion. This splitting is calculated based upon known metrics such as Gini, Residual sum of squares or Information Gain to choose a feature at each step that best splits the items in the dataset at that node (Breiman, Friedman, Stone, & Olshen, 1984). These use well known mathematical formulae to decide the split.

Decision Trees can also over-fit like other models. To this end, some of the techniques that we mentioned earlier can be utilised to mitigate for over-fitting. Decision Trees, if left to learn as many leaf nodes as possible, could end up fitting every instance to a leaf node - thereby over-fitting to each value. To combat this, we can use Early Stopping by specifying a minimum factor per node for it to be considered for splitting. One such technique is minimum samples per leaf. In this approach, if the node has less than the least the number of samples, then that node is a leaf node - where no further splits can occur.

L1 and L2 regularization can be thought to be achieved based upon the criterion specified. For Decision Tree regression we can look at a number of criterion by which to decide a split including Mean Absolute Error (MAE) and Mean Squared Error (MSE). These can be thought of as minimizing the L1 and L2 losses respectively. Indeed, we can use Grid Search to search for the best values for the minimum samples per leaf and the criterion.

Decision Trees are attractive for a number of reasons. It has in-built feature selection - more irrelevant features will be used less often. Also it is a white box approach, where the decisions are often easily explained by boolean logic. This can be useful in cases where being explainable is legally required. They are often simple to understand.

However, small changes in the training data can result in big changes and the reasons for splits. As the decisions are made at local node level, there is no guarantee that it is globally the correct decision.

Linear Regression

Linear Regression is a statistical modelling approach to determining a target variable from a set of input features. It models the relationship that the input features have with the target variable. It is a supervised learning technique that predicts an output variable based on a perceived linear relationship with the input features, much like the equation of a line. The error is often fitted using least squared error, measuring the sum of the squared error for each target variable to the line (linear regression value). The lowest sum of the squared errors provides the best fit.

Much like Decision Trees, Linear Regression models can over-fit. To that end, regularization can be applied to the linear model to mitigate for over-fitting. Both L1 (Lasso) and L2 (Ridge) can both be used with Linear Regression. Indeed SciKit Learn, a popular Python Machine Learning library, has implementations of Linear Regression for both L1 and L2. We can use Grid Search to optimise the regularization parameter. For the model, L2 Ridge Regression will be used as the Linear Regression implementation. It penalizes the size of the coefficients in line with L2 regularization outlined earlier [3].

Linear Regression has the advantage of being simple to understand and explain, given that the weights can be provided. However, it can also be sensitive to outliers and assumes that the inputs are not dependent upon each other.

[3] `https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html`

SVR

Support Vector Machines (SVM) have been used widely for classification problems. They try to find a line in multidimensional space which can separate targets into appropriate classes based on which side of the line they reside. They have been quite effective to this purpose. Support Vector Regression (SVR) works on a similar principal. It tries to find an appropriate line in multidimensional space that fits the data. However, it also allows for us to decide how much margin we have for loss or error in the model. This error is known as the maximum error either side of the line (epsilon ϵ) that can be configured by the user, or via Grid Search.

As an example a simple linear regression may be of the form: $y = m_i x_i + c$. SVR adds the epsilon ϵ either side of this equation to give an error margin:

$y = m_i x_i + c + \epsilon$

and

$y = m_i x_i + c - \epsilon$

While ϵ determines the margin of error around the line in multidimensional space, there are a number of other parameters that can be used to optimise the model. Standardizing the input features is one technique we can use as it is recommended that the data is scaled for use with SVRs. Also in terms of determining how to determine the line, a kernel is chosen. There are a number of different kernels which can improve the performance, depending on the data. In our case we will look into both rbf and sigmoid. We can also look to regularize the model, as we have with other models. The hyperparameter C, can be thought of as the regularization parameter, and the use of MSE minimizes it as L2 and MAE as L1 regularization. In other models we have also looked at mitigating against over-fitting by using Early Stopping. For SVRs we can also use Early Stopping. In this case we can set a tolerance for stopping criterion as we have in other models.

XGBoost

XGBoost is a modern implementation of gradient boosting. It is a distributed implementation of the gradient boosting framework - allowing for it to be used across a number of distributed processing frameworks to improve performance. While it can be used in a distributed environment, it can also be run on a single computer. Gradient boosting is a supervised learning technique based upon ensemble based techniques (Friedman, 2001). These usually use Decision Trees as their prediction models. It does this by training many models in a sequence where the next model in the sequence builds upon the previous model to build a progressively stronger model based upon a regression error. This occurs for a specified number of iterations. The final prediction of the ensemble model is then the weighted sum of the predictions from all the models in the ensemble.

XGBoost uses a max_depth parameter to prune the trees in the ensemble and enforce early stopping. Regularization is a concern as it is with the other models. Both L1 and L2 regularization are supported through the reg_alpha and reg_lambda parameters. These parameters are XGBoost specific parameters which can be adjusted to make the model more conservative. GridSearch can also be used to search through the number of estimators (or trees) to be used as well as the learning rate which should be used - this gives a weighting to new trees that are added to the model.

While it can perform well on structured datasets, it doesn't allow for the results to be explained clearly as it is a black box technique.

Neural Network

Artificial Neural Networks are a supervised learning technique which is loosely modelled on the human brain consisting of multiple nodes that are connected - mimicking the behaviour of the brain. While there are multiple architectures of neural networks we will focus on two of the more straight forward - the single layer neural network and later we will look at Multi Layer Perceptron or multi-layer neural network.

In the fully connected single layer neural network, a layer of inputs feeds into a single hidden layer, which performs computations before feeding forward into an output layer to calculate the prediction. This hidden layer is made up of a number of neurons which receive all the inputs as input and whose computation feeds forward into the output layer. A neuron receives each input which has an associated weight (random at the start). The sum of result of the product of the weights with the inputs is then used in a function to determine the output of the node. This function is known as an activation function. Some common activation functions include linear, tanh and Relu. Once the output has been calculated it is then passed to the node(s) in the output layer for the target to be calculated in a similar manner to the hidden layer.

Once the output layer calculates the target, this target value is compared to the actual target via a cost function to determine the error. In the case of our regression this will be RMSE. This error is then propagated back through the layers, adjusting the weights associated with the nodes using calculus. This pushes the error back from the output layer through the hidden layer(s) and is known as back-propagation. For every input pushed through the system, the error is calculated and pushed back through the network, resulting in a constantly changing set of weights for all the inputs to all the nodes.

The benefit of neural networks lies in the fact that they can learn patterns in the data - especially when using deeper networks with multiple layers. This can be seen in the application of neural networks to the classification of images and text using neural network architectures such as Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN). They often, however, require a lot of data to train and can take time to train.

For the single layer neural network, a number of activations will be considered - Linear and relu. Relu is one of the more popular activation functions used today, often in classification applications, while linear will deal with the data in a similar manner to a linear regression. We will standardise all the input data before passing it into the network as many of the values in the input are of different scale. Leaving the input unscaled could result in changes to larger values to cause the weights to change dramatically.

As in previous models, grid search will be used to tune the learning of the model. As mentioned previously, two activation functions will be tested as will two solvers for weight optimization - adam and sgd (stochastic gradient decent). The number of epochs used to learn will vary so that the best number of iterations can be achieved.

To this end early stopping will also be used by specifying a tolerance level of change as well as a number of iterations similar to previous models. Finally regularization will also be applied to model by specifying a regularization parameter (alpha) as well as use MSE to minimize L2.

Multi-Layer Perceptron

In the previous section single layer neural networks were discussed. Multi-layer deep neural networks are an extension of this technique. Deep learning introduces multiple hidden layers which allow for the learning of patterns in the data, allowing for the flexibility to address non-linear problems. It continues to learn through the data passed through, adding more patterns - allowing it to fit more complex functions. It is not limited to the input provided as it can discover patterns at various levels. For example, in CNNs, patterns can be discovered in images, even though the input does not provide that in it's raw form.

Again grid search will be used to find the best fit for the parameters provided. The same parameters as specified for the single layer neural network will be used here with the addition of the rmsprop optimizer. MSE will once again be used as the loss metric. However, we will also have to consider a parameter to determine the best fit for number of layers to be used.

Adding more layers may uncover more patterns in the data and give us a more accurate result, but also may tend to over fit the data. To this end, regularization will once again need to be considered. In addition to the regularization and early stopping used in previous models, multi-layer neural networks can also use the concept of dropout that was mentioned previously. The use of dropout should allow for more sparse networks, with independent neurons within the layers - mitigating for over-fitting. The dropout level can be configured via grid search.

2.5.2 Feature Extraction

In order to obtain the most relevant features when evaluating a players performance for a given position, two approaches will be taken - Sequential Feature Selection and

Permutation Importance. Each will result in a list of the most relevant features, the intersection of the two lists will then be used as the features with most relevance as they will have appeared in both approaches.

Sequential Feature Selection

Sequential Feature Selection (SFS) is an approach which attempts to reduce the dimensionality in a set of features so as to only include those features that are most relevant to the problem, thus reduce the generalization error. In the case of this dissertation three main parameters were passed in - the regressor, number of features (K) and the scoring function. The number of features tells SFS the number of most relevant features to return that fit the supplied regressor best. The means by which to evaluate the best K subset is given in the scoring function, which in this case is r2. SFS works by searching the features in the data set and collating all the subsets of size K within the data set. It then uses those subsets for the supplied regressor, evaluating each using the scoring function r2.

While in normal operation SFS is used as a feature reduction technique (Aha & Bankert, 1996), in this case it will be used as a wrapper method technique similar to that used by Nsolo, Lambrix and Carlsson (Nsolo et al., 2019). In this dissertation it is used to identify the most relevant features in evaluating a player for a particular position for the best performing approach.

Permutation Importance

Permutation Importance using the eli5[4] package is also often used for feature selection. It works slightly different to SFS in that it evaluates the importance of every feature in the data set by removing a feature from the dataset, retraining the estimator (regressor) and measuring how much the score (r2) decreases as a result. It then performs the same operation for every other feature in the data set, always evaluating the estimator with one feature missing. This can be computationally expensive for

[4] `https://eli5.readthedocs.io/en/latest/blackbox/permutation_importance.html#eli5-permutation-importance`

data sets with large numbers of features. As mentioned on the eli5 website a similar method is described by Breiman (Breiman, 2001).

As with SFS, this is often used in feature selection but is used here to identify the most relevant feature similar to the wrapper method used by Nsolo, Lambrix and Carlsson (Nsolo et al., 2019).

2.6 Summary

In this chapter, some context was presented in relation to the experiment. Background was presented in relation to the NFL and the positions under review. Research into similar problems in the NFL as well as in other sports was presented before finally discussing the methods and approaches that were researched to be used in the experiment. In the next chapter, the experiment design and methodology are discussed, presenting and analysing the data as well as the methods and error evaluation to be used.

Chapter 3

Experiment design and methodology

In this chapter, some of the approaches learned and discussed through the Literature Review chapter are used in terms of designing and describing the experiment to be undertaken. The data used in this thesis is described at the start of this chapter in section 3.1. The data sources themselves as well as the data they contain are outlined. As the data comes from multiple sources, the methods for merging these data sources are outlined for each position in section 3.1. In addition to this, some investigation is done in section 3.2 into the features on the data sets to find interesting insights as well as to eliminate duplicate and highly correlated features.

Finally the methods and technical approaches used to make the predictions and determine the most significant features are introduced and discussed in section 3.3. Error evaluation approaches are also introduced in section 3.4.

3.1 Data Collection and Preparation

3.1.1 Sources

The data for the work done in this thesis comes from two sources:

- Pro Football Reference (PFR) [1]
- Pro Football Focus (PFF) [2]

The first source - pro football reference - is operated by the Sports Reference LLc group, which operates a number of sports related statistical sites, including ice hockey, baseball, basketball and soccer. PFR provides data on both retired and active players, with data covering at least some data from seasons all the way back to pre superbowl days, including 1920. Data for the site is provided by sportradar which is the official stats partner of the NFL for current NFL seasons. Newer seasons provide additional statistics not found previously, due to the thirst for knowledge in the game by professionals and fans alike - advanced statistics on different play types such as RPO (Run Pass Option) and Play Action are only available from 2019 but give greater insight into the approach taken by modern offenses.[3]

Pro football reference contains data that covers different levels of statistics. It supplies data on a team-by-team basis as well as player-by-player data. Indeed, within the player-by-player data, statistics can be further broken down by season-by-season, game-by-game as well as some limited play-by-play data. This thesis will concern itself with season-by-season data.

The second source comes from PFF. Pro Football Focus is a company which focuses on thorough analysis of both college and professional football. It provides grades from 0-100 for players as well as creating and providing advanced statistics throughout the season. The company supplies custom data to all 32 NFL teams as well as a large number of college football teams. Their data and grades are also used by multiple media outlets and sports agencies. Some of their analysts are former NFL players and assistant coaches.

The company is known for the grades that it provides as they are based upon context and performance. As opposed to being purely quantitative assessments, PFF grades every play taking into account the circumstances as not all statistics may be

[1] `https://www.pro-football-reference.com/`
[2] `https://www.pff.com/`
[3] `https://www.pro-football-reference.com/years/2019/passing_advanced.htm`

equal. For example, if one running back makes 10 yards breaking a number of tackles, that may rank differently to a run of 10 yards with no pressure on the running back. This can lead to criticism as to how objective the grading is as well as other issues such as consistency. However, in spite of this the grades are widely used including in media broadcasts including TV.

PFF first achieved complete data in 2006. This thesis will focus on 10 years worth of data - the most recent 10 years from 2010-2019. This will be used in conjunction with the data from pro football reference to obtain a data set that we will look to predict an evaluation and figure out the most relevant features for a player to achieve that evaluation.[4]

3.1.2 Position

Quarterback

Sources For the quarterback position, data from both pro football reference and PFF will be used. Both sources have some overlapping features that will need to be addressed, but both also have their own data that is not present in the other. For example, in pro football reference it outlines whether the player made the pro bowl, all pro team, how many 4th quarter comebacks they were involved in or game winning drives that they lead. From a PFF point of view, some of the data was more detailed, particularly in relation to deep passing, throw-aways and also contains the grades from PFF analysts.

From PFR, the passing stats - those statistics which measure a QBs ability to throw a pass to a receiver - per year are used. While there are advanced passing stats, these are only applicable to the last 3 years. The passing stats are those most often used when talking about QBs. In addition, once the data has been collected it has been filtered down to just those players who are QBs. There are times where trick plays are used and a tight-end or running back may throw a play, but they account for a very small minority of passes made.

[4] `https://www.pff.com/grades`

Within PFF there are a lot of different sources of data per player, but for this thesis three sources of data are used. There are two sources for passing - passing with grades and deep passing. This will allow us to look at the grades of the player, how they handle pressure as well as deep throwing statistics to show us how often they attempt long passes. In addition to the passing statistics, rushing statistics are also considered. In recent years, more mobile quarterbacks have entered the league such as Deshaun Watson, Lamar Jackson and Patrick Mahomes. Their ability to use the running game has transformed how offensive plays are constructed. As a result, rushing statistics will also be included to observe these newer attributes of a quarterback. Data files were downloaded as CSV files from the websites.

The target variable for the thesis is the offensive grade as given by PFF. This is a ranking from 0-100 as judged by the analysts employed by PFF. There are 600+ analysts used by PFF, including many former players and coaches with 10% of analysts trained so that they can grade plays.[5] They use a mix of statistical data as well as expert knowledge to grade plays within a game, such as grading an incompletion slightly differently if it is as a result of catcher error as opposed to a bad throw by the QB.

While the data comes from multiple sources (PFR and PFF), each source provides records which represent the data for one player in a given year. This data is compiled and provided on their websites, collated for the year and no additional data wrangling was needed. Only records which match up in both sources is included for analysis and use within the models. All features are positive numerical features (either integer or real numbered) - with only the grades and percentage features having a bound of 100. All other features have no enforced upper bound.

Merge Methods and Justification Given that there are 4 data sets, across 2 different sources, merging of the data sets is necessary. The four data sets are as follows: PFR Passing, PFF Passing, PFF Deep Passing and PFF Rushing. Each data set is only concerned with QB players. The data sets are merged per year and then

[5] `https://www.pff.com/grades`

the resulting data is appended together.

PFR Passing encompasses statistics related to the QBs performance when throwing the ball such as completions, yards per attempt, passing touchdowns, etc. PFF Passing also encompasses measurements of the QB throwing ability over the year adding in additional features such as the grades, throw aways (balls thrown away under pressure so as not to lose yards on a play) and bats (balls batted down by the defense). The PFF Deep Passing statistics also represent throwing statistics, but only in relation to deep passes (20+ yards). Completed deep passes are of a higher risk as the ball is in the air for longer and accuracy is key. These can have a big impact on a game and require a higher level of skill. Finally, the PFF Rushing statistics were included as QBs run more in the current game. These statistics measure how often they run with the ball and how effective they are when they do run.

First, PFF Passing and PFF Deep Passing are merged. The two data sets have overlapping data. As the Deep Passing data set pertains mainly to deep passing statistics, only the player ID and those fields with 'deep' in their name are kept from the deep passing data source. Every player in PFF is given a unique ID. As a result, the two data sets are merged on the players unique PFF ID. Each subset of data is for a single year and so the records for each PFF ID should be unique - the year will be kept as part of the resulting mered record.

Next, the PFF Rushing data is added to the merged data set. This data set contains a lot of features, but from the QB point of view only the following features are considered as designed rushing plays for QBs are still not as common: player_id, run_plays, attempts, yards, ypa, touchdowns and avoided_tackles. Once again the players unique PFF ID can be used to merge the data sets together.

Finally, the PFR Passing data is incorporated into the final data set. In contrast to the previously merged PFF data sets, there is no unique ID on which to join. As a result, another feature needs to be used as the joining feature. No two quarterbacks have had the same name in the same year, so it was taken that the player name was the unique feature on which to join. Care had to be taken to make sure that the names in the two data sources lined up. In the PFR data, the name can have additional characters that are added to signify other features. For example, '*' is used to signify whether the player made the Pro Bowl (an accolade given to the best players in a given year), '+' is used to signify whether the player made the All Pro team (another less prestigious accolade given to top players at the end of the year). Both of these were removed from the player's name, but used to derive the corresponding features. Also postfixes such as 'II' (the second) are also removed. Just the first and last names are used.

Features Tables 3.1 and 3.2 below outline the final set of features obtained from the sources outlined above. Once the complete set of records have been collated, the PFF Id is no longer required as each record is unique based upon year and name. It could also influence the learning and so was removed.

Table 3.1: List of QB features - PFR

Column	Description
player	Player Name
Tm	Team Played For
Age	Age in the given year
GS	Number of Games Started
TD%	Percentage of passes thrown that were TDs
Int%	Percentage of passes that were Intercepted
1D	Number of 1st Downs passed
Lng	Longest completed pass thrown
AYA	Adjusted Yards gained per Attempt
YC	Average Passing Yards per completed passing catch
YG	Average Passing Yards per game
Yds.1	Yards lost due to Sacks
NY/A	Net Yards gained per Attempt
ANY/A	Adjusted Net Yards gained per Attempt
Sk%	Sack Percentage
4QC	4th Quarter Comebacks led by QB
GWD	Game Winning Drives led by QB
Year	Year observed
proBowl	Made Pro Bowl?
allPro	Made All Pro Team?

Table 3.2: List of QB features - PFF

Column	Description
player_game_count	Number of Games played
dropbacks	Number of times QB dropped back to pass
attempts_passing	Number of passing attempts
completions	Number of completed passes
completion_percent	Completion Percentage
yards_passing	Yards from passing
ypa_passing	Number of yards gained per attempt
touchdowns_passing	Number of touchdowns
interceptions	Number of Interceptions
grades_offense	PFF Offensive Grade
grades_pass	PFF Passing Grade
grades_run	PFF Running Grade
grades_hands_fumble	PFF Hands Fumble Grade
sacks	Number of sacks
bats	Batted Passes
drops	Drops by Receiver
thrown_aways	Number of times intentionally threw away
hit_as_threw	Number of times hit as thrown
qb_rating	NFL Passer Rating
scrambles	Number of undesigned runs by QB
first_downs	Number of first downs
deep_attempts	Number of deep passing attempts
deep_completions	Number of deep passing completions
deep_drops	Number of deep passing drops
deep_yards	Number of deep passing yards
deep_touchdowns	Number of deep passing TDs
deep_interceptions	Number of deep passing interceptions
deep_attempt_percent	Deep Passing Attempt Percentage
deep_accuracy_percent	Deep Passing Accuracy Percentage
run_plays	Number of run plays
attempts_rushing	Number of designed rushing attempts
yards_rushing	Number of rushing yards
ypa_rushing	Number of rushing yards per attempt
touchdowns_rushing	Number of rushing TDs
avoided_tackles	Number of avoided tackles
wins	Number of wins
losses	Number of losses
draws	Number of draws

Whilst the majority of the features outlined in Tables 3.1 and 3.2 above are exactly as prescribed in the online sources, there were a number of features that needed to be derived from the sourced data.

The proBowl and allPro features were derived from the PFR data as it was provided via special characters on the name. To that end, these special characters were also removed from the name feature. This was achieved by parsing the name text for the name as well as the special characters which denoted that the player was omitted to the pro bowl and/or the all pro team. The proBowl and allPro features are Boolean features set to True when the special character in question is present in the name of the player provided by PFR - otherwise it is false.

The Year feature was added based upon the year in question that we were adding to the data set. Every year of data was one data set and these needed to be combined together to construct the final data set. As each player has a unique id and name, these could not be used to identify a particular seasons set of data. As a result the Year feature was added per row.

In PFR, there is a feature which indicates the record of the QB in a particular season, with one year containing one season. It does this in the following format: Wins-Losses-Draws. Each piece of information in that feature was extracted out into it's own feature and the original feature discarded.

There were two features in the data that contained null values: 4QC and GWD.

4QC indicates the number of 4th quarter comebacks lead by the QB. This means that the QBs team was trailing going into the 4th quarter of the game and the QB was involved in leading the team to a comeback victory. There are 27 null values across 311 entries. Given that the numbers are very small and indicate an actual example of the feature occurring, it was decided to substitute 0 for the null values - no credit given for any 4th quarter comebacks.

GWD indicates the number of game winning drives (including overtime) lead by the QB. This means that the team were trailing in the 4th quarter of the game (or overtime) and that the QB lead the team on a drive (set of plays) that resulted in the game winning score. There are 27 null values across 311 entries. Given that the numbers are very small and an event such as a game winning drive is such a notable occurrence that it would be recorded it was decided to substitute 0 for the null values - no credit given for any game winning drives.

Authenticity There were a number of features that were dropped after the data sets were merged, as they were equivalent features. Where duplicate features existed, those provided by PFF were chosen as they were the numbers taken into account by the PFF analysts when grading the players. The differences between the results often only involved a small subset of records, which varied by a small margin which will be outlined below.

The following features were dropped: Rk (Rank - PFR rank not relevant), Pos and Position as all records are QBs, QBR is an ESPN evaluation.

'Rate' is dropped in favour of 'qb_rating'. These are the NFL passer ratings. 39 of 311 were different with a difference of either 0.1 or 0.2, for example 87.4 vs 87.5.

'G' is dropped in favour of player_game_count. It is the number of games played by the QB. 5 of 311 were different with the majority a difference of 1, for example 10 vs 9.

'Cmp' is dropped in favour of 'completions'. This is the number of completions passing by the QB. 8 of 311 were different with the majority a difference of 1, for example 353 vs 354.

'Att' is dropped in favour of 'attempts_passing'. These are the number of passing attempts. 24 of 311 were different with the majority a difference of 1, for example 569 vs 570.

'Cmp%' is dropped in favour of 'completion_percent'. This the completion percentage passing for he QB. 24 of 311 were different with the majority a difference of 0.1, for example 64.2 vs 64.1.

'Yds' is dropped in favour of 'yards_passing'. This is the total number of yards thrown and is the largest difference with 47 of 311 different with the largest range from 1 to 12 i.e. 4054 vs 4055.

'TD' is dropped in favour of 'touchdowns_passing'. This is the number of touchdowns thrown by the QB. 3 of 311 were different, all with a difference of 1, for example

20 vs 19.

'Int' is dropped in favour of 'interceptions'. This is the total number of interceptions thrown by the QB. 4 of 11 were different, all with a difference of 1, for example 23 vs 24.

'Y/A' is dropped in favour of 'ypa_passing'. This is the average number of yards per attempt. 8 of 311 were different with differences of 0.1, for example 8.1 vs 8.2.

'Sk' is dropped in favour of 'sacks'. This is the number of sacks of the QB. 31 of 311 are different with a difference of 1, for example 46 vs 45.

As well as dropping duplicate features, there were a number of features that were also dropped because they were primarily not quantitative - in this case the other ratings apart from the grades_offense. In addition to that grade, there were three other grades: grades_pass, grades_rush and grades_hands_fumble. These are all grades from the PFF analysts, and in particular the grades_pass has a very high correlation (0.98) with grades_offense. By including that grade, it would match it very closely and mask the other features. These were removed as a result.

Wide Receiver

Sources As with the quarterback position, data from both PFR and PFF will be used. There are some overlapping features between the two data sources that will need to be looked at. However, both bring value as they add additional data unique to that source. For example, in pro football reference it outlines whether the player made the pro bowl or made the all pro team. PFF also adds some more advanced statistics including contested catch rate, yards after catch and average depth of target.

From PFR, the basic receiving stats were used. While there are advanced receiving stats such as broken tackles and QB passer rating targeting the wide receiver, these are only applicable to the last 3 years and so weren't considered. Receiving statistics aren't specific to just the wide receiver position - indeed it covers the entire offensive position group. As a result the results had to be filtered down to just those players who are classified as wide receivers. There are times where trick plays may occur and the wide receiver may act as a running back, but they account for a very small minority of the runs played and so were not considered for this evaluation.

Within PFF there are a number of different data sources applicable to receiving that cover a number of different categories such as statistics of the player against different defensive strategies or within particular offensive strategies. However,for the purposes of this evaluation just the data source that provides the grades and most of the receiving stats is used. By using all the other data sources, there is a possibility that the model might overfit to the data given the number of records per year - approximately 220 players per year. Some of the other data sources are summarized within the main data source which provides approximately 40 features.

As with the QB position, the target variable for the thesis for wide receivers is the offensive grade as given by PFF. These grades are achieved using the same process as with the QB position, using multiple analysts who are often ex-players and coaches to evaluate players on a play-by-play basis.

As with the QB position, while the data comes from multiple sources (PFR and PFF), each source provides records which represent the data for one player in a given year. This data is compiled and provided on their webites collated for the year and no additional data wrangling was needed. Only records which match up in both sources is included for analysis and use within the models. All features are positive numerical features (either integer or real numbered) - with only the grades and percentage features having a bound of 100. All other features have no enforced upper bound.

Merge Methods and Justification While the QB position used 4 data sets, across 2 different sources, the WR position is using just 2 data sets from those same 2 data sources: PFR Receiving and PFF Receiving. Each data set is concerned only with WR players and any other positions are filtered out. The data sets are merged per year and then the resulting data is appended together.

PFR Receiving encompasses basic receiving statistics related to a wide receiver such as Yards per game, receptions per game, total yards for the year, total receptions per year, targets per year, etc. PFF Receiving also encompasses measurements of the WRs receiving ability over the year adding in additional features such as the grades, avoided tackles, targeted QB rating, yards achieved after a catch, etc.

The two data sets are merged, but there is no common unique ID on which to join. PFF has a unique ID but this is specific to PFF. As a result, another feature needs to be used as the joining feature. No two wide receivers have had the same name in the same year, so it was taken that the player name was the unique feature on which to join. Care had to be taken to make sure that the names in the two data sources lined up. In the PFR data, the name can have additional characters that are added to signify other features. For example, '*' is used to signify whether the player made the Pro Bowl (an accolade given to the best players in a given year), '+' is used to signify whether the player made the All Pro team (another less prestigious accolade given to top players at the end of the year). Both of these were removed from the player's name, but used to derive the corresponding features. Also postfixes such as 'II' (the second) are also removed. Just the first and last names are used.

Features Tables 3.3 and 3.4 below outline the final set of features obtained from the sources outlined above. Once the complete set of records have been collated, the PFF Id is no longer required as each record is unique based upon year and name. It could also influence the learning and so was removed.

Table 3.3: List of WR features - PFR

Column	Description
player	Player Name
Tm	Team Played For
Age	Age in the given year
GS	Number of Games Started
Tgt	Number of Pass Targets
Rec	Number of Receptions
Yds	Receiving Yards
Y/R	Receiving Yards per Reception
TD	Number of Receiving Touchdowns
1D	Number of First Downs Receiving
Lng	Longest reception in yards
Y/Tgt	Number of Receiving Yards per Target
R/G	Number of Receptions per Game
Y/G	Number of Receiving Yards per Game
Fmb	Number of Fumbles
proBowl	Did player make Pro Bowl?
allPro	Did player make All Pro Team?

Table 3.4: List of WR features - PFF

Column	Description
Year	Year Observed
player_game_count	Number of Games Played
targets	Number of times Targeted
receptions	Number of receptions
caught_percent	Number of passes caught
yards	Number of Receiving Yards
touchdowns	Number of receiving Touchdowns
grades_offense	PFF Offensive Grade
grades_pass_route	PFF Passing Routes/Receiving Grade
grades_hands_drop	PFF Hands Drop Grade
grades_hands_fumble	PFF Hands Fumble Grade
grades_pass_block	PFF Pass Block Grade
yards_per_reception	Number of yards per reception
yards_after_catch	Number of yards after catch
yards_after_catch_per_reception	Number of yards after catch per reception
longest	Longest Reception in yards
first_downs	Number of First Downs Receiving
drops	Number of on target passes dropped
interceptions	Number of Receiving interceptions
fumbles	Number of fumbles
avoided_tackles	Missed tackles forced after a reception
targeted_qb_rating	NFL Passer Rating when targeted
penalties	Number of penalties
declined_penalties	Number of declined penalties

Whilst the majority of the features outlined in Tables 3.3 and 3.4 above are exactly as prescribed in the online sources, there were a number of features that needed to be derived from the sourced data.

The proBowl and allPro features were derived from the PFR data as it was provided via special characters on the name. To that end, these special characters were also removed from the name feature. This was achieved by parsing the name text for the name as well as the special characters which denoted that the player was omitted to the pro bowl and/or the all pro team. The proBowl and allPro features are Boolean features set to True when the special character in question is present in the name of the player provided by PFR - otherwise it is false.

The Year feature was added based upon the year in question that we were adding to the data set. Every year of data was one data set and these needed to be combined together to construct the final data set. As each player has a unique id and name, these could not be used to identify a particular seasons set of data. As a result the Year feature was added per row.

Authenticity There were a number of features that were dropped after the data sets were merged, as they were equivalent features. Where duplicate features existed, those provided by PFF were chosen as they were the numbers taken into account by the PFF analysts when grading the players. The differences between the results often only involved a small subset of records, which varied by a small margin which will be outlined below.

The following features were dropped: Rk (Rank - PFR rank not relevant), Pos and Position as all records are WRs.

'Yds' is dropped in favour of 'yards'. This is the total number of yards achieved by the wide receiver. 74 of 1877 were different with the majority a difference of 3 or 4 yards.

'TD' is dropped in favour of 'touchdowns'. This is the total number of touchdowns achieved by the wide receiver. 15 0f 1877 were different but there was some difference between those that differed for example 7 vs 1.

As well as dropping duplicate features, there were a number of features that were also dropped because they were primarily not quantitative - in this case the other ratings apart from the grades_offense. In addition to that grade, there were four other grades: grades_pass_route, grades_hands_drop, grades_pass_block and grades_hands_fumble. These are all grades from the PFF analysts, and in particular the grades_pass_route has a very high correlation (0.98) with grades_offense. By including that grade, it would match it very closely and mask the other features. These were removed as a result.

As well as dropping some features that are effectively the same features, there were a number of features that had null values. Both 'Tgt' and 'Y/Tgt' had null values for 2 records. Both of these records had null values for both features. As a result the records that had those null values were removed from the data set.

Running Back

Sources As with the previous QB and WR positions, data from both PFR and PFF will be used. Both sources have some overlapping features that will need to be addressed, but both also have their own data that is not present in the other. For example, PFR outlines whether a player was selected for Pro Bowl or All Star team honours. From a PFF point of view, it is able to provide insight into both the running and passing games allowing for some additional metrics.

From PFR, basic rushing (running) statistics are provided such as longest run, yards per attempt rushing and yards per game rushing. A lot of these can also be found in the PFF statistics, although features such as rushing yards per game and selection for Pro Bowl and All Star team honours are unique to this data set. While there are advanced rushing stats such as broken tackles and yards before contact targeting the running back, these are only applicable to the last 3 years and so weren't considered. Rushing statistics aren't specific to just the running back position - indeed it covers the entire offensive position group. As a result the results had to be filtered down to just those players who are classified as running backs.

Using the rushing statistics, PFF can provide information in relation to yards after contact when rushing as well as avoiding tackles as a rusher (runner). Using the same passing statistics as the WR position, PFF also allows for receiving statistics to be used. As a result there are two different data sources used from PFF. Traditionally running backs would be handed the ball by the QB and they would attempt to run as far towards the opponents end zone as possible before being tackled. However, in recent years there have been additional tactics used with running backs where they are more and more asked to catch passes as well as their rushing duties. These passes aren't long passes but short passes designed to find more space for the running backs to use their running skills in.

As with the QB and WR positions, the target variable for the thesis for wide receivers is the offensive grade as given by PFF. These grades are achieved using the same process as with the QB and WR positions, using multiple analysts who are often ex-players and coaches to evaluate players on a play-by-play basis.

As with the QB and WR positions, while the data comes from multiple sources (PFR and PFF), each source provides records which represent the data for one player in a given year. This data is compiled and provided on their webites collated for the year and no additional data wrangling was needed. Only records which match up in both sources is included for analysis and use within the models. All features are positive numerical features (either integer or real numbered) - with only the grades and percentage features having a bound of 100. All other features have no enforced upper bound.

Merge Methods and Justification Given that there are 3 data sets, across 2 different sources, merging of the data sets is necessary. The three data sets are as follows: PFR Rushing, PFF Rushing and PFF Receiving. Each data set is filtered so that it is only concerned with RB players. The data sets are merged per year and then the resulting data is appended together.

PFR Rushing encompasses basic statistics related to a RBs performance when rushing the ball such as yards made rushing, yards per game, yards per attempt and fumbles. PFF Rushing also incorporates additional features such as yards after contact and avoided tackles. These metrics indicate how productive the RB can be and how much they can make plays, get extra yards for the team and enhance the production of the team. PFF Receiving details are the same as those for the WR position. This allows for analysis into the productivity of the RB in a passing game, allowing for more options to the team. It shows how versatile they can be and shows an additional skill set.

First, PFF Rushing and PFF Receiving are merged. The two data sets have overlapping data. Where features with the same name are found in both data sets, the rushing features will be appended with _rushing and the receiving features will be appended with _receiving. Every player in PFF is given a unique ID. As a result, the two data sets are merged on the players unique PFF ID.

Finally, the PFR Passing data is incorporated into the final data set. In contrast to the previously merged PFF data sets, there is no unique ID on which to join. As a result, another feature needs to be used as the joining feature. No two running backs have had the same name in the same year, so it was taken that the player name was the unique feature on which to join. Care had to be taken to make sure that the names in the two data sources lined up. In the PFR data, the name can have additional characters that are added to signify other features. For example, '*' is used to signify whether the player made the Pro Bowl (an accolade given to the best players in a given year), '+' is used to signify whether the player made the All Pro team (another less prestigious accolade given to top players at the end of the year). Both of these were removed from the player's name, but used to derive the corresponding features. Also postfixes such as 'II' (the second) are also removed. Just the first and last names are used.

Features Tables 3.5 and 3.6 below outline the final set of features obtained from the sources outlined above. Once the complete set of records have been collated, the PFF Id is no longer required as each record is unique based upon year and name. It could also influence the learning and so was removed.

Table 3.5: List of RB features - PFR

Column	Description
player	Player Name
Tm	Team Played For
Age	Age of player
G	Number of Games played
GS	Number of Games started
Y/A	Yards per attempt rushing
Y/G	Yards per game rushing
Fmb	Number of rushing fumbles
Year	Year observed
proBowl	Made Pro Bowl?
allPro	Made All Pro Team?

Table 3.6: List of RB features - PFF

Column	Description
player_game_count_rushing	Number of games played
yards_rushing	Number of yards rushed
ypa	Yards per Attempt
touchdowns_rushing	Number of rushing touchdowns
grades_offense	PFF Offensive Grade
grades_run	PFF Running Grade
grades_hands_fumble_rushing	PFF Rushing Fumble Grade
yards_after_contact	Yards after contact
yco_attempt	Number of Attempts with yards after contact
longest_rushing	Longest rushing yards
avoided_tackles_rushing	Number of tackles avoided while rushing
fumbles_rushing	Number of fumbles while rushing
penalties_rushing	Number of Penalties on rushing plays
declined_penalties_rushing	Number of Declined Penalties on rushing plays
caught_percent	Percentage of passes caught
yards_receiving	Number of yards receiving
touchdowns_receiving	Number of touchdowns receiving
grades_offense_receiving	PFF Offensive Grade
grades_pass_route	PFF Receiving Grade
grades_hands_fumble_receiving	PFF Grade Receiving Fumbles
yards_per_reception	Number of yards per reception
yards_after_catch	Number of yards after catch
yards_after_catch_per_reception	Number of yards after catch per reception
longest_receiving	Longest reception in yards
first_downs_receiving	Number of first downs receiving
drops	Number of drops receiving
interceptions	Number of interceptions on passes thrown to
fumbles_receiving	Number of fumbles while receiving
avoided_tackles_receiving	Number of tackles avoided after receiving a pass
targeted_qb_rating	QB Rating of QB throwing to RB
penalties_receiving	Number of Penalties on receiving plays
declined_penalties_receiving	Number of Declined Penalties on receiving plays

Whilst the majority of the features outlined in Tables 3.5 and 3.6 above are exactly as prescribed in the online sources, there were a number of features that needed to be derived from the sourced data.

The proBowl and allPro features were derived from the PFR data as it was provided via special characters on the name. To that end, these special characters were also removed from the name feature. This was achieved by parsing the name text for the name as well as the special characters which denoted that the player was omitted to the pro bowl and/or the all pro team. The proBowl and allPro features are Boolean features set to True when the special character in question is present in the name of the player provided by PFR - otherwise it is false.

The Year feature was added based upon the year in question that we were adding to the data set. Every year of data was one data set and these needed to be combined together to construct the final data set. As each player has a unique id and name, these could not be used to identify a particular seasons set of data. As a result the Year feature was added per row.

Authenticity There were a number of features that were dropped after the data sets were merged, as they were equivalent features. Where duplicate features existed, those provided by PFF were chosen as they were the numbers taken into account by the PFF analysts when grading the players. The differences between the results often only involved a small subset of records, which varied by a small margin which will be outlined below.

The following features were dropped: Rk (Rank - PFF Rank not relevant), Pos and position_rushing and position_receiving as all records are RBs, team_name_rushing and team_name_receiving, player_id as this is a PFF specific ID.

'TD' is dropped in favour of 'touchdowns_rushing'. This is the total number of touchdowns achieved by the running back while rushing. 8 0f 1266 records were different but there was small differences between those that differed for example 6 vs 7.

'Lng' is dropped in favour of 'longest_rushing'. This is the longest run in yards that the RB achieved while rushing. 10 of 1141 records were different but there was some difference between the two.

As well as dropping duplicate features, there were a number of features that were also dropped because they were primarily not quantitative - in this case the other ratings apart from the grades_offense. In addition to that grade, there were eight other grades: grades_run, grades_hands_fumble_rushing, grades_run_block, grades_offense_receiving, grades_pass_route, grades_hands_drop, grades_hands_fumble_receiving and grades_pass_block. These are all grades from the PFF analysts, and in particular the grades_pass_route has a very high correlation (0.98) with grades_offense. By including that grade, it would match it very closely and mask the other features. These were removed as a result.

As well as dropping some features that are effectively the same features, there were a number of features that had null values. There were a considerable number of features that had null values in the compiled data set. Some of these were PFF grades and so they were removed from the data set. These features were based on grades given by PFF and so are subject to their grading and evaluations. There were two other scenarios where null values occurred. The first is where there was no receiving records for the RB. In this case all the PFF receiving values were null. As a result these records were removed from the data set. The second is where the RB had null values for yards per reception and yards after the catch per reception. A number of records had 0 for both, so any records with null values for this scenario were also removed.

3.2 Data Analysis

3.2.1 Data Types

The data collected in the NFL predominantly consists of numeric data - integers and real values. These values are often used to compare player and team performance. Within the evaluation process of players in the NFL and in the draft where players can select players coming out of college, numbers are very important and small margins are often used when negotiating contracts with existing players or in selecting players in the draft. For example, in the draft the 40 yard dash is one of the most talked about metrics and a few hundredths of a second difference can have big repercussions on a players evaluation. Other tests include broad jump length, number of bench press reps, physical measurements, etc.

In the data sets obtained, the majority of the data is unbounded. Whilst there is the possibility that a negative value can exist, the majority of values have a lower limit of zero, with no upper bound. However, there are a number of features that are included that are bounded. These come in two varieties. The grades presented in the data are bounded from 0-100. We will use the offensive grade as our target variable and remove the other grades for the models as they are from analyst evaluation and in the case of passing grade, would correlate very closely to our target feature. In the case of passing grade, the pearson correlation is 0.984449. The other features that are bounded are those that deal in percentages. Some of the features provide an absolute number, but these may tell just one side of the story, where the percentage may provide a different angle. For example, a QB may complete a high number of passes, but if he also attempts a high number of passes, then he may not be as efficient as another QB who has less pass completions but they are of a higher percentage completion.

There are two features which are categorical: All Pro and Pro Bowl. Both of these features are accolades given in a particular year in recognition of good play by players. The All Pro is voted by the press and declares that player the best in their position that year. The Pro Bowl is voted by coaches, players and fans and signifies the best players in each conference of the NFL of which there are two. The players represent their conference in a game at the end of the season and recognises players as being the best in their position within their conference.

For the most part, the majority of features follow a normal distribution. The distributions can be seen in Appendix A.

For QBs, we can see the feature distributions in Appendix A.1. If we remove the grades, what we can see is that the majority of the features are similar to a normal distribution. Some of the features are skewed, such as GS (Games started) and player game count and is something that we would expect as QBs are normally used in every offensive play. Also dropbacks and scrambles are skewed as we would expect in the last 10 years that QBs would drop back to pass more often, while scrambles are not something that is designed (in comparison to QB Runs). Some of the other skewed variables are expected and are usually on features that contain lower numbers such as running touchdowns (touchdowns_y) as QBs traditionally would not score a lot of running touchdowns.

For WRs, we can see the feature distributions in Appendix A.2. Unlike with QBs, when the grades are removed, the majority of the remaining features do not fit into normal distributions. Some features such as caught percentage, yards per reception, longest reception and targeted QB rating follow a normal distribution showing that per reception, values follow a normal distribution. However, as can be seen from the feature distribution for 'receptions', not all WRs get targeted as often. It is heavily skewed to the left and shows that there are quite a lot of WRs who get few receptions, and this is the majority. When it comes to throwing a ball to a WR, there are a few WRs who stand above the others. These WRs are trusted to get free of their defenders, make the catch (even contested catches) and make yards after the catch. This fact results in a lot of the other features also being skewed such as receptions per game, yards per game, yards, yards after catch.

For RBs, we can see the feature distributions in Appendix A.3. Unlike with QBs, when the grades are removed, the majority of the remaining features do not fit into normal distributions. Some features such as yards per attempt, yards after contact per attempt and targeted QB rating follow a normal distribution showing that per reception, values follow a normal distribution.

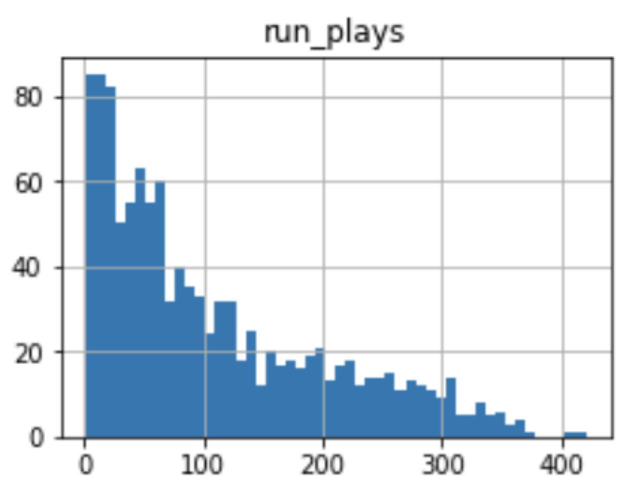


Figure 3.1: RB Distribution - Number of Run Plays

Looking at the distribution for number of run plays in Figure 3.1 above it can be seen that not all RBs are used equally. It is heavily skewed to the left and shows that there are quite a lot of RB who get few runs, and this is the majority. However, there are a few who are trusted to make plays such as avoiding, breaking through tackles and making yards after contact over the course of the season. This results in a number of other features also being skewed such as yards after contact, yards rushing and touchdowns rushing. The same can be seen for the receiving features - much like with the WR position.

3.2.2 Data Grouping

Within the data obtained from the two sources, some features can be thought of as related. In relation to QBs features, from the Heatmap generated in Figure 3.3 below we can see that there are a number of features that are closely aligned. Positive Deep Passing features correlate closely with each other as do QB designed running plays. This is to be expected.

Number of games played by a QB, Dropbacks, Passing Attempts, Passing Completions, First Downs and Passing Yards are all closely correlated. Number of games played by a QB is not closely correlated to runs by QBs showing that QBs main concern in the last 10 years has been to pass the ball to the teams offensive playmakers, as opposed to making yards through their own running. Later it will be shown that this has begun to change with the evolution of newer, younger QBs entering the league.

3.2.3 Correlations

Quarterbacks

In Figure 3.2 below any feature with _x is _passing and anything _y is _rushing.

grades_offense	1.000000
grades_pass	0.984449
ANY/A	0.799381
qb_rating	0.789406
AY/A	0.766944
NY/A	0.734192
touchdowns_x	0.694695
ypa_x	0.692252
completion_percent	0.677155
TD%	0.649679
Y/G	0.642582
wins	0.636260
yards_x	0.621635
first_downs	0.616693
1D	0.616168
deep_yards	0.582607
deep_completions	0.580672
proBowl	0.566526
deep_touchdowns	0.545929
completions	0.542906
deep_accuracy_percent	0.537586
attempts_x	0.427953
GS	0.422527
dropbacks	0.417024
deep_attempts	0.402634
run_plays	0.392115
player_game_count	0.389682
Age	0.334477

(a) QB Correlation Numbers Top

GWD	0.316852
grades_hands_fumble	0.291388
allPro	0.288502
4QC	0.277458
Y/C	0.276317
Lng	0.241471
deep_drops	0.234085
drops	0.202779
thrown_aways	0.177866
grades_run	0.163124
Year	0.126754
attempts_y	0.120946
deep_attempt_percent	0.066800
touchdowns_y	0.064560
draws	0.037826
scrambles	0.035359
yards_y	0.021475
avoided_tackles	0.013419
sacks	0.004885
deep_interceptions	0.000785
Yds.1	-0.000492
hit_as_threw	-0.011803
bats	-0.087321
ypa_y	-0.125560
interceptions	-0.177880
Sk%	-0.328019
losses	-0.389745
Int%	-0.493248

(b) QB Correlation Numbers Bottom

Figure 3.2: QB Correlation Numbers against grades_offense

The correlations are measured against the target feature "grades_offense".As can be seen from the correlation numbers in Figures 3.2a and 3.2b shown above, the passing grade and the target feature offensive grade are very closely correlated and so we will remove this grade from the model, given that it is also given from analysts. From a positive correlation point of view that any of the metrics that cover yards per attempt correlate highly, as does the completion percentage. This is interesting as it shows that despite the fact that a QB may complete a lot of passes, it is the ability to consistently complete a pass when thrown that is more important. Touchdowns are always important in the sport and it is no surprise that it is one of the top correlated features. What these features indicate is that the ability to correctly choose the correct target and be able to execute a complete pass to that target, are important skills for the QB.

From the negative correlation point of view, we can see that percentages are once again more relevant than the count of instances. Losses will always count against a

QB as they are the leader and considered the player with most influence on a team. The interception percentage show that consistently poor reads and throws that result in interceptions are more relevant than the raw numbers, although the raw numbers are also relevant as an interception can at best result in ending the offensive drive and giving the opponent an opportunity to attack, and worst case it can result in the opponent running in for a touchdown directly off that interception. The sack percentage is interesting as the raw sack numbers actually correlate positively with the target variable, whereas the percentage is negatively correlated. It is the percentage of times that the QB has received the ball and been sacked. This shows that the prevalence of a QB to throw an interception consistently is a bigger issue.

The correlation plots for the Quarterback position can be found in Appendix B.1. Plotting a basic scatterplot for each feature against the target feature, we can see their corresponding correlation strength. What is interesting to see is that there are some features that appear to be more scattered apart than others. Some features such as AY/A (Adjusted Yards per Attempt) appear to fit the line a lot more closely than others such as Y/C (Yards per catch). This indicates a greater variance in performance for QBs for the features like Y/C and that features such as AY/A may be a more reliable indicator of performance. Other such features include ANY/A (Adjusted Net Yards per Attempt), completion percent, passing touchdowns and of course the passing grade.

Correlation heat map show the correlation between features. Due to the number of features for the QB position, it is difficult to show the entire correlation heat map. However, the heat maps provided below (3.3, 3.4 and 3.5) show some insights into some highly correlated features. As a result a number of features were removed from the data set as they correlated very closely to other features.

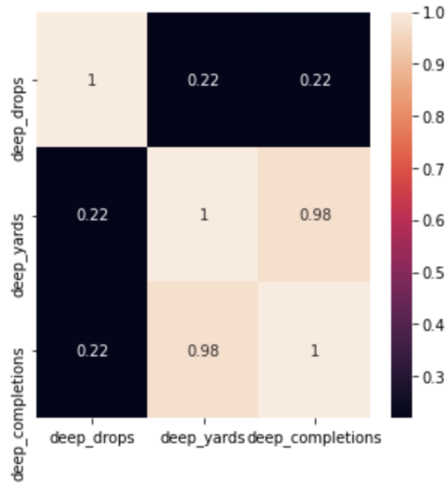


Figure 3.3: QB Correlation Numbers - Deep Passing

As can be seen in Figure 3.3 above, the features deep_yards and deep_completions are highly correlated at 0.98. This makes sense as in order to achieve yards from deeps passes, then the pass needs to be completed. As a result, the deep_completions feature was removed from the data set as the yards tells us more information than the number of completions.

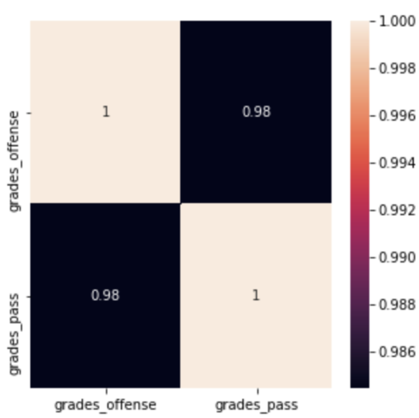


Figure 3.4: QB Correlation Numbers - Grades

As can be seen in Figure 3.4 above, the features grades_offense and grades_pass are highly correlated at 0.98. It shows the high correlation over the last 10 years between how high a QB is graded and how good a passer that QB is. This has been consistent over the 10 years, but in the section looking at correlation trends over time, it will be shown that the ability for a QB to also possess a running game is becoming more and more important. In this instance, the grades_pass feature will be removed as the grades_offense is our target feature and including the grades_pass feature would overly skew the predictions.

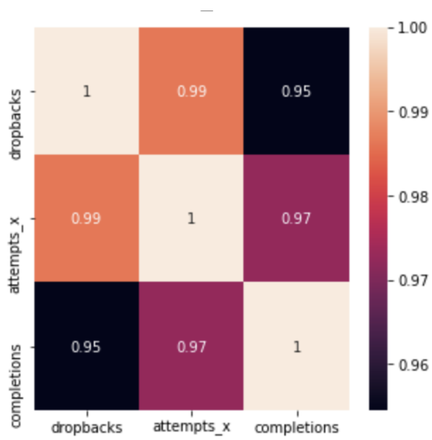


Figure 3.5: QB Correlation Numbers - Completions

As can be seen in Figure 3.5 above, there are a number of features which are highly correlated. The features dropbacks and attempts_passing are correlated at 0.99, attempts_x and completions are correlated at 0.97 and dropbacks and completions are correlated at 0.95. All three are highly correlated to each other. As a result, only one will be kept. Whenever a QB drops back there is a high correlation that they will attempt a pass. What is more interesting is how many completions were made and it is for this reason that the completions feature is kept above the other two.

Wide Receivers

grades_offense	1.000000
Y/G	0.749501
first_downs	0.720248
yards	0.715688
R/G	0.696039
receptions	0.679659
yards_after_catch	0.653631
touchdowns	0.630266
avoided_tackles	0.550331
targeted_qb_rating	0.524162
GS	0.514323
longest	0.511996
Y/Tgt	0.452388
proBowl	0.432753
grades_hands_drop	0.424441
interceptions	0.373096
caught_percent	0.327252
drops	0.315055
player_game_count	0.310103
penalties	0.292824
yards_per_reception	0.262590
Y/R	0.259573
G	0.251428
fumbles	0.239839
allPro	0.199658
Fmb	0.141048
yards_after_catch_per_reception	0.135225
declined_penalties	0.120006
Age	0.096577
grades_hands_fumble	0.040238
Year	-0.060361

Figure 3.6: WR Correlation Numbers against grades_offense

The correlations are measured against the target feature "grades_offense". There are a number of PFF grades in the data set that are compiled from the analysts at PFF. These features will be removed from the data set before progressing with the machine learning. From a positive correlation point of view, a pattern can be seen. The most highly correlated features are those that represent the values collated over the course of the season or per game as opposed to the numbers per reception. What this shows is that consistent performances over a prolonged period are more valued than per reception averages. The reason that this may occur is that the most productive WRs will be marked and covered a lot more by a defence which can result in more space for a less productive WR. The feature distributions show that the per reception values follow a more normal distribution.

Four additional features offer interesting insights. Touchdowns are key to any offense and it is of no surprise that this correlates with the offensive grade. Two after catch features show highly as well and indicate the skill of an individual WR - yards after catch and avoided tackles. The ability for a WR to make a defender miss or break a tackle is something that can lead to additional yards gained as well as getting the team into a better position or score a touchdown. These are effective plays. In addition, the passer rating of the QB who is throwing the pass to the WR is considered - it makes sense that WRs will find it difficult to perform without an effective QB in the team.

The correlation plots for the Wide Receiver position can be found in Appendix B.2. Plotting a basic scatterplot for each feature against the target feature, we can see their corresponding correlation strength. What is interesting to see is that in comparison to the QB scatterplots, the WR scatterplots are a lot tighter and closer together. There are 1877 WR records in comparison to the 311 QB records. Following on from the correlations above it can be seen that those features related to a per reception basis have a lower correlation and that their scatterplots fit the line closely with low correlation. The highly correlated features on the other hand have a different pattern. Indeed that pattern for the more highly correlated features looks like a a slice of pizza where the lower values in the feature such as yards or yards after catch are close together. However, while the values follow the line for the most part, there are some outliers which correlate with a high offensive grade despite a low feature value. That could be as a result of a number of reasons. It could be interpreted that the WR may have a number of different facets which contribute to a good grade, which may not necessarily be present together. Another interpretation is that there is additional data that is missing from the data set that may be available to the analysts either from statistics or their own experiences.

Correlation heat map show the correlation between features. Due to the number of features for the WR position, it is difficult to show the entire correlation heat map. However, the heat maps provided in Figure 3.7 below show some insights into some highly correlated features. As a result a number of features were removed from the data set as they correlated very closely to other features.

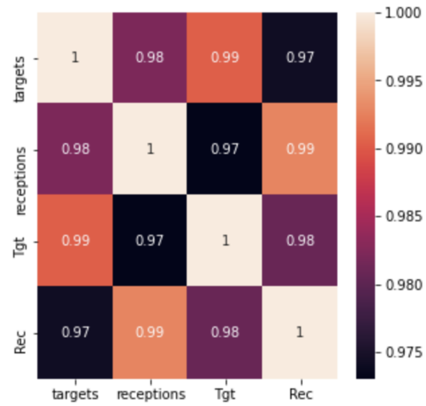


Figure 3.7: WR Correlation Numbers - Receptions

As can be seen in Figure 3.7 above, there are four features which are highly correlated to each other. ‘Tgt’ and ’targets’ are duplicates and show a high correlation of 0.99 here. ‘Rec’ and ’receptions’ are also duplicates and highly correlated with a correlation of 0.99. However, all four features are all highly correlated between 0.97 and 0.99. As a result of the high correlation three of the features can be removed. In this instance ‘Tgt’, ‘targets’ and ‘Rec’ are removed. The feature ‘receptions’ is kept for two reasons. The first is that it is from the PFF data set and most likely to be used in the grade. The second is that receptions were thought to be more relevant as it signifies that the pass was actually caught.

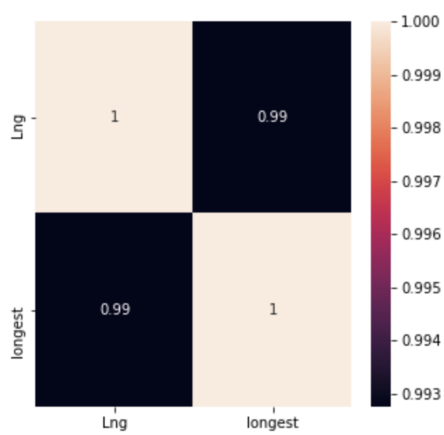


Figure 3.8: WR Correlation Numbers - Longest

As can be seen in Figure 3.8 above, the features ‘Lng’ and ‘longest’ are highly correlated at 0.99. These are effectively duplicate features and as such only one will be kept. In this instance the feature 'longest' will be kept as it is from the PFF data set associated with the grade.

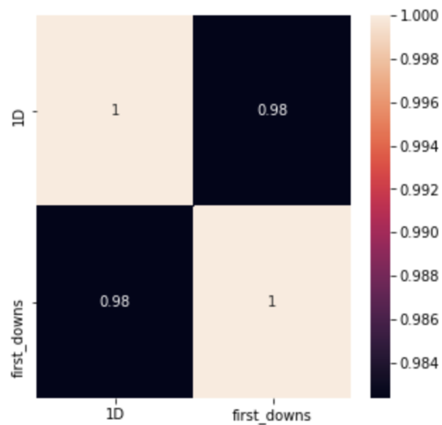


Figure 3.9: WR Correlation Numbers - First Downs

As can be seen in Figure 3.9 above, the features '1D' and 'first_downs' are highly correlated at 0.99. These are effectively duplicate features and as such only one will be kept. In this instance the feature 'first_downs' will be kept as it is from the PFF data set associated with the grade.

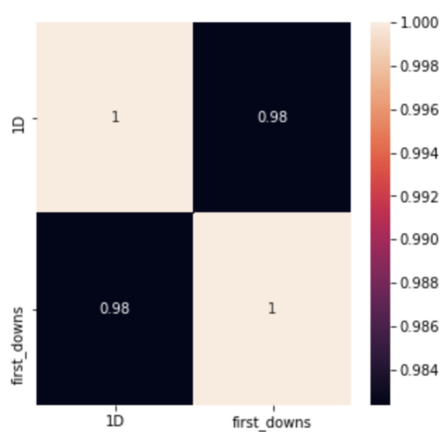


Figure 3.10: WR Correlation Numbers - Grades

As can be seen in Figure 3.10 above, the features 'grades_offense' and 'grades_pass_route' are highly correlated at 0.98. This shows the high correlation between the offensive grade overall and the passing grade. While RBs and running is an important part of the offense, passing plays an important part of any NFL offense and is reflected in it's correlation to the offensive grade. As the 'grades_offense' feature is the target variable, 'grades_pass_route' will be removed.

Running Back

grades_offense	1.000000
grades_offense_receiving	1.000000
grades_run	0.849403
avoided_tackles_rushing	0.506723
yards_after_contact	0.470862
yards_rushing	0.470284
Y/G	0.467851
grades_pass_route	0.467563
longest_rushing	0.441866
avoided_tackles_receiving	0.422819
yards_after_catch	0.416547
first_downs_receiving	0.412564
touchdowns_rushing	0.412348
yards_receiving	0.411737
yco_attempt	0.405516
ypa	0.374881
Y/A	0.361733
GS	0.345666
longest_receiving	0.339046
touchdowns_receiving	0.320652
proBowl	0.309653
targeted_qb_rating	0.306302
player_game_count_rushing	0.294353
G	0.238141
grades_hands_fumble_rushing	0.224331
grades_hands_fumble_receiving	0.224331
caught_percent	0.183383
allPro	0.182897
yards_after_catch_per_reception	0.176434
yards_per_reception	0.160524
penalties_receiving	0.157808
penalties_rushing	0.157808
fumbles_rushing	0.146136
Fmb	0.136000
drops	0.116364
interceptions	0.095108
fumbles_receiving	0.086320
declined_penalties_receiving	0.027279
declined_penalties_rushing	0.027279
Year	-0.005586
Age	-0.072084

Figure 3.11: RB Correlation Numbers against grades_offense

The correlations are measured against the target feature "grades_offense". There are a number of PFF grades in the data set that are compiled from the analysts at PFF. These features will be removed from the data set before progressing with the machine learning. From a positive correlation point of view, a pattern can be seen. The most highly correlated features are those that represent the values collated over the course of the season or per game as opposed to the numbers per run. What this shows is that consistent performances over a prolonged period are more valued than per run averages. Even though player A may have a higher per run average than player B, if player B is a high performing RB, then they will get more attempts with more defensive strategies and players deployed against him.

A number of highly correlated features offer insight. The highest correlated features outside of the PFF grades include avoiding tackles while rushing, yards after contact, number of yards rushing in total and longest rushing yards. We can also see that some receiving features (avoided tackles, yards after catch) are amongst the most highly correlated features and follow the pattern that production after the ball has been received is what is most highly thought of. In both cases it can be seen that the features that correlate most with the offensive grade are those that measure the ability of the player to produce after the ball has been given to them. It is similar to the WR position in that the ability to break tackles, make defenders miss and be a playmaker are the most relevant. From a negative correlation point of view it can be seen that there is a slight negative correlation between age and offensive grades. The running back position is a physically demanding position where there are a lot of big physical hits from big players, so this isn't surprising.

The correlation plots for the Running Back position can be found in Appendix B.3. Plotting a basic scatterplot for each feature against the target feature, we can see their corresponding correlation strength. The scatterplots for the RB position follow a very familiar pattern as those for the WR position - especially in relation to the receiving features. Similar to the WR position there are considerably more records for RB than there is for QB - 1141 RB records versus 311 QB records. As a result there are more data points and some are more observed to be much closer together. Similar to the WR position, per attempt features follow a normal distribution, both for rushing and receiving. However, for the highly correlated features that encompass data not per action, look similar to those of the WR position for a very similar reason.

When the higher grade is achieved there are far more with higher values for example for yards after contact. The higher the value the better correlation with a higher grade - however there are records with low yards after contact but receive high grades. As there are different styles of RB this isn't surprising. Where some may not catch but be very good at running between tackles and big defences, others may use a combination of running against those big defensive lines and catching short passes and running.

Correlation heat maps show the correlation between features. Due to the number of features for the RB position, it is difficult to show the entire correlation heat map. However, the heat maps provided in Figure 3.12 below show some insights into some highly correlated features. As a result a number of features were removed from the data set as they correlated very closely to other features.

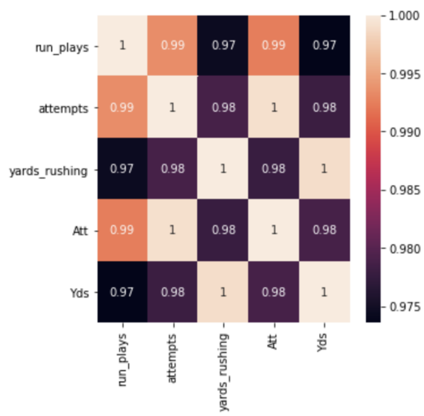


Figure 3.12: RB Correlation Numbers - Rushing Yards

As can be seen in Figure 3.12 above, there are five features that are highly correlated to each other - 'run_plays', 'attempts', 'yards_rushing', 'Att' and 'Yds'. 'attempts' and 'Att' are duplicates and show a high correlation of exactly 1. 'yards_rushing' and 'Yds' are also duplicates and show a high correlation of exactly 1. The final feature 'yards_rushing' is also highly correlated and the minimum correlation between any two of the features is 0.97 showing that every feature is highly correlated to each other. As such it is not needed to use all five and four of the features can be removed. In this instance 'run_plays', 'attempts', 'Att' and 'Yds' are all removed. 'yards_rushing' is kept for two reasons. The first is that it is from the PFF data set and most likely to be used in the grade. The second is that yards rushing were thought to be more relevant as it signifies that the number of yards that the RB actually obtained.

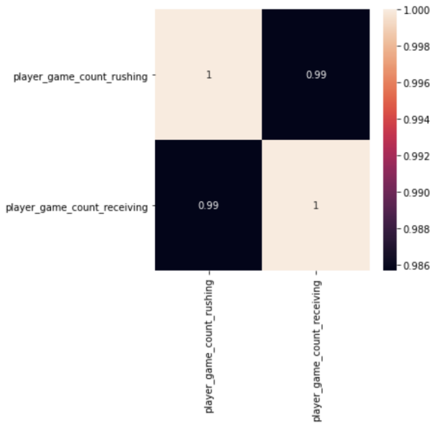


Figure 3.13: RB Correlation Numbers - Game Count

As can be seen above, the features 'player_game_count_rushing' and 'player_game_count_receiving' are highly correlated at 0.99. These are effectively duplicate features and only one will be kept. In this instance the feature 'player_game_count_rushing' will be kept as it is related to the number of games for rushing which is our primary focus with RBs.

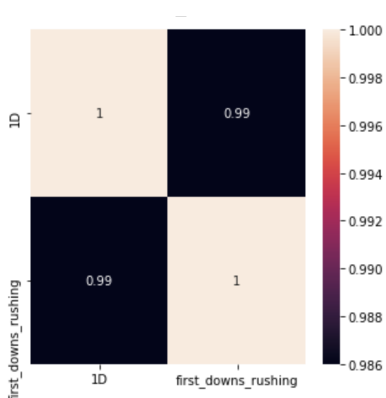


Figure 3.14: RB Correlation Numbers - First Downs Rushing

As can be seen in Figure 3.14 above, the features 'first_downs_rushing' and '1D' are highly correlated at 0.99. These are effectively duplicate features and only one will be kept. In this instance the feature 'first_downs_rushing' will be kept as it is from the PFF data set associated with the grade.

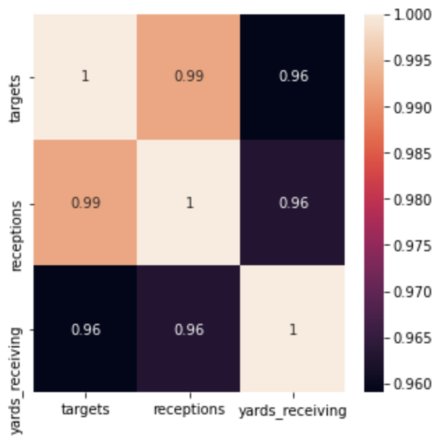


Figure 3.15: RB Correlation Numbers - Receiving Yards

As can be seen in Figure 3.15 above, there are three features which are highly correlated to each other - 'yards_receiving', 'targets' and 'receptions'. None of the features are duplicates of each other but are all highly correlated to each other with the lowest correlation between any two of the three being 0.96. In this instance 'targets' and 'receptions' will be removed. In this instance the feature 'yards_receiving' will be kept as it is from the PFF data set associated with the grade. It also signifies the yards from receiving and and is thought to be more relevant especially as it would be a larger number.

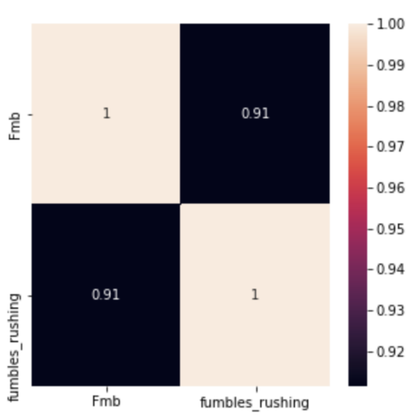


Figure 3.16: RB Correlation Numbers - Rushing Fumbles

As can be seen in Figure 3.16 above, the features 'fumbles_rushing' and 'Fmb' are highly correlated at 0.91. These are effectively duplicate features and only one will be kept. In this instance the feature 'fumbles_rushing' will be kept as it is from the PFF data set associated with the grade.

3.2.4 Correlation Trends over Time

Quarterbacks

Analysis of the correlation of the features against the target variable can be seen in the plots in Appendix C.1. By looking at the correlation of features over the last ten years we hope to observe some trends that are appearing in the game. Looking at some of the features shows a trend in the modern quarterback play in the last 2 years. While QBs who could run have played in the past, it wasn't a skill set that was often used by offensive coordinators as it increases the likely of injury to the team's most important player. However, the figures below indicate that there has been a shift in design of offensive plays to take advantage of the physical attributes and approaches to playing the quarterback position brought by a new breed of college quarterbacks including Deshaun Watson, Patrick Mahomes and Lamar Jackson[6]. These players have the physical traits including physicality and speed to play a new style of game where there is an increased focus on quarterback runs. These players are not just running QBs, but can also pass the ball efficiently. Selected trends over time are discussed below to show the change to the correlation of some features in the previous season that show a shift in the attributes valued in a quarterback. The figures that follow plot the correlation of a particular feature against the target feature "grades_offense" over a period of 10 years to observe the change. In Figures 3.17 and 3.18 below attempts_x is the passing attempts i.e. attempts_passing.

[6] `https://www.wsj.com/articles/the-pocket-passer-is-dying-long-live-the-mobile-quarterback-11578318795`

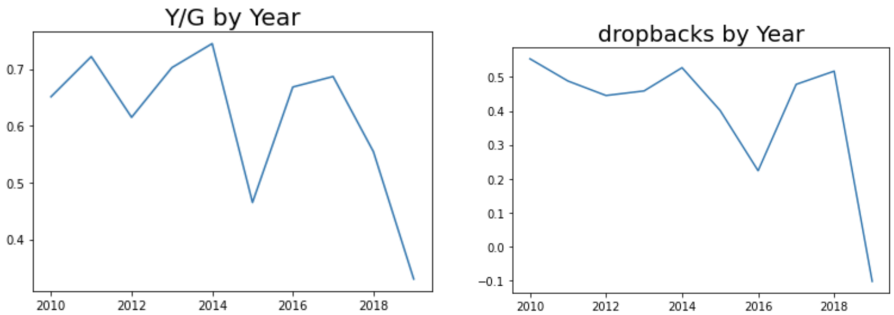


Figure 3.17: Selected Correlation Trend over Time for Yards Per Game (Y/G) and Dropbacks

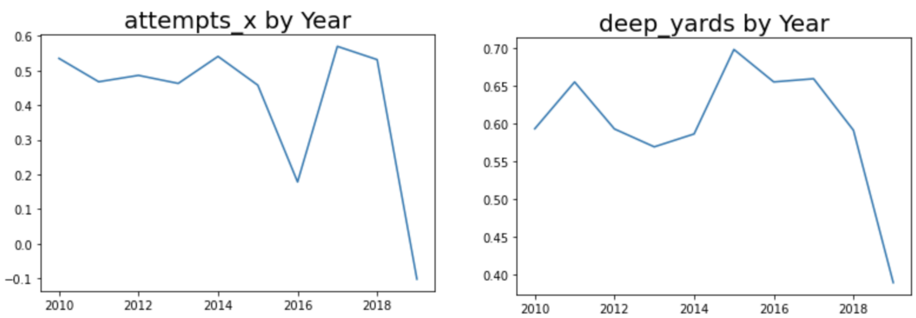


Figure 3.18: Selected Correlation Trend over Time for passing attempts (attempts_x) and deep yards

In Figures 3.17 and 3.18 above, a decline can be seen in the correlation between the target variable and the following features: passing yards per game, dropbacks, passing attempts (attempts_passing) and deep yards thrown. This indicates that passing attempts is becoming less of a factor, with QBs with consistent average passing yards less favoured in comparison to QBs who are efficient and able to get similar yards with less attempts using more efficient throws as well as being able to read the play to choose the right receiver to pass to, a receiver who is more likely to gain more ground.

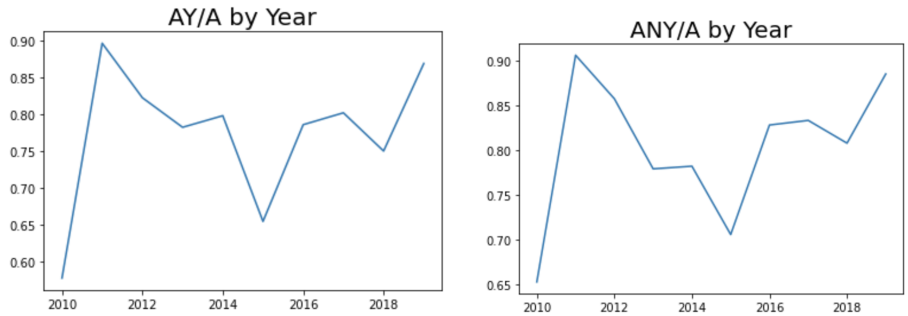


Figure 3.19: Selected Correlation Trend over Time for Adjusted Yards per Attempt (AY/A) and Adjusted Net Yards per Attempt(ANY/A)

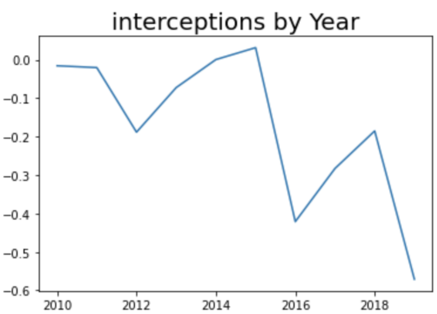


Figure 3.20: Selected Correlation Trend over Time for Interceptions

Figures 3.19 and 3.20 above highlight that increased average yard per attempt is trending upwards, indicating that passers who can make big plays and get larger yards per attempt are more favoured. While larger passing yards per attempt is more desirable, there is a trade-off - risk. Longer passes result in higher risk of a turnover or incompletion. In the modern game, where teams are very even in terms of performance (thanks to the salary cap), every opportunity is vital. To this end the increase in negative correlation between the target variable and interceptions, highlights how important the follow two factors are to a football team: accurate passers over multiple distances and the ability to protect the ball. In Figure 3.21 below ypa_y is the yards per attempts for rushing i.e. ypa_rushing.

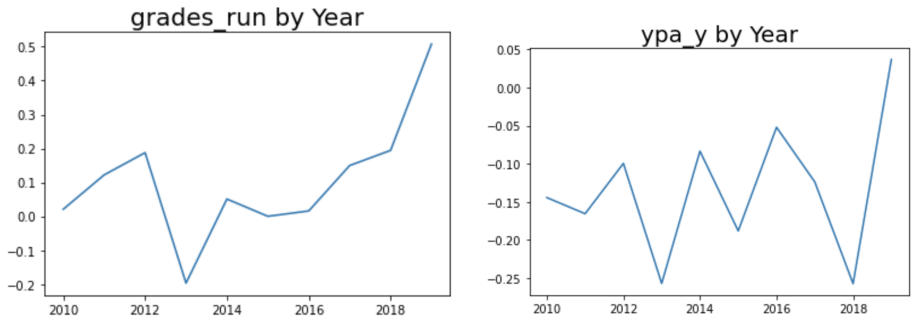


Figure 3.21: Selected Correlation Trend over Time for run grades and Yards per Attempt Rushing (ypa_y)

One method to protect the ball is not to throw the ball, but to run the ball. This decreases the risk of losing the ball as the player running the ball should have more control of the ball than when the ball is in the air. However, this often results in less yards gained. Traditionally the QB would hand the ball to the running back would run the ball. However, recently younger QBs are physically larger and faster than before. This has allowed offensive coordinators to design plays where the QB can run the ball. This can result with the defence having to worry about an extra running threat while the offense will gain a blocker. This can be seen in figures 3.17, 3.18,3.19,3.20 and 3.21 above. Yards per attempt running for a QB has gone from being a negatively correlated feature to being positively correlated, show the desire for QBs who can also incorporate running into their game. This can be seen by the running grade for QBs steadily increasing in recent years.

Wide Receivers

Analysis of the correlation of the features against the target variable can be seen in the plots in Appendix C.2. By looking at the correlation of features over the last ten years we hope to observe some trends that are appearing in the game. With QBs there has been a notable trend to add a running game to modern QBs. With WRs there hasn't been as big a focus on a new style of play. However, what can be seen from some of the trends is that results from the receptions and targets of the WR are key and the ability for the WR to positively impact the play is key.

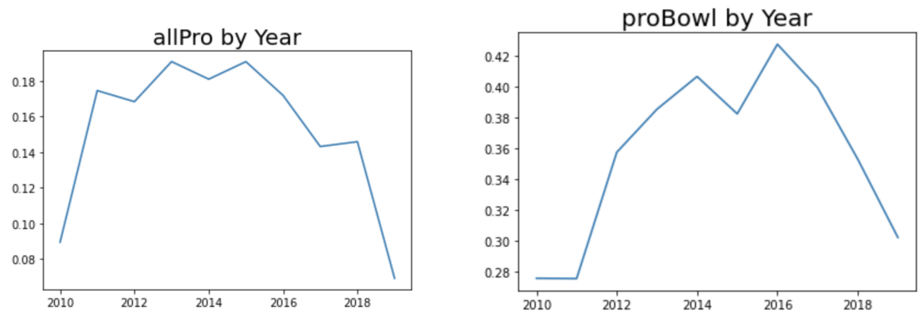


Figure 3.22: Selected Correlation Trend over Time for All Pro and Pro Bowl selection

In Figure 3.22 above a decline can be seen in the correlation between whether a WR made the Pro Bowl or All Pro team and their offensive grade. It can be seen that the in recent years that the correlations has been in decline. This is interesting as both Pro Bowl selection is decided by a vote of coaches, players and fans (each have one third of the vote). And the All Pro team is voted for by the press. As such both of these are voted on by people who follow the game closely, but it is also partly a popularity vote. It is interesting to see the reduction in correlation between the votes and the offensive grades over the course of the season. Where the grades are accumulated on a play-by-play basis, the votes are a once off evaluation at the end of the season.

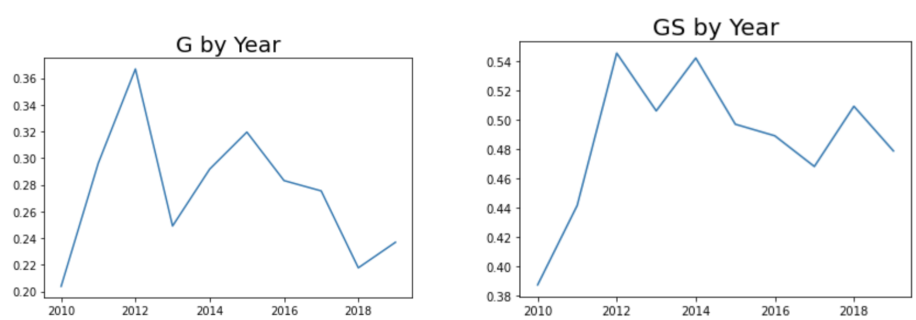


Figure 3.23: Selected Correlation Trend over Time for Games Played (G) and Games Started (GS)

In Figure 3.23 above, the correlation of games played and games started show a decreasing trend. This indicates that starting and playing in games is less of a factor in recent years to the overall grade and that alternative factors may be of influence.

It indicates that other features are now of more influence and that WRs have to show more in their game than previously.

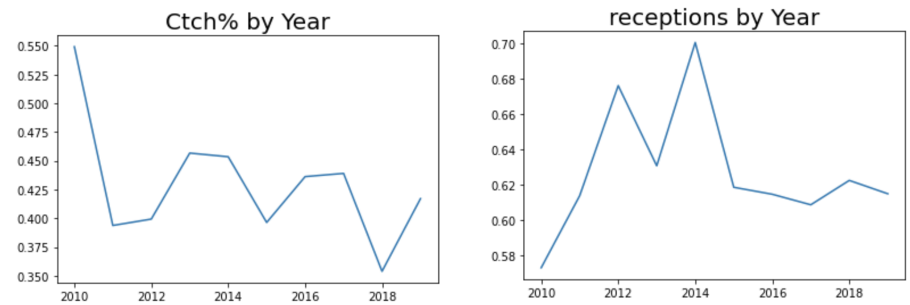


Figure 3.24: Selected Correlation Trend over Time for Successful Catch % (Ctch$) and Receptions

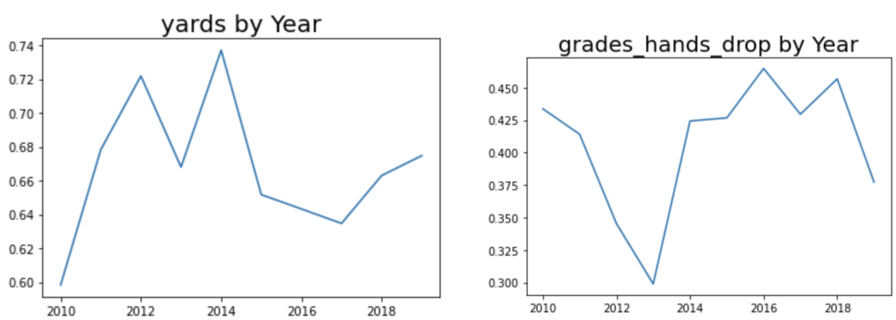


Figure 3.25: Selected Correlation Trend over Time for yards gained and drops grade

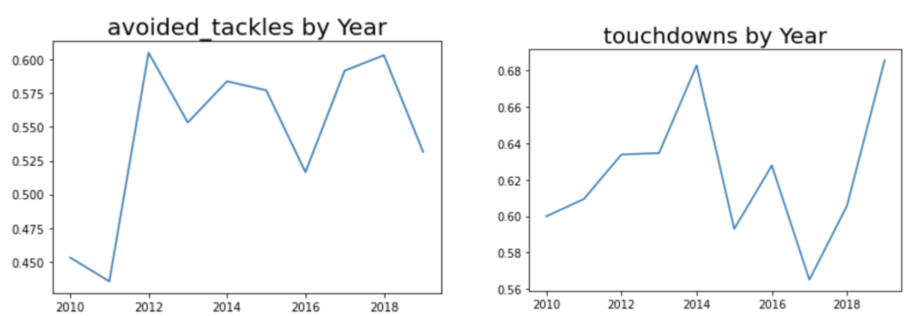


Figure 3.26: Selected Correlation Trend over Time for avoided tackles and touchdowns

Following on from the observation above that appearances in games no longer correlate as high, Figures 3.24, 3.25 and 3.26 above show a trend towards a higher correlation for actions taken after or at point of a catch. Catch%, Yards and receptions, while highly correlated seem to have plateaued in recent years. They are still some of the highest correlated features and show that production by a WR in terms of catches and end result is still an important factor. Indeed the final three figures (Figures 3.17,3.18 and 3.19) showed an increase generally and are closely related to accuracy and after catch production. The grade given for hands in terms of drops indicates that the ability of the WR to not drop the ball and be relied upon to catch the passes has been more important in recent years. The final two figures (Figures 3.20 and 3.21) also show an increase in correlation over the past few years. They are both features which give an indication of the ability of the WR to perform after the catch has been performed. They are both highly correlated to the offensive grade showing that despite appearances as shown above, performance and production are key features of a WR. The ability to make avoid a tackle and make a defender miss can result in additional yards for the team or indeed the scoring of a touchdown. Both of which are key. The ability of the WR to consistently and accurately catch the ball and then add value to the play are becoming more and more relevant to a good grade.

Running Back

Analysis of the correlation of the features against the target variable can be seen in the plots in Appendix C.3. By looking at the correlation of features over the last ten years we hope to observe some trends that are appearing in the game. With QBs there has been a notable trend to add a running game to modern QBs. With WRs there didn't appear to be a big trend. For RBs there has been a noticeable use of running backs as pass catchers on short passes to expose running lanes outside the big defensive linemen. Figures 3.27 and 3.28 plot the correlation of a particular feature against the target feature "grades_offense" over a period of 10 years to observe the change.

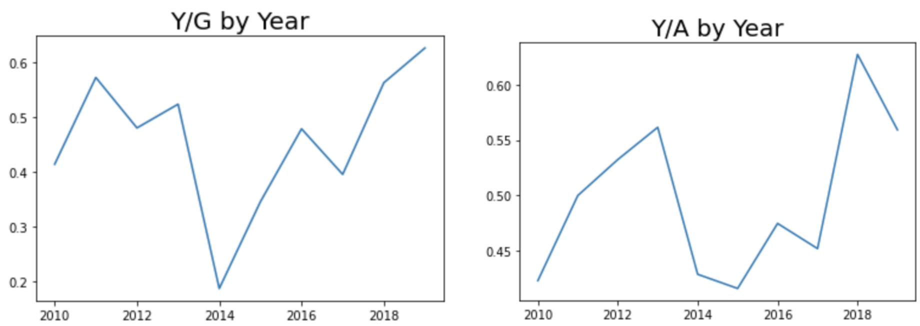


Figure 3.27: Selected Correlation Trend over Time for Yards per Game (Y/G) and yards per attempt (Y/A)

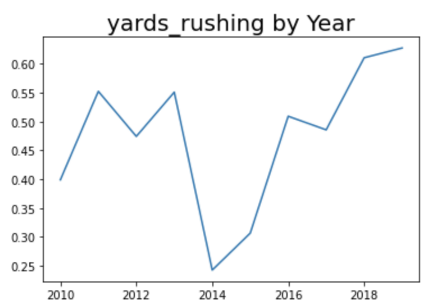


Figure 3.28: Selected Correlation Trend over Time for Yards Rushing

In Figures 3.27 and 3.28 above, it can be noticed that there was a dip in correlation between various rushing yards features and the offensive grade around 2014 and has been steadily rising ever since. This indicates that the production of yards are becoming more relevant to the offensive grades. The dip could be interesting to investigate, but it can be seen that the correlation of some receiving features rose during that period. In addition, part of a RBs job is also to block which is not represented in the data. However, in recent years the ability to make yards through rushing has risen in relevance.

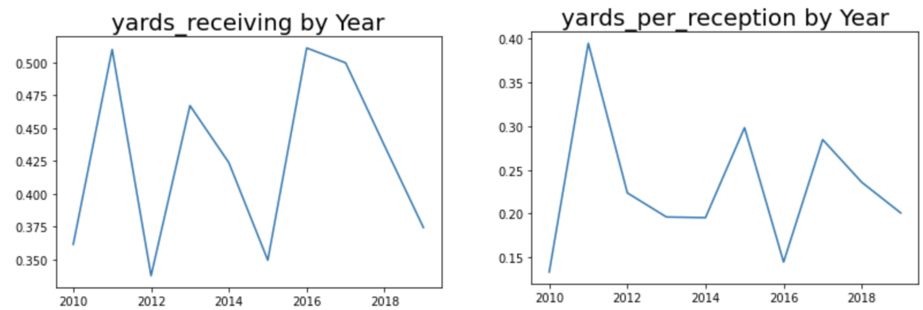


Figure 3.29: Selected Correlation Trend over Time for Yards Receiving Total and Yards Per Reception

Figure 3.29 above show a decrease in the correlation of yards for receiving over the past number of years after some years where it had high correlation. The use of RBs as pass catchers has been noticeable over the last ten years. As an offensive strategy becomes effective so defensive coaches look at effective ways to stop it. This is possibly what is being seen here. As can be seen in the yards after catch per reception in Figures 3.30 and 3.31 below, while receptions correlations have seemed to go down, effective production after the catch is reflected in the increase in correlation.

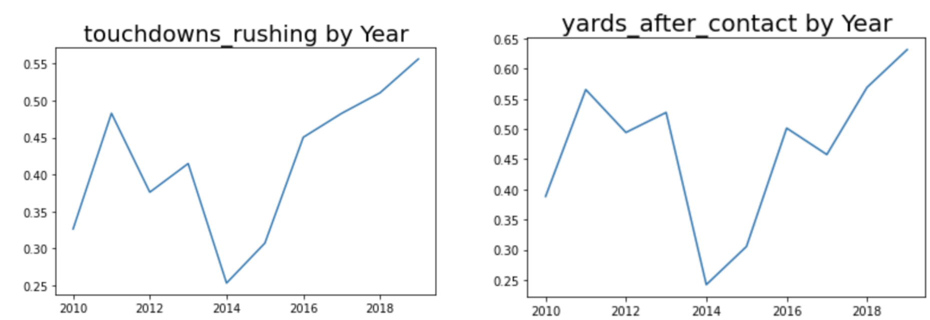


Figure 3.30: Selected Correlation Trend over Time for Rushing Touchdowns and Yards after Contact Gained

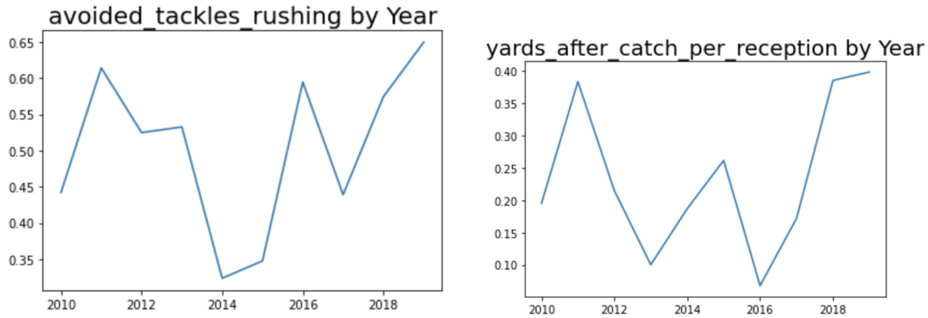


Figure 3.31: Selected Correlation Trend over Time for Tackles Avoided Rushing and Yards after Catch Per Reception

As mentioned above, the ability of the RB to produce results after receiving the ball is paramount. As new tactics appear in the game, so do defensive strategies to nullify them appear. As a result it is ever more important for playmakers such as RBs to make the most of the situations that they find themselves in. This can be seen in figures 3.27, 3.28,3.29,3.30 and 3.31 above, where each of the features represent outcomes such as touchdowns and performance related features showing increasing correlations to the offensive grade, to where they are now among the highest correlated features.

3.3 Methods

In the following sections, the methods used in this thesis to implement the models as well as to extract the most relevant features (Regularization, Early Stopping, Drop Out, Cross Validation, Grid Search, Decision Tree, Linear Regression, SVR, XGBoost, Neural Network and MLP) are outlined.

3.3.1 Model Creation

The following sections outline the machine learning algorithms used in this thesis. Also detailed are some of the techniques used to optimise said algorithms and to avoid over-fitting.

Decision Tree

For Grid Search, only one parameter is chosen - minimum samples per leaf. Mean Squared Error is chosen as the splitting criterion. Mean Squared Error also minimizes the L2 loss per node. Minimum samples per leaf enforces that after a split each left and right branch should have at least that minimum number of leaves in them, helping to reduce overfitting.

Linear Regression

For Grid Search, two parameters were chosen - alpha and solver. Alpha is the regularization strength used as part of the Ridge Regression. The Ridge Regression variant is used so that L2 regularization can be applied to reduce overfitting. Alpha determines the strength of L2 regularization to apply, the values used known as the regularization parameter (R). The larger the value the greater the regularization. The value is used in the formula as $\frac{1}{2R}$.

SVR

For Grid Search, three parameters were chosen - C, gamma and tol. The kernel used is rbf (radial basis function). C is the regularization parameter used to implement L2 regularization. The strength of the regularization used is inversely proportional to the value specified for C. Gamma is the kernel coefficient used for the chosen kernel and is used to adjust the sensitivity in the kernel to input features. Five kernel coefficients were chosen - $2^{-9}, 2^{-7}, 2^{-5}, 2^{-3}, 2^{-1}$.

The tol parameter specifies the tolerance for the stopping condition for early stopping. This should help mitigate against overfitting in the model. Five values for the tolerance were chosen - 1e$^{-5}, 1e^{-4}, 1e^{-3}, 1e^{-2}, 1e^{-1}$.

XGBoost

For Grid Search, two parameters were chosen - number of estimators and the learning rate. Squared error was used as the objective by which to evaluate. Number of estimators is the number of boosting rounds used to learn the function. The learning rate is similar to regression in that it is a weighting to slow down the learning and helps mitigate against overfitting.

Neural Network

For Grid Search, six parameters were chosen - solver, maximum iterations, activation, alpha, number of iterations with no change and tol. There is a default hidden layer size of 100 neurons. Solver is the optimizer to be used to change the weights of the network and the learning rate so as to minimize the loss and achieve results quicker. Maximum iterations is the number of iterations that the solver should go through until convergence is found as determined by the tolerance. It determines the number of epochs that are used for the neural networks training. Activation is the activation function used by the network to apply a mathematical formula to a set of inputs to a neuron/node and generate an output. Alpha is the L2 regularization parameter used to mitigate against overfitting. Number of iterations with no change is used to provide early stopping in the network. It specifies the maximum number of epochs where the tolerance improvement was not met. Tol is the tolerance that was mentioned in the previous parameters and is the tolerance used for optimizing the network.

Multi-Layer Perceptron

For Grid Search, five parameters were chosen - number of neurons, epochs, optimizer, activation and dropout rate. A batch size of 128 was chosen. Even though this may reduce accuracy, it allowed for the learning to finish quicker. Through trial and error, three layers worked well. Both accuracy and speed of training had to be taken into account. Number of neurons is the number of neurons in each layer. Epochs is the number of iterations to use while training the model until convergence is found - similar to the maximum iterations in the previous approach. Optimizer is the same as the solver in the previous approach. Activation is again the same as in the previous approach. Finally the drop out rate is adjusted. As mentioned previously drop out helps to learn more independent features and helps mitigate against overfitting. The rate determines what fraction of the neurons are held out at that layer. Dropout has been applied to each layer in the network.

3.4 Error Evaluation

The evaluation of the models will be conducted across the positional groups. Each machine learning approach (Linear Regression, Decision Tree, SVM, Gradient Boost and the ANNs) will be run against each positional group using K-fold cross validation with K set to 10. The r-squared, root mean squared error (rmse) and mean absolute error (mae) will be retrieved for each approach. The r-squared will be our main metric as this measures the best fit - it denotes the proportion of the variance in the target (dependent variable) that is predictable from the independent variables using the model. The hypothesis will be accepted if the results from the neural networks are comparable to those achieved by the traditional models. That is to say that after running the models using K-Fold cross validation with K equal to 10 (along with a t-test), and if significant we should evaluate the models based upon their r-squared values. A neural network model should be comparable to the best fitting traditional model (within 5 percent) for the hypothesis to be accepted. Also the best fitting model for each position group should be used along with the wrapper method to determine the most significant features that lead to the evaluation of the player by the model.

3.5 Summary

In this chapter the data and its sources were introduced. The merging of the sources for each position was detailed and the resulting data sets and features were investigated for interesting insights. The hyper-parameters for the different approaches were discussed along with the error evaluation. In the next chapter, the results of the experiment will be presented and discussed.

Chapter 4

Results, evaluation and discussion

Following the presentation of the approaches in chapter 3, the approaches are implemented and their results for each of the three positions discussed in this chapter. In sections 4.1, 4.2 and 4.3 the implementations are discussed and their results compared. The results of the investigations into the most significant features are presented for each position. In section 4.4, the results are presented from the point of view of the approaches. Finally, a discussion on the results is presented for each position.

4.1 Quarterback Results

The following models were run on the QB data allowing for the models to be compared and the most significant features extracted.

4.1.1 Model Results and Evaluation

As mentioned in the previous section, Methods , 5 different approaches were taken to predict the PFF ranking of the quarterback for a given year. In order to achieve the best result, each approach will use Grid Search to obtain the best parameters from a range of parameters provided. Each approach will also use 10-Fold Cross Validation to help reduce overfitting, allow for all the data to be used and to get more metrics by which to evaluate the approaches. Each approach will also apply standard scaling to the input features - scaling to unit variance for each feature.

Linear Regression

The first approach that was evaluated was Linear Regression. This approach gives a mathematical calculation across all the 50 parameters in the input feature data set. As well as an intercept value, the Linear Regression algorithm learns an appropriate weight for each feature.

For Grid Search the following values were used for the parameters.In this case the Alpha values of 0.5, 0.8 and 1.0 were used as the regularization parameter (R). Therefore the values range from no regularization (0.5) to regularization of a half (1.0). The other parameter is the solver. This represents the computational functionality used in determining the target. Two solvers were tested - lsqr (least squares resolver) and sag (stochastic average gradient). Both are commonly used solvers.

Using Grid Search it was found that the best parameters were an alpha of 1.0 and sag was the best solver. This combination resulted in a mean squared error of 40.1148 and an accuracy of 0.6786.

Use of a higher alpha would be recommended, to see at what point too much regularization affects the result in a negative sense. The accuracy of 0.6786 indicates that there is plenty of work to be done to achieve a similar grade as that provided by PFF.

From the weights learned by the model, it can be seen that the feature that most positively affects the grade is the yards per attempt passing (ypa_x).

```
'1D': 4.247804376970762,
'4QC': -0.7404737333214361,
'ANY/A': 0.2643855976247282,
'AY/A': 2.4036182153610506,
'Age': 0.7742034117074661,
'GS': 1.3112468168564737,
'GWD': 1.4145375652744712,
'Int%': -1.8790454196475515,
'Lng': -0.3598535729491476,
'NY/A': 0.3461983979960585,
'Sk%': -1.4133542638925143,
'TD%': -0.2034094022616564,
'Y/C': -1.6635887185071712,
'Y/G': -0.004784249950890272,
'Yds.1': 2.6006720356171353,
'Year': -0.7019233920776335,
'allPro': 0.005147874062123075,
'attempts_y': -1.182854597968451,
'avoided_tackles': 0.06800599256068546,
'bats': -1.528032619470236,
'completion_percent': 2.107898746472693,
'completions': -3.1104865393674004,
'deep_accuracy_percent': 0.7221239030290939,
'deep_attempt_percent': 0.9673298448956406,
'deep_attempts': -1.1278125915581494,
'deep_drops': 1.1771194928249773,
'deep_interceptions': -0.045366530467295815,
'deep_touchdowns': -1.5515853594505227,
'deep_yards': 0.7272038320326467,
'draws': 0.9681257728152066,
'drops': 1.6930915228818701,
'first_downs': 2.44907092453826,
'grades_hands_fumble': 1.1577315302353486,
'hit_as_threw': 0.42136460364560474,
'interceptions': 1.4134113016416887,
'losses': 0.3840543088029575,
'player_game_count': -2.31451938635951,
'proBowl': 0.7566525841812911,
'qb_rating': 1.257873374574428,
'run_plays': 1.1035816017515703,
'sacks': -1.9014407942845488,
'scrambles': 0.03317904835873411,
'thrown_aways': 0.48154648310693604,
'touchdowns_x': 0.3470858131178369,
'touchdowns_y': -0.20423418137026042,
'wins': 0.45040434179177474,
'yards_x': -3.5643907849080865,
'yards_y': 0.998678652856447,
'ypa_x': 3.4783434834398452,
'ypa_y': 0.7581268809120808}
```

Figure 4.1: QB Linear Regression Coefficients

Each of the coefficients in Figure 4.1 above matches to each of the input features shown. For any single prediction, the value for each of the features are multiplied by their corresponding coefficient and they are all added together. The result is added to the intercept of 72.0621 to predict the grade/target value. Some of the top features to positively affect the prediction include yards per attempt passing, first downs, completion percentage amongst others. If the input for a prediction provided 0 for all values, the model would produce a better grade of approx. 72 versus the 67.8 achieved on average. This is interesting as the QBs age provides a positive increase.

SVR

SVR was the next approach evaluated. This approach again uses mathematical calculations to arrive at a prediction from the input feature set.

For Grid Search the following values were used for the parameters. Four values were tried for C in the Grid Search - $1\mathrm{e}^0, 1e^1, 1e^2, 1e^3$.

Five kernel coefficients were chosen for Gamma - $2^{-9}, 2^{-7}, 2^{-5}, 2^{-3}, 2^{-1}$.

Five values for the tolerance were chosen - $1\mathrm{e}^{-5}, 1e^{-4}, 1e^{-3}, 1e^{-2}, 1e^{-1}$.

Using Grid Search it was found that the best parameters were a C of 10, gamma of 0.0019 and tol of $1\mathrm{e}^{-5}$.

This combination resulted in a mean squared error of 32.1267 and accuracy of

0.7426. This is a significant improvement on the results returned by Linear Regression. However, SVR doesn't expose the importance of features within the model - it is a black box.

XGBoost

As described previously, XGBoost is a modern implementation of gradient boosting, using ensembles of decision trees to progressively build stronger and stronger models based upon regression error.

For Grid Search the following values were used for the parameters. Four values for number of estimators were chosen - 50, 100, 500 and 1000. Three values for learning rate were chosen - 0.01, 0.05 and 0.1.

Using Grid Search it was found that the best parameters were number of estimators of 1000 and learning rate of 0.01.

This combination resulted in a mean squared error of 38.4277 and accuracy of 0.6921. While it is an improvement on the Linear Regression model, the SVR model performs considerably better on this data set. Similar to SVR, XGBoost is also a black box. Given the iterative nature of the improving model, it is difficult to reverse engineer to find the impact of each feature.

Decision Tree

Like Linear Regression, Decision Trees are one of the older more established approaches. It also relies upon mathematics, in this case to eagerly determine the most appropriate splits at a particular level depending on the feature and its value.

For Grid Search the following values were used for the parameter.Five values were chosen - 2, 4, 6, 8, 10.

Using Grid Search it was found that the best parameter was minimum samples per leaf of 10, indicating less overfitting to the test data set.

This combination resulted in a mean squared error of 57.4905 and accuracy of 0.5394. This is significantly worse than even the Linear Regression approach. As mentioned previously, as Decision Trees are eager and make decisions at a local level, there is no guarantee that it is the globally correct decision and therefore can result in less accurate models. One advantage of Decision Trees is that they can explain the choices made.

From the graph of the decision tree shown in Appendix D.1, it can be seen that any prediction that is made starts with a decision on the value of the Adjusted Net Yards per Attempt (ANY/A) feature. Any value for this over 72.352 will result in a higher grade, indicating that this is the most important feature in the data set for the Decision Tree. Other critical features include the QB Rating as determined by the NFL, the QBs age, Yards Per Game (Y/G) and the number of first downs they obtain. It is also interesting to note that the QB Rating is used a second time down one of the branches in the tree.

Single Layer Neural Network

Single Layer Neural Network is the first neural network approach used in this evaluation. In this case only one layer is considered when training the model. Weights are randomly assigned on initiation of the model and a series of feed forward with back propagation update the weights of the neurons in the single layer of the network.

For Grid Search the following values were used for the parameters. Two optimizers were chosen - stochastic gradient descent and adam. They are both stochastic optimizers. For maximum iteration four values were chosen - 110, 100, 90 and 80. For activation two activation functions were chosen - Identity and Relu. The Identity activation function acts in a similar manner to Linear Regression, except per neuron. The Relu activation function is a popular activation function in deep neural networks and outputs the input if it is positive but sets all negative input to zero. For alpha 6 values were chosen 0.00001, 0.0001,0.001,0.01,0.1 and 0.5. For number of iterations with no change three values were chosen - 10,15,20. For the tolerance three values were chosen - 1e^{-5}, $1e^{-4}, 1e^{-3}$.

Using Grid Search it was found that the best parameters were solver of sgd, maximum iterations of 80, activation function of identity, alpha of 0.01, number of iterations with no change of 15 and tolerance of 1e^{-3}.

The combination resulted in a mean squared error of 38.5951 and accuracy of 0.6908. This is comparable to the XGBoost approach with a similar mean squared error but marginally worse accuracy. Similar to the XGBoost approach, it is difficult to find the impact of each feature, although the coefficients and weights are available. However, as every input is connected to every neuron in the hidden layer this is a difficult task.

Deep Neural Network

The Deep Neural Network approach is very similar to the previous Single Layer Neural Network approach except that it uses multiple layers to discover patterns in the data to help improve prediction. Also a different technology is used to implement deep neural networks here. Keras with Tensorflow is used instead of SciKit Learn as it is one of the most common and well known frameworks for buildling deep learning networks.

For Grid Search the following values were used for the parameters. For number of neurons two values were chosen - 80 and 100. For number of epochs Three values were chosen - 1000,3000 and 5000. Optimizer is the same as the solver in the previous approach and three values were chosen - sgd, adam and rmsprop. Rmsprop was not available in the previous approach, but provides an additional option. Activation is again the same as in the previous approach and the same values are chosen again, with Identity now known as linear. Adam was also used again. For drop out two values were chosen - 0.3 and 0.5.

Using Grid Search, it was found that the best parameters were number of neurons of 80, epochs of 5000, optimizer of rmsprop, activation of linear and dropout rate of 0.5.

The combination resulted in a mean squared error of 38.2680 and accuracy of 0.6934. This again is similar to the XGBoost approach except with slightly better results for both mean squared error and accuracy. As with the Single Layer Neural Network, the ability to explain how a prediction arose from it's inputs is a very difficult task.

The summary of the layers in the network can be seen in Figure 4.2.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_40 (Dense)             (None, 80)                4080
_________________________________________________________________
dropout_30 (Dropout)         (None, 80)                0
_________________________________________________________________
dense_41 (Dense)             (None, 80)                6480
_________________________________________________________________
dropout_31 (Dropout)         (None, 80)                0
_________________________________________________________________
dense_42 (Dense)             (None, 80)                6480
_________________________________________________________________
dropout_32 (Dropout)         (None, 80)                0
_________________________________________________________________
dense_43 (Dense)             (None, 1)                 81
=================================================================
Total params: 17,121
Trainable params: 17,121
Non-trainable params: 0
_________________________________________________________________
```

Figure 4.2: QB Deep Neural Network Summary

4.1.2 Feature Extraction

When predicting grades of players it can be useful to highlight what features influence the prediction the most. However, that can be a challenge when the model uses an approach that doesn't easily explain how it came to that prediction. While Decision Trees can explain the decisions made to get to a prediction, and Linear Regression can give the coefficients for each feature along with the intercept term, neither is the most performant approach.

To this end, two approaches will be used to explore the model to see the most significant features in the input feature set. Wrapper methods have been used to identify significant features in previous machine learning work with sport (Nsolo et al., 2019).Both approaches taken are often used as feature reduction techniques and can be classified as wrapper methods but can also help provide insight into significant features. One approach is that of Sequential Feature Selection as outlined and implemented by mlxtend[1]. The other approach is to use Permutation Importance as outlined by eli5[2]. With Permutation Importance measures how loss or score varies based upon the omission of a feature.

By using the two approaches mentioned above, significant features can be found, with those found in both gaining extra significance. Only the top features of each will be considered. In addition to this, a comparison will also be made to the features and their weights arising from both the Linear Regression and Decision Tree models. The top performing model - SVR - will be analysed for the most significant features.

Table 4.1: Significant Features for QB

Permutation Importance	**SFS**
ypa_passing	1D
completion_percent	ANY/A
qb_rating	completion_percent
losses	grades_hands_fumble
AY/A	drops
deep_accuracy_percent	qb_rating
interceptions	deep_drops
Age	deep_attempt_percent
grades_hands_fumble	deep_accuracy_percent
allPro	losses
ANY/A	

There are six common significant features arising from the overlap of the two wrapper methodologies:

- completion_percent
- qb_rating
- losses
- deep_accuracy_percent

[1] http://rasbt.github.io/mlxtend/user_guide/feature_selection/SequentialFeatureSelector/#sequential-feature-selector

[2] https://eli5.readthedocs.io/en/latest/blackbox/permutation_importance.html#eli5-permutation-importance

- grades_hands_fumble
- ANY/A

Completion Percent is the percentage of passes that were completed by the QB to their target. QB Rating is the NFLs Passer Rating for QBs. This is published frequently by the NFL. Losses are the number of losses the QB was involved in. Deep accuracy percent is the percentage of deep passes that were completed - similar to the completion percent except solely looking at deep (long - 20+ yards) passes. Grades hands fumble is a grade given based on the QB fumbling the ball when receiving a snap at the start of a play or from hand the ball to a running back. ANY/A is adjusted net yards per attempt. This is a more advanced statistic which is a calculation based on a number of other features. It rewards touchdowns thrown while punishing interceptions and is all calculated into one statistic using the following formula[3] (pass yards + 20*(passing touchdown) - 45*(interceptions thrown) - sack yards)/(passing attempts + sacks).

4.1.3 Discussion

The table below summarizes the results found above.

Table 4.2: QB Model Comparisons

Model	MSE	Accuracy
SVR	32.126725	0.742643
Deep Neural Network	38.268055	0.693447
XGBoost	38.427732	0.692168
Single Layer Neural Network	38.595139	0.690827
Linear Regression	40.114897	0.678653
Decision Tree	57.490511	0.539462

We can see from Table 4.2 above that the SVR approach clearly outperforms the other approaches. While both Linear Regression and Decision Tree both come with an element of observability and explainability, their results are the worst performing of

[3] https://www.pro-football-reference.com/about/glossary.htm

all the approaches. This raises an interesting discussion and trade-off of performance versus explainability. Depending on the audience a different approach may be necessary. If the goal is to solely provide a metric, then SVR is clearly the best performing approach for a data set of this size. Deep Neural Networks perform best with a larger data set and it is possible that with a larger data set or a fine tuned pre-trained model that the Deep Neural Network could improve it's performance (Feng, Zhou, & Dong, 2019).

Each of the approaches performed Grid Search with 10 Fold Cross Validation. As such each had different parameters to tune and optimize. Where possible techniques such as Drop out, Regularization and early stopping were included to mitigate against overfitting. Many of the starting points for the parameters used were achieved through manual trial and error and this can be seen in the difference in parameter values for similar parameters for Single Layer Neural Network and Deep Neural Network. Indeed in these cases trade-offs were needed to address learning time. GPUs help speed up the training, but resources can be limited in usages of GPUs.

The difference in performance of each approach depending on the parameters used in the Grid Search with Cross Validation can be seen in the box plots in Figures 4.3 and 4.4.

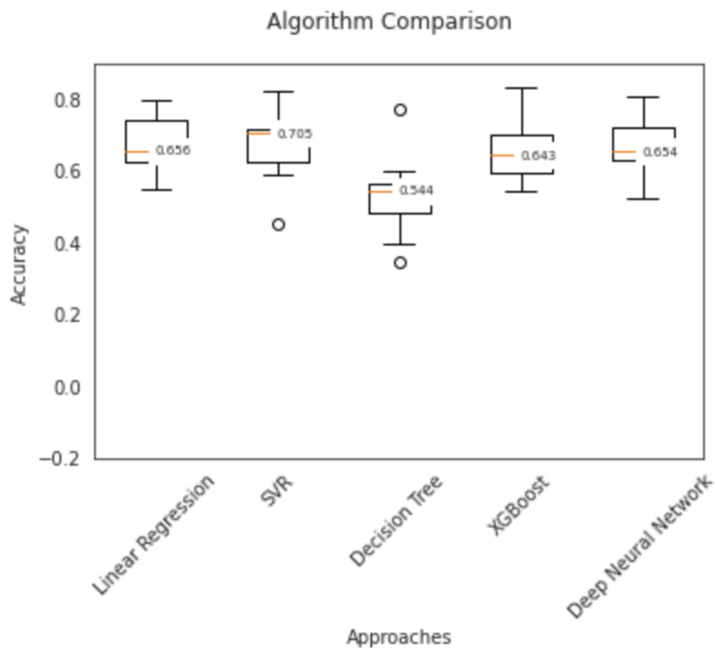


Figure 4.3: QB Best Performing Search Parms Per Approach

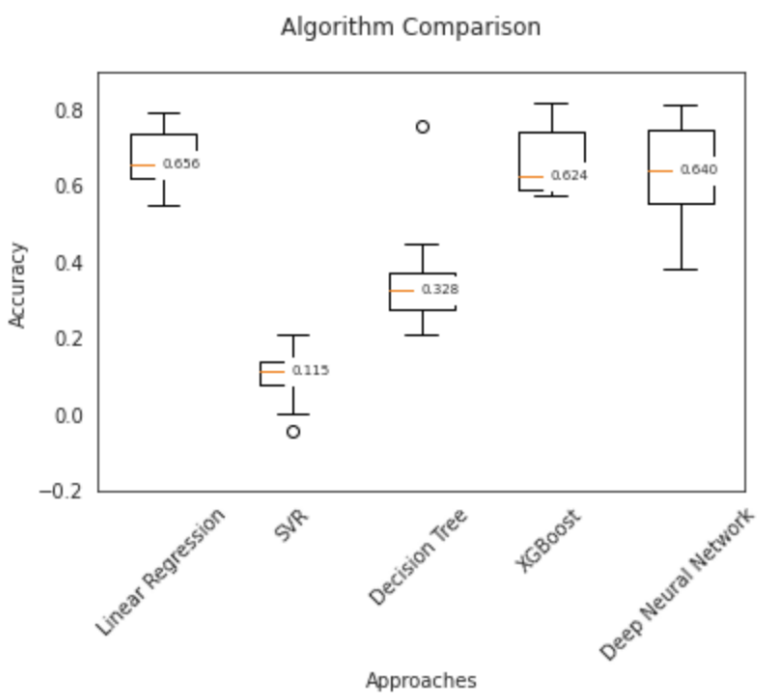


Figure 4.4: QB Worst Performing Grid Search Parms Per Approach

The box plots in Figures 4.3 and 4.4 above represent the best and worst performing parameters for each approach using Grid Search Cross Validation. Each box plot represents the 10 accuracy numbers as achieved using 10 Fold Cross Validation for that parameter set for the accuracy metric. The values for the Single Layer Neural Network approach could not be shown at time of writing. They do highlight some interesting observations.

The first interesting observation is that depending on the difference in parameters, some approaches will be significantly more affected than others. As an example, the worst performing models for Linear Regression, XGBoost and Deep Neural Network do not appear to vary significantly from the best performing model. However, both Decision Tree and SVR vary significantly from their best performing model. The is especially the case with SVR. The difference in parameters results in the SVR being significantly the worst performing approach with it's worst set of parameters, to being significantly the best with it's best performing set of parameters. This could be interpreted as the SVR approach being more sensitive to parameter changes, or that the best set of values and/or parameters were not found for the other approaches. Further investigation would be required for this.

The second interesting observation surrounds the box plots themselves and their ranges for the best performing models. From the box plots it can be seen that the box plots of Linear Regression, SVR, XGBoost and Deep Neural Networks are much more similar to each other than to Decision Trees. This reinforces the findings that those models tend to have a difference to the Decision Tree model that is statistically significant - p values of 0.027 for Decision Tree versus Neural Networks, 0.002 for decision tree versus Linear Regression with SVM at 0.094 and XGBoost at 0.163. None of the other approaches had a difference to another that was statistically significant.

It is also interesting that XGBoost appears to have achieved the highest accuracy during one of it's Cross Validation runs. And indeed it's range seems to be higher than that of Deep Neural Networks. However, it's inter-quartile range is longer and it's median is slightly lower than that in the Deep Neural Network. This may result in a wider range of accuracy values when using XGBoost versus the Deep Neural Network model. The Linear Regression model does appear to have a large inter-quartile range, but never gets a maximum that compares to the other three highest models. While

Decision Tree has one outlier at the top, it is significantly lower. SVR is interesting as it has an outlier at the lower end. This may appear at the start of the training cycle, whereas the median is almost at the top of the interquartile range, indicating that there are a considerable number of higher values above that point and that indeed that low outlier may have skewed the inter-quartile range somewhat. Despite achieving the highest accuracy and lowest mean squared error across the approaches, there is a marked difference in accuracy between the SVR prediction and that of the grade given by PFF. This indicates that there could be many more optimizations that could be made. However, additional data points might also add additional patterns that could be observed within the data. PFF use many analysts to look at plays in the game and to give scores based upon the outcome (as defined in the statistics) as well as situational evaluations. This situational awareness provides a level of detail that the data used here does not provide. An example for QBs might be a well thrown, accurate deep pass to a receiver who makes a mistake and the opposition team turning over the ball and returning the play for a touchdown. In this scenario, the QB made a good play but was let down by the receiver. In the data collected in this data set, this would be represented as a negative for deep passing metrics, an additional interception and would negatively affect the QBs grade. However, with additional data, such as PFFs analysis, then the QB may actually get a more favourable grade.

However, SVR outperformed the other approaches - especially those that can give some indication as to how they came to a prediction. To tackle this some wrapper method techniques were applied to the SVR as it was the best performing approach. As a result some insight was observed into what were the most significant features for SVR. One prominent theme that shows is that accuracy of both throwing and execution is a key metric as shown by the completion percent, deep accuracy percent features as well as the grades hands fumble feature. These, coupled with the externally calculated values of QB Rating and ANY/A is important as it shows that QBs who can reduce unforced errors and be accurate grade highest. As shown previously, in recent years QBs who can run have seen their grades rise as a result. However, over the span of the last ten years accuracy has been of most importance.

An interesting insight is to compare the significant features used in the prediction of the SVR model to those explained by the Decision Tree and Linear Regression models. For the Linear Regression model only 1D (first downs), Age and completion percent overlap with the significant features of both wrapper methods for SVR. Indeed, only completion percent is in all three. For Decision Trees there is much more overlap with the following features appearing prominently in the graph: ANY/A, QB Rating, Age, 1D (first Downs), Deep Accuracy Percent and Deep Drops. Three of these overlap with both the wrapper methods for the SVR - ANY/A, Deep Accuracy Percent and QB Rating. It is noticeable that the features that are in common are based upon accuracy and completing passes, as shown in Deep Accuracy Percent and completion Percent as well as in the calculated features of ANY/A and QB Rating.

4.2 Wide Receiver Results

The following models were run on the WR data allowing for the models to be compared and the most significant features extracted.

4.2.1 Model Results and Evaluation

As in the previous section, Section 4.1, 5 different approaches were taken to predict the PFF ranking of the wide receiver for a given year. In order to achieve the best result, each approach will use Grid Search to obtain the best parameters from a range of parameters provided. Each approach will also use 10-Fold Cross Validation to help reduce overfitting, allow for all the data to be used and to get more metrics by which to evaluate the approaches. Each approach will also apply standard scaling to the input features - scaling to unit variance for each feature.

Linear Regression

The first approach that was evaluated was Linear Regression. This approach gives a mathematical calculation across all the 50 parameters in the input feature data set.

As well as an intercept value, the Linear Regression algorithm learns an appropriate weight for each feature.

For Grid Search the following values were used for the parameters.In this case the Alpha values of 0.5, 0.8 and 1.0 were used as the regularization parameter (R). Therefore the values range from no regularization (0.5) to regularization of a half (1.0). The other parameter is the solver. This represents the computational functionality used in determining the target. Two solvers were tested - lsqr (least squares resolver) and sag (stochastic average gradient). Both are commonly used solvers.

Using Grid Search it was found that the best parameters were an alpha of 1.0 and sag was the best solver. This combination resulted in a mean squared error of 26.4427 and an accuracy of 0.7525.

Use of a higher alpha would be recommended, to see at what point too much regularization affects the result in a negative sense. The accuracy of 0.7525 indicates that there is plenty of work to be done to achieve a similar grade as that provided by PFF. This will be discussed later.

From the weights learned by the model, it can be seen that the feature that most positively affects the grade is the number of first downs as a result of the play (first_downs).

```
'Age': -0.14776468645548493,
'Ctch%': 0.9018535421485782,
'Fmb': 0.10279553250076776,
'G': 0.46971804591528576,
'GS': -2.2352622549921866,
'R/G': 0.27656239404126154,
'Y/G': 5.199009290665431,
'Y/R': -4.309360608012859,
'Y/Tgt': 1.4255474222504088,
'Year': -0.7240881283737494,
'allPro': -0.06018830408612927,
'avoided_tackles': 1.940981364865342,
'caught_percent': 0.17286963657585527,
'declined_penalties': -0.19719759580061838,
'drops': -2.4116396436032472,
'first_downs': 9.085711000744823,
'fumbles': -0.9036697556710078,
'interceptions': 0.29689445770186784,
'longest': -0.5576861353092278,
'penalties': -0.3460820867031004,
'player_game_count': -0.4445111070382823,
'proBowl': 0.4610273234479633,
'receptions': -1.5074630129014994,
'targeted_qb_rating': 1.2058557213380325,
'touchdowns': -0.3385374118248813,
'yards': -1.4527269133871714,
'yards_after_catch': -1.4360126082493625,
'yards_after_catch_per_reception': -0.32098515868808003,
'yards_per_reception': 5.152331164233034}
```

Figure 4.5: WR Linear Regression Coefficients

Each of the coefficients matches to each of the input features in Figure 4.5. For any single prediction, the value for each of the features are multiplied by their corresponding coefficient and they are all added together. The result is added to the intercept of 66.4356 to predict the grade/target value. Some of the top features to positively affect the prediction include yards per game, first downs, yards per reception amongst others.

SVR

SVR was the next approach evaluated. This approach again uses mathematical calculations to arrive at a prediction from the input feature set.

For Grid Search the following values were used for the parameters. Four values were tried for C in the Grid Search - $1\mathrm{e}^0, 1e^1, 1e^2, 1e^3$.

Five kernel coefficients were chosen for Gamma - $2^{-9}, 2^{-7}, 2^{-5}, 2^{-3}, 2^{-1}$.

Five values for the tolerance were chosen - $1\mathrm{e}^{-5}, 1e^{-4}, 1\mathrm{e}^{-3}, 1e^{-2}, 1e^{-1}$.

Using Grid Search it was found that the best parameters were a C of 100, gamma of 0.001953125 and tol of $1\mathrm{e}^{-4}$.

This combination resulted in a mean squared error of 25.4618 and accuracy of 0.7617. This is a slight improvement on the results returned by Linear Regression. However, SVR doesn't expose the importance of features within the model - it is a black box.

XGBoost

As described previously, XGBoost is a modern implementation of gradient boosting, using ensembles of decision trees to progressively build stronger and stronger models based upon regression error.

For Grid Search the following values were used for the parameters. Four values for number of estimators were chosen - 50, 100, 500 and 1000. Three values for learning rate were chosen - 0.01, 0.05 and 0.1.

Using Grid Search it was found that the best parameters were number of estimators of 500 and learning rate of 0.05.

This combination resulted in a mean squared error of 29.2021 and accuracy of 0.7267. While it is an improvement on the Linear Regression model, the SVR model performs considerably better on this data set. Similar to SVR, XGBoost is also a black box. Given the iterative nature of the improving model, it is difficult to reverse engineer to find the impact of each feature.

Decision Tree

Like Linear Regression, Decision Trees are one of the older more established approaches. It also relies upon mathematics, in this case to eagerly determine the most appropriate splits at a particular level depending on the feature and its value.

For Grid Search the following values were used for the parameter.Five values were chosen - 2,4,6,8,10.

Using Grid Search it was found that the best parameter was minimum samples per leaf of 10, indicating less overfitting to the test data set.

This combination resulted in a mean squared error of 42.2984 and accuracy of 0.6041. This is significantly worse than even the Linear Regression approach. As mentioned previously, as Decision Trees are eager and make decisions at a local level, there is no guarantee that it is the globally correct decision and therefore can result in less accurate models. One advantage of Decision Trees is that they can explain the choices made.

From the graph of the decision tree shown in Appendix D.2, it can be seen that any prediction that is made starts with a decision on the value of the Yards Per Game (Y/G) feature. Any value for this over 66.167 will result in a higher grade, indicating that this is the most important feature in the data set for the Decision Tree. Other critical features include the Yards per Target (Y/Tgt), catch percentage, caught percentage, touchdowns, drops and the number of first downs they obtain. It is also interesting to note that the yards per game Rating is used multiple times down many of the branches in the tree. This tree is much deeper and wider than that seen with the QB, possibly as a result of considerably more records to split on.

Single Layer Neural Network

Single Layer Neural Network is the first neural network approach used in this evaluation. In this case only one layer is considered when training the model. Weights are randomly assigned on initiation of the model and a series of feed forward with back propagation update the weights of the neurons in the single layer of the network.

For Grid Search the following values were used for the parameters. Two optimizers were chosen - stochastic gradient descent and adam. They are both stochastic optimizers. For maximum iteration four values were chosen - 110,100,90 and 80. For activation two activation functions were chosen - Identity and Relu. The Identity activation function acts in a similar manner to Linear Regression, except per neuron. The Relu activation function is a popular activation function in deep neural networks and outputs the input if it is positive but sets all negative input to zero. For alpha 6 values were chosen 0.00001, 0.0001,0.001,0.01,0.1 and 0.5. For number of iterations with no change three values were chosen - 10,15,20. For the tolerance three values were chosen - 1e^{-5}, $1e^{-4}$, $1e^{-3}$.

Using Grid Search it was found that the best parameters were solver of sgd, maximum iterations of 90, activation function of identity, alpha of 0.001, number of iterations with no change of 20 and tolerance of 1e^{-5}.

The combination resulted in a mean squared error of 27.7681 and accuracy of 0.7401. This is comparable to the XGBoost approach with a similar mean squared error but marginally worse accuracy. Similar to the XGBoost approach, it is difficult to find the impact of each feature, although the coefficients and weights are available. However, as every input is connected to every neuron in the hidden layer this is a difficult task.

Deep Neural Network

The Deep Neural Network approach is very similar to the previous Single Layer Neural Network approach except that it uses multiple layers to discover patterns in the data to help improve prediction. Also a different technology is used to implement deep neural networks here. Keras with Tensorflow is used instead of SciKit Learn as it is one of the most common and well known frameworks for buildling deep learning networks.

For Grid Search the following values were used for the parameters. For number of neurons three values were chosen - 50, 80 and 100. For number of epochs Three values were chosen - 1000,3000 and 5000. Optimizer is the same as the solver in the previous approach and three values were chosen - sgd, adam and rmsprop. Rmsprop was not available in the previous approach, but provides an additional option. Activation is again the same as in the previous approach and the same values are chosen again, with Identity now known as linear. Adam was also used again. For drop out two values were chosen - 0.3 and 0.5.

Using Grid Search, it was found that the best parameters were number of neurons of 100, epochs of 5000, optimizer of rmsprop, activation of linear and dropout rate of 0.5.

The combination resulted in a mean squared error of 26.5896 and accuracy of 0.7511. This again is similar to the XGBoost approach except with slightly better results for both mean squared error and accuracy. As with the Single Layer Neural Network, the ability to explain how a prediction arose from it's inputs is a very difficult task.

The summary of the layers in the network can be seen in Figure 4.6.

```
Model: "sequential_10"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_40 (Dense)             (None, 80)                2400
_________________________________________________________________
dropout_30 (Dropout)         (None, 80)                0
_________________________________________________________________
dense_41 (Dense)             (None, 80)                6480
_________________________________________________________________
dropout_31 (Dropout)         (None, 80)                0
_________________________________________________________________
dense_42 (Dense)             (None, 80)                6480
_________________________________________________________________
dropout_32 (Dropout)         (None, 80)                0
_________________________________________________________________
dense_43 (Dense)             (None, 1)                 81
=================================================================
Total params: 15,441
Trainable params: 15,441
Non-trainable params: 0
_________________________________________________________________
```

Figure 4.6: WR Deep Neural Network Summary

4.2.2 Feature Extraction

As with the QB feature extraction, both Sequential Feature Selection (SFS) and Permutation Importance with eli5 will be used to identify significant features. Using the two approaches, significant features can be found, with those found in both gaining extra significance. Only the top features of each will be considered. In addition to this, a comparison will also be made to the features and their weights arising from both the Linear Regression and Decision Tree models. The top performing model - SVR - will be analysed for the most significant features.

Table 4.3: Significant Features for WR

Permutation Importance	**SFS**
first_downs	GS
Y/G	Y/Tgt
drops	Y/G
GS	player_game_count
Y/Tgt	first_downs
avoided_tackles	drops
penalties	fumbles
fumbles	avoided_tackles
longest	targeted_qb_rating
yards_after_catch	penalties

There are eight common significant features arising from the overlap of the two wrapper methodologies:

- first_downs
- Y/G
- drops
- GS
- Y/Tgt
- avoided_tackles
- penalties
- fumbles

First downs is the number of first downs that were achieved after a passing play to the player. Y/G is the number of yards gained by the WR through passing plays per game. The Drops feature is a measure of the number of times that the WR dropped a pass that they were a target of. Y/Tgt is the receiving yards per target to the WR. The number of tackles that the WR avoided is also recorded. It measures some of the elusiveness of the WR as it measures both the speed of the WR and their ability to run past players as well as how they can send defensive players in different directions just from their body movement. Penalties measures the number of penalties that were recorded on defensive players that were defending them. It shows how much defensive players need to illegally defend in order to try to minimize the effect of the WR. Finally, the number of fumbles recorded by the WR when they might fumble the ball after catching it - resulting in the opposition gaining possession of the ball.

4.2.3 Discussion

The table below summarizes the results found above.

Table 4.4: WR Model Comparisons

Model	MSE	Accuracy
SVR	25.4618	0.7617
Deep Neural Network	26.5896	0.7511
XGBoost	29.2021	0.7267
Single Layer Neural Network	27.7681	0.7401
Linear Regression	26.4427	0.7525
Decision Tree	42.2984	0.6041

As with QB position, SVR is again the best performing approach. However, now the gap to the next best performing approach is significantly less. In this case both Neural Network approaches are close to achieving the same results as SVR. Interestingly, so is the Linear Regression approach, and this is particularly interesting as there is a level of explainability to the Linear Regression approach. This is in contrast to the QB results. Similar to the QB results, the Decision Tree approach is the worst performing. XGBoost reduces the gap with SVR but not to the same extend as the other approaches. SVR is clearly once again the best performing approach, but the Deep Neural Network approach appears to be a consistent second best performer. There are significantly more records to learn on in this data set so it is interesting to see that the gap in approaches close.

Each of the approaches performed Grid Search with 10 Fold Cross Validation. As such each had different parameters to tune and optimize. Where possible techniques such as Drop out, Regularization and early stopping were included to mitigate against overfitting. Many of the starting points for the parameters used were achieved through manual trial and error and this can be seen in the difference in parameter values for similar parameters for Single Layer Neural Network and Deep Neural Network. Indeed in these cases trade-offs were needed to address learning time. GPUs help speed up the training, but resources can be limited in usages of GPUs.

The difference in performance of each approach depending on the parameters used in the Grid Search with Cross Validation can be seen in the following box plots of Figures 4.7 and 4.8.

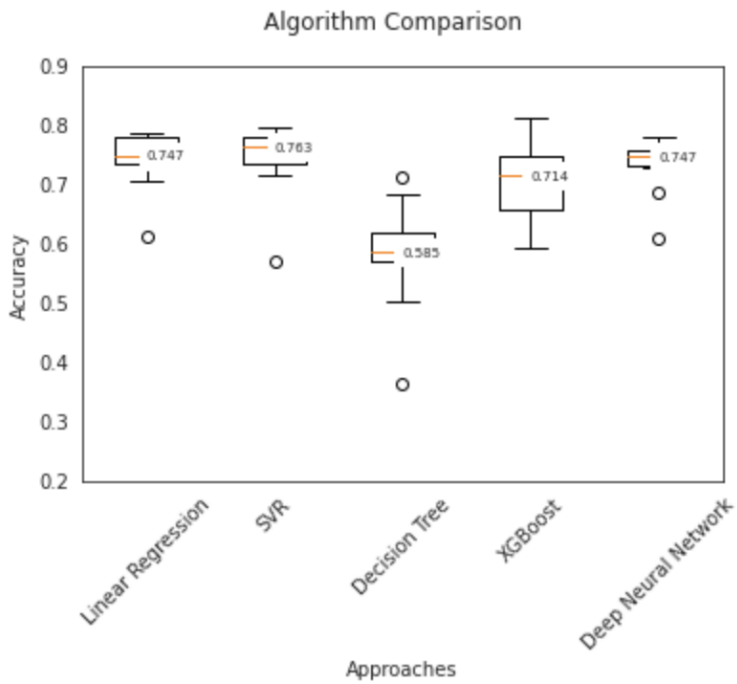


Figure 4.7: WR Best Performing Grid Search Parms per Approach

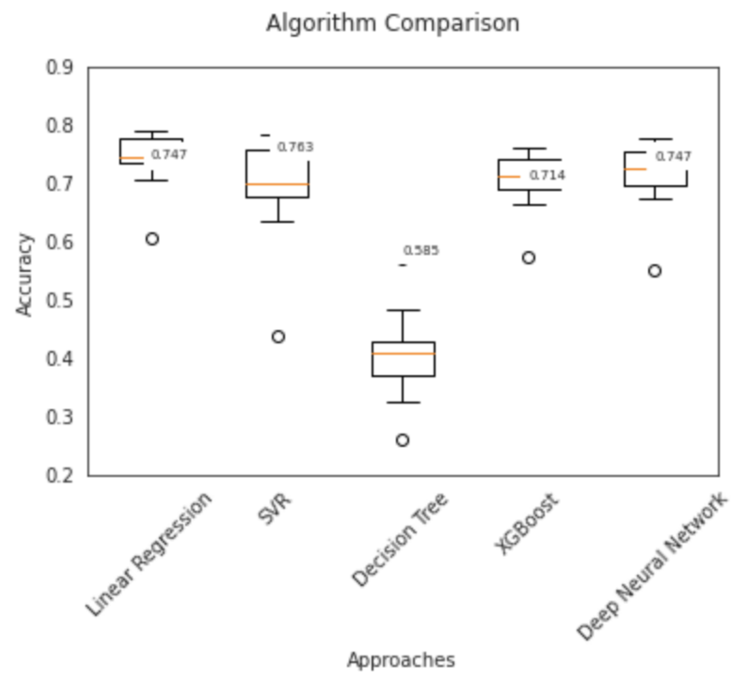


Figure 4.8: WR Worst Performing Grid Search Parms per Approach

The box plots in Figures 4.7 and 4.8 above represent the best and worst performing parameters for each approach using Grid Search Cross Validation. Each box plot represents the 10 accuracy numbers as achieved using 10 Fold Cross Validation for that parameter set for the accuracy metric. The values for the Single Layer Neural Network approach could not be shown at time of writing. They do highlight some interesting observations.

The first interesting observation relates to the difference between best and worst performances for each approach. The Decision Tree remains the worst performing in this approach. On this occasion, however, the SVR approach doesn't experience as big a decrease in it's worst performing approach. There is still a noticeable decrease however with the other approaches appearing to perform better in their worst performance. Again, Linear Regression and Deep Neural Network are relatively consistent. XGBoost again is consistent, but in it's best performing run, it sees a wider spread of results, including once again the highest individual result on one of it's cross validation runs.

Again the ranges are of interest. The ranges for the four main approaches, Linear Regression, Deep Neural Network, XGBoost and SVR all appear to be relatively similar. The XGBoost approach does appear to be lower down on it's best performing run. While having a wider range and the highest prediction in one of it's cross validation runs, it appears to sit slightly lower than the other three. This is also represented in the statistical tests where a statistical difference was found between the XGBoost approach and the other three. This was not found in the QB approach. This is despite the fact that the accuracy is only a few percentage points away from them. Decision Trees again are statistically different from the rest and can be found not to be preferment.

Despite achieving the highest accuracy and lowest mean squared error across the approaches, there is a marked difference in accuracy between the SVR prediction and that of the grade given by PFF. This indicates that there could be many more optimizations that could be made. However, additional data points might also add additional patterns that could be observed within the data. PFF use many analysts to look at plays in the game and to give scores based upon the outcome (as defined in the statistics) as well as situational evaluations. This situational awareness provides a level of detail that the data used here does not provide. For example, a wide receiver might run a well run route, and lose their defender, but the QB may overthrow the receiver and not give them a chance to catch it. However, this will go down as a negative statistic for the WR. Similarly with the statistics all catches are equal, whereas with PFF a difficult contested catch would show up better for the WR than an easy catch.

SVR outperformed the other approaches - especially those that can give some indication as to how they came to a prediction. To tackle this some wrapper method techniques were applied to the SVR as it was the best performing approach. Two main patterns appear when looking at the significant features that emerge. The first is that production is important for wide receivers. Number of yards gained shows up twice - per game and per target. The further the WR can get down the field and win catches ahead of the defenders marking them the higher the grade. The ability of the WR to deliver a new first down and ability to avoid tackles to get extra yards, shows how catching is only part of the equation. The ability to make plays and gain extra yards is also important. The second pattern that can be seen is how accuracy of the basics of the wide receiver is also important. The basics of catching the ball and holding onto it are important and can be seen by the presence of drops and fumbles in the significant features.

It is also interesting to compare the significant features from the SVR to those explained by the Decision Tree and Linear Regression models. For the Linear Regression model first downs, Y/G, avoided tackles, drops, GS and yards per target were the features that overlapped with both wrapper methods. This is significantly more than for the QB approach. This again highlights the first pattern in relation to production as all the features in common are in relation to producing more yards and avoiding tackles. For Decision Trees, they similarly have four features in common with both wrapper methods - Y/G, Y/Tgt, First Downs and Drops. Again with Decision Trees, production shows prominently with other features also prevalent such as yards after catch and yards after catch per reception. This highlights the importance of production in the evaluation of wide receivers.

4.3 Running Back Results

The following models were run on the RB data allowing for the models to be compared and the most significant features extracted.

4.3.1 Model Results and Evaluation

As in the previous sections, Sections 4.1 and 4.2, 5 different approaches were taken to predict the PFF ranking of the wide receiver for a given year. In order to achieve the best result, each approach will use Grid Search to obtain the best parameters from a range of parameters provided. Each approach will also use 10-Fold Cross Validation to help reduce overfitting, allow for all the data to be used and to get more metrics by which to evaluate the approaches. Each approach will also apply standard scaling to the input features - scaling to unit variance for each feature.

Linear Regression

The first approach that was evaluated was Linear Regression. This approach gives a mathematical calculation across all the 50 parameters in the input feature data set. As well as an intercept value, the Linear Regression algorithm learns an appropriate weight for each feature.

For Grid Search the following values were used for the parameters.In this case the Alpha values of 0.5, 0.8 and 1.0 were used as the regularization parameter (R). Therefore the values range from no regularization (0.5) to regularization of a half (1.0). The other parameter is the solver. This represents the computational functionality used in determining the target. Two solvers were tested - lsqr (least squares resolver) and sag (stochastic average gradient). Both are commonly used solvers.

Using Grid Search it was found that the best parameters were an alpha of 1.0 and lsqr was the best solver. This combination resulted in a mean squared error of 47.3491 and an accuracy of 0.5460.

Use of a higher alpha would be recommended, to see at what point too much regularization affects the result in a negative sense. The accuracy of 0.5460 indicates that there is plenty of work to be done to achieve a similar grade as that provided by PFF. This will be discussed later.

From the weights learned by the model, it can be seen that the feature that most positively affects the grade is avoided tackles rushing (avoided_tackles_rushing).

```
'Age': -0.14776468645548493,
'Ctch%': 0.9018535421485782,
'Fmb': 0.10279553250076776,
'G': 0.46971804591528576,
'GS': -2.2352622549921866,
'R/G': 0.27656239404126154,
'Y/G': 5.199009290665431,
'Y/R': -4.309360608012859,
'Y/Tgt': 1.4255474222504088,
'Year': -0.7240881283737494,
'allPro': -0.06018830408612927,
'avoided_tackles': 1.940981364865342,
'caught_percent': 0.17286963657585527,
'declined_penalties': -0.19719759580061838,
'drops': -2.4116396436032472,
'first_downs': 9.085711000744823,
'fumbles': -0.9036697556710078,
'interceptions': 0.29689445770186784,
'longest': -0.5576861353092278,
'penalties': -0.3460820867031004,
'player_game_count': -0.4445111070382823,
'proBowl': 0.4610273234479633,
'receptions': -1.5074630129014994,
'targeted_qb_rating': 1.2058557213380325,
'touchdowns': -0.3385374118248813,
'yards': -1.4527269133871714,
'yards_after_catch': -1.4360126082493625,
'yards_after_catch_per_reception': -0.32098515868808003,
'yards_per_reception': 5.152331164233034}
```

Figure 4.9: RB Linear Regression Coefficients

Each of the coefficients in Figure 4.9 matches to each of the input features shown. For any single prediction, the value for each of the features are multiplied by their corresponding coefficient and they are all added together. The result is added to the intercept of 65.4584 to predict the grade/target value. Some of the top features to positively affect the prediction include yards per game, first downs, yards per reception amongst others. If the input for a prediction provided 0 for all values, the model would produce a better grade of approx. 65.4584 versus the 54.60 achieved on average.

SVR

SVR was the next approach evaluated. This approach again uses mathematical calculations to arrive at a prediction from the input feature set.

For Grid Search the following values were used for the parameters. Four values were tried for C in the Grid Search - $1e^0, 1e^1, 1e^2, 1e^3$.

Five kernel coefficients were chosen for Gamma - $2^{-9}, 2^{-7}, 2^{-5}, 2^{-3}, 2^{-1}$.

Five values for the tolerance were chosen - $1e^{-5}, 1e^{-4}, 1e^{-3}, 1e^{-2}, 1e^{-1}$.

Using Grid Search it was found that the best parameters were a C of 100, gamma of 0.001953125 and tol of 0.1.

This combination resulted in a mean squared error of 42.3328 and accuracy of 0.5941. This is an improvement on the results returned by Linear Regression. However, SVR doesn't expose the importance of features within the model - it is a black box.

XGBoost

As described previously, XGBoost is a modern implementation of gradient boosting, using ensembles of decision trees to progressively build stronger and stronger models based upon regression error.

For Grid Search the following values were used for the parameters. Four values for number of estimators were chosen - 50, 100, 500 and 1000. Three values for learning rate were chosen - 0.01, 0.05 and 0.1.

Using Grid Search it was found that the best parameters were number of estimators of 1000 and learning rate of 0.01.

This combination resulted in a mean squared error of 43.5915 and accuracy of 0.5820. While it is an improvement on the Linear Regression model, the SVR model performs better on this data set. Similar to SVR, XGBoost is also a black box. Given the iterative nature of the improving model, it is difficult to reverse engineer to find the impact of each feature.

Decision Tree

Like Linear Regression, Decision Trees are one of the older more established approaches. It also relies upon mathematics, in this case to eagerly determine the most appropriate splits at a particular level depending on the feature and its value.

For Grid Search the following values were used for the parameter.Five values were chosen - 2,4,6,8,10.

Using Grid Search it was found that the best parameter was minimum samples per leaf of 10, indicating less overfitting to the test data set.

This combination resulted in a mean squared error of 69.9093 and accuracy of 0.3297. This is significantly worse than even the Linear Regression approach. As mentioned previously, as Decision Trees are eager and make decisions at a local level, there is no guarantee that it is the globally correct decision and therefore can result in less accurate models. One advantage of Decision Trees is that they can explain the choices made.

From the graph of the decision tree shown in Appendix D.3, it can be seen that any prediction that is made starts with a decision on the value of the Yards Per Attempt (ypa) feature. Any value for this over 65.409 will result in a higher grade, indicating that this is the most important feature in the data set for the Decision Tree. Other critical features include the first downs receiving (first_downs_receiving), avoided tackles rushing, yards after contact per attempt and yards after catch. This tree is much deeper and wider than that seen with the QB, possibly as a result of considerably more records to split on.

Single Layer Neural Network

Single Layer Neural Network is the first neural network approach used in this evaluation. In this case only one layer is considered when training the model. Weights are randomly assigned on initiation of the model and a series of feed forward with back propagation update the weights of the neurons in the single layer of the network.

For Grid Search the following values were used for the parameters. Two optimizers were chosen - stochastic gradient descent and adam. They are both stochastic optimizers. For maximum iteration four values were chosen - 110,100,90 and 80. For activation two activation functions were chosen - Identity and Relu. The Identity activation function acts in a similar manner to Linear Regression, except per neuron. The Relu activation function is a popular activation function in deep neural networks and outputs the input if it is positive but sets all negative input to zero. For alpha 6 values were chosen 0.00001, 0.0001,0.001,0.01,0.1 and 0.5. For number of iterations with no change three values were chosen - 10,15,20. For the tolerance three values were chosen - 1e^{-5}, $1e^{-4}, 1e^{-3}$.

Using Grid Search it was found that the best parameters were solver of sgd, maximum iterations of 80, activation function of identity, alpha of 0.001, number of iterations with no change of 20 and tolerance of 0.001.

The combination resulted in a mean squared error of 47.2606 and accuracy of 0.5469. This is comparable to the Linear Regresssion approach with a similar mean squared error but marginally better accuracy. Similar to the XGBoost approach, it is difficult to find the impact of each feature, although the coefficients and weights are available. However, as every input is connected to every neuron in the hidden layer this is a difficult task.

Deep Neural Network

The Deep Neural Network approach is very similar to the previous Single Layer Neural Network approach except that it uses multiple layers to discover patterns in the data to help improve prediction. Also a different technology is used to implement deep neural networks here. Keras with Tensorflow is used instead of SciKit Learn as it is one of the most common and well known frameworks for buildling deep learning networks.

For Grid Search the following values were used for the parameters. For number of neurons three values were chosen - 50, 80 and 100. For number of epochs Three values were chosen - 1000,3000 and 5000. Optimizer is the same as the solver in the previous approach and three values were chosen - sgd, adam and rmsprop. Rmsprop was not available in the previous approach, but provides an additional option. Activation is again the same as in the previous approach and the same values are chosen again, with Identity now known as linear. Adam was also used again. For drop out two values were chosen - 0.3 and 0.5.

Using Grid Search, it was found that the best parameters were number of neurons of 100, epochs of 5000, optimizer of rmsprop, activation of linear and dropout rate of 0.5.

The combination resulted in a mean squared error of 46.9450 and accuracy of 0.5499. This again is similar to the Linear Regression approach except with slightly better results for both mean squared error and accuracy. As with the Single Layer Neural Network, the ability to explain how a prediction arose from it's inputs is a very difficult task.

The summary of the layers in the network can be seen in Figure 4.10.

```
Model: "sequential_10"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_40 (Dense)             (None, 80)                2880
_________________________________________________________________
dropout_30 (Dropout)         (None, 80)                0
_________________________________________________________________
dense_41 (Dense)             (None, 80)                6480
_________________________________________________________________
dropout_31 (Dropout)         (None, 80)                0
_________________________________________________________________
dense_42 (Dense)             (None, 80)                6480
_________________________________________________________________
dropout_32 (Dropout)         (None, 80)                0
_________________________________________________________________
dense_43 (Dense)             (None, 1)                 81
=================================================================
Total params: 15,921
Trainable params: 15,921
Non-trainable params: 0
_________________________________________________________________
```

Figure 4.10: RB Deep Neural Network Summary

4.3.2 Feature Extraction

As with the QB and WR feature extractions, both Sequential Feature Selection (SFS) and Permutation Importance with eli5 will be used to identify significant features. Using the two approaches, significant features can be found, with those found in both gaining extra significance. Only the top features of each will be considered. In addition to this, a comparison will also be made to the features and their weights arising from both the Linear Regression and Decision Tree models. The top performing model - SVR - will be analysed for the most significant features.

Table 4.5: Significant Features for RB

Permutation Importance	**SFS**
avoided_tackles_rushing	Y/A
fumbles_rushing	Y/G
ypa	Year
first_downs_receiving	ypa
drops	yco_attempt
Y/G	avoided_tackles_rushing
GS	fumbles_rushing
avoided_tackles_receiving	first_downs_receiving
yco_attempt	targeted_qb_rating
touchdowns_rushing	drops

There are seven common significant features arising from the overlap of the two wrapper methodologies:

- avoided_tackles_rushing
- Y/G
- fumbles_rushing
- ypa
- first_downs_receiving
- drops

- yco_attempt

Avoided tackles for rushing is the number of tackles that the RB managed to avoid while in a rushing play. Y/G is the number of yards per game that the RB managed to obtain from rushing plays. Fumbles rushing represents the number of times that the RB fumbled the ball while in a rushing play. This could be from losing the ball in a tackle where a defensive player may knock it from their grasp, or it could also be a poor hand off between the QB and the RB. Ypa represent yards per attempt that the RB achieved. First Downs Receiving is a measure of the number of First Downs that the player achieves as a result of a passing play to the player. The number of drops in a passing play is also recorded. Finally, yco_attempt represents the yards after contact that the RB managed to achieve per attempt. It represents how productive the RB can be despite having momentum stalled a little as a result of contact from a defensive player.

4.3.3 Discussion

The table below summarizes the results found above.

Table 4.6: RB Model Comparisons

Model	MSE	Accuracy
SVR	42.3328	0.5941
Deep Neural Network	46.9450	0.5499
XGBoost	43.5915	0.5820
Single Layer Neural Network	47.2606	0.5469
Linear Regression	47.3491	0.5460
Decision Tree	69.9093	0.3297

As with the QB and WR positions, SVR is the best performing approach. In this instance XGBoost is a close second with only approx. 1% in the difference. Similar to the WR position, Linear Regression and the two Neural Network approaches offer similar approaches, but are all clearly not performing as well as SVR or XGBoost. Decision Tree as seen in the previous approaches is clearly the worst performing. In this case the number of records for RB sits in between the number of records for

WR and QB and for the neural networks approaches the accuracy has regressed. The accuracy for RB appears to be significantly less than those for the other approaches.

Each of the approaches performed Grid Search with 10 Fold Cross Validation. As such each had different parameters to tune and optimize. Where possible techniques such as Drop out, Regularization and early stopping were included to mitigate against overfitting. Many of the starting points for the parameters used were achieved through manual trial and error and this can be seen in the difference in parameter values for similar parameters for Single Layer Neural Network and Deep Neural Network. Indeed in these cases trade-offs were needed to address learning time. GPUs help speed up the training, but resources can be limited in usages of GPUs.

The difference in performance of each approach depending on the parameters used in the Grid Search with Cross Validation can be seen in the box plots in Figures 4.11 and 4.12.

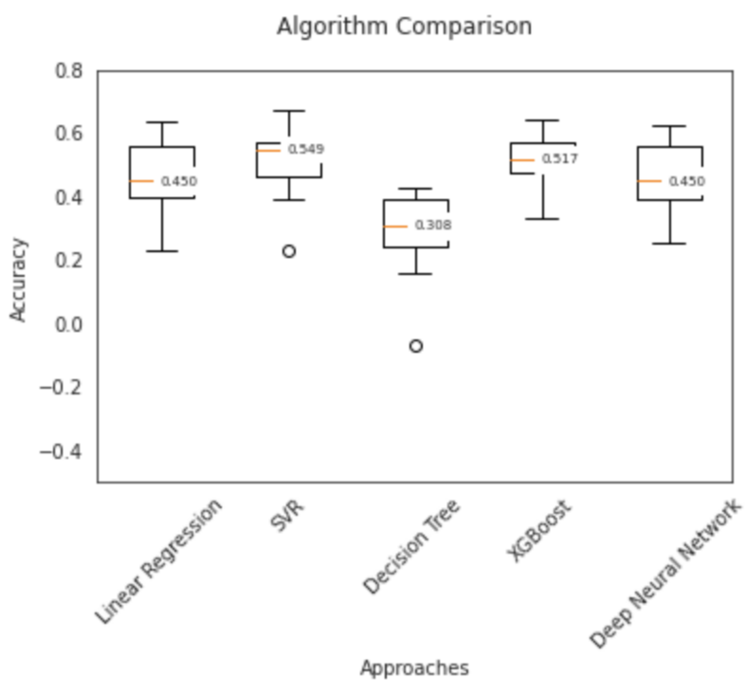


Figure 4.11: RB Best Performing Grid Search Parms per Approach

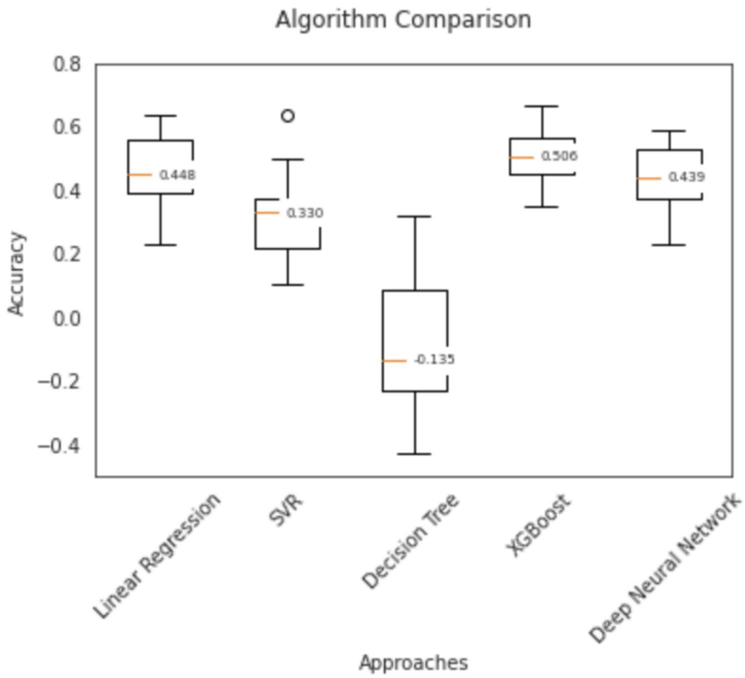


Figure 4.12: RB Worst Performing Grid Search Parms per Approach

The box plots in Figures 4.11 and 4.12 above represent the best and worst performing parameters for each approach using Grid Search Cross Validation. Each box plot represents the 10 accuracy numbers as achieved using 10 Fold Cross Validation for that parameter set for the accuracy metric. The values for the Single Layer Neural Network approach could not be shown at time of writing. They do highlight some interesting observations.

The first observation is similar to that observed in the QB and WR positions.Linear Regression, XGBoost and Deep Neural Networks didn't actually change a lot in comparison to the changes in SVR and Decision Tree. While there were differences they weren't as big as those experienced by SVR and Decision Trees. Once again it was observed that the SVR was very susceptible to changes in the hyper-parameters tuning it.

The next observation is in relation to the differences between the best performing approaches themselves. While the top 4 all appear to be around the same area a few things are noticeable. Both of the highest performing approaches, SVR and XGBoost, have tight interquartile ranges in comparison to Linear Regression and Deep Neural

Networks. Also their medians appear to be noticeably higher. This is reflected in the tests for statistical significance, where there is a statistically significant difference between the two groups. There is no statistically significant difference between XG-Boost and SVR or between Linear Regression and Deep Neural Network. However, there is a difference between the two, showing that XGBoost and SVR do provide a statistically better prediction for this position group. Decision Trees are statistically different (and worse) than all of the other approaches. Statistical Significance in this case is a p-value of less than 0.05.

Despite achieving the highest accuracy and lowest mean squared error across the approaches, there is a marked difference in accuracy between the SVR prediction and that of the grade given by PFF. This indicates that there could be many more optimizations that could be made. However, additional data points might also add additional patterns that could be observed within the data. PFF use many analysts to look at plays in the game and to give scores based upon the outcome (as defined in the statistics) as well as situational evaluations. This situational awareness provides a level of detail that the data used here does not provide. For example, for RB a rushing gain of 10 yards may not be the same for every run or for every team. For a team with a very good Offensive Line (group of Offensive players who try to make a path for the running back), big gaps may appear in the defense where it may be easier for a RB to make 10 yards. In contrast, for a RB on a team with a poor Offensive Line, 10 yards may involve more skill and/or power. The features collected tries to observe some of this through features such as avoided tackles and yards after contact. However, that doesn't tell the entire story. PFF analysts may be able to grade based upon more contextually aware information.

SVR outperformed the other approaches - especially those that can give some indication as to how they came to a prediction. To tackle this some wrapper method techniques were applied to the SVR as it was the best performing approach. Two main patterns appear when looking at the significant features that emerge. Similar to WR it comes down to production after receiving the ball and sound fundamentals of receiving the ball either through rushing or pass reception. Again production is to the forefront with yards per game, yards per attempt showing up on both approaches. Very similar to the WR first downs from passes and avoided tackles are key indicators with RB as well as yards after contact metrics. The ability of the RB to avoid tackles and make yards after a contact show the difference between the elite RBs and other RBs. What it also shows is that there is a passing element to the RBs game that is important despite the traditional role of just running the ball from a hand off from the QB. As with the WR position, sound fundamentals such as carrying and catching the ball are also key. Increased drops and fumbles result in poorer grades and have negative effects on the team as well as the grading of the RB.

Finally, interesting insights can be gained from comparing to the explanations given from the Decision Trees as well as the Linear Regression approaches. For the Linear Regression approach all the overlapping features identified by the wrapper methods above arise. It is interesting that they overlap, indeed the Linear Regression approach overlaps with each of the models on other features such as yards after catch with permutation importance and target QB rating for SFS. From the Decision Tree point of view it is similar. Six of the seven features that are common between the two wrapper methods are in common with the Decision Tree decisions. For RBs it appears as though the the different methods are much more closely aligned than for previous positions. However, it is also noticeable that the accuracy is significantly less than for the other two positions. Possibly the approaches such as SVR, Neural Networks, etc. are not finding as many patterns in the data to decide on. As with the other positions, additional data around the individual plays may provide greater insight and accuracy.

4.4 Results Summary

Tables 4.7, 4.8, 4.9, 4.10, 4.11 and 4.12 below summarise the results obtained by the various approaches. An overall discussion of the results obtained as it related to each position is discussed in sections 4.1.3, 4.2.3 and 4.3.3, but what the results below show is that the WR position consistently achieves higher accuracy and lower MSE than the other two approaches, with QB performing better than RB. With 1,877 records, the WR position presents with far more records than in either of the other two positions - 311 for QB and 1,141 for RB. With 50 records, the QB position presents with far more records than in either of the other two positions - 29 for WR and 35 for RB. With the greatest number of records, the WR position is able to iterate and adapt more often to get closer to the target. While QB having more features also increases it's accuracy. If more records and more quality features were available, then perhaps better accuracy could be achieved.

Table 4.7: SVR Model Comparisons

Position	**MSE**	**Accuracy**
QB	32.126725	0.742643
WR	25.4618	0.7617
RB	42.3328	0.5941

Table 4.8: Deep Neural Network Model Comparisons

Position	**MSE**	**Accuracy**
QB	38.268055	0.693447
WR	26.5896	0.7511
RB	46.9450	0.5499

Table 4.9: XGBoost Model Comparisons

Position	**MSE**	**Accuracy**
QB	38.427732	0.692168
WR	29.2021	0.7267
RB	43.5915	0.5820

Table 4.10: Single Layer Neural Network Model Comparisons

Position	MSE	Accuracy
QB	38.595139	0.690827
WR	27.7681	0.7401
RB	47.2606	0.5469

Table 4.11: Linear Regression Model Comparisons

Position	MSE	Accuracy
QB	40.114897	0.678653
WR	26.4427	0.7525
RB	47.3491	0.5460

Table 4.12: Decision Tree Model Comparisons

Position	MSE	Accuracy
QB	57.490511	0.539462
WR	42.2984	0.6041
RB	69.9093	0.3297

4.5 Summary

In this chapter the results of the experiment were presented. Initially the results were presented from the point of view of the approaches used - comparing the approaches across each position and how the data sets may influence the results. The results were then discussed and compared on a position by position basis, so as to compare the performance of each approach with respect to the other approaches for the position. In the next chapter, an overview of the experiment and problem definition will be presented. Further analysis of the results will be conducted and possible future work and approaches will be presented.

Chapter 5

Conclusion

Following the results presented in chapter 4, this chapter will finish up with a review of the observed research in section 5.1, followed by a discussion on the problem definition in section 5.2. Further discussion of the results and the impact of the results are discussed in sections 5.3 and 5.4. Finally suggestions for improving the experiment and suggestions for alternative approaches are presented in section 5.5.

5.1 Research Overview

The main objectives of this research was to evaluate the performance of NFL players, in particular QB, WR and RB using a number of different approaches and observing whether a neural network approach would perform to the same level as other approaches. In previous research, researchers have used a variety of different approaches in the evaluation of player evaluation a wide range of sports. In the NFL neural networks haven't been used to date with researchers favouring tradition methods such as regression, decision trees, SVM, etc. (Byanna & Klabjan, 2016; Porter, 2018; Devarakonda & Colson, 2019). However, in other sports neural networks approaches were investigated (Oytun et al., 2020). The ability of neural networks to evaluate player performance in comparison to other approaches was a focus of this research. Another focus of the research was to discover significant features in the evaluation of a player for a given position. Using wrapper methods to determine the most significant features for a position like Nsolo (Nsolo et al., 2019) allowed for insight into black box approaches such as SVR.

5.2 Problem Definition

Current evaluation of players by well known NFL Analytics company PFF is done using a combination of statistics as well as analysis performed by analysts. While there are strict controls and quality control in use by PFF, there is always the human element involved. This study looks at regression evaluation using variety of ML supervised learning techniques, most notably to see if neural network approaches stack up to other approaches in evaluating player performance. While providing an evaluation as a number can be useful, the reasoning behind the evaluation is key for different stakeholders such as owners and players during contract negotiations. To this end, discovery of the most significant features for the most performant approach is key.

Based on the results of the experiment, SVR is the most performant approach, exceeding the values obtained by XGBoost and Neural Network. While the Neural Network approaches don't outperform the SVR approach, they exceed the performance of other approaches in the majority of cases - except for one position against XGBoost. Even then SVR is not close to the evaluations as determined by PFF.

5.3 Design/Experimentation, Evaluation & Results

To answer the proposed research question, data was obtained from two main sources - PFR and PFF, using the offensive grade from PFF as the target variable. Because the data comes from multiple sources, it needs to be merged together with duplicated and highly correlated features removed. Six different approaches were used in evaluating the performance of a player given their yearly data. Among the six different approaches were two neural network approaches which were being evaluated against the other approaches as feasible approaches to the problem of evaluating player performance. There are a considerable number of positions in an NFL team and as a result three were chosen to provide a representation of how the approaches would work - they were QB, WR and RB. Each would have their own data sets, with some position specific features, which would be used in developing the models for each position.

For each approach GridSearchCV was used to get the best hyper-parameters and to get a relatively accurate evaluation of each approach by using the entire data set. This was done for each of the three positions. Having found the optimal hyper-parameters, it was discovered that the SVR approach was the best performing across all three positions (Accuracy: 74% for QB, 76% for WR and 59% for RB). Not only did it always produce the best RSME and accuracy, but also was statistically better that all other approaches in at least one position. SVR allows for non-linear patterns to be mapped into a linear plane. Comparing the performance of neural network approaches against SVR, they came close in terms of accuracy in a number of the tests and were only not statistically worse from a p-value in one position. It can be seen that Neural Networks, in this data set, do not perform as well as SVR, but are comparable to XGBoost, slightly outperforms Linear Regression and significantly outperforms Decision Trees.

As SVR is a black box in terms of explainability, additional work was performed to obtain the most significant feature involved in the evaluation of a players performance. This can be important when players look back at where they could possibly improve. Similar to the approach by Nsolo et al (Nsolo et al., 2019), wrapper method techniques were used to identify the significant features.The significant features were consistent in terms of two clusters of features. The first group of features in each position was in relation to the ability of the player to make positive impacts when gaining the ball. Examples include the ability to get a first down on the play, avoided tackles after receiving the ball from a pass or as part of a rush, as well as the amount of yards achieved after contact was made (particularly relevant for RBs). The other group of features that are prevalent surround the ability of the player to perform the fundamentals to a high level. High impact plays do occur but do not necessarily happen on every play. As a result, players with a high level of consistency in the fundamentals coupled with the ability to make big, high impact plays evaluate best.

5.4 Contributions and Impact

From the evaluation, it can be seen that the SVR approach outperformed all the other approaches for this data set. However, it also shows that neural networks still outperform most of the other approaches and would be a viable option if the SVR approach was not available. The data sets did not contain large amounts of records for the QB position and there was a significant difference in accuracy. However, as larger numbers of records were observed for the WR position, accuracy metrics were observed that were a lot closer to the SVR model. It would, therefore, be interesting to observe the results when applied to a larger data set, for example, on a football (soccer) data set where there are thousands of players as opposed to the smaller number of professional NFL players.

It can be seen that the accuracy values for the evaluations for all the approaches were considerably less than the actual PFF evaluations, with the highest accuracy being SVR for WR and it gave an accuracy of 76%. Given that the data for the evaluation came from two data sets that collated data for the year, this difference may not be surprising. The PFF evaluation involves analysts which grade the performance of the player for a particular play and so works on more granular play-by-play data as opposed to the data collated over the year. In addition to this, the PFF analysts are able to take into account aspects of the play that is not currently recorded in the available data. Such information could include the following:

- For RB and QB, how effective was the offensive line?
- Did the offensive line make space for the RB to run through - not all 10 yard runs are equal?
- For QB, did the offensive line protect the QB from tackles, allowing the QB to have more time to make a throw?
- For WRs, was the catch a contested catch or did the WR have a lot of room and little pressure when catching the ball?

- Did a QB throw a very good pass to a WR, only for the WR to fumble the catch, which results in an interception for the opposing team?

These are just some of the aspects to a play that are currently not represented in the data used in this evaluation. If such data could be recorded, or indeed learned by an ML algorithm, then a more accurate model could be constructed.

5.5 Future Work & Recommendations

In the current implementations, reduction of dimensions using methods were not used. Such methods include the two wrapper methods discussed previously in the Feature Extraction section as well as PCA. While the two wrapper methods were used, they weren't used as dimension reduction. The reason was so that the most significant features could be obtained from the entire set where possible (without duplicates or high correlation). To further increase the accuracy, dimension reduction could be implemented.

As mentioned in the previous section, Contributions and Impact, there is a lot of additional information that is not captured in the data sets used and the volume of records on which to learn could be increased. In order to further evaluate the neural network implementations against the other approaches, it would be interesting to implement them against a larger data set, such as European professional soccer players.

To address the shortfall in additional information there are a number of options to investigate. PFR has some additional advanced metrics that it has been recording, but only for the last 3 years of data. If there were additional years worth of data then it would be interesting to include those features and see if that would help the accuracy. Similarly, PFF has some additional metrics that it records and fills out for previous years. Using that additional data would be an interesting investigation. It should be noted that with additional data and comes the danger of overfitting so this would need to be considered.

The current data set covers the data for a particular year. Additional data sets that investigate game-by-game data (which are now available on PFF) could give greater granularity and accuracy, as it can show where a player may be consistent throughout the season, where a player may be excellent for one or two games but mediocre for the others, or perhaps it could show a pattern of a particular teams defense being very good, so a good performance against that team may result in a higher evaluation. Some black box approaches such as SVR and Neural Networks can be good when applied to such non-linear data where patterns may need to be uncovered.

It would be ideal if the same information that was available for the analysts was available for the machine learning. An approach similar to Liu et al (Liu & Schulte, 2018; Liu et al., 2020, 2019) could be used. This works on play-by-play data to evaluate player performance by using reinforcement learning to compute the Q function. This in turn could measure the impact the player had on the play. Indeed no such data source could be found in the research to date. What could also be possible would be to use deep learning techniques to gather relevant play-by-play data from video of each play to produce such a data set and use the approach mentioned above to evaluate the players impact on the game. This would be an interesting contrast to the analyst grades from PFF.

In terms of the most significant features, analysis could be performed on smaller subsets of data such as contrasting the significant features from the last 5 years versus the previous 5 years. The data set that is currently available may be too small to provide a good evaluation, so using it in combination with one of the approaches above may provide a better insight into the most relevant features in the current game.

References

Aalbers, B., & Van Haaren, J. (2019). Distinguishing Between Roles of Football Players in Play-by-Play Match Event Data. In U. Brefeld, J. Davis, J. Van Haaren, & A. Zimmermann (Eds.), *Machine Learning and Data Mining for Sports Analytics* (pp. 31–41). Cham: Springer International Publishing. doi: 10.1007/978-3-030-17274-9_3

Aha, D. W., & Bankert, R. L. (1996). A Comparative Evaluation of Sequential Feature Selection Algorithms. In D. Fisher & H.-J. Lenz (Eds.), *Learning from Data: Artificial Intelligence and Statistics V* (pp. 199–206). New York, NY: Springer. Retrieved 2021-06-05, from `https://doi.org/10.1007/978-1-4612-2404-4_19` doi: 10.1007/978-1-4612-2404-4_19

Breiman, L. (2001, October). Random Forests. *Machine Learning*, *45*(1), 5–32. Retrieved 2021-06-05, from `https://doi.org/10.1023/A:1010933404324` doi: 10.1023/A:1010933404324

Breiman, L., Friedman, J., Stone, C., & Olshen, R. (1984). *Classification and Regression Trees*. Taylor & Francis. Retrieved from `https://books.google.ie/books?id=JwQx-WOmSyQC`

Bunker, R. P., & Thabtah, F. (2019, January). A machine learning framework for sport result prediction. *Applied Computing and Informatics*, *15*(1), 27–33. Retrieved 2020-12-14, from `http://www.sciencedirect.com/science/article/pii/S2210832717301485` doi: 10.1016/j.aci.2017.09.005

Byanna, N., & Klabjan, D. (2016, March). *Evaluating the Performance of Offensive Linemen in the NFL.* Retrieved 2020-03-17, from `https://deepai.org/`

`publication/evaluating-the-performance-of-offensive-linemen-in-the-nfl` (Library Catalog: deepai.org)

Chang, J., & Zenilman, J. (2013). *A Study of Sabermetrics in Major League Baseball : The Impact of Moneyball on Free Agent Salaries.* Retrieved 2021-06-12, from `/paper/A-Study-of-Sabermetrics-in-Major-League-Baseball-%3A-Chang-Zenilman/e7900907fcadfa4ae1652dcbaf2d53b5a38eaa86`

Cortes, C., Mohri, M., & Rostamizadeh, A. (2009). ¡i¿l¡/i¿¡sub¿2¡/sub¿ regularization for learning kernels. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence* (p. 109–116). Arlington, Virginia, USA: AUAI Press.

David, J., Pasteur, R., Ahmad, M., & Janning, M. (2011, January). NFL Prediction using Committees of Artificial Neural Networks. *Journal of Quantitative Analysis in Sports*, *7*, 9–9. doi: 10.2202/1559-0410.1327

Devarakonda, D. S., & Colson, M. (2019). *Predicting NFL Wide Receiver Performance using Machine Learning.*

Feng, S., Zhou, H., & Dong, H. (2019, January). Using deep neural network with small dataset to predict material defects. *Materials & Design*, *162*, 300–310. Retrieved 2021-05-17, from `https://www.sciencedirect.com/science/article/pii/S0264127518308682` doi: 10.1016/j.matdes.2018.11.060

Friedman, J. H. (2001). Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics*, *29*(5), 1189–1232. Retrieved 2021-05-30, from `https://www.jstor.org/stable/2699986` (Publisher: Institute of Mathematical Statistics)

Hore, S., & Bhattacharya, T. (2018, April). A Machine Learning Based Approach Towards Building a SustainabilityModel for NBA Players. In *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)* (pp. 1690–1694). doi: 10.1109/ICICCT.2018.8473102

Hsu, C.-W., Chang, C.-C., & Lin, C.-J. (2003). *A practical guide to support vector classification* (Tech. Rep.). Department of Computer Science, National Taiwan University. Retrieved from http://www.csie.ntu.edu.tw/~cjlin/papers.html

Iyer, S. R., & Sharda, R. (2009, April). Prediction of athletes performance using neural networks: An application in cricket team selection. *Expert Systems with Applications*, *36*(3, Part 1), 5510–5522. Retrieved 2020-11-11, from http://www.sciencedirect.com/science/article/pii/S095741740800420X doi: 10.1016/j.eswa.2008.06.088

Ji, B., & Li, J. (2013, December). NBA All-Star Lineup Prediction Based on Neural Networks. In *2013 International Conference on Information Science and Cloud Computing Companion* (pp. 864–869). doi: 10.1109/ISCC-C.2013.92

Kataoka, Y., & Gray, P. (2019). Real-Time Power Performance Prediction in Tour de France. In U. Brefeld, J. Davis, J. Van Haaren, & A. Zimmermann (Eds.), *Machine Learning and Data Mining for Sports Analytics* (pp. 121–130). Cham: Springer International Publishing. doi: 10.1007/978-3-030-17274-9_10

Landers, J. R., & Duperrouzel, B. (2019, June). Machine Learning Approaches to Competing in Fantasy Leagues for the NFL. *IEEE Transactions on Games*, *11*(2), 159–172. (Conference Name: IEEE Transactions on Games) doi: 10.1109/TG.2018.2841057

Liu, G., Luo, Y., Schulte, O., & Kharrat, T. (2020, September). Deep soccer analytics: learning an action-value function for evaluating soccer players. *Data Mining and Knowledge Discovery*, *34*(5), 1531–1559. Retrieved 2020-10-23, from https://doi.org/10.1007/s10618-020-00705-9 doi: 10.1007/s10618-020-00705-9

Liu, G., & Schulte, O. (2018, July). Deep Reinforcement Learning in Ice Hockey for Context-Aware Player Evaluation. *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, 3442–3448. Retrieved 2020-04-02, from http://arxiv.org/abs/1805.11088 (arXiv: 1805.11088) doi: 10.24963/ijcai.2018/478

Liu, G., Zhu, W., & Schulte, O. (2019). Interpreting Deep Sports Analytics: Valuing Actions and Players in the NHL. In U. Brefeld, J. Davis, J. Van Haaren, & A. Zimmermann (Eds.), *Machine Learning and Data Mining for Sports Analytics* (pp. 69–81). Cham: Springer International Publishing. doi: 10.1007/978-3-030-17274-9_6

Ljung, D., Carlsson, N., & Lambrix, P. (2019). Player Pairs Valuation in Ice Hockey. In U. Brefeld, J. Davis, J. Van Haaren, & A. Zimmermann (Eds.), *Machine Learning and Data Mining for Sports Analytics* (pp. 82–92). Cham: Springer International Publishing. doi: 10.1007/978-3-030-17274-9_7

Maszczyk, A., Gołaś, A., Pietraszewski, P., Roczniok, R., Zajac, A., & Stanula, A. (2014, March). Application of Neural and Regression Models in Sports Results Prediction. *Procedia - Social and Behavioral Sciences*, *117*, 482–487. doi: 10.1016/j.sbspro.2014.02.249

McCabe, A., & Trevathan, J. (2008, April). Artificial Intelligence in Sports Prediction. In *Fifth International Conference on Information Technology: New Generations (itng 2008)* (pp. 1194–1197). doi: 10.1109/ITNG.2008.203

Mccullagh, J. (2010). Data Mining in Sports: A Neural Network Approach. In *IJCA TM : www.ijcaonline.org* (pp. 131–138).

McCullagh, J., & Whitfort, T. (2013, July). An Investigation into the Application of Artificial Neural Networks to the Prediction of Injuries in Sport. *International Journal of Sport and Health Sciences*, *7*(7), 356–360. Retrieved 2020-12-14, from `https://publications.waset.org/16308/an-investigation-into-the-application-of-artificial-neural-networks-to-the-prediction-of-injuries-in-sport`

Muazu Musa, R., Abdul Majeed, A. P. P., Taha, Z., Abdullah, M. R., Husin Musawi Maliki, A. B., & Azura Kosni, N. (2019, September). The application of Artificial Neural Network and k-Nearest Neighbour classification models in the scouting of high-performance archers from a selected fitness and motor skill performance parameters. *Science & Sports*, *34*(4), e241–e249. Retrieved 2020-11-09, from `http://www`

.sciencedirect.com/science/article/pii/S0765159719300656 doi: 10.1016/j.scispo.2019.02.006

Namatevs, I., Aleksejeva, L., & Polaka, I. (2016, December). Neural Network Modelling for Sports Performance Classification as a Complex Socio-Technical System. *Information Technology and Management Science*, *19*. doi: 10.1515/itms-2016-0010

Nsolo, E., Lambrix, P., & Carlsson, N. (2019). Player Valuation in European Football. In U. Brefeld, J. Davis, J. Van Haaren, & A. Zimmermann (Eds.), *Machine Learning and Data Mining for Sports Analytics* (pp. 42–54). Cham: Springer International Publishing. doi: 10.1007/978-3-030-17274-9_4

Oytun, M., Tinazci, C., Sekeroglu, B., Acikada, C., & Yavuz, H. U. (2020). Performance Prediction and Evaluation in Female Handball Players Using Machine Learning Models. *IEEE Access*, *8*, 116321–116335. (Conference Name: IEEE Access) doi: 10.1109/ACCESS.2020.3004182

Pelechrinis, K., Winston, W., Sagarin, J., & Cabot, V. (2019). Evaluating NFL Plays: Expected Points Adjusted for Schedule. In U. Brefeld, J. Davis, J. Van Haaren, & A. Zimmermann (Eds.), *Machine Learning and Data Mining for Sports Analytics* (pp. 106–117). Cham: Springer International Publishing. doi: 10.1007/978-3-030-17274-9_9

Perl, J. (2001, July). Artificial Neural Networks in Sports: New Concepts and Approaches. *International Journal of Performance Analysis in Sport*, *1*, 106–121. doi: 10.1080/24748668.2001.11868253

Porter, J. W. (2018). Predictive Analytics for Fantasy Football: Predicting Player Performance Across the NFL. *undefined*. Retrieved 2021-06-12, from /paper/Predictive-Analytics-for-Fantasy-Football%3A-Player-Porter/fa3ca8064b29aa75f4c3b568a95de5eba82140db

Prechelt, L. (1998). Early stopping - but when? In G. B. Orr & K.-R. Müller (Eds.), *Neural networks: Tricks of the trade* (pp. 55–69). Berlin, Heidelberg: Springer Berlin

Heidelberg. Retrieved from https://doi.org/10.1007/3-540-49430-8_3 doi: 10.1007/3-540-49430-8_3

Purucker, M. C. (1996, August). Neural network quarterbacking. *IEEE Potentials*, *15*(3), 9–15. (Conference Name: IEEE Potentials) doi: 10.1109/45.535226

Refaeilzadeh, P., Tang, L., & Liu, H. (2009). Cross-Validation. In L. LIU & M. T. ÖZSU (Eds.), *Encyclopedia of Database Systems* (pp. 532–538). Boston, MA: Springer US. Retrieved from https://doi.org/10.1007/978-0-387-39940-9_565 doi: 10.1007/978-0-387-39940-9_565

Saikia, H., Bhattacharjee, D., & Lemmer, H. H. (2012, April). Predicting the Performance of Bowlers in IPL: An Application of Artificial Neural Network. *International Journal of Performance Analysis in Sport*, *12*(1), 75–89. Retrieved 2020-11-11, from https://www.tandfonline.com/doi/full/10.1080/24748668.2012.11868584 doi: 10.1080/24748668.2012.11868584

Silva, A. J., Costa, A. M., Oliveira, P. M., Reis, V. M., Saavedra, J., Perl, J., ... Marinho, D. A. (2007, March). The Use of Neural Network Technology to Model Swimming Performance. *Journal of Sports Science & Medicine*, *6*(1), 117–125. Retrieved 2020-12-14, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3778687/

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014, June). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, *15*, 1929–1958.

Yurko, R., Ventura, S., & Horowitz, M. (2019, August). nflWAR: a reproducible method for offensive player evaluation in football. *Journal of Quantitative Analysis in Sports*, *15*(3), 163–183. Retrieved 2020-10-23, from https://www.degruyter.com/view/journals/jqas/15/3/article-p163.xml (Publisher: De Gruyter Section: Journal of Quantitative Analysis in Sports) doi: 10.1515/jqas-2018-0010

Appendix A

Feature Distribution

A.1 Quarterback

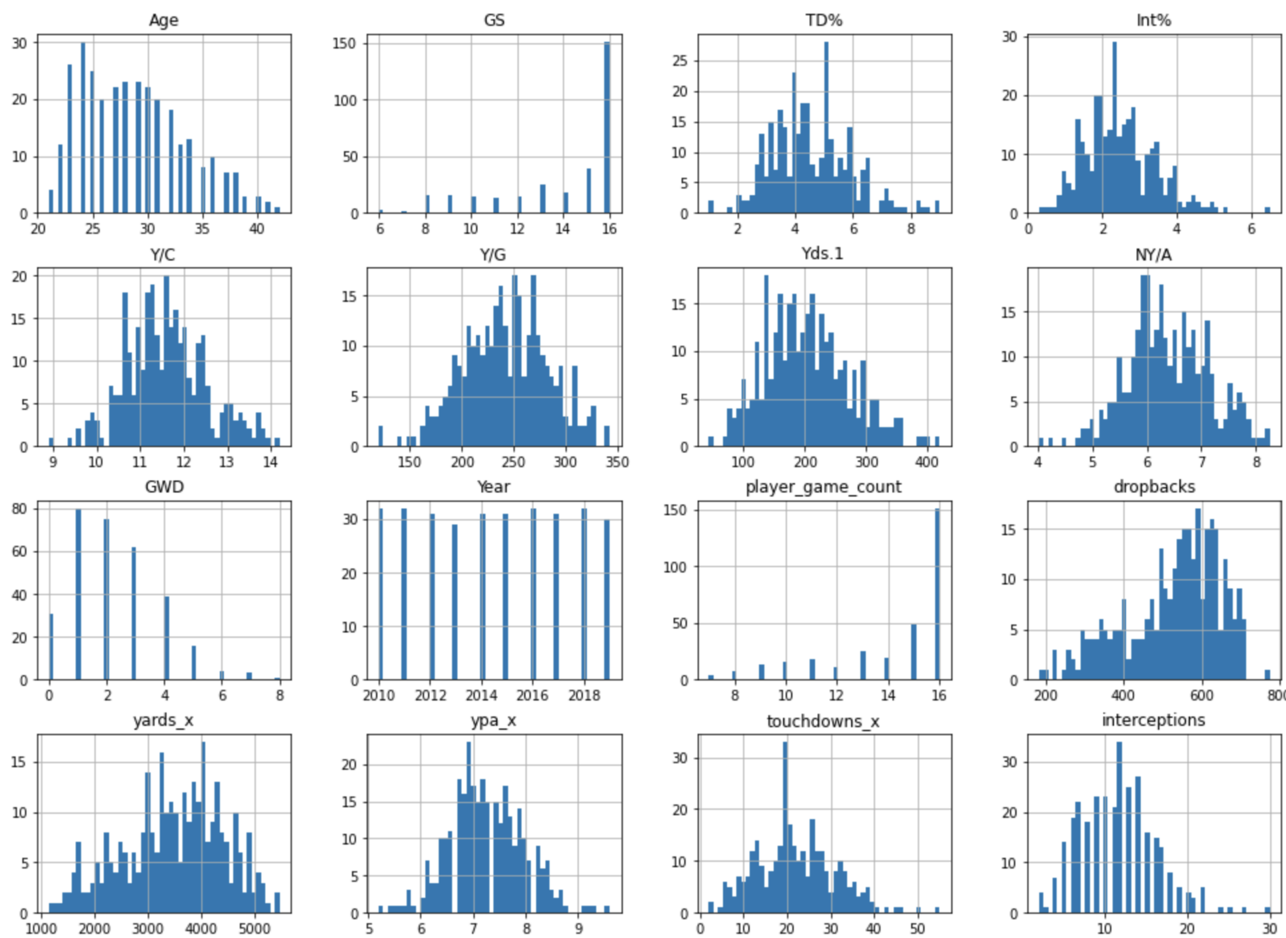


Figure A.1: QB Feature Distribution 1

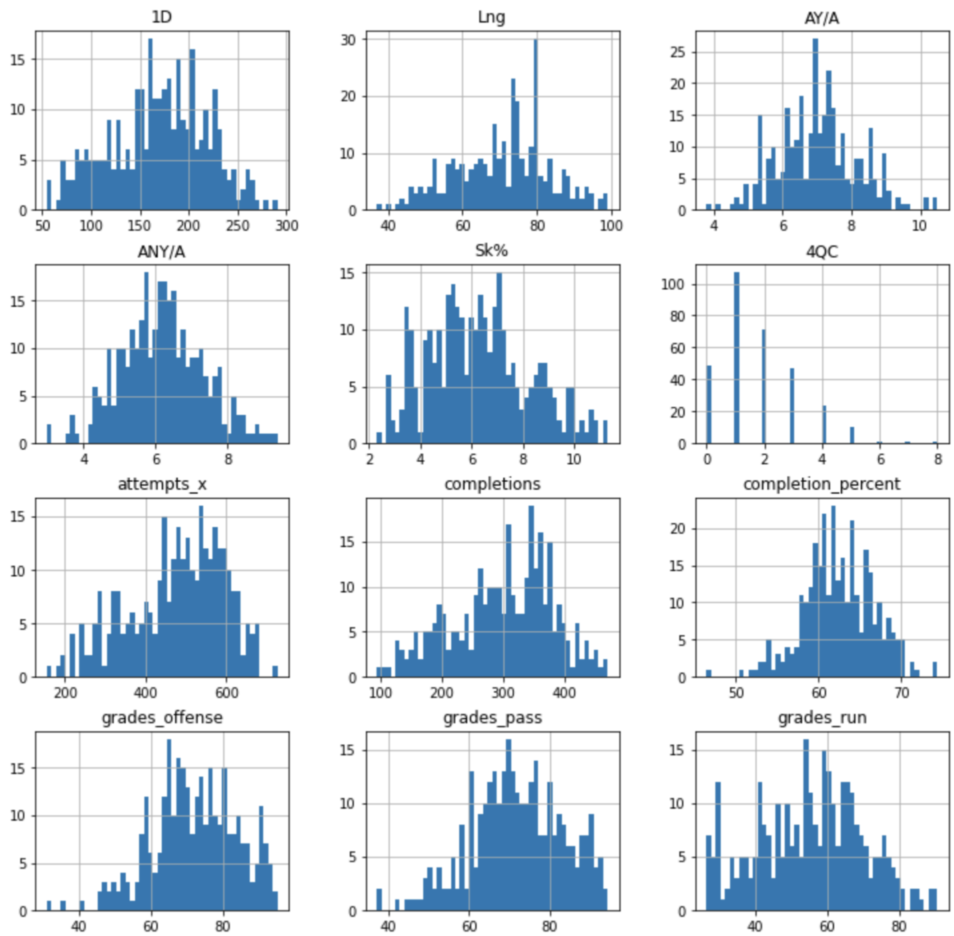


Figure A.2: QB Feature Distribution 2

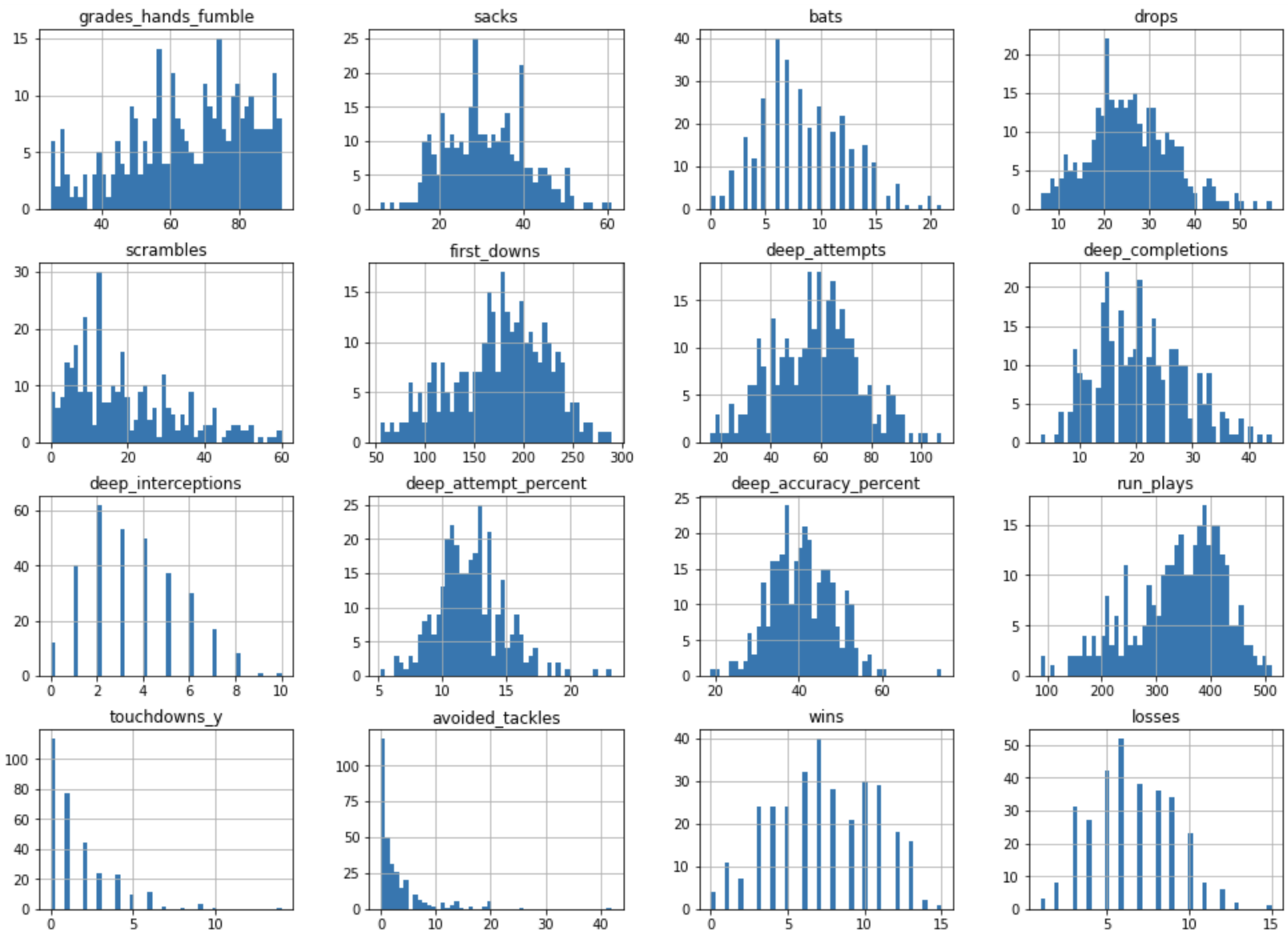


Figure A.3: QB Feature Distribution 3

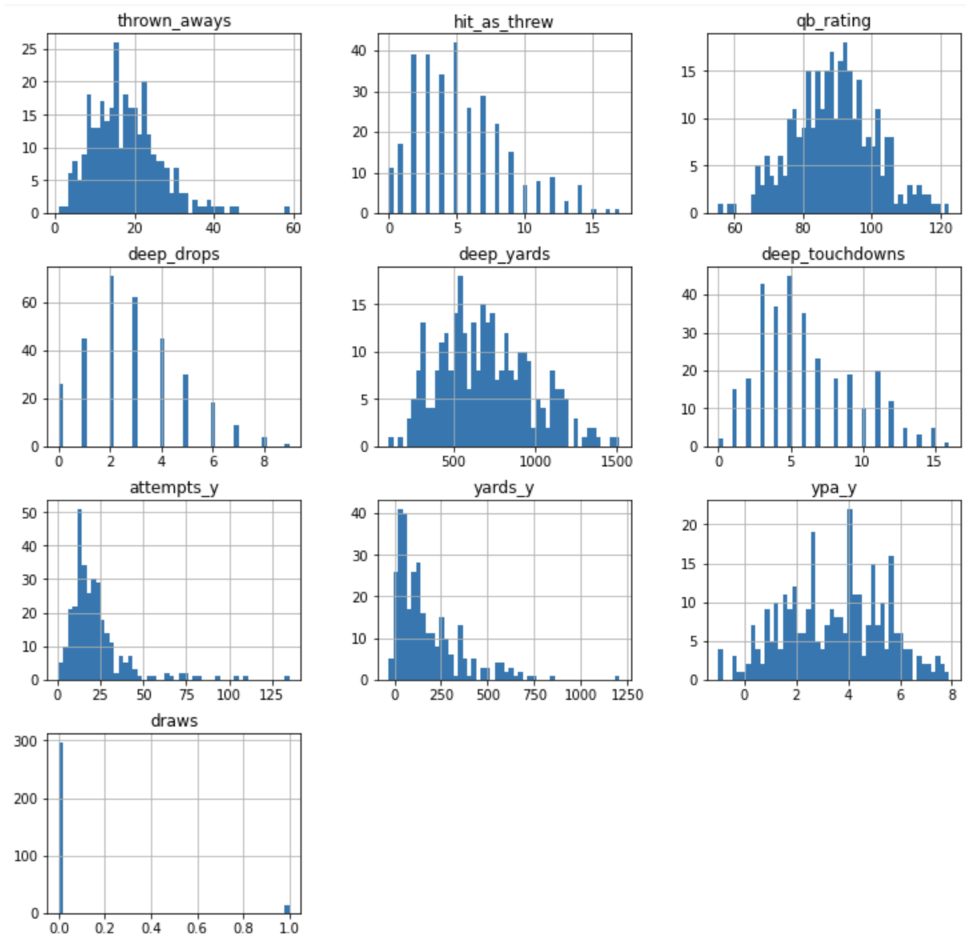


Figure A.4: QB Feature Distribution 4

A.2 Wide Receiver

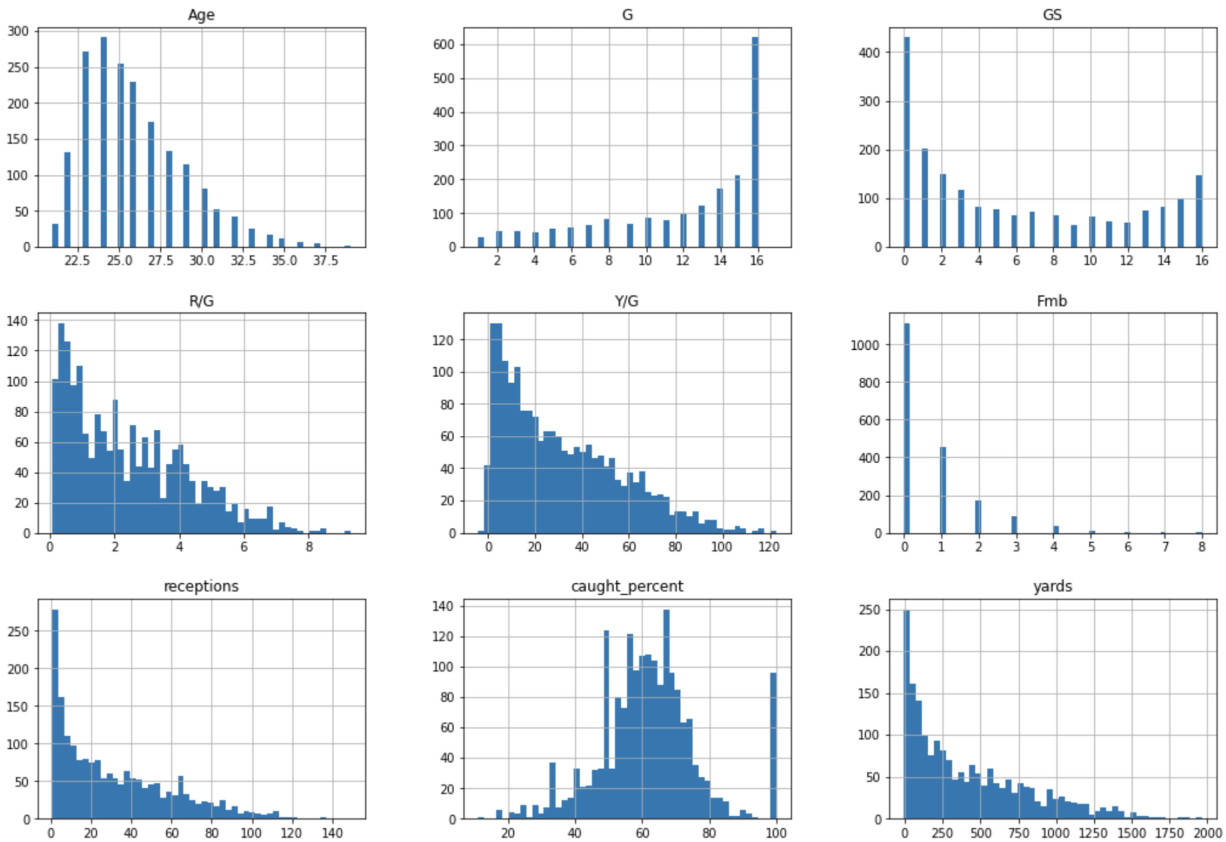


Figure A.5: WR Feature Distribution 1

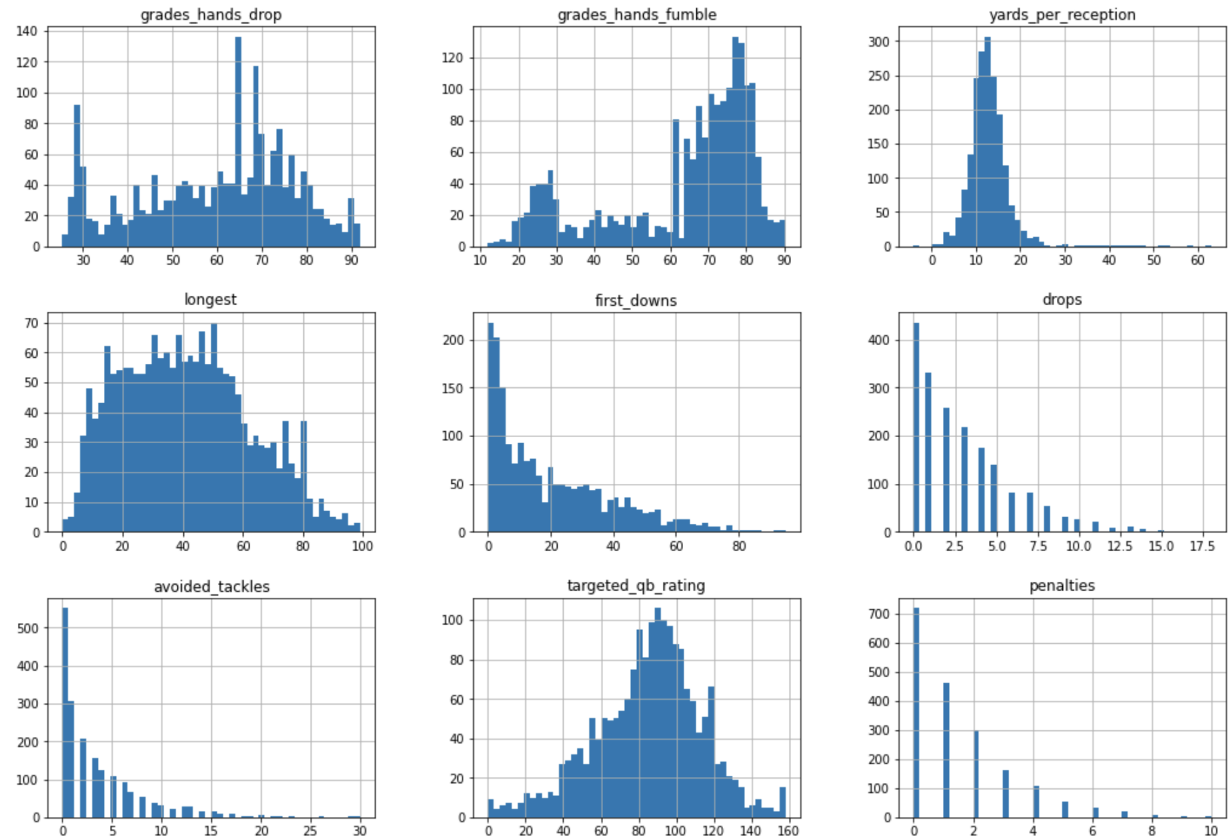


Figure A.6: WR Feature Distribution 2

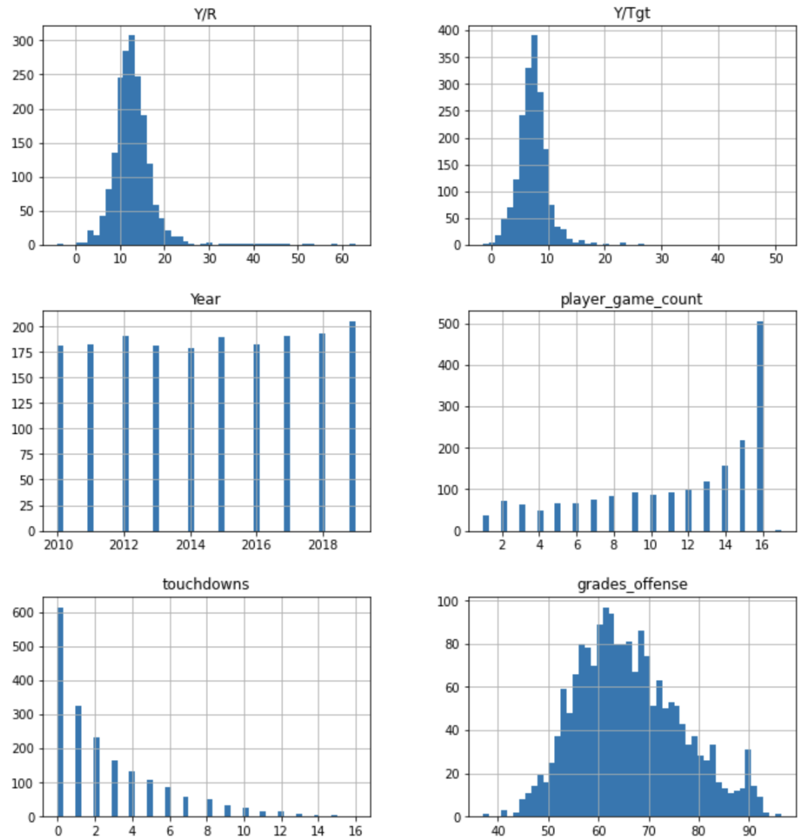


Figure A.7: WR Feature Distribution 3

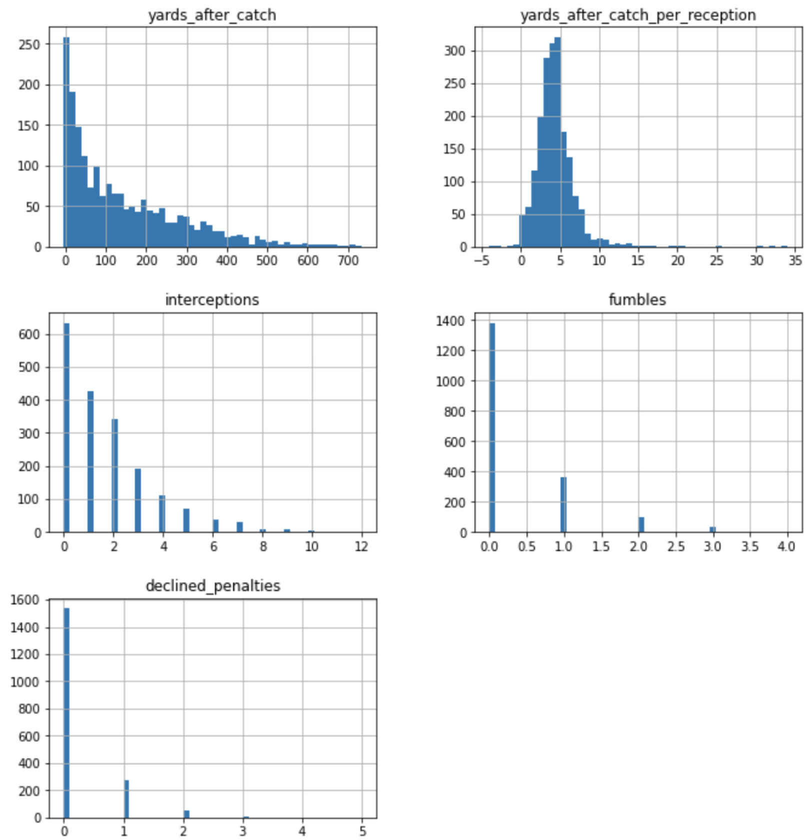


Figure A.8: WR Feature Distribution 4

A.3 Running Back

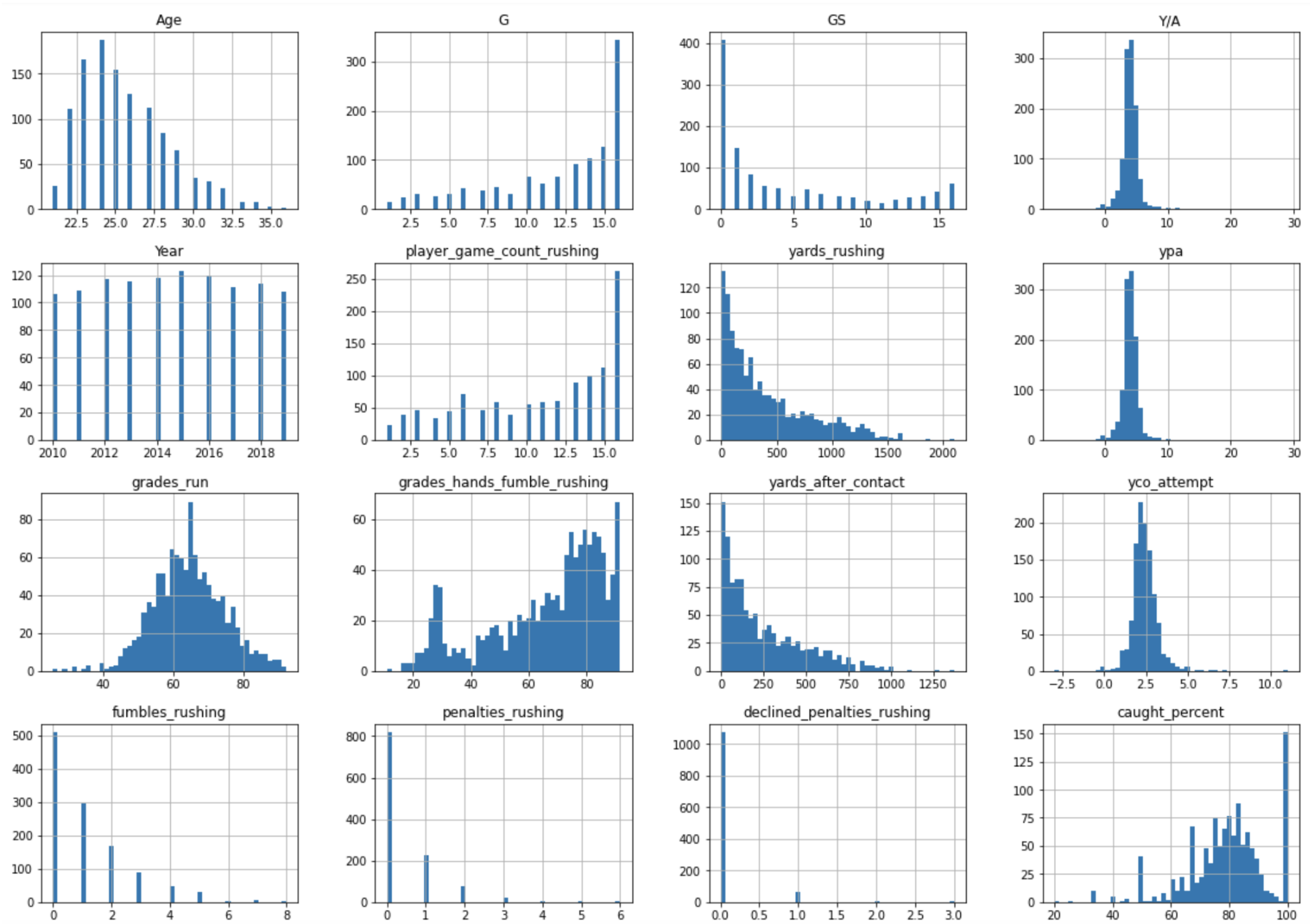


Figure A.9: RB Feature Distribution 1

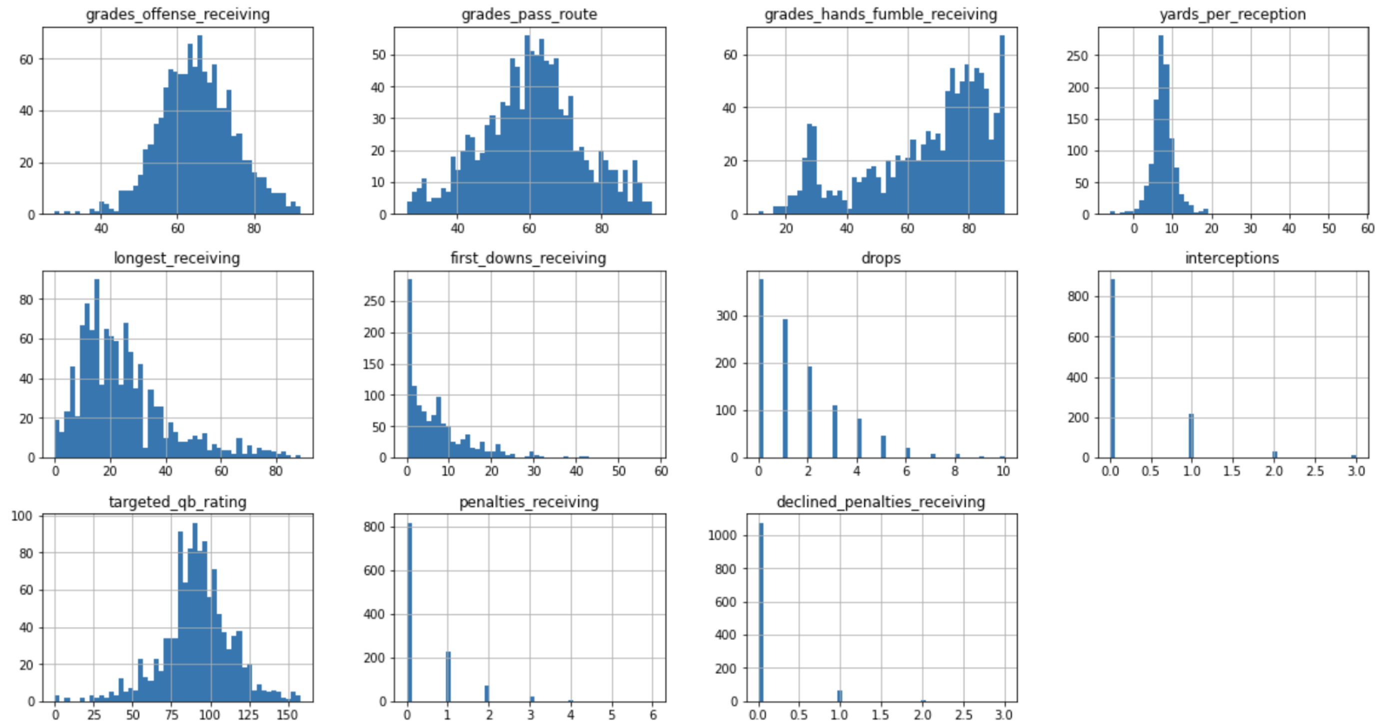


Figure A.10: RB Feature Distribution 2

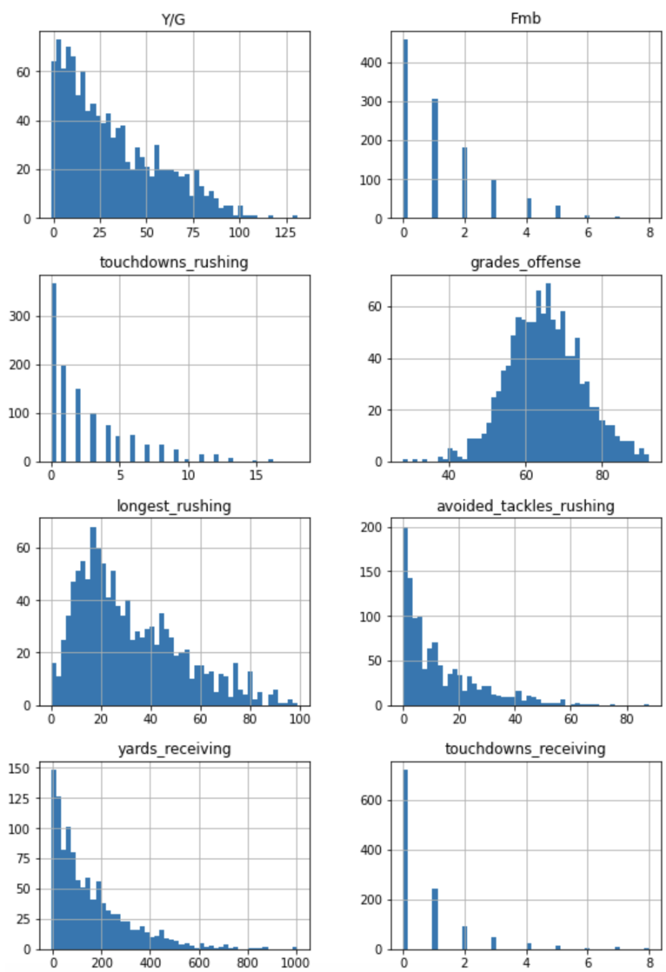


Figure A.11: RB Feature Distribution 3

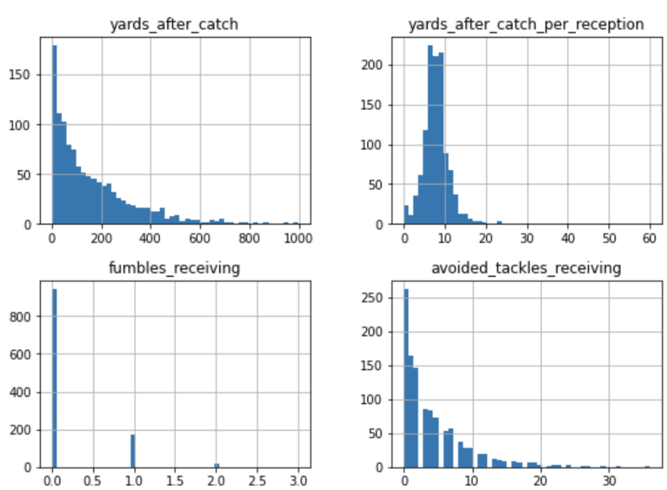


Figure A.12: RB Feature Distribution 4

Appendix B

Correlation Plots

B.1 Quarterback

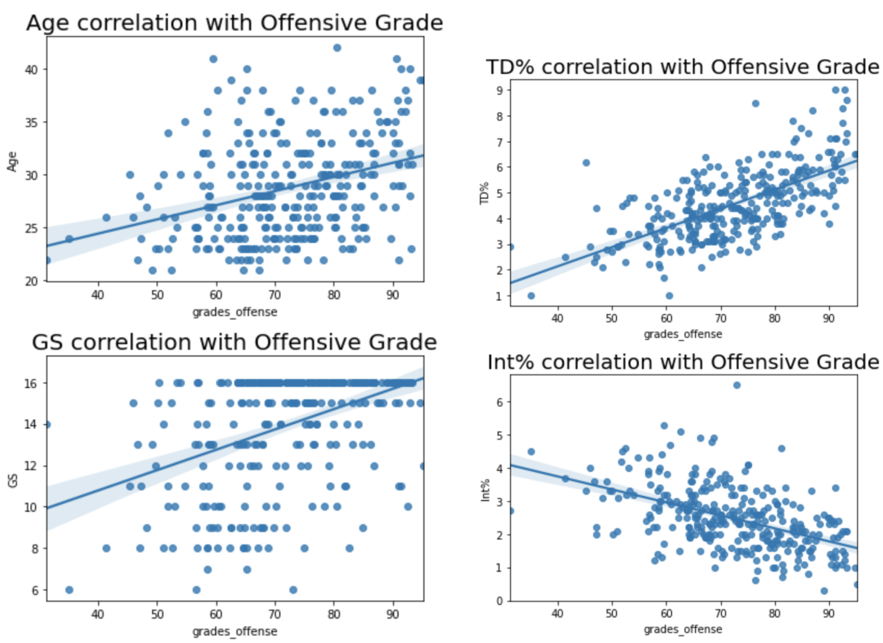


Figure B.1: QB Correlation Numbers - Age, TD%, GS,Int%

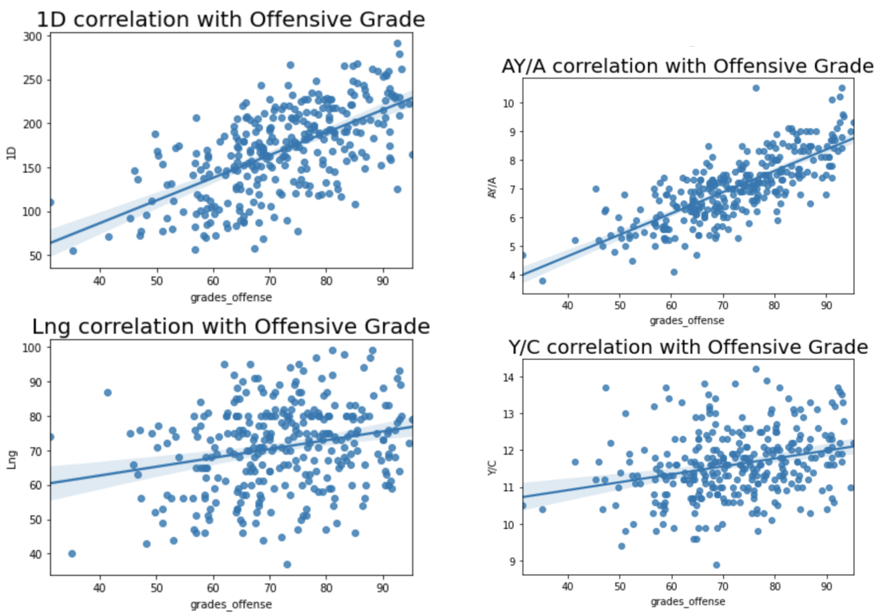


Figure B.2: QB Correlation Numbers - 1D,AY/A, Lng, Y/C

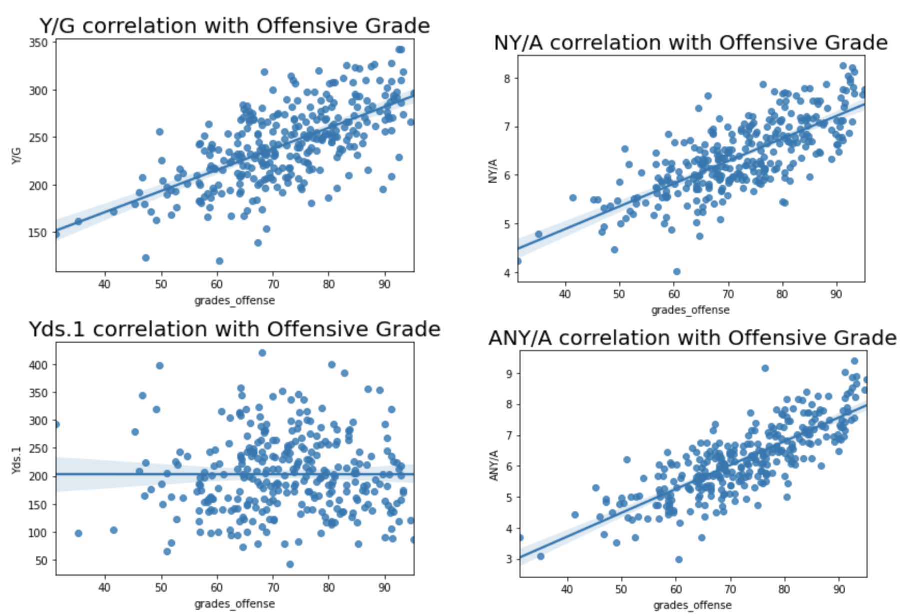


Figure B.3: QB Correlation Numbers - Y/G, NY/A, Yds, ANY/A

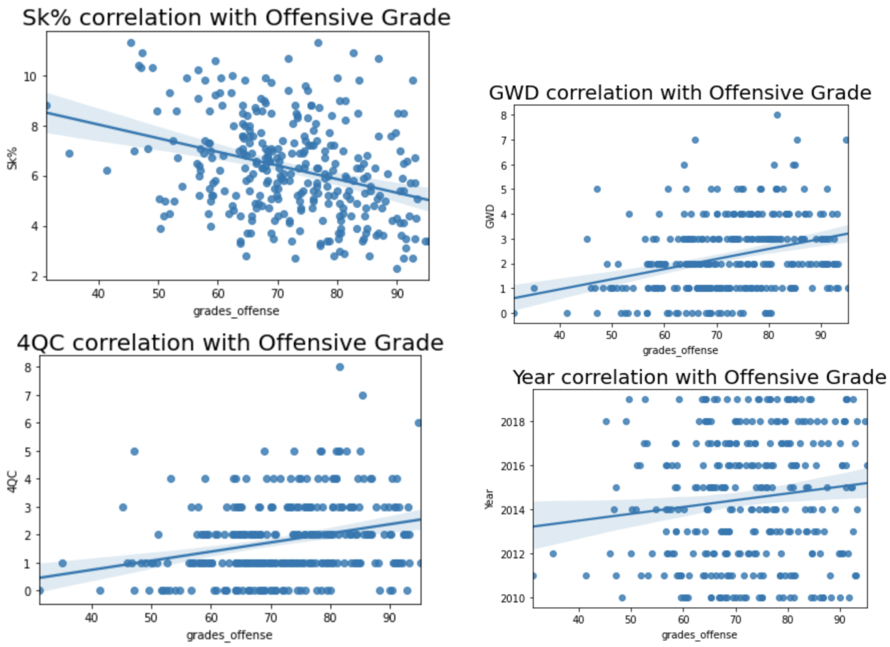


Figure B.4: QB Correlation Numbers - Sk%, GWD, 4QC, Year

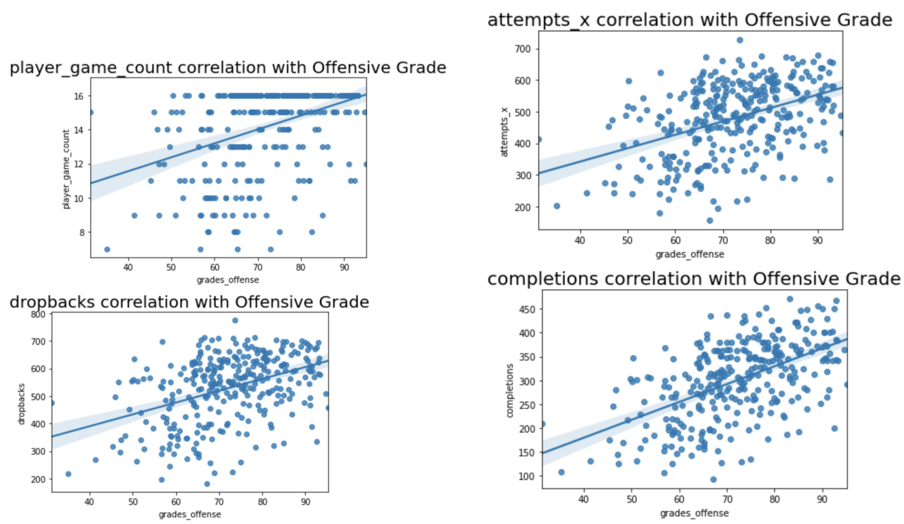


Figure B.5: QB Correlation Numbers - Player Game Count, Attempts Passing, Dropbacks, Completions

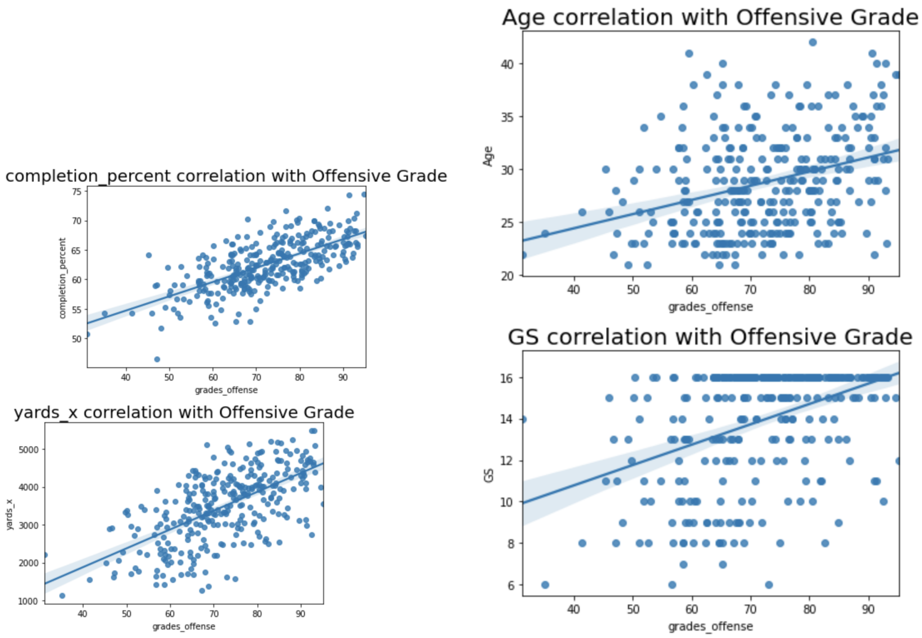


Figure B.6: QB Correlation Numbers - Completion%, Age, Yards Passing, GS

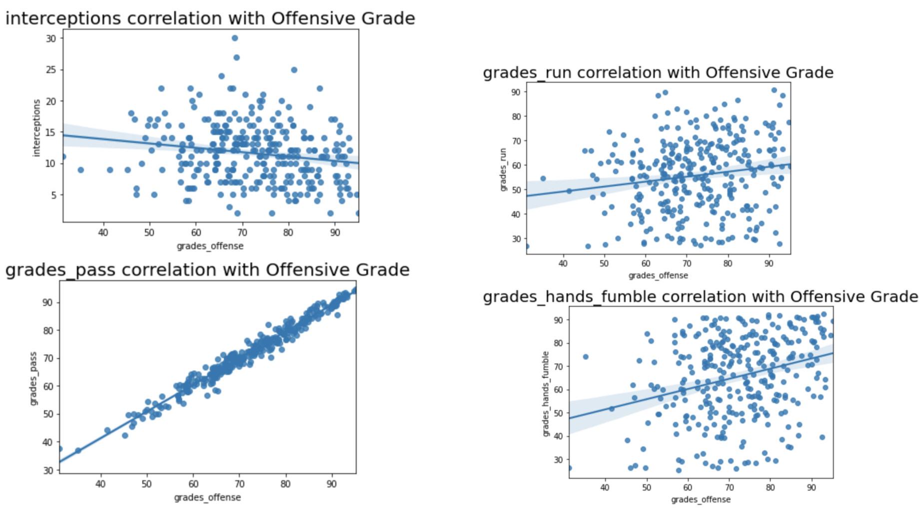


Figure B.7: QB Correlation Numbers - Interceptions, Grades Run, Grades Pass, Grades Hands Fumble

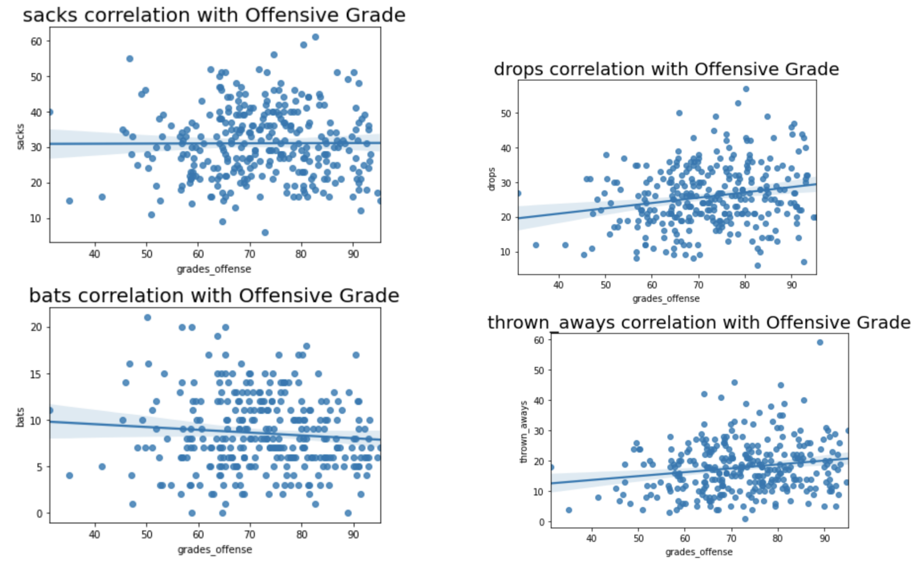


Figure B.8: QB Correlation Numbers - Sacks, Drops, Bats, Throw Aways

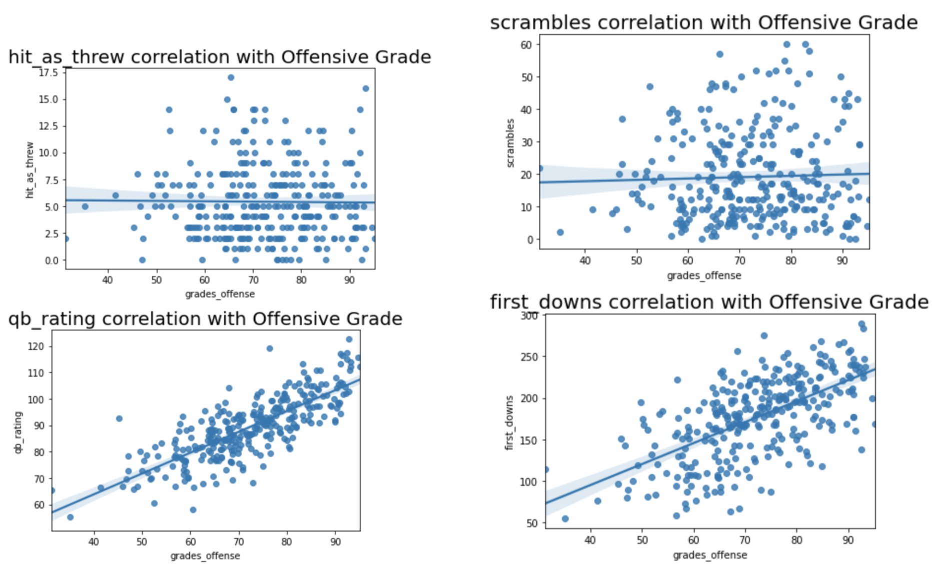


Figure B.9: QB Correlation Numbers - Hit As Threw, Scrambles, QB Rating, First Downs

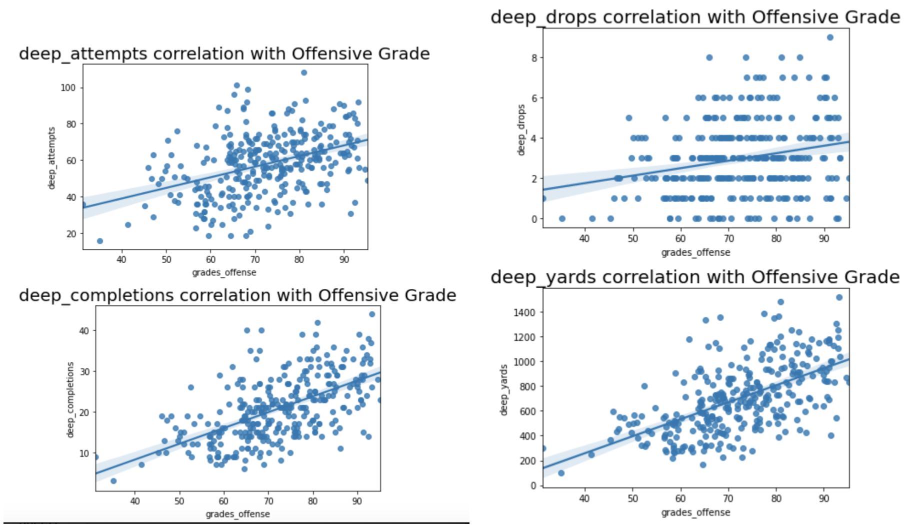


Figure B.10: QB Correlation Numbers - Deep Attempts, Deep Drops, Deep Completions, Deep Accuracy

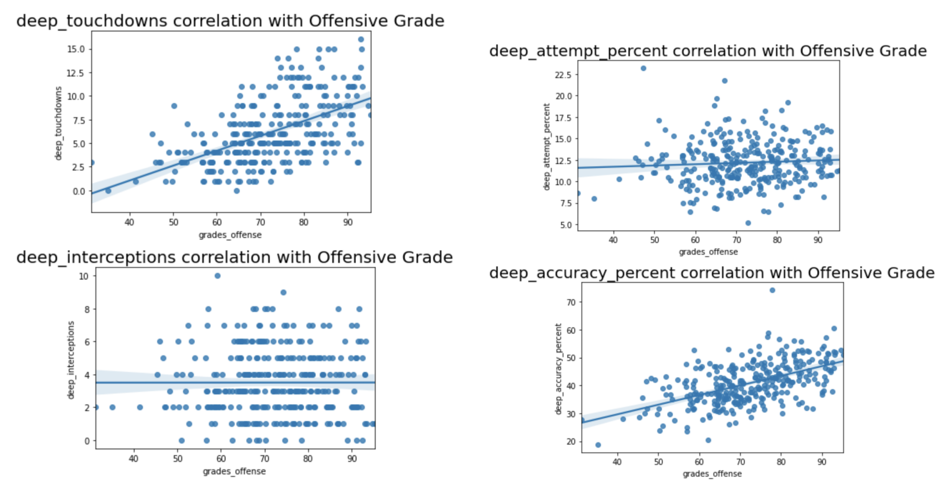


Figure B.11: QB Correlation Numbers - Deep Touchdowns, Deep Attempt Success, Deep Interceptions, Deep Accuracy%

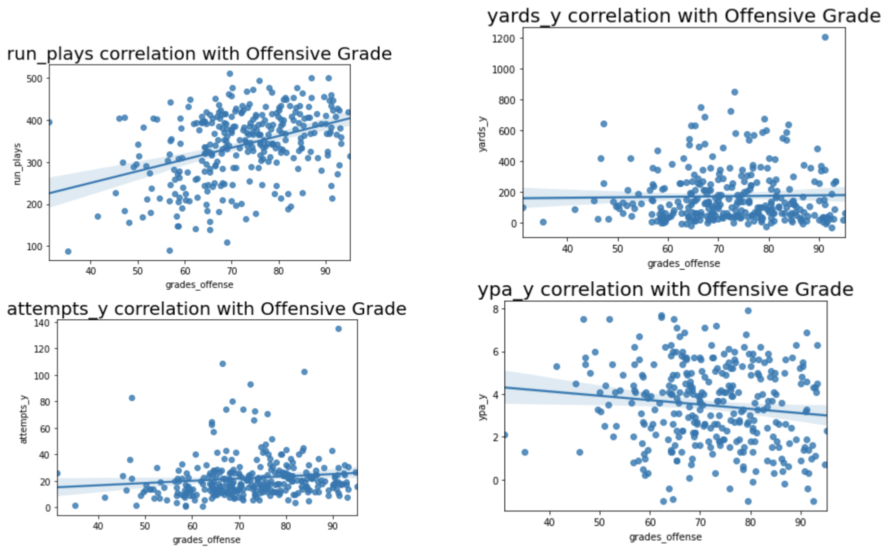


Figure B.12: QB Correlation Numbers - Run Plays, Yards Rush, Attempts Rush, YPA Rush

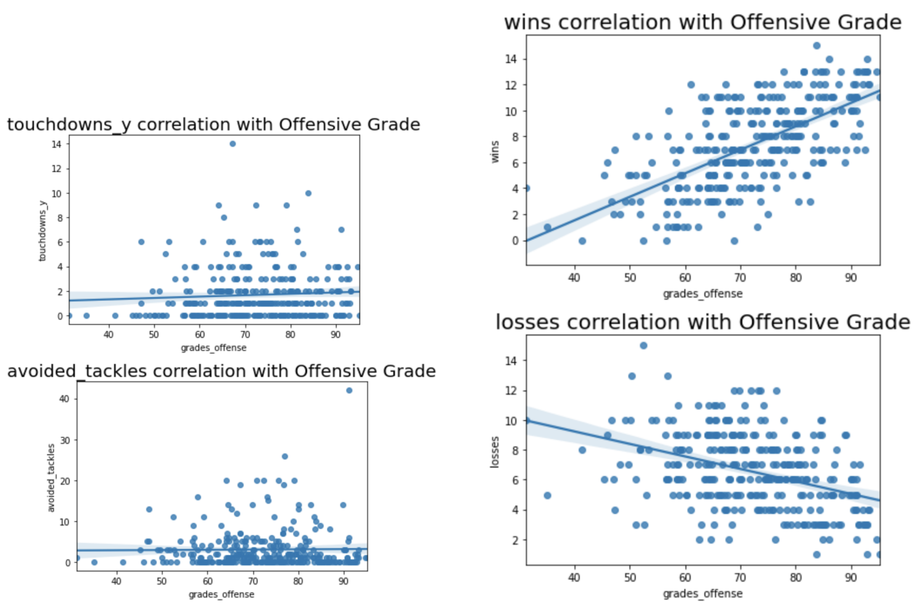


Figure B.13: QB Correlation Numbers - Touchdowns Rush, Wins, Avoided Tackles, Losses

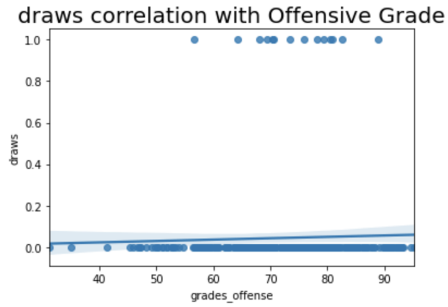


Figure B.14: QB Correlation Numbers - Draws

B.2 Wide Receiver

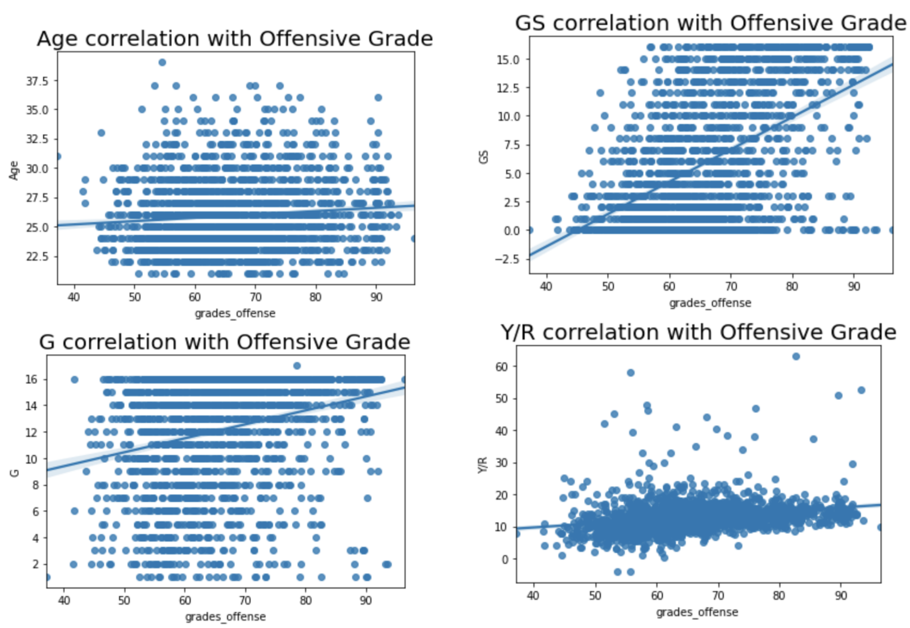


Figure B.15: WR Correlation Numbers - Age, GS, G, Y/R

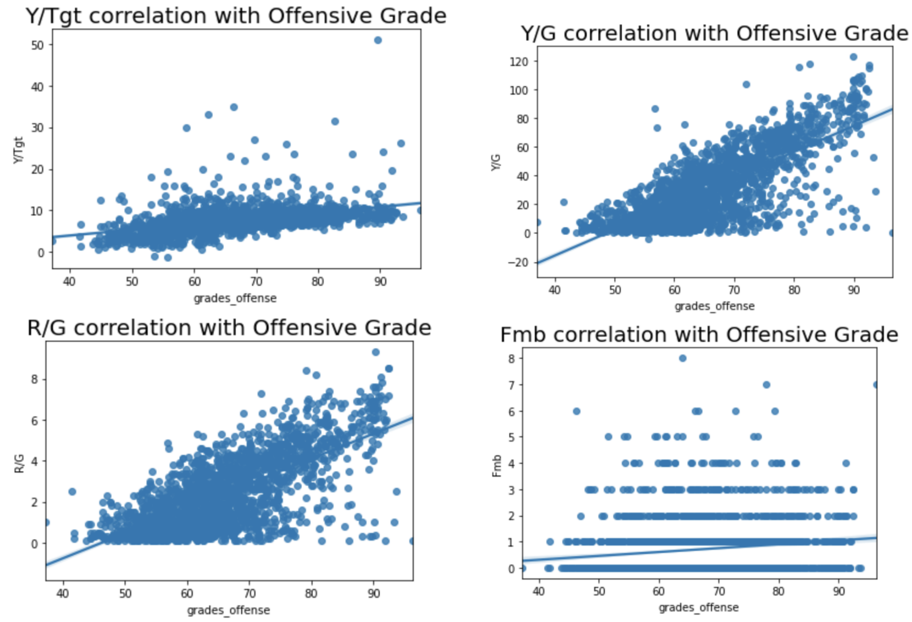


Figure B.16: WR Correlation Numbers - Y/Tgt, Y/G, R/G, Fmb

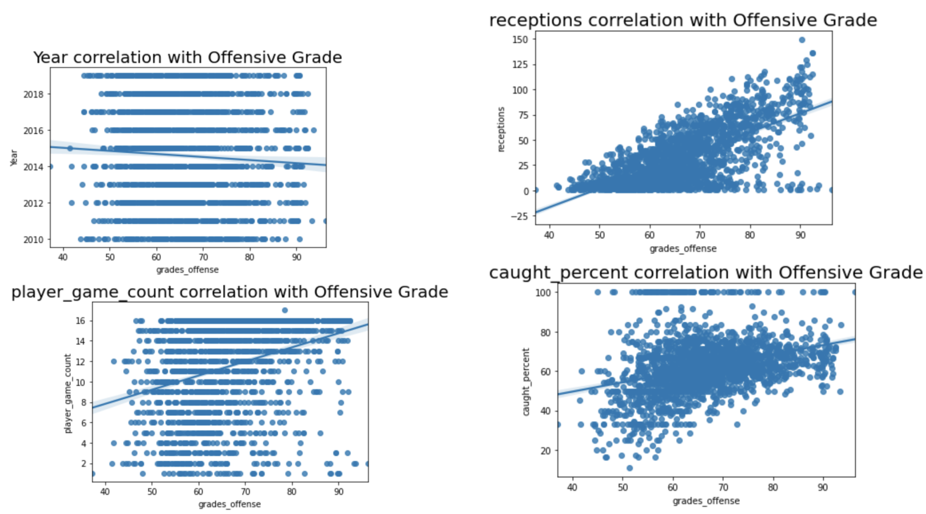


Figure B.17: WR Correlation Numbers - Year, Receptions, Player Game Count, Caught%

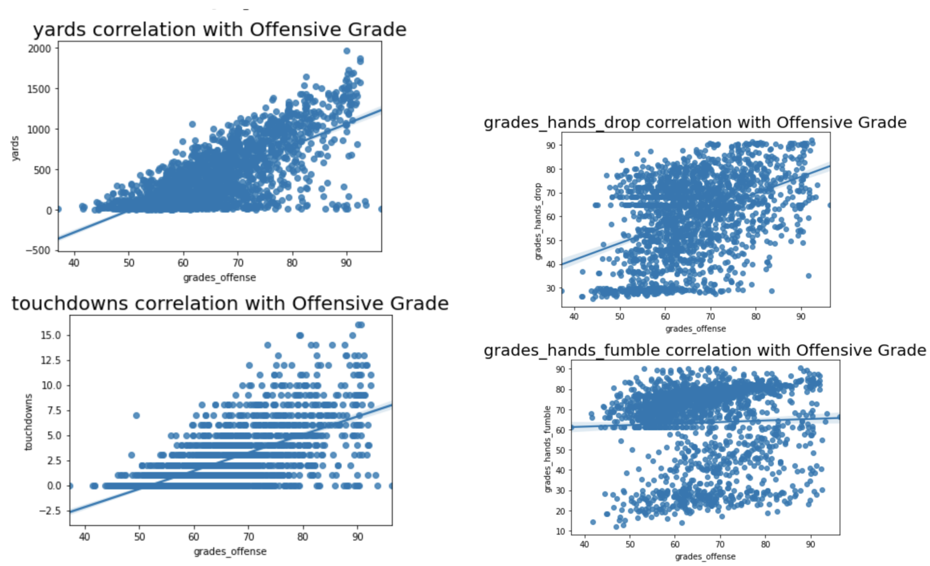


Figure B.18: WR Correlation Numbers - Yards, Grades Hands Drop, Touchdowns, Grades Hands Fumble

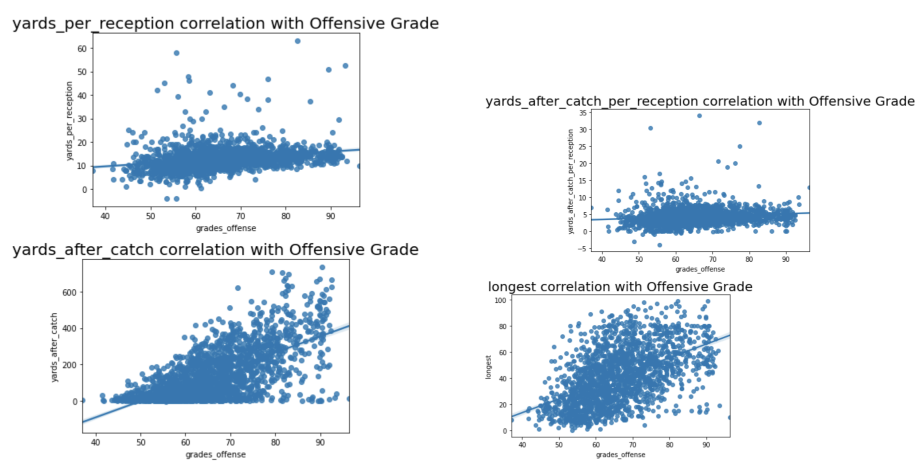


Figure B.19: WR Correlation Numbers - Yards Per Reception, Yards After Catch Per Rec, Yards After Catch, Longest

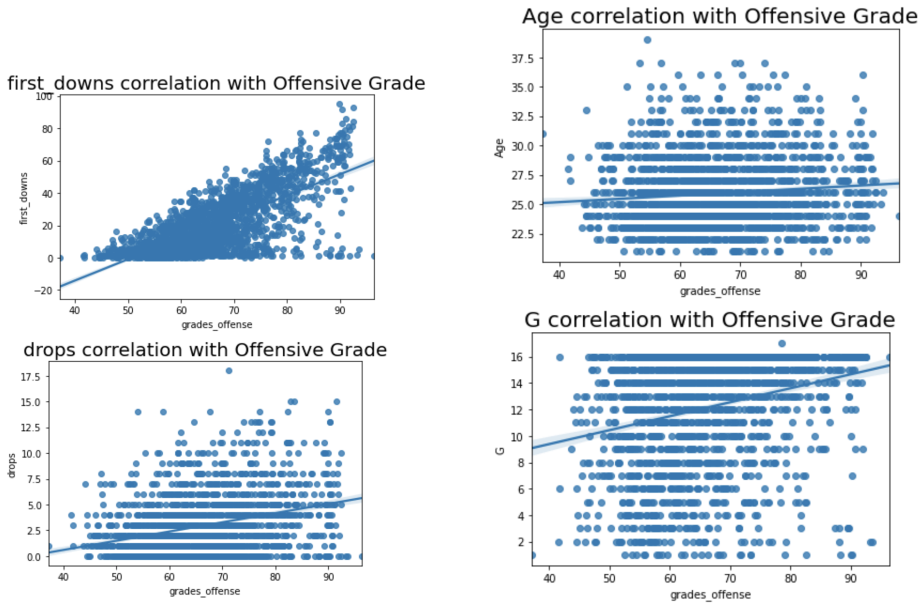


Figure B.20: WR Correlation Numbers - First Downs, Age, Drops, G

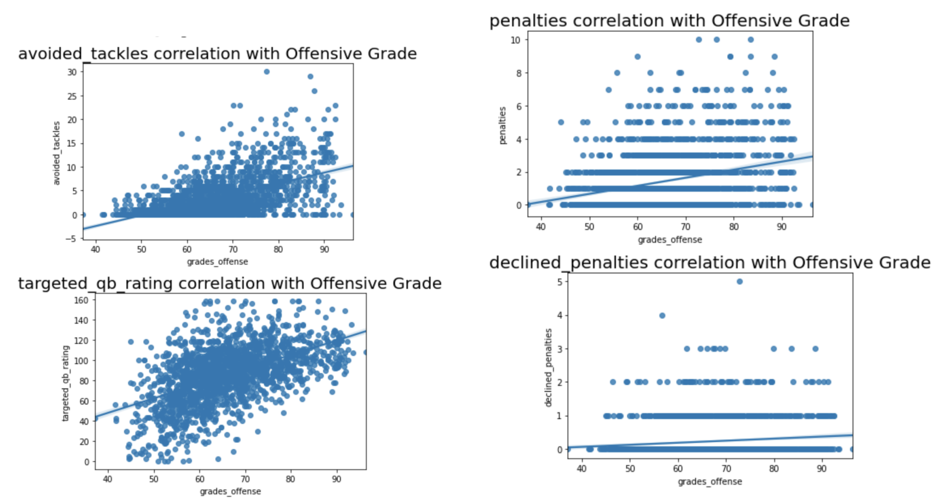


Figure B.21: WR Correlation Numbers - Avoided Tackles, Penalties, Targeted QB Rating, Declined Penalties

B.3 Running Back

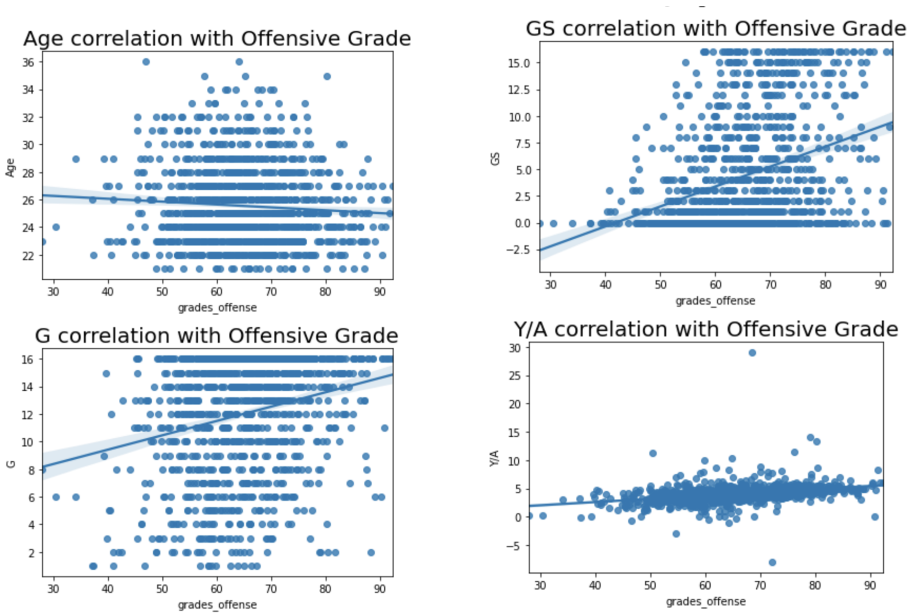


Figure B.22: RB Correlation Numbers - Age, GS, G, Y/A

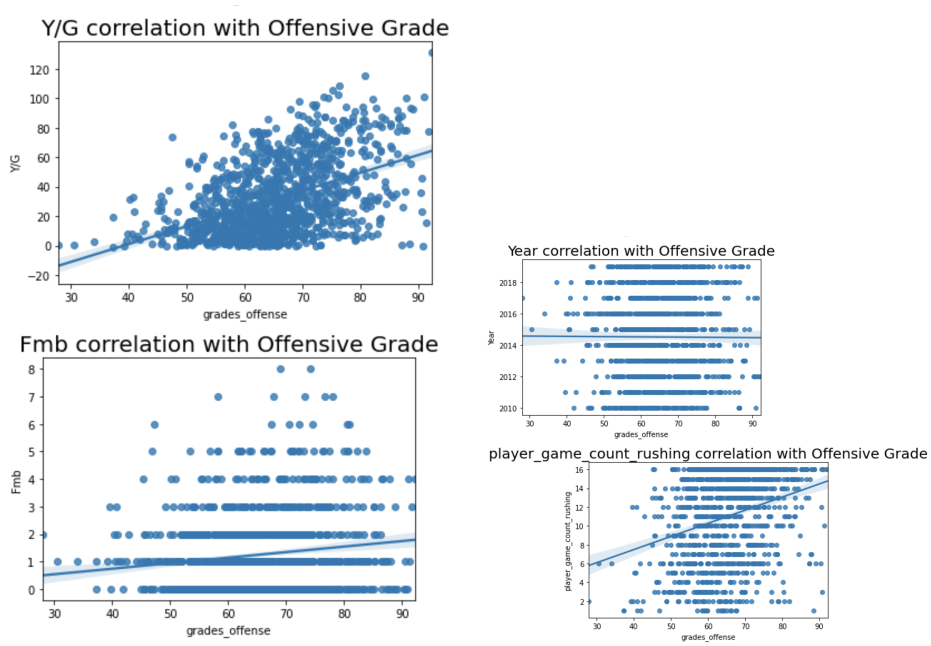


Figure B.23: RB Correlation Numbers - Y/G, Year, Fmb, Player Game Count

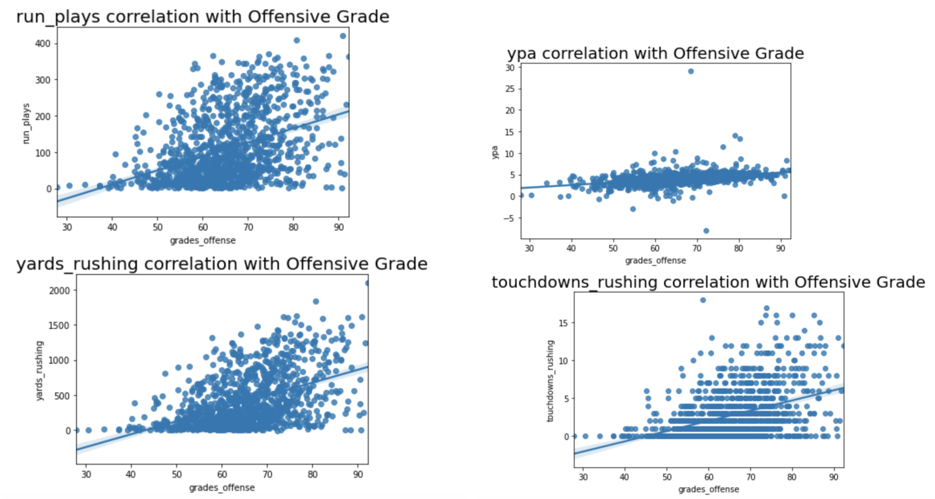


Figure B.24: RB Correlation Numbers - Run Plays, YPA, Yards Rushing, Touchdowns Rushing

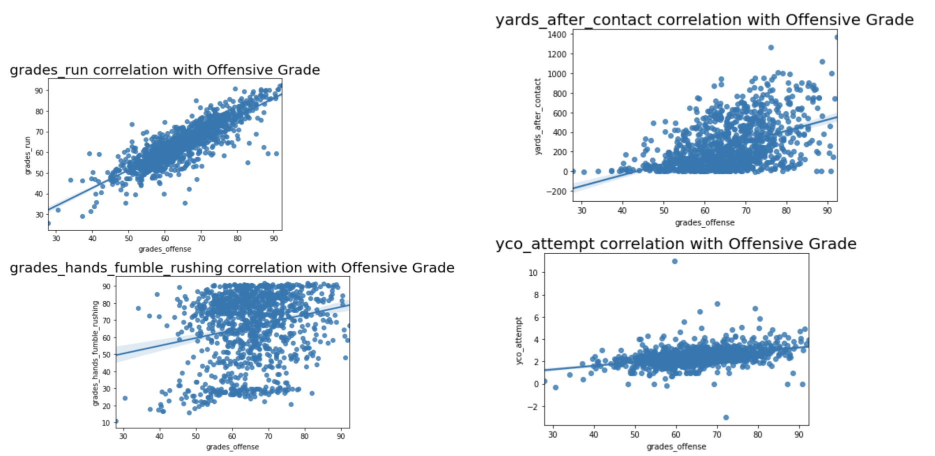


Figure B.25: RB Correlation Numbers - Grades Run, Yards After Contact, Grades Hands Fumble Rushing, Yco Attempt

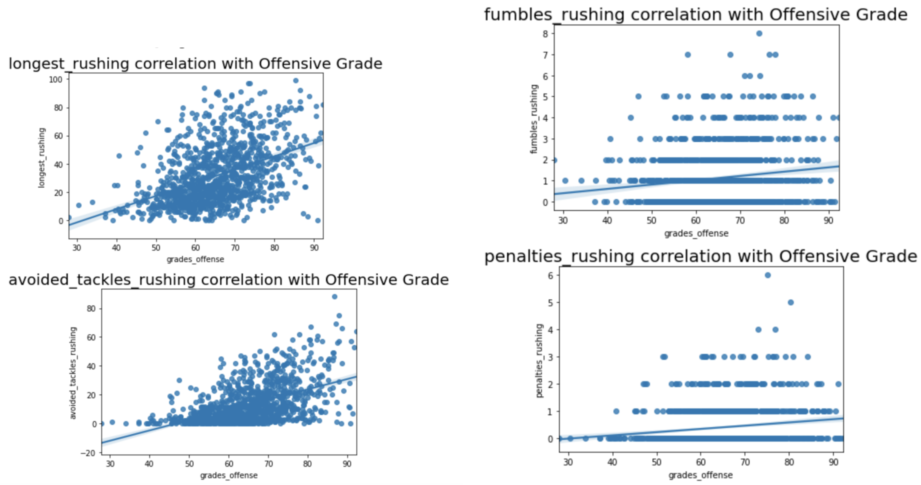


Figure B.26: RB Correlation Numbers - Longest Rushing, Fumbles Rushing, Avoided Tackles Rushing, Penalties Rushing

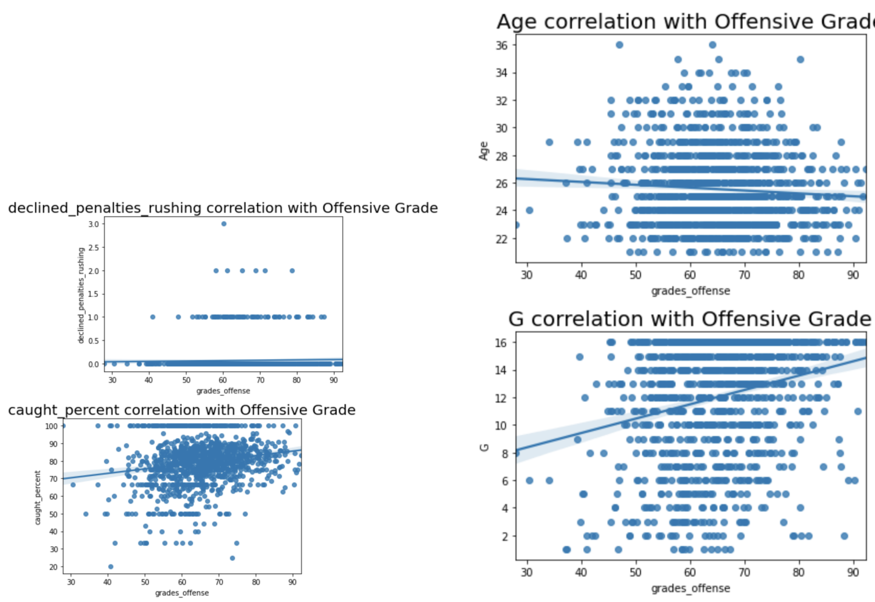


Figure B.27: RB Correlation Numbers - Declined Penalties, Age, Caught%, G

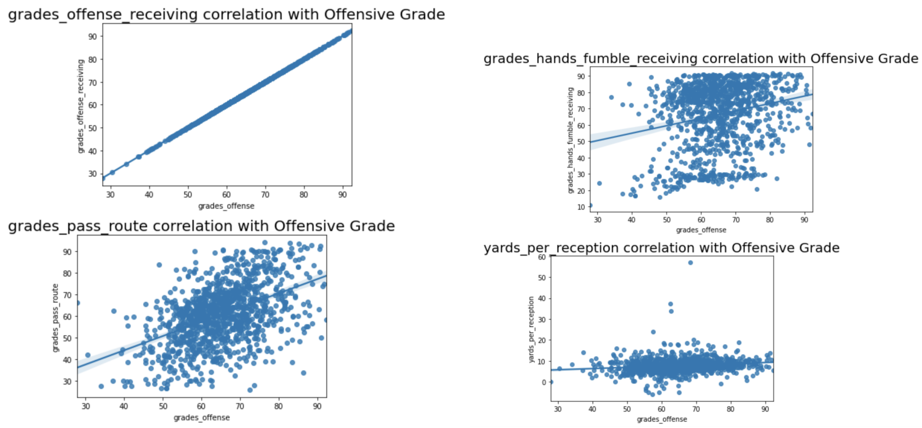


Figure B.28: RB Correlation Numbers - Grades Offense Receiving, Grades Hands Fumble Receiving, Grades Pass Route, Yards Per Reception

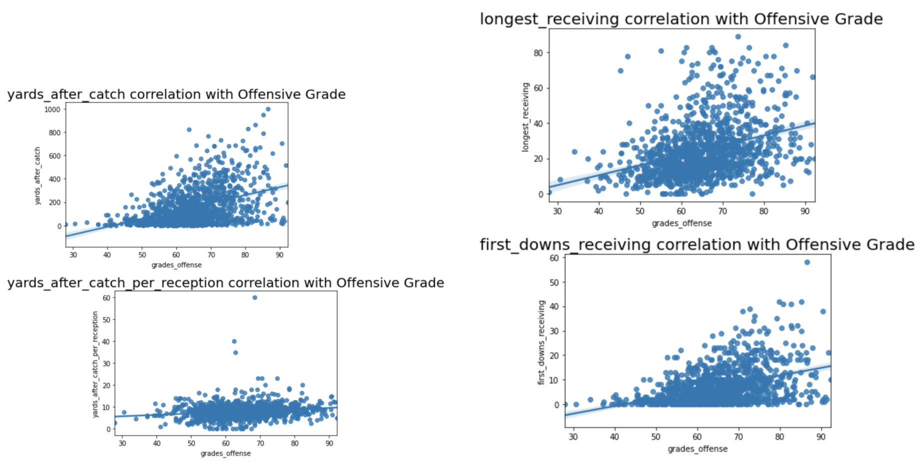


Figure B.29: RB Correlation Numbers - Yards After Catch, Longest Receiving, Yards Per Catch Per Reception, First Downs Receiving

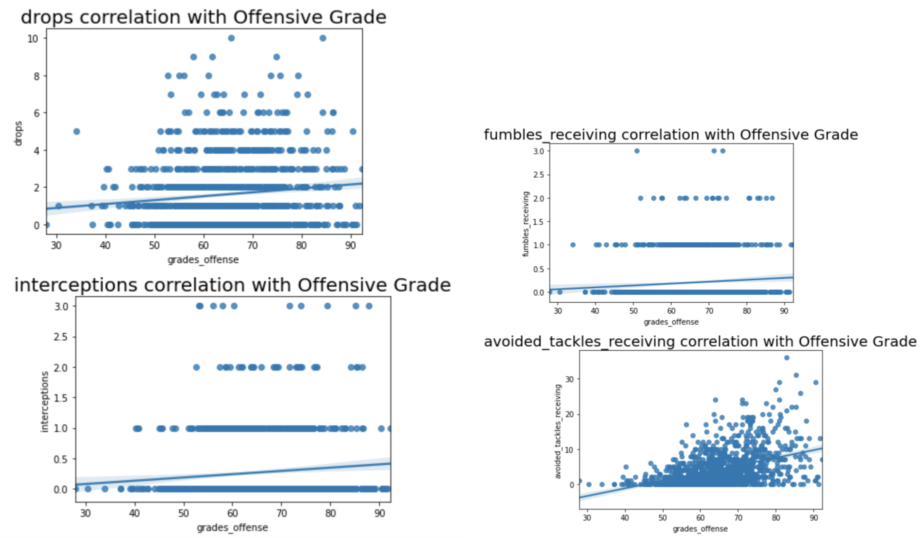


Figure B.30: RB Correlation Numbers - Drops, Fumbles Receiving, Interceptions, Avoided Tackles Receiving

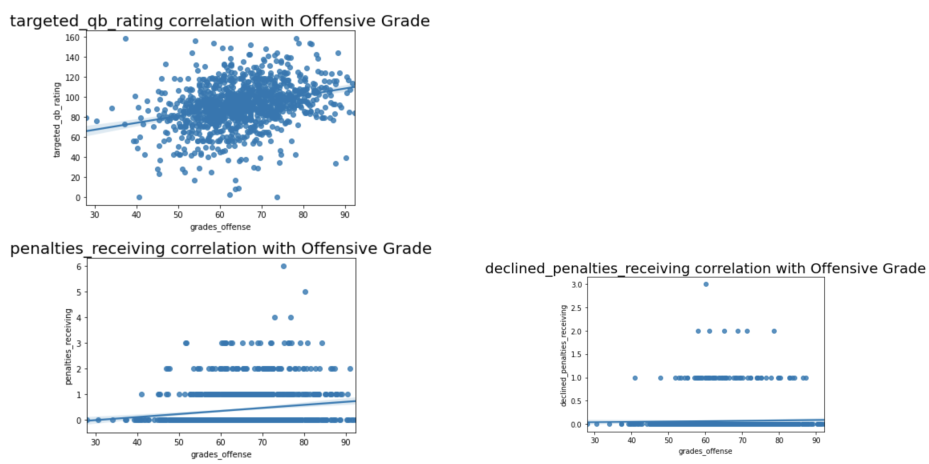


Figure B.31: RB Correlation Numbers - Targeted QB Rating, Penalties Receiving, Declined Penalties Receiving

Appendix C

Correlation Over Time

C.1 Quarterback

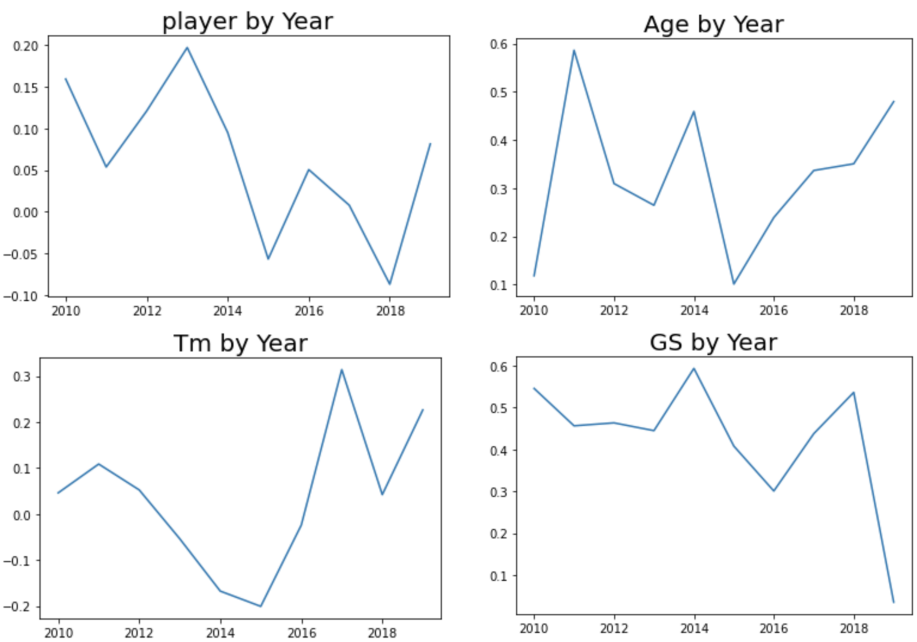


Figure C.1: QB Correlation over Time - Player, Age, Tm, GS

Figure C.2: QB Correlation over Time - TD%, 1D, Int%, Lng

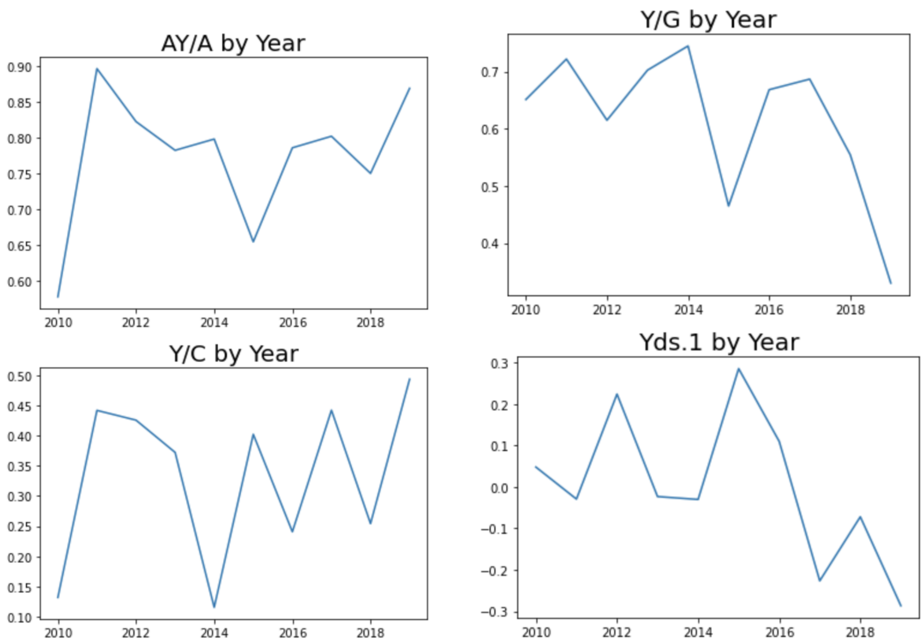


Figure C.3: QB Correlation over Time - AY/A, Y/G, Y/C, Yds

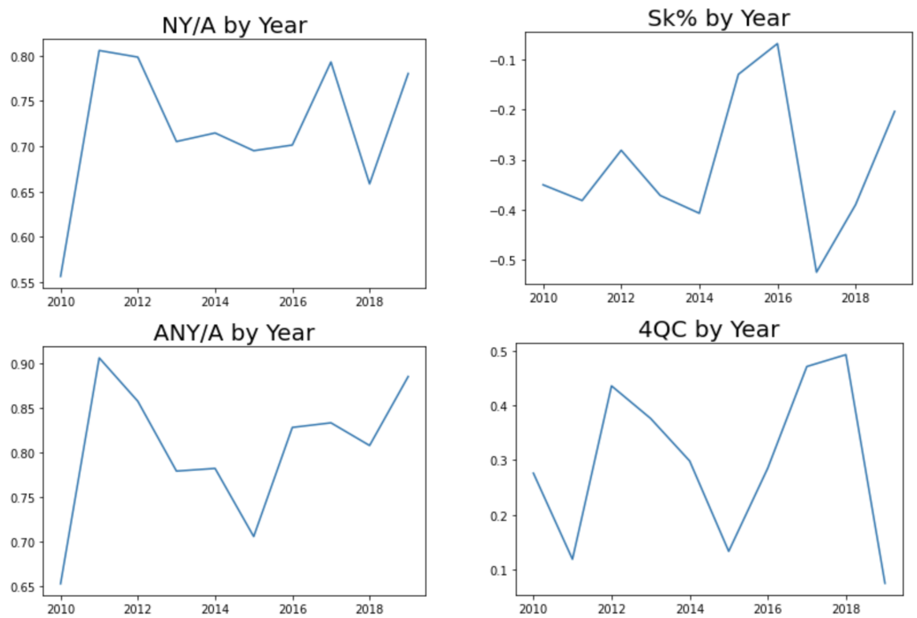


Figure C.4: QB Correlation over Time - NY/A, Sk%, ANY/A, 4QC

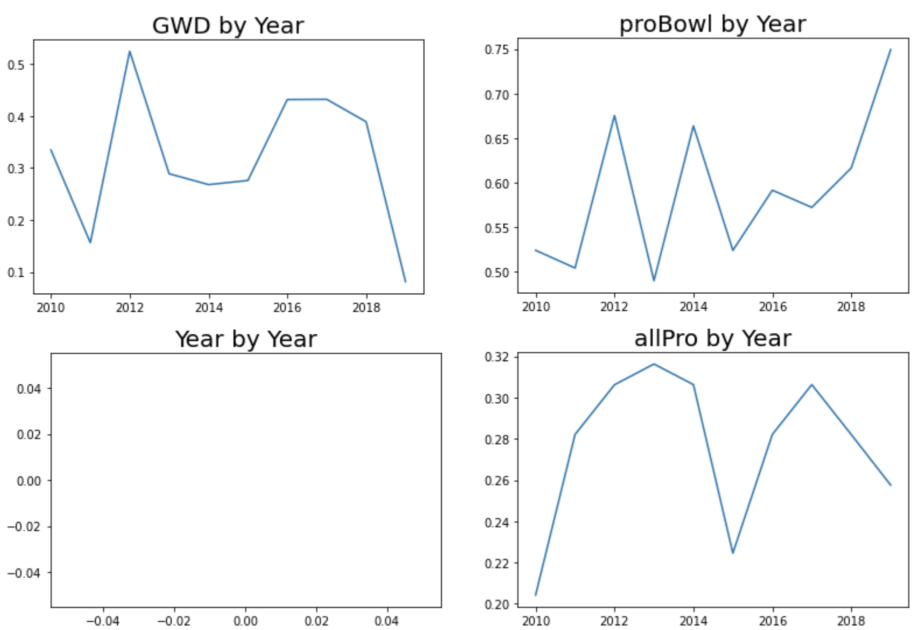


Figure C.5: QB Correlation over Time - WD, Pro Bowl, Year, All Pro

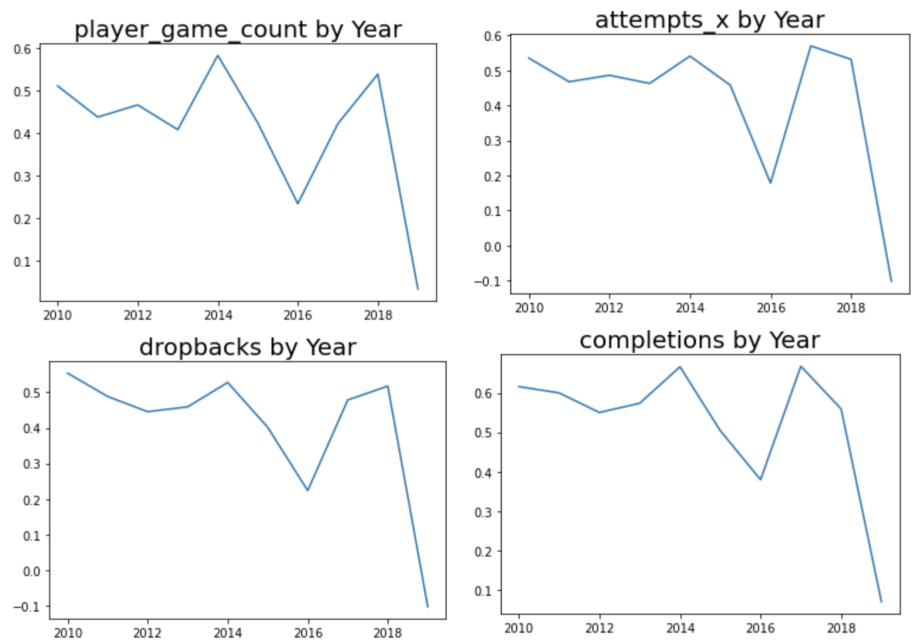


Figure C.6: QB Correlation over Time - Player Game Count, Attempts Pass, Dropbacks, Completions

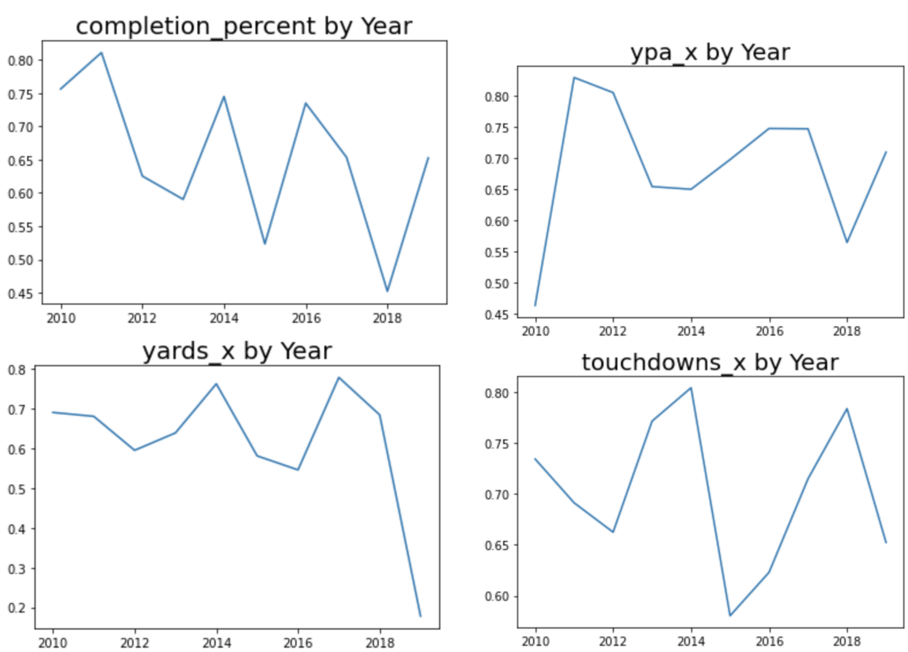


Figure C.7: QB Correlation over Time - Completion%, YPA Pass, Yards Pass, Touchdowns Pass

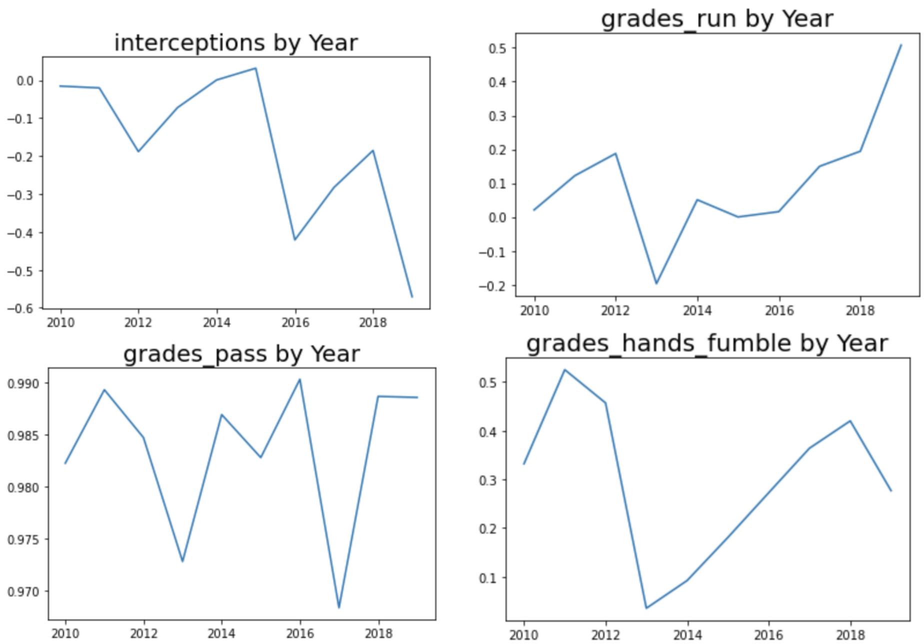


Figure C.8: QB Correlation over Time - Interceptions, Grades Run, Grades Pass, Grades Hands Fumble

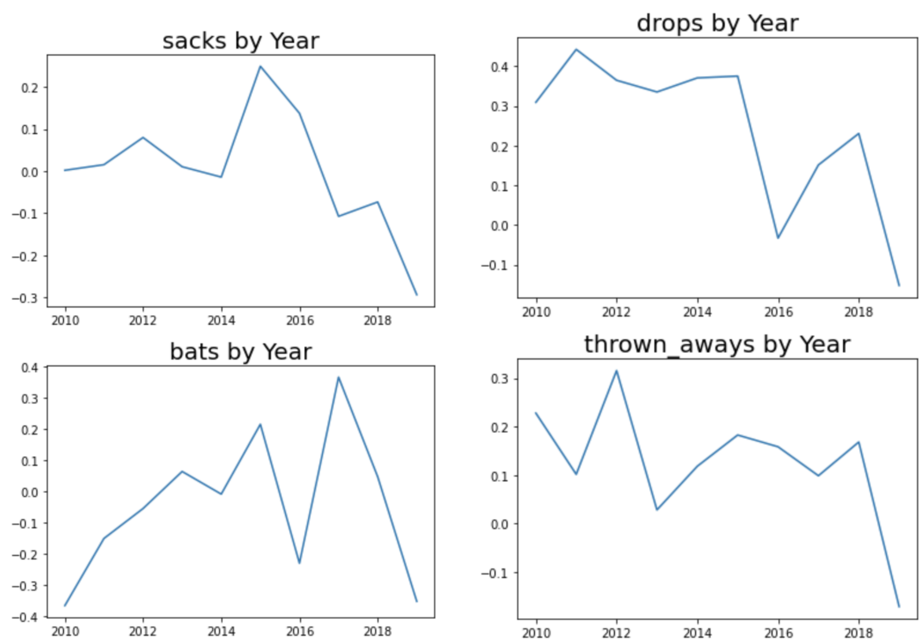


Figure C.9: QB Correlation over Time - Sacks, Drops, Bats, Throw Aways

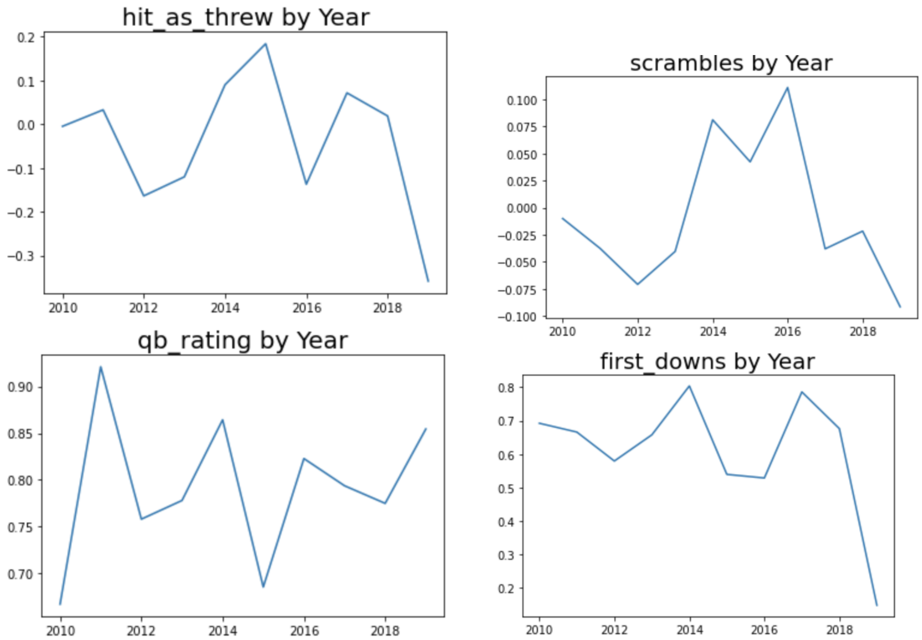


Figure C.10: QB Correlation over Time - Hit As Threw, Scrambles,QB Rating, First Downs

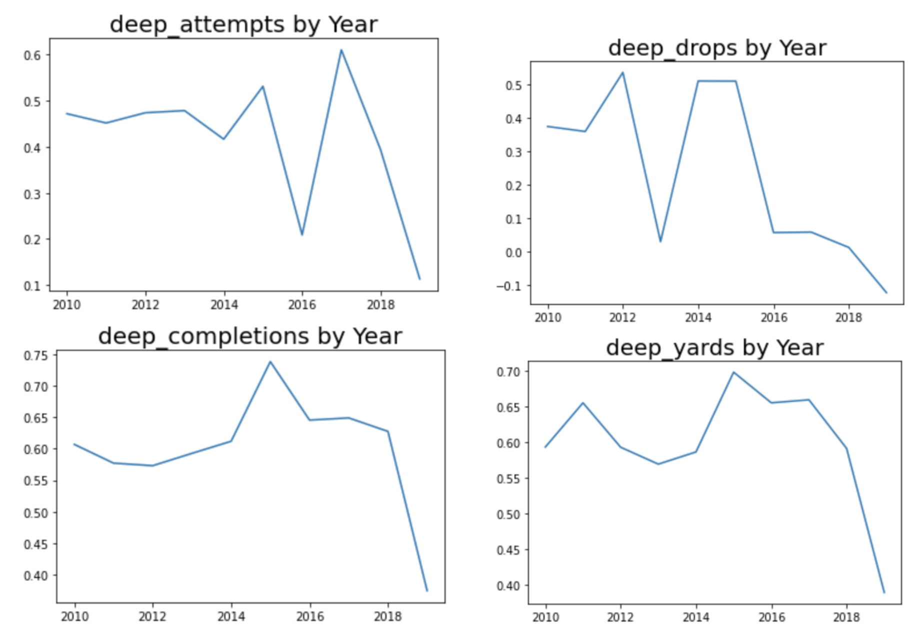


Figure C.11: QB Correlation over Time - Deep Attempts, Deep Drops, Deep Completions, Deep Yards

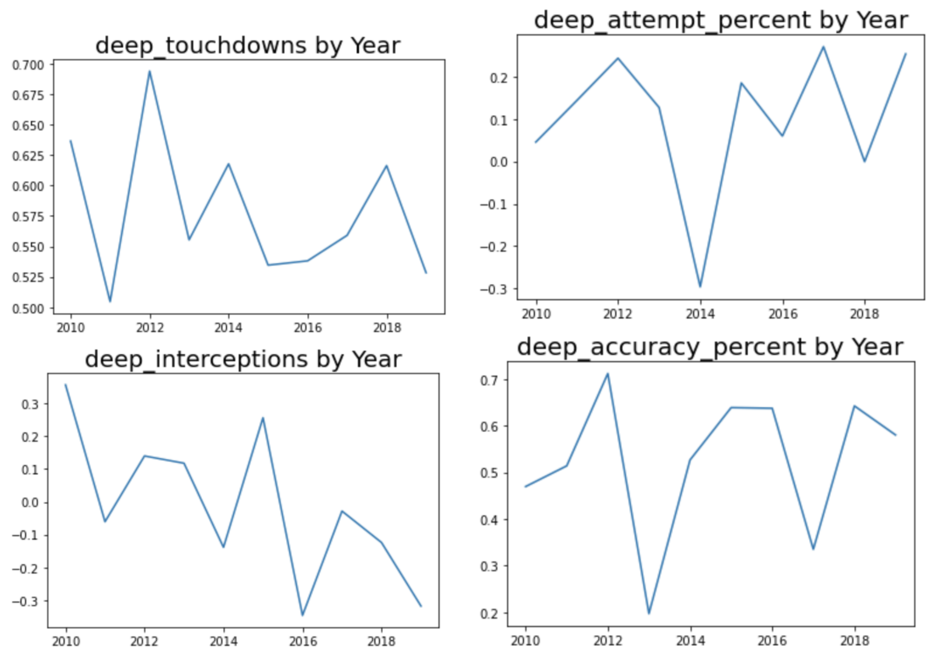


Figure C.12: QB Correlation over Time - Deep Touchdowns, Deep Attempt%, Deep Interceptions, Deep Accuracy%

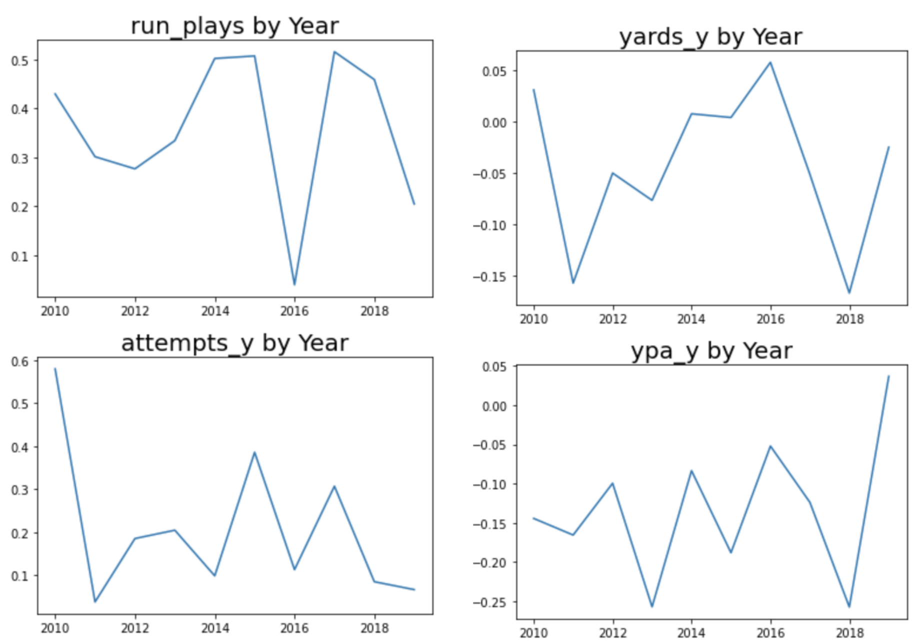


Figure C.13: QB Correlation over Time - Run Plays, Yards Rush, Attempts Rush, YPA Rush

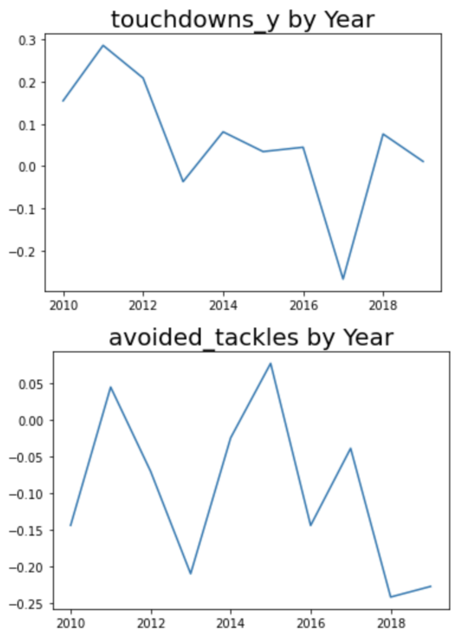


Figure C.14: QB Correlation over Time - Touchdowns Rush, Avoided Tackles

C.2 Wide Receiver

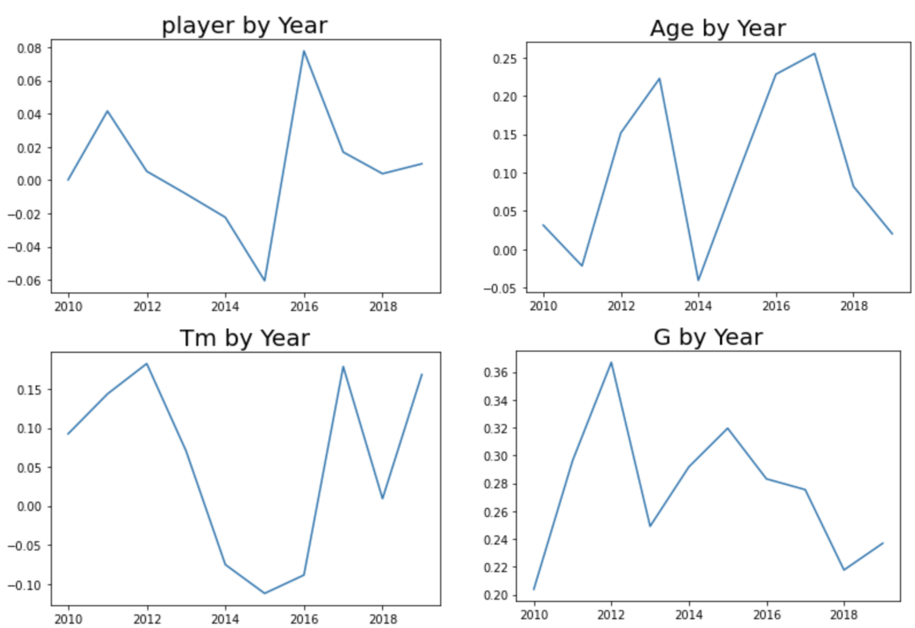


Figure C.15: WR Correlation over Time - Player, Age, Tm, G

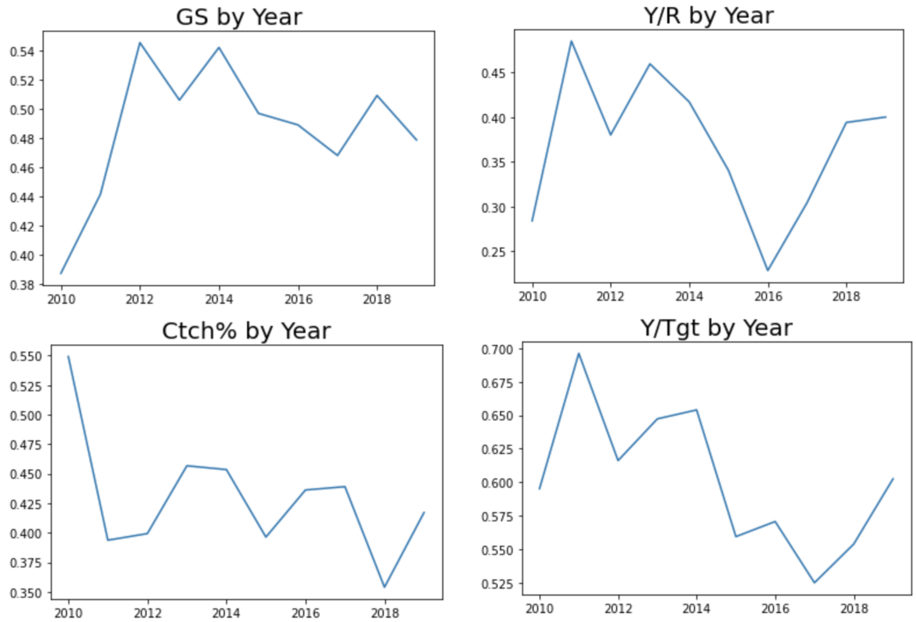


Figure C.16: WR Correlation over Time - GS, Y/R, Ctch%, Y/Tgt

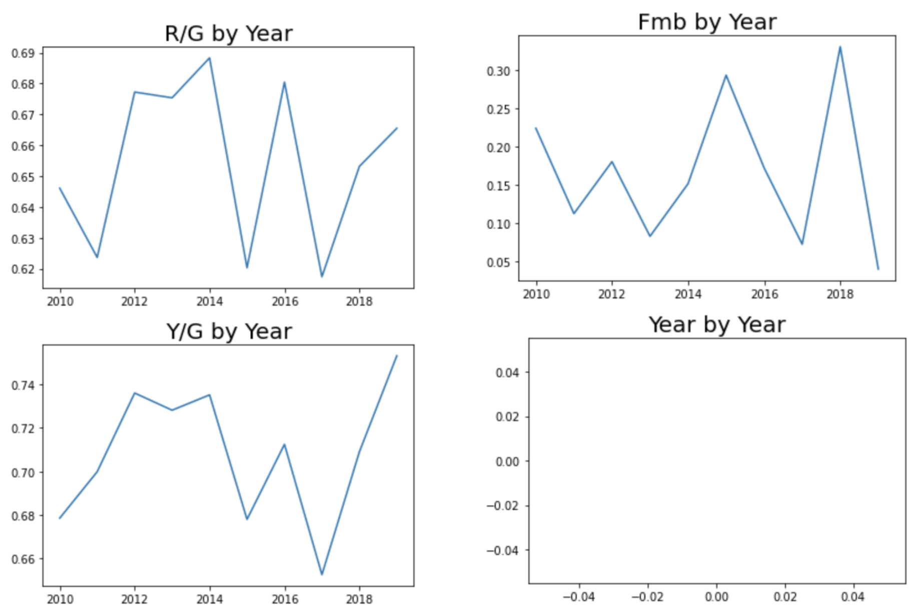


Figure C.17: WR Correlation over Time - R/G, Fmb, Y/G, Year

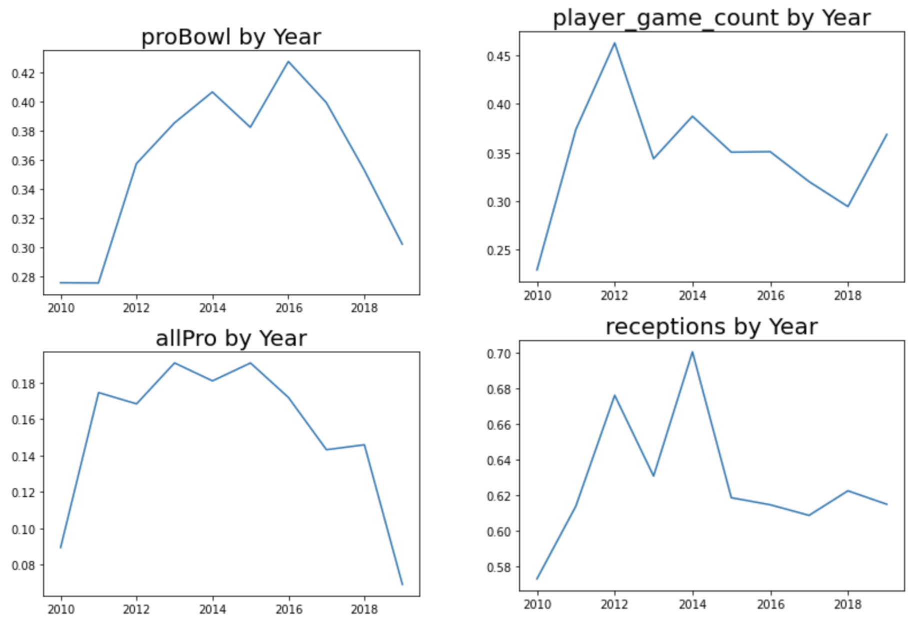


Figure C.18: WR Correlation over Time - Pro Bowl, Player Game Count, All Pro, Receptions

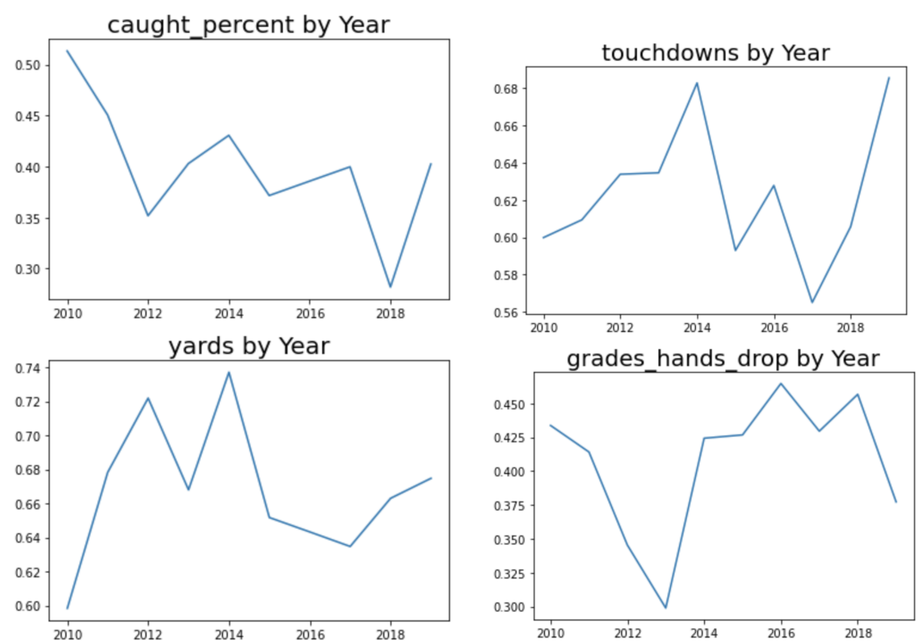


Figure C.19: WR Correlation over Time - Caught%, Touchdowns, Yards, Grades Hands Drop

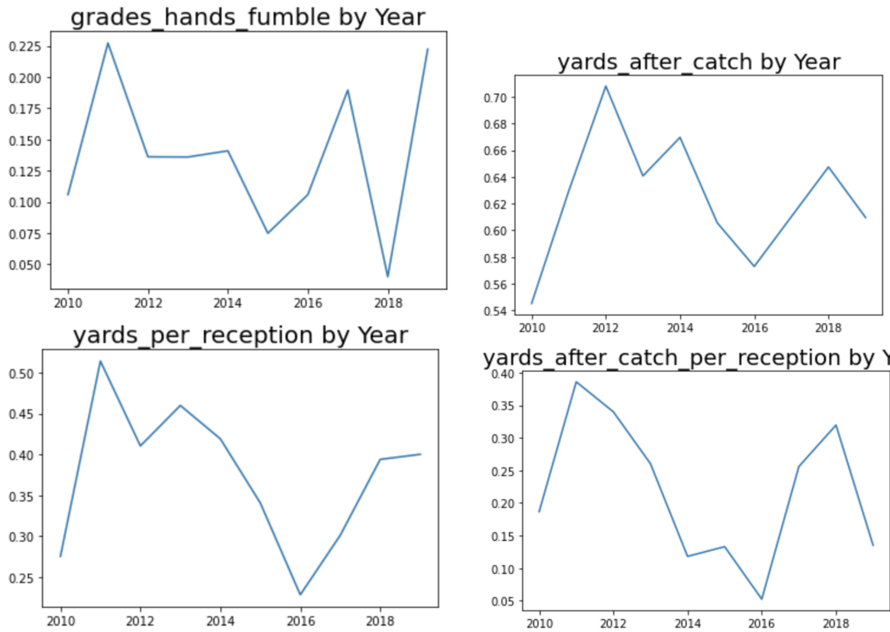


Figure C.20: WR Correlation over Time - Grades Hands Fumble, Yards After Catch, Yards Per Reception, Yards After Catch Per Reception

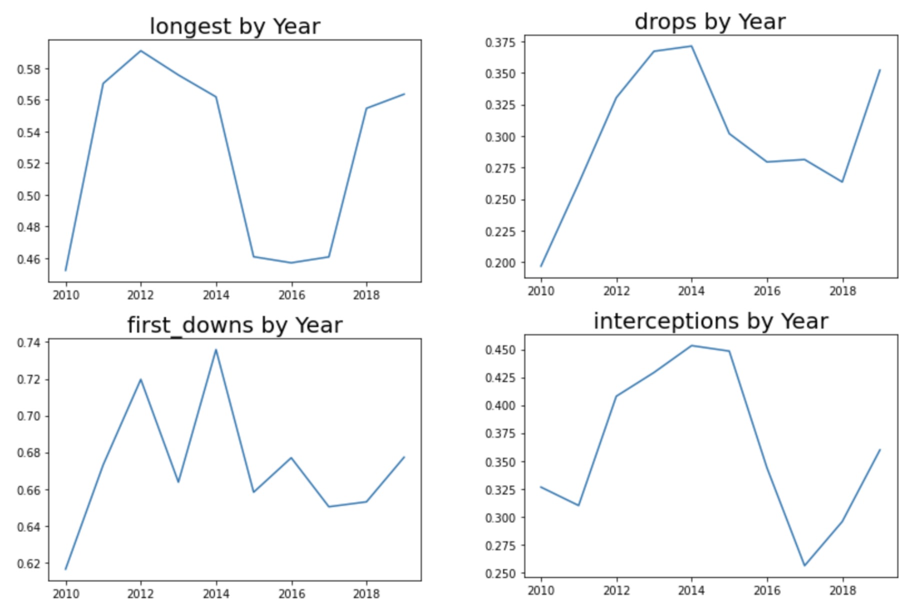


Figure C.21: WR Correlation over Time - Longest, Drops, First Downs, Interceptions

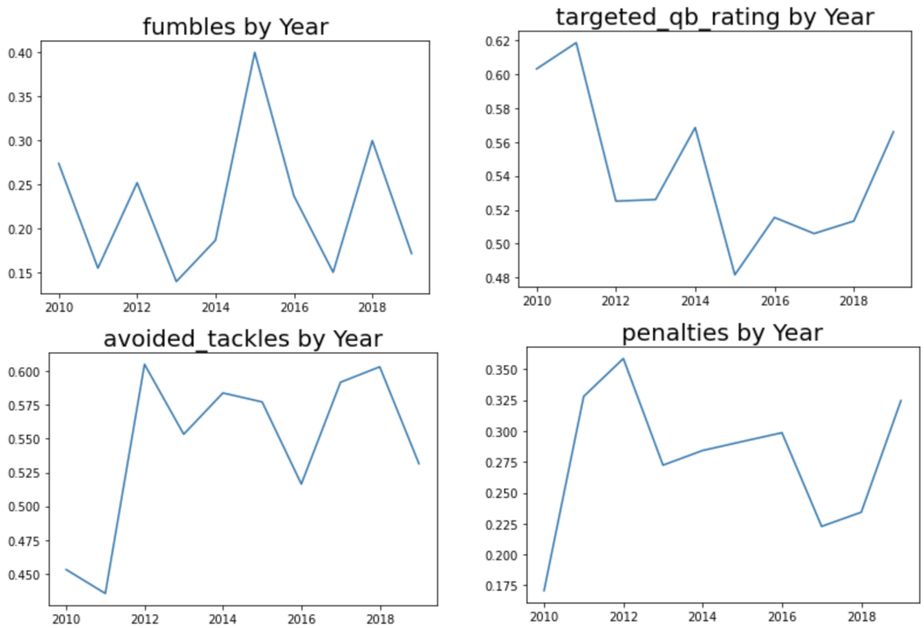


Figure C.22: WR Correlation over Time - Fumbles, Targeted QB Rating, Avoided Tackles, Penalties

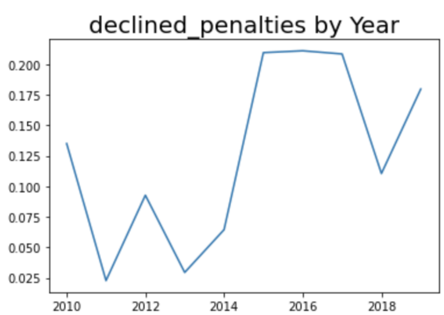


Figure C.23: WR Correlation over Time - Declined Penalties

C.3 Running Back

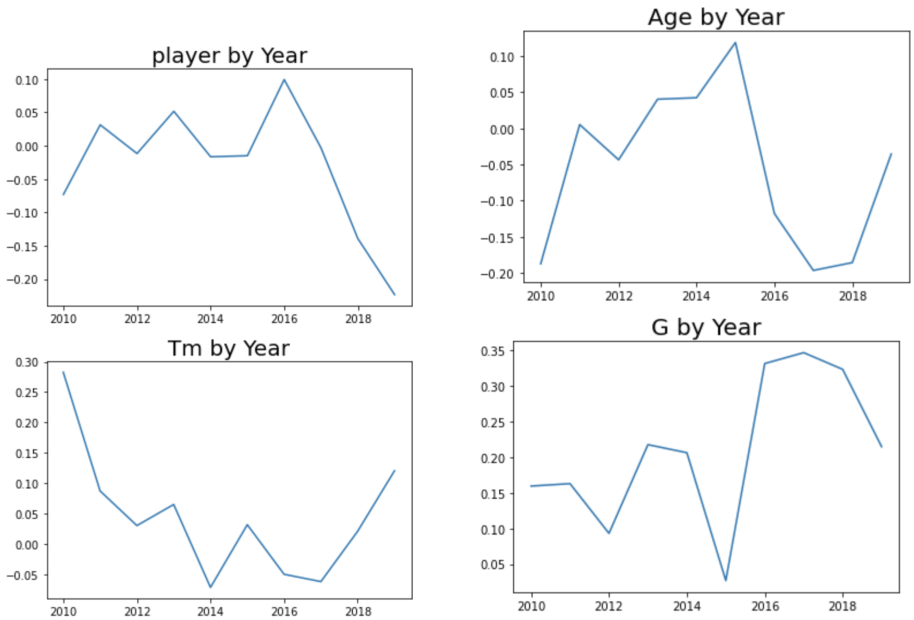


Figure C.24: RB Correlation over Time - Player, Age, Tm, G

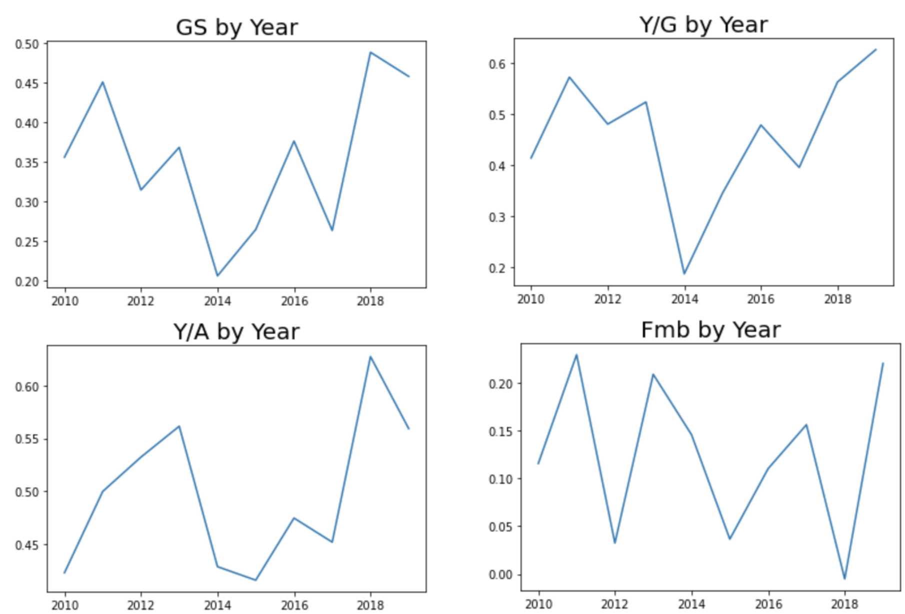


Figure C.25: RB Correlation over Time - GS, Y/G, Y/A, Fmb

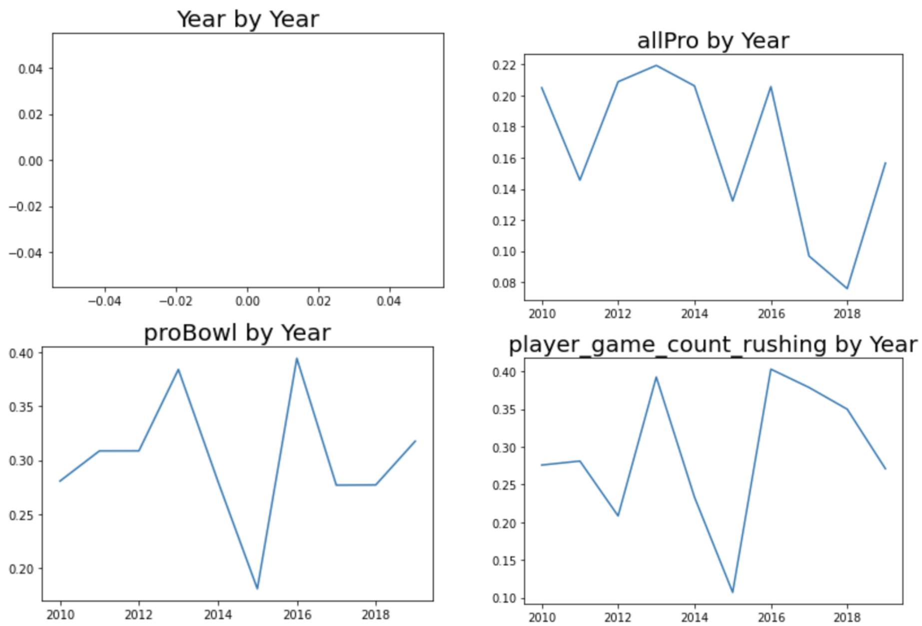


Figure C.26: RB Correlation over Time - Year, All Pro, Pro Bowl, Player Game Count

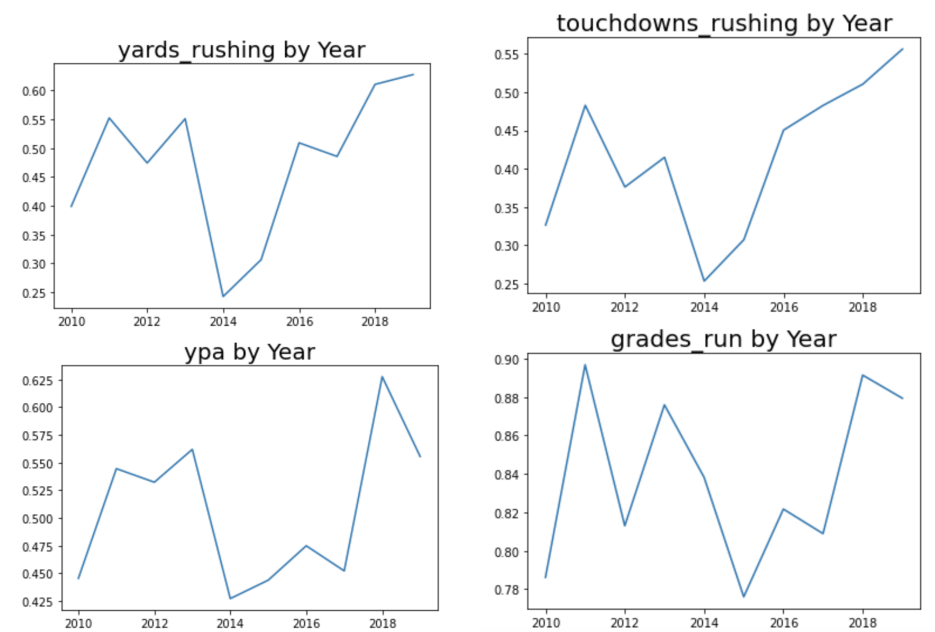


Figure C.27: RB Correlation over Time - Yards Rushing, Touchdowns Rushing, YPA, Grades Run

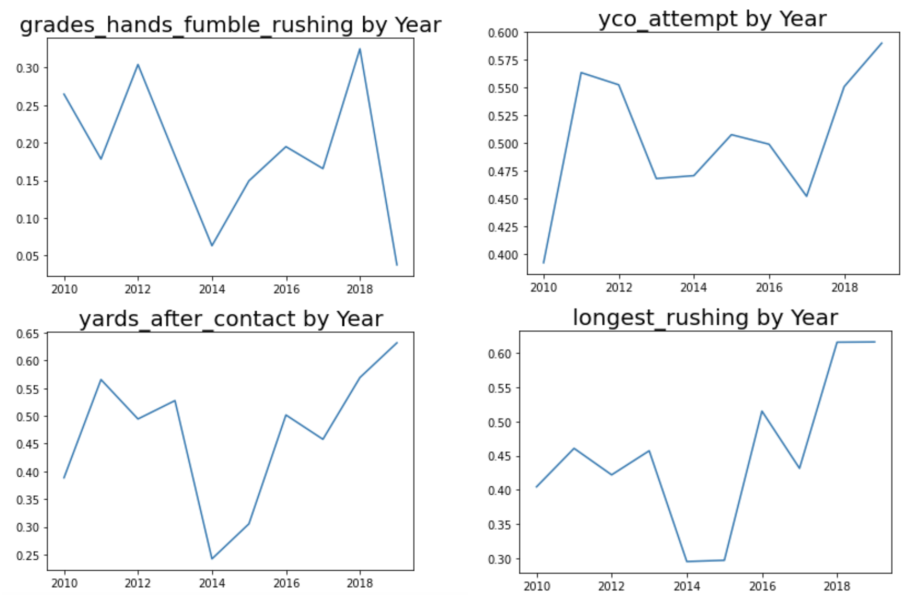


Figure C.28: RB Correlation over Time - Grades Hands Fumble Rush, Yco Attempt, Yards After Contact, Longest Rushing

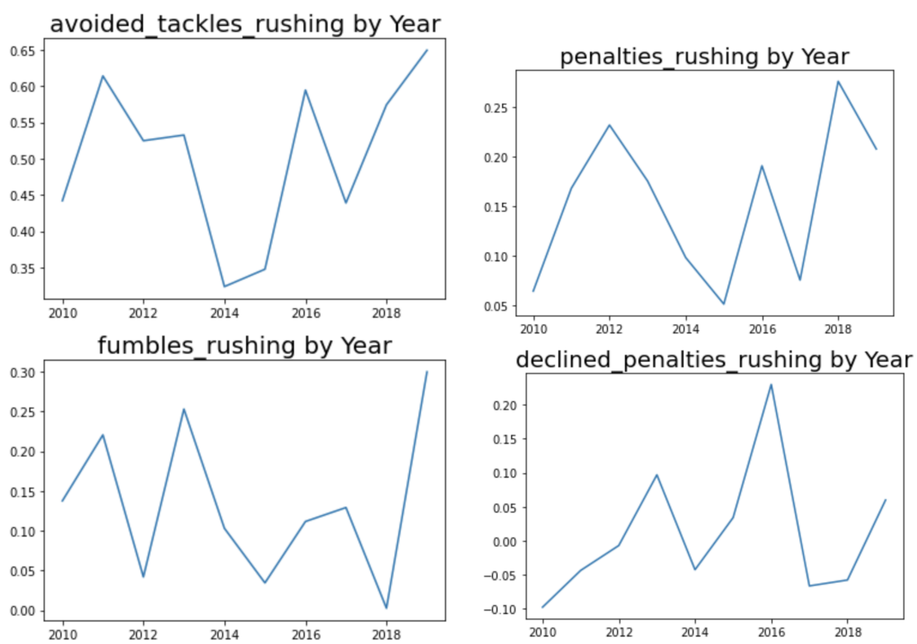


Figure C.29: RB Correlation over Time - Avoided Tackles Rushing, Penalties Rushing, Fumbles Rushing, Declined Penalties Rushing

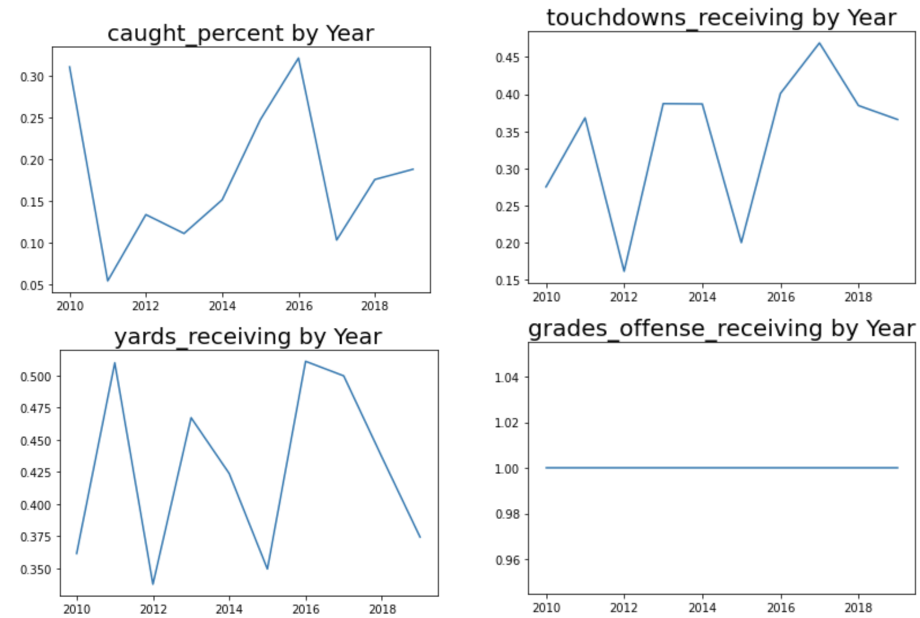


Figure C.30: RB Correlation over Time - Caught%, Touchdowns Receiving, Yards Receiving, Grades Offense Receiving

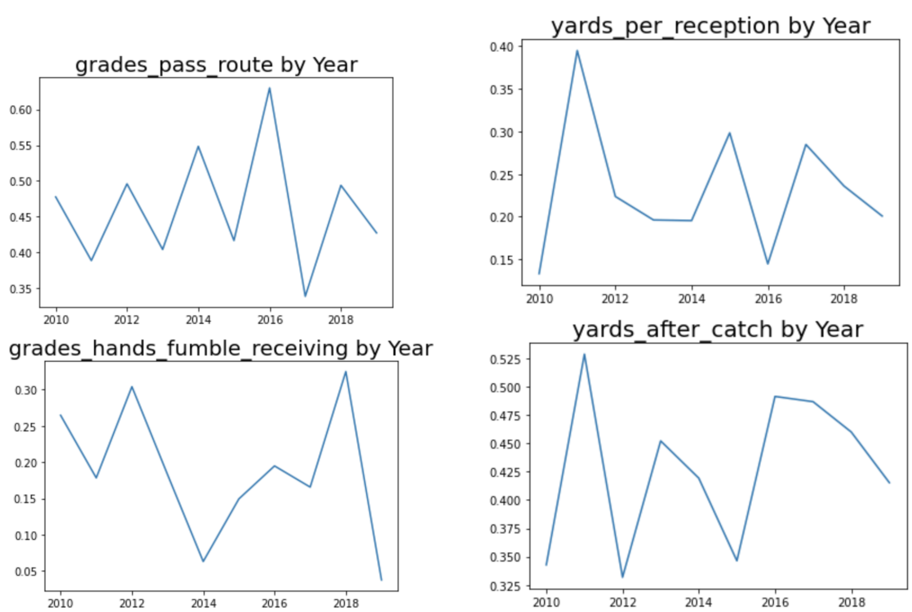


Figure C.31: RB Correlation over Time - Grades Pass Route, Yards Per Reception, Grades Hands Fumble Receiving, Yards After Catch

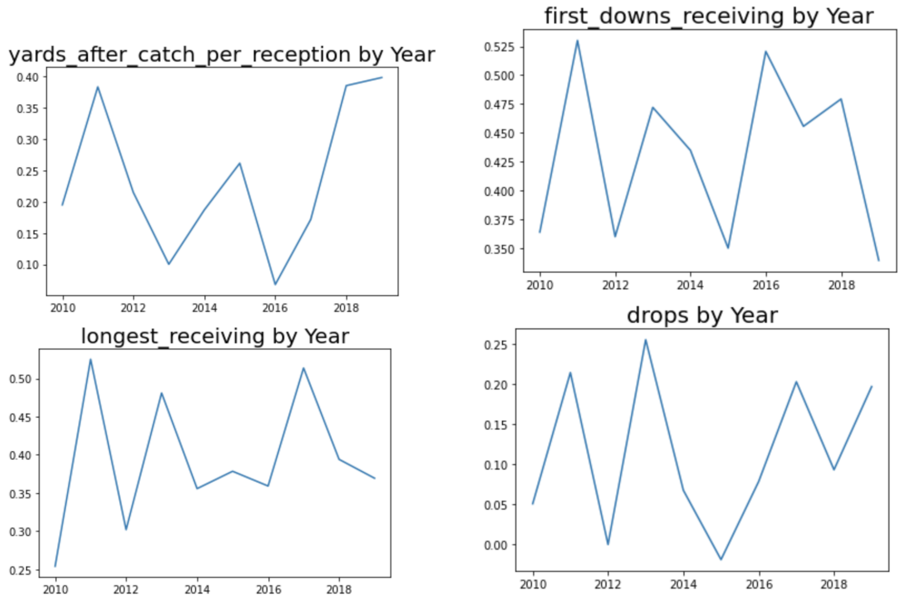


Figure C.32: RB Correlation over Time - Yards After Catch Per Reception, First Downs Receiving, Longest Receiving, Drops

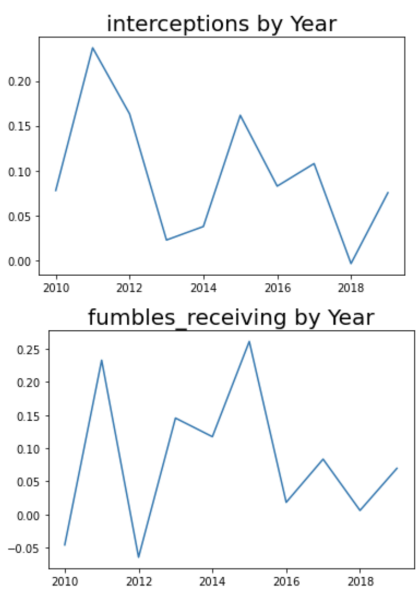


Figure C.33: RB Correlation over Time - Interceptions, Fumbles Receiving

Appendix D

Decision Tree Graphs

D.1 Quarterback

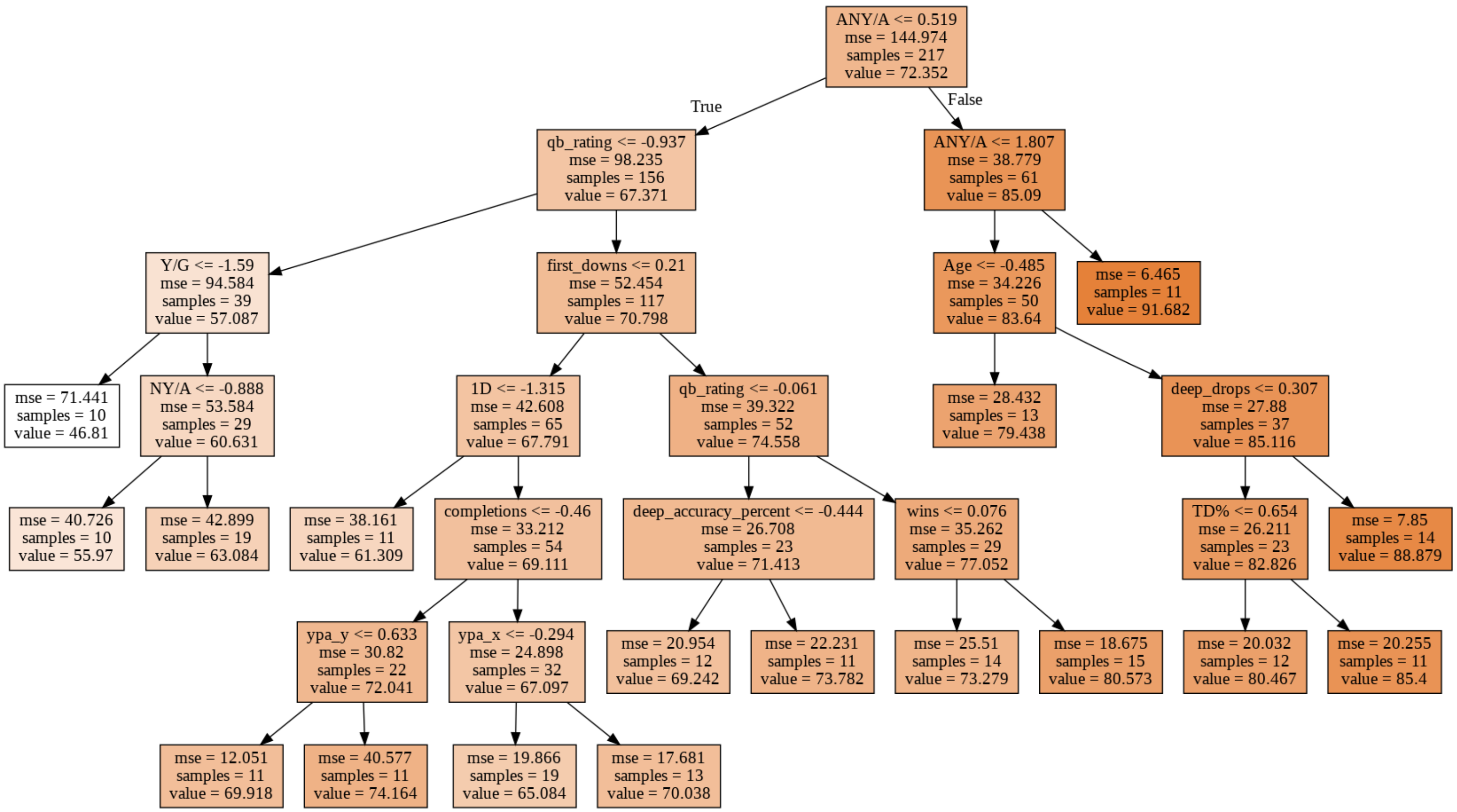


Figure D.1: QB Decision Tree Graph

D.2 Wide Receiver

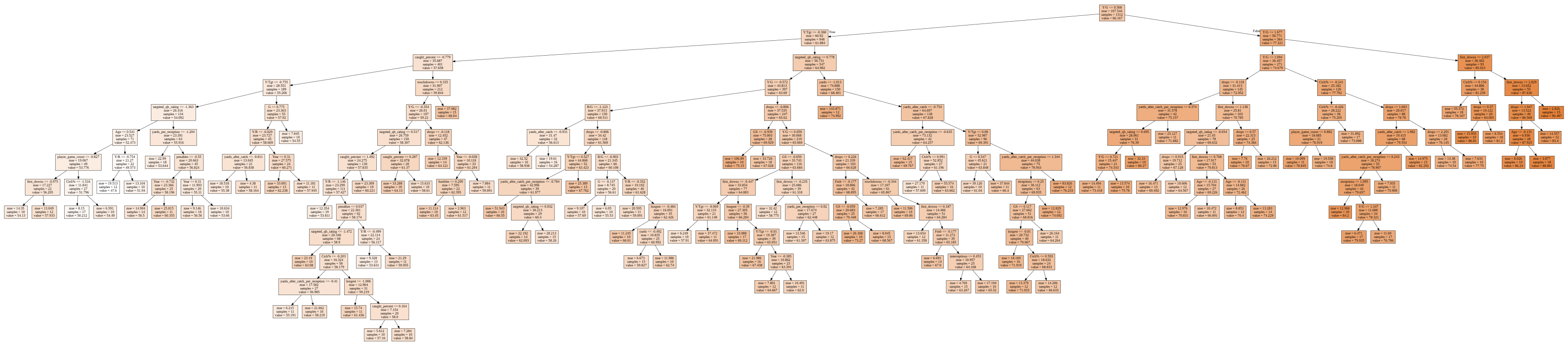


Figure D.2: WR Decision Tree Graph

D.3 Running Back

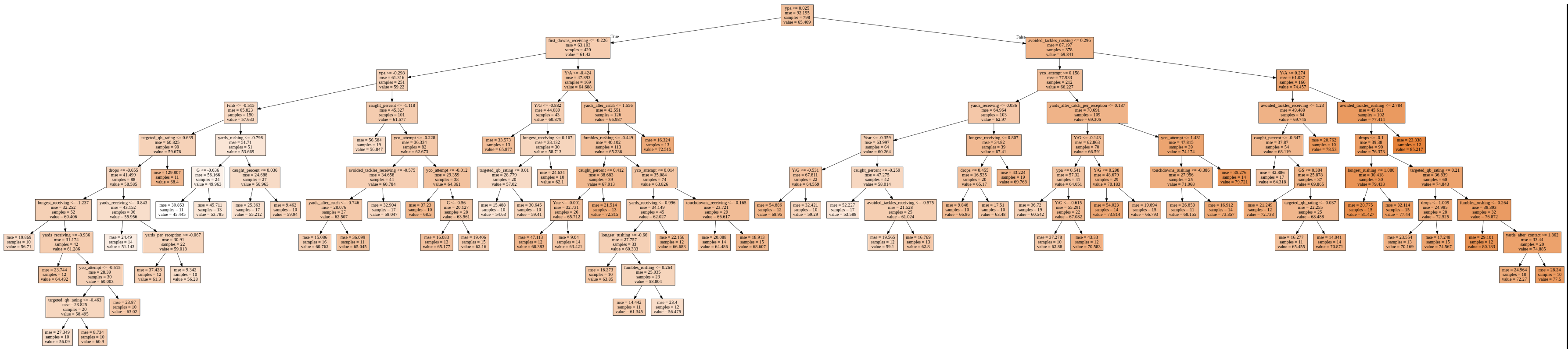


Figure D.3: RB Decision Tree Graph