Student - Animesh

Instructor - Ivan Bajic

Subject - Deep Learning System in Engineering

Date - 17 April 2021

Prediction of NBA winner using Logistic Regression and Neural Network

Abstract

Deep learning is one of the advance machine learning tool that have shown promising results in the domains of classification and prediction. One of the expanding area which require good prediction accuracy is sports prediction. Team managers and coaches are striving for machine learning models so that they can understand and formulate strategies needed to win matches. In addition, betting websites are interested in models which can increase their profits margins. These models are based on numerous factors involved in the games, such as result of historical matches, box score statistics, and opposition information. This paper provides an analysis of the literature in Machine learning for sports prediction, focusing on the application of Artificial Neural Network (ANN) along with other machine learning algorithms and their implementation on sport result prediction. In doing so, we identify the learning methodologies, dataset utilised, appropriate means of model evaluation, and specific challenges of predicting sport results. This then leads us to propose a sport prediction framework through which deep learning can be used as a learning strategy. This research will hopefully be informative and of use to those performing future research in this application area.

1. Introduction

National Basketball Association is one of the top three most popular sports in USA and a global sensation, NBA is the most followed basketball league worldwide. Over the last three decades, NBA has extended its reach to engage an increasingly larger audience, according to the Nielsen media research the viewership of NBA for 2020-21 season average around 2 million viewers per game **[1].**The NBA has capitalised on its success by continually improving the business model. For example, The NBA All-Star weekend which takes place in February each year was once just considered a midseason showcase for the top rated and most popular players. However, the event has developed from a single event into a three-day, weekend-long extravaganza, which includes a rookie game, skills challenge, three-point shootout, and a slam dunk contest. NBA All-Star weekend attracts global media attention and has become an enormous event for the sport. According to Ed Dixon from sports media, the NBA all star weekend viewership for the season 2020-21 is approximately 6 millions, though it got affected due to Covid and get reduced by 18% from last year **[2]**.

The increased popularity of the NBA has translated to a successful business model where revenue is following an increasing trend every year. Due to Covid the revenue for 2020-21 season dropped by 10% but it still manage to capture a big sum of 8.3 billion dollars **[3]**. In the season 2020-21 the top salary of a NBA player is around 43 million dollars for Stephen Curry who plays for Golden state warriors and is one of the best 3 point shooter in the NBA history. Salary of top 10 players of NBA including all big names in the league like LeBron James, Russel Westbrook, James Harden etc. is approximately 400 million dollars **[4]**.

Let us take a look into the format of NBA, total 30 teams takes part in NBA out of which 29 teams are from United States and 1 team is from Canada. All 30 teams are

divided into 2 conference of three divisions with five teams each. In a regular season, each team plays total of 82 games, 41 at home and 41 away games. A team faces opponents in its own division four times in a season (16 games). Each team played six team from other 2 divisions in its conference four times (24 games), and the remaining 4 teams three times (12 games). Finally, each team played all the teams in the other conference twice (30 games). NBA playoff begins in April, with eight teams participating from each conference, all teams have a same objective which is to win the championship. The three division winners, along with the team with the next best record from the conference are given the top four seeds. The next four teams in terms of record are given the lower four seeds. Playoff follows a knockout format. Each team plays an opponent in a best of 7 game series and winner promotes to next round while the loser gets eliminated from playoffs. Next round follows the same analogy and all but one team from each conference remains. Conference winner are called as Eastern conference champions for eastern division and Western conference champions for western division. In the final playoff round, a best-of-seven series between the champions of both conferences, is known as the NBA Finals, and is held annually in June **[5]**.

The rest of this paper is structured as follows. In section 2, we presented the motivation of the project work . In section 3, a review of the previous work done in this field. In section 4, we describe the data used for the model. In section 5, project implementation is shown which is divided into three main parts. In section 6, we displayed the experiment results. In section 7, we discussed the conclusion. Acknowledgment, Appendix and bibliography are added towards the end.

2. Motivation

Because of the worldwide viewership and revenue focused model of NBA, there is always lot of competitiveness involved in the game not only between teams but among game fans as well. That is why, predicting outcome of a game beforehand is a hot topic and everyone what's to know if their favourites are winning the next game or not, even team management and team coaches also wants to look ahead for the up-coming games and want to know the prediction of their team's result in order to analyse and modify the gameplays or training technique or line-up for a particular match for getting results in their favour.

Another community which is very much interested in sports prediction is the field of sports betting. Websites or bookmakers involved in sports betting also wants to know the game prediction for increasing their individual profits. Game analysts working with television industry or betting websites predicts a game winner and then these websites set the bid on winning or losing teams accordingly. Current NBA sport analyst for Sports Line website who have the maximum accuracy is Matt severance having prediction accuracy of 68% with 46 correct and 22 incorrect predictions out of total 68 games **[6]**. Most of the game experts have an accuracy of approximately 60% in pre-dicting more than 50 games.

Inspired by these factors the main aim of this project is to implement a deep learning model to predict the NBA game result. Along with deep learning models, Lo-gistic regression model is also implemented for comparison between the accuracy of deep learning and other machine learning model. In this project we will try to overcome the prediction accuracy of top sports analyst which is around 68% approximately as mentioned in the last para.

3. Related Work

We will review related work in terms of basketball outcome prediction along with match prediction methods used for other sports, as these methods for different sports may inspire research in basketball prediction models. Among many studies and research which has been conducted related to field of sports, forecasting the outcomes of future matches has been being investigated extensively by researchers and different statistical models have been proposed.

One of the example is from Renato Amorim Torres, he wrote a paper titled *Prediction of NBA games based on Machine Learning Methods* **[7]**. The goal of the paper was to predict the winner of a NBA game using machine learning models. He compared the results from Maximum likelihood classifier along with linear Regression model. Linear regression model achieved a performance of 67.89% which was better than the likelihood method that achieved a performance of 66.81%. For the linear regression model he used the features like win-loss percentage for both teams, point differential per game for both teams, win-loss percentage for previous games for both teams, and win-loss percentage as a visitor and home team for both teams. Another Linear Regression algorithm to predict the winner was used in a paper by M.Beckler, H.Wang, and M.Papamichael titled as *NBA Oracle* **[8]**. They achieved a result of 73% accuracy in NBA game outcome prediction, which was the best result found. G. Avalon, B. Balci and J.Guzman wrote a paper *Various Machine Learning Approaches to Predicting NBA Score Margins* **[9]**. Their goal was to predict the margin of score spread for the game and using the margin they also tried to see how accurate they were to predict the winner of a game. They used Gaussian discriminant analysis and achieved an accuracy of 65.54%, with Linear Regression their accuracy was 64.26% and Random forest helped them to attain an accuracy of 61.35%. They have used a wide range of features for

their model, total features used for training was 218 which is much more than the feature we will be using in our model. From the above past researches, it's observed that linear and logistic regression are one of the best machine learning model for sports prediction and that is the main reason we used logistic regression as a comparison model in our project.

Various Deep learning models have also been implemented in the past for sports prediction. Purucker conducted his study on predicting results in National Football League(NFL) using Artificial neural network titled *Neural network quarterbacking* **[10]**. He used data from first eight rounds, consisting of multiple features like yard Gained, turnover margin etc. ANN with backward propagation was used and he achieved an accuracy of 61%. A limitation of this approach is that only a relatively small number of features were used. McCabe and Trevathan attempted to predict the results of four different sports: National Football League, American Football League, Super Rugby and English Premier League using Multilayer perceptron **[11]**. The perceptron with 20 nodes in input layer, 10 hidden layer nodes and 1 node in output layer(20-10-1) is the most efficient configuration which gave the best accuracy for the model. The average performance of the multilayer perceptron in predicting results was around 67.5%, compared with expert tipster predictions that achieved around 60–65% accuracy. Artificial neural networks has also been applied to predict the outcome of Horse races by Davoodi and Khanteymoori in their research *Horse racing prediction using Artificial Neural Networks* **[12]**. They used the data from 100 races held in 2010, eight features has been used to train the model. This optimal network architecture (8-2-1) was the best structure when used with multiple training algorithms gradient-descent using back propagation, momentum gradient, Levenberg-Marquadt (LM), and conjugate gradient descent (CGD). It was found that with 400 epochs, the back propagation and the momentum algorithms

were most effective at predicting the winner of the race, with BP obtaining an accuracy of 77%. However, the disadvantage of back propagation was that the training time was lengthy.

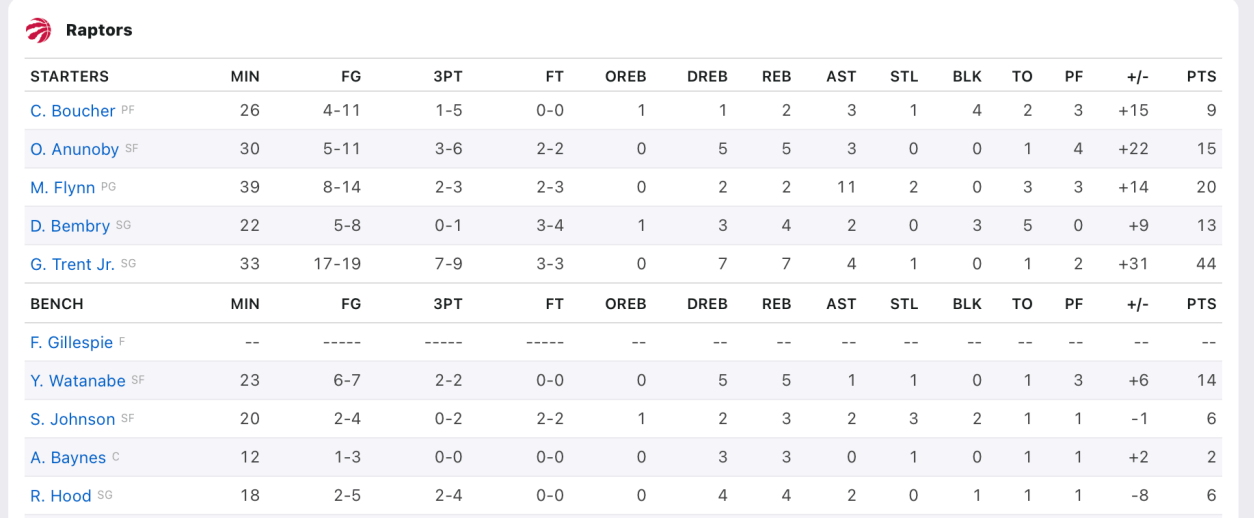
A study comparing Logistic Regression model and neural network was done by Adam Maszczyk titled *Application of Neural and Regression Models in Sports Results Prediction* **[13]**. The main focus of the study was to compare regression and neural models with respect to their accuracy of predicting sports results. Study involved a group of 116 javelin throwers from Polish national team. To verify the models, the sports results were predicted for the group of 20 javelin throwers and tested by comparing the model generated predictions with their actual data. Their results showed that the neural network models offered much higher quality of prediction than the nonlinear regression model. The absolute network error was found to be 16.77 m, versus the absolute regression error of 29.45 m.

4. Box Score Dataset

Currently, basketball is one of the most analysed sport disciplines. These methods evolved from simple stat sheets, filled out by hand during the game by assistant coaches to fully computerised procedures that automatically register all of the significant statistics of the game and calculate the necessary results.

Data for the project is Box score data taken from Paul Rossetti dataset available on Kaggle **[14]**, dataset scrapped earlier from official NBA website. Dataset is based on box score and standing statistics from the NBA season 2012 to 2018. Dataset contains added features other than general Box score stats. Figure 1 shows an example of Box score from a game between Toronto Raptors and Cleveland Cavaliers, the box score is taken from ESPN website **[15]**.

Figure - 1



Raptors

STARTERS	MIN	FG	3PT	FT	OREB	DREB	REB	AST	STL	BLK	TO	PF	+/-	PTS
C. Boucher PF	26	4-11	1-5	0-0	1	1	2	3	1	4	2	3	+15	9
O. Anunoby SF	30	5-11	3-6	2-2	0	5	5	3	0	0	1	4	+22	15
M. Flynn PG	39	8-14	2-3	2-3	0	2	2	11	2	0	3	3	+14	20
D. Bembry SG	22	5-8	0-1	3-4	1	3	4	2	0	3	5	0	+9	13
G. Trent Jr. SG	33	17-19	7-9	3-3	0	7	7	4	1	0	1	2	+31	44
BENCH	**MIN**	**FG**	**3PT**	**FT**	**OREB**	**DREB**	**REB**	**AST**	**STL**	**BLK**	**TO**	**PF**	**+/-**	**PTS**
F. Gillespie F	--	-----	-----	-----	--	--	--	--	--	--	--	--	--	--
Y. Watanabe SF	23	6-7	2-2	0-0	0	5	5	1	1	0	1	3	+6	14
S. Johnson SF	20	2-4	0-2	2-2	1	2	3	2	3	2	1	1	-1	6
A. Baynes C	12	1-3	0-0	0-0	0	3	3	0	1	0	1	1	+2	2
R. Hood SG	18	2-5	2-4	0-0	0	4	4	2	0	1	1	1	-8	6

Main fields in a general Box score is shown in Table 1. Box score contains statistics for individual players for both the teams playing the game.

Table 1

Min	Minutes played
FG	Field goal
3PT	3 Pointers
FT	Free throws
OREB	Offensive rebound
DREB	Defensive rebound
REB	Rebound
AST	Assist
STL	Steal
BLK	Block
TO	Turnover
PF	Personal foul
PTS	Points

As discussed in introduction part, there are 82 matches for each team in a season so roughly there are 1250 game each season including playoffs. Our dataset had around 44k rows for 7 season while it should be around 7000 plus. After closely studying the dataset it is observed that for each match in our dataset we have 6 rows associated with it. First 3 rows had stats for home team and last 3 rows have stats for away team, also these 3 rows have duplicate data with only change in official line man name. So we removed the duplicate entries and merged home and away teams in one row to

make it efficient to use as input to our models. Home team result is used as an output of the model, i.e. if home team wins the result is assigned a value of 1 or else 0 if away team wins. Model accuracy and loss is defined with home team result labels. Figure 2 shows data after we removed all the discrepancies.

Figure - 2

	gmDate	gmTime	seasTyp	offLNm	offFNm	teamAbbr	teamConf	teamDiv	teamLoc	teamRslt	teamMin	teamDayOff	teamPTS	teamAST	tea
0	2012-10-30	19:00	Regular	Brothers	Tony	WAS	East	Southeast	Away	Loss	240	0	84	26	
1	2012-10-30	19:00	Regular	Smith	Michael	WAS	East	Southeast	Away	Loss	240	0	84	26	
2	2012-10-30	19:00	Regular	Workman	Haywoode	WAS	East	Southeast	Away	Loss	240	0	84	26	
3	2012-10-30	19:00	Regular	Brothers	Tony	CLE	East	Central	Home	Win	240	0	94	22	
4	2012-10-30	19:00	Regular	Smith	Michael	CLE	East	Central	Home	Win	240	0	94	22	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
44279	2018-04-11	10:30	Regular	Orr	J.T.	HOU	West	Southwest	Away	Loss	241	1	83	11	
44280	2018-04-11	10:30	Regular	Foster	Scott	HOU	West	Southwest	Away	Loss	241	1	83	11	
44281	2018-04-11	10:30	Regular	Tiven	Josh	SAC	West	Pacific	Home	Win	240	2	96	22	
44282	2018-04-11	10:30	Regular	Orr	J.T.	SAC	West	Pacific	Home	Win	240	2	96	22	
44283	2018-04-11	10:30	Regular	Foster	Scott	SAC	West	Pacific	Home	Win	240	2	96	22	

Now one last data modification required to convert the string values to float or integer as our model only able to understand the mathematical values. After the final adjustment data looked like showed in figure 3.

Figure - 3

	TEAMRSLT	TEAMFGM	TEAMFGA	TEAMFG%	TEAM3PM	TEAM3PA	TEAM3P%	TEAMFTM	TEAMFTA	TEAMFT%	TEAMORB	TEAMDRB	TEAMTRB
0	1	36	79	0.4557	7	20	0.3500	15	22	0.6818	18	36	54
1	1	43	79	0.5443	8	16	0.5000	26	32	0.8125	5	31	36
2	0	38	77	0.4935	3	13	0.2308	12	31	0.3871	15	31	46
3	1	30	85	0.3529	7	25	0.2800	17	21	0.8095	14	33	47
4	0	33	91	0.3626	6	17	0.3529	16	19	0.8421	15	27	42
...	...	...	...	...	...	...	...	...	...	...	...	...	...
7376	1	33	77	0.4286	13	32	0.4063	22	27	0.8148	6	36	42
7377	1	52	105	0.4952	15	36	0.4167	11	15	0.7333	17	40	57
7378	0	41	83	0.4940	6	18	0.3333	12	28	0.4286	8	33	41
7379	1	41	89	0.4607	9	24	0.3750	11	16	0.6875	7	39	46
7380	1	38	80	0.4750	7	26	0.2692	13	20	0.6500	6	42	48

Data set is having 7381 rows, each representing one match including playoff games for season 2012 to 2018. We will change the number of features used to train the model as required by the different approaches we have taken in out project, all the implementations are explained in detail on next section of the report.

5. Experiments

All the experiments performed during the implementation phase are divided into three parts.

- Part A - Naive Approach towards sports prediction
    1. Gathering data and labels
    2. Basic machine learning models implementation
        I. K Nearest Neighbour
        II. Support Vector Machines
        III. Multilayer Perceptron
    3. Problem with Naive approach
- Part B - Data remodelling for experienced learning
    1. Data modification as mean values for both teams playing the game
    2. Machine learning models implementation
        I. Logistic Regression
        II. Neural Network
- Part C - Model tuning for better performance
    1. Feature Engineering
    2. Machine learning models implementation
        I. Logistic Regression
        II. Neural Network

Part A - Naive Approach

First approach is termed as Naive approach because it produces a model which is highly accurate but with the post game data, which means the model needs data from box score to predict the outcome of a game and as already discussed the stats in box score like points scored, assists etc are all available once the game is completed. So the model is of no use if it only predict the outcome of a game using post game stats. For this approach various machine learning model are implemented like K nearest neighbours, support vector machines along with multilayer perceptron. Maximum accuracy was approximately 99% though the state of the art system have an accuracy of 74% only. We used a default network without tuning any parameters and still the accuracy was very high, reason behind such a higher accuracy is that the model is training and predicting the outcome of a game with the stats available in box score which consist the data directly associated with result of the game. For example number of three pointers and two pointers made along with free throws gives the total score of home as well as away team and it became very easy for machine learning models to learn the relationship and figure out the actual outcome of the games. Features used for the model training involves all the columns shown in table 2.

Table 2

FGM	Field goals made	**OREB**	Offensive rebound
FGA	Field goals attempt	**DREB**	Defensive rebound
FG%	Field goals %	**TREB**	Total rebound
3PM	3 Pointers made	**AST**	Assist
3PA	3 Pointers attempt	**TO**	Turnover
3P%	3 Pointers %	**STL**	Steal
FTM	Free throw made	**BLK**	Block
FTA	Free throw attempt	**PF**	Personal Fouls
FT%	Free throw %		

There are total of 34 features selected for model training, out of these 34 features, 17 are for homes team stats and next 17 columns represents away team stats. Out of 7381 rows in the dataset, only test set is splitted from the training data. Model training is performed on 6879 rows while model testing is done on 502 rows. We successfully able to capture correct value for almost all the testing labels and accuracy was more than 99%.

Part B - Experienced Learning

Moving to next part of the experimentation, data remodelling for experienced learning is the most important part of the project implementation. After unsuccessful attempt of creating a prediction model in Naive approach, it's clear that in the field of sports prediction data remodelling is an important aspect and if not performed carefully may land the result in wrong direction. It's necessary to modify the data accordingly such that all the features are known before the game begin. After remodelling the data as an average of last 5 games for both the teams playing the match, data looked as shown in Figure 4.

Figure - 4

	TEAMFGM	TEAMFGA	TEAMFG%	TEAM3PM	TEAM3PA	TEAM3P%	TEAMFTM	TEAMFTA	TEAMFT%	TEAMAST	TEAMSTL	TEAMBLK	TEAMPF	OPI
0	32.0	90.0	0.35560	8.0	32.0	0.25000	12.0	20.0	0.60000	26.0	11.0	10.0	19.0	
1	35.0	80.5	0.43600	6.0	18.5	0.36185	13.5	21.5	0.64005	20.5	4.0	7.5	16.0	
2	38.0	75.0	0.50455	7.5	18.0	0.42500	18.5	22.5	0.82935	21.5	6.0	5.0	19.0	
3	0.0	0.0	0.00000	0.0	0.0	0.00000	0.0	0.0	0.00000	0.0	0.0	0.0	0.0	
4	38.5	74.0	0.52770	3.0	9.5	0.31110	24.0	31.5	0.76215	27.5	7.5	7.0	20.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
7351	38.8	87.8	0.44146	8.0	27.0	0.29708	12.2	19.0	0.62942	26.2	6.6	5.2	15.4	
7352	46.6	92.0	0.50654	12.2	30.8	0.39534	14.2	20.4	0.70070	31.2	9.4	6.4	20.6	
7353	40.4	85.6	0.47198	5.0	18.4	0.26390	19.6	27.6	0.69194	23.4	6.6	3.4	20.8	
7354	38.0	87.6	0.43570	9.4	32.2	0.29780	15.2	18.4	0.81384	20.0	7.4	2.6	20.0	
7355	35.4	86.6	0.40956	8.0	24.2	0.33584	11.8	17.0	0.67900	20.6	6.2	4.6	22.4	

For better understanding of how average data is calculate we will take an example of one game in the dataset. To begin with, we will start with the point where the dataset had 2 rows for each game played. First step is to assign a game id to each game(row) to be able to work on all games by a game id, one important thing is that mean calculation must start from 51st row, so we skipped first 50 rows from dataset so that each team will have at least 1 game played previously for the mean calculation. Now let's take 26th game, this game is between Sacramento kings and Cleveland Cavaliers, game id is 26. Home team abbreviation is SAC and away team is CLE, two functions are used for the mean calculation. First function is get_team_mean where we have to initialise the columns for both Home and Away teams for which we want to calculate the mean. Next step is to find previous 5 games for SAC as well as CLE and assign that to HOME and AWAY variable. Then with another function get_column_mean which uses a predefined function DataFrame.describe() from Panda library in python to find the mean values of all the columns for HOME and AWAY variables, get_column_mean then return that mean values corresponding to all the columns which is then appended to a list type variable named game and results are separately stored in another variable result. Last step is to convert these lists to a NumPy array to be used conveniently inside the program. For our example result of game 26 was 0 that means home team loses the match and Cleveland Cavaliers won so we appended result with a value of 0 and game variable is appended with 34 values of individual columns for both SAC and CLE (17each).

Another important aspect of the project is selection of Models. Primary aim of the project is using a deep learning models to predict the result, Artificial neural network is implemented for this purpose along with one non deep learning model. From literature review it is clear that Linear or logistic regression models are proved to be highly

efficient while working with sports prediction, having two different models also help to verify the models accuracy on all the different points so logistic regression models are used alongside ANN for achieving higher accuracy on the dataset through various experiments.

Logistic Regression :

General linear regression models are a set of regression methods for which the output value is assumed to be a linear combination of input values **[16]**. Mathematical formula for Linear regression is:

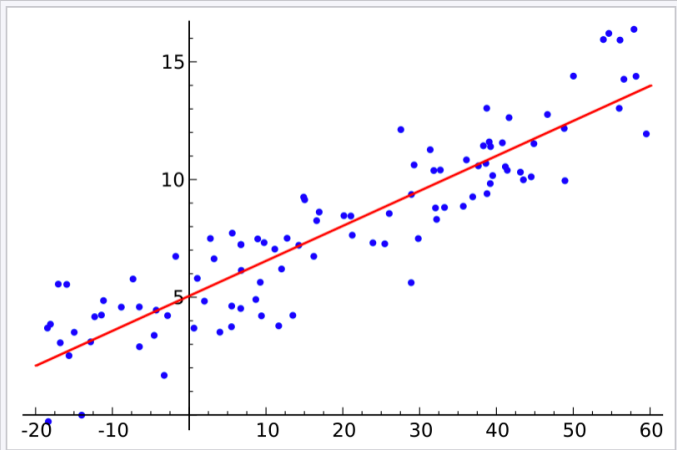
$$y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i = \mathbf{x}_i^{\mathsf{T}} \boldsymbol{\beta} + \varepsilon_i, \qquad i = 1, \ldots, n,$$

Where:

- y is the observed value of dependent variable
- xi represents input values
- β is the weight
- ε_i is the error term

Linear regression fits a linear model with weights β by minimising the residual sum of square between the actual response Y in the dataset and predicted response y which is equal to Xβ. This is called as Ordinary Least Square method. Once the model is fitted with best possible weight the linear regression can generate continuous output variables given a set of input variables. Figure 5 shows general representation of Simple linear regression which have one independent variable.

Figure - 5



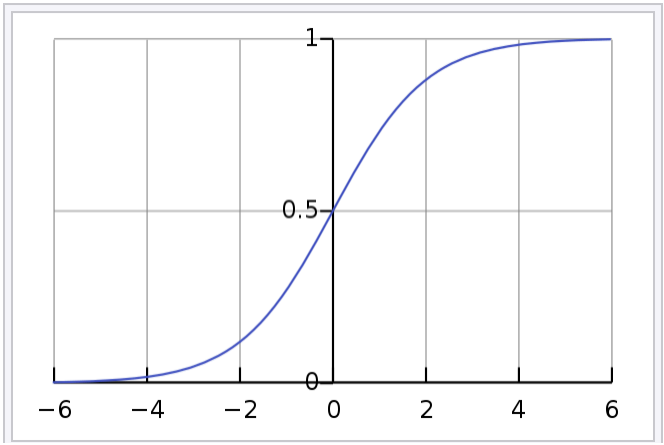
Credit - Wikipedia [16]

On the other hand Logistic Regression is a variation for linear regression but it is a classification model for a binary output variable problem **[17]**. The output variable of the logistic model represents the log odds score of the positive outcome in the classification.

$$\operatorname{logit} p = \ln \frac{p}{1-p} \quad \text{for } 0 < p < 1 \,.$$

Log odd score is a continuous variable which is used as input in a logistic function which is similar to sigmoid function, an example of a logistic function is showing in figure 6.

Figure - 6



Credit - Wikipedia [17]

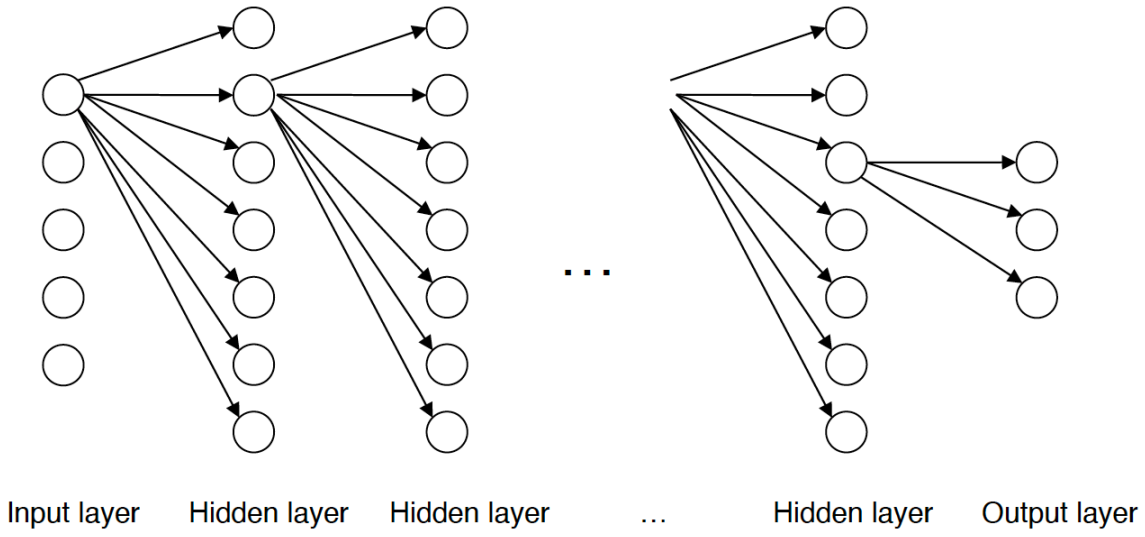
$$\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$$

Logistic function output a number between 0 and 1 which is chosen to be the probability of the positive classification outcome given the input variable. Logistic function never reaches a value of exact 0 or 1, it's always in between and any outcome above 0.50 is taken as positive output while any outcome below 0.51 is taken as negative class output.

Artificial Neural Network:

Artificial neural network also called as Neural network are machine learning model which is inspired by biological neural network inside our brain. Neural network is a collection of interconnected neurons, in a layered structure topology which is shown in figure 7.

Figure - 7

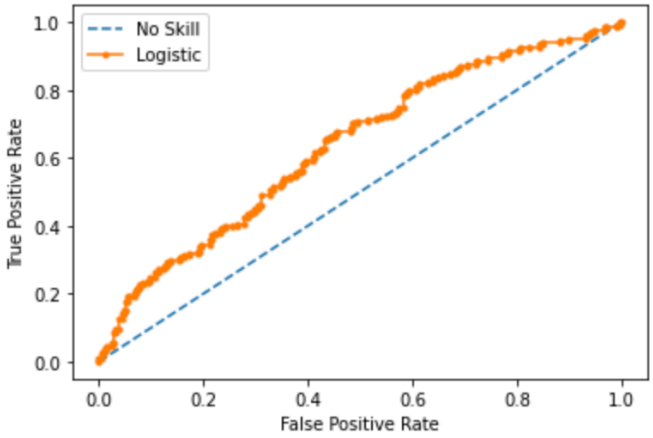


Credit - Course notes

Neural network follows a similar concept of firing as used by neurons in our brain, whenever a neuron gets an input above threshold it fires to the next neuron and this is how the communication happens. Weights associated with interconnected components are continuously changing to accomplish high levels of Neural network prediction accuracy. Back propagation is the technique used to update the weight matrix of different layers. Every time a full set of data passes through the network also called an iteration or epoch, network will update the weight matrix associated with each layer according to back propagation. Final models with desired accuracy is then saved which consist of weight matrices from trained model, saved model can be used to predict the result on the new data.

Next part is to fit the data, we are going to implement the logistic regression and neural network model on the mean average data we derived earlier. First we implemented a simple logistic regression model along with a K fold cross validation variation to find our the best among these and then compare it with the neural net. Simple logistic model with 'lgbfs' algorithm is used on input data with 6855 rows and Receiver operating characteristics curve is used to predict the probability of outcome **[18]**. It is a plot of the false positive rate (x-axis) versus the true positive rate (y-axis) for a number of different candidate threshold values between 0.0 and 1.0. Put another way, it plots the false alarm rate versus the hit rate . Logistic regression with K fold cross validation is used with 10 folds cross validation and repeat set value as 3. Figure 8 shows a representation of how ROC curve is plotted for Simple regression model along with a dummy model of no learning. No learning model with dotted lines is used to understand how much the model is learning as compare to a no skill model.

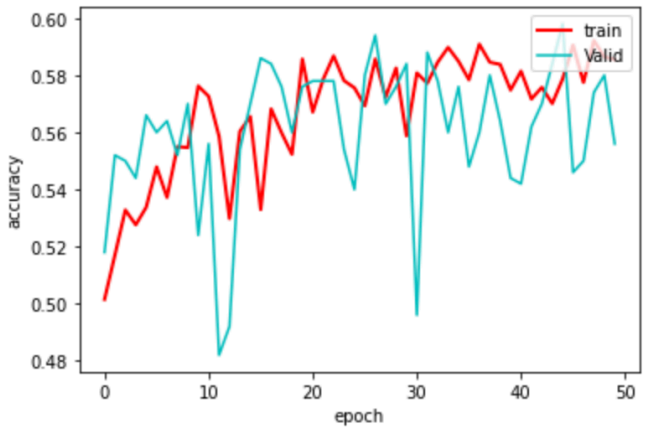
Figure - 8



We got higher accuracy on the simple Logistic regression model as compare to logistic model with K fold cross validation which is used for 10 folds. There is a drawback in using a K fold model is that it randomly used the input data for training set and validation set which is not best in the case of sports prediction. We must need to train the model on the dataset in line with the date and not shuffle or use the data randomly.

Training Neural Network with one hidden layer of 130 neurons at 50 epoch and learning rate of 0.01 gave the best accuracy for neural net which is slightly less than logistic regression model. Figure 9 shows a plot of Training accuracy with test accuracy of the neural network model.

Figure - 9



Now this accuracy from simple logistic regression model and neural network will be used as a baseline for our project. We have a minimum value now and we want to overcome sports analysts accuracy of 68%. Next part of the experiment section is data modelling and model improvements in which we will try to increase the model accuracy as much as possible.

Part C - Model Tuning

This section of experiments comprises of Model tuning for increasing the accuracy of the models. There are 3 main modification done on the Model which we will see one by one. First attempt was to increasing the Features in the model, additional features are introduced in the input data which were not used previously. Also, with the help of basketball realtime website [18] there are some additional features already available in our dataset. New features added as the input data for both home and away teams are shows in table 3.

Table 3

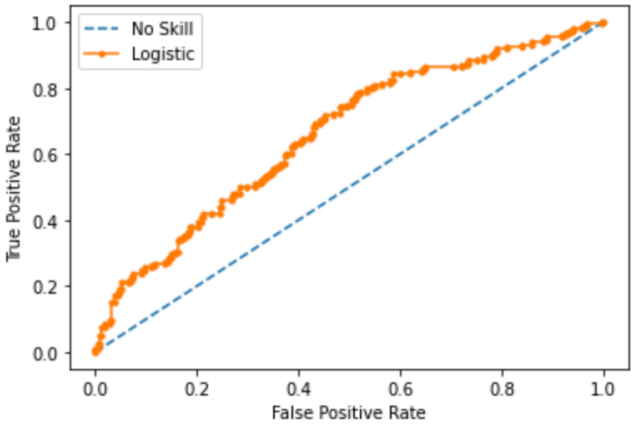
DayOff	Day off
2PA	2 Pointer attempted
2PM	2 Pointer made
2P%	2 Pointers %
FIC	Floor Impact ratio

Floor Impact counter is a mathematical approach to encompass all aspects of the ox score into a single statistic **[19]**. Mathematical formula is shown below :

$$Formula : (Points + ORB. + 0.75DRB + AST + STL + BLK - 0.75FGA - 0.375FTA - TO - 0.5PF)$$

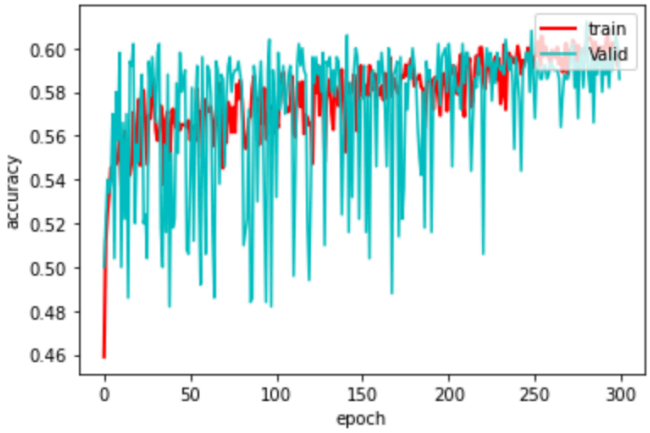
All the other columns added are just from box score stats. Logistic regression model with ‘lbfgs’ algorithm is used on 6855 rows of training data and 501 rows of test data. ROC curve is used for model accuracy, Figure 10 shown below is the ROC curve for this model. Yellow curve shows the actual learning of the model with respect to no skill model. Accuracy for this model was around 66.3%.

Figure - 10



Neural network trained with the same dataset with one hidden layer of 130 neurons. Having ReLU and Sigmoid activation on hidden and output layers along with Binary cross entropy loss with 0.001 learning rate and 300 epochs have an accuracy of 60.28% on test data. Figure 11 shows the accuracy plot of training and test data of neural network.

Figure - 11



Next approach is to normalise the input data to increase the accuracy. We used a Min Max Scalar function from SKlearn library **[20]** and full data sheet is converted to normalised data in between the range of (0,1) so the new data after normalization looked as shown in figure 12 shown below:

Figure - 12

	TEAMFGM	TEAMFGA	TEAMFG%	TEAM3PM	TEAM3PA	TEAM3P%	TEAM2PA	TEAM2PM	TEAM2P%	TEAMFTM	TEAMFTA	TEAMFT%	TEAMORB	T
0	0.645161	0.865385	0.637985	0.404040	0.650407	0.473664	0.736041	0.6000	0.656846	0.352941	0.483092	0.636578	0.810811	
1	0.705645	0.774038	0.782231	0.303030	0.376016	0.685582	0.786802	0.7250	0.741055	0.397059	0.519324	0.679069	0.518018	
2	0.766129	0.721154	0.905217	0.378788	0.365854	0.805229	0.723350	0.7625	0.845582	0.544118	0.543478	0.879910	0.270270	
3	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000	0.000000	0.000000	0.000000	0.000000	0.000000	
4	0.776210	0.711538	0.946751	0.151515	0.193089	0.589428	0.818528	0.8875	0.889631	0.705882	0.760870	0.808613	0.427928	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
7351	0.782258	0.844231	0.792027	0.404040	0.548780	0.562865	0.771574	0.7700	0.806057	0.358824	0.458937	0.667791	0.414414	
7352	0.939516	0.884615	0.908788	0.616162	0.626016	0.749034	0.776650	0.8600	0.888060	0.417647	0.492754	0.743417	0.423423	
7353	0.814516	0.823077	0.846783	0.252525	0.373984	0.500000	0.852792	0.8850	0.837900	0.576471	0.666667	0.734123	0.594595	
7354	0.766129	0.842308	0.781693	0.474747	0.654472	0.564229	0.703046	0.7150	0.832407	0.447059	0.444444	0.863454	0.486486	
7355	0.713710	0.832692	0.734795	0.404040	0.491870	0.636302	0.791878	0.6850	0.699578	0.347059	0.410628	0.720394	0.495495	

Logistic regression model with similar structure as last part is implemented and it reduced the logistic regression model accuracy by approximately 1% to 65.4%. Neural network configuration with learning rate as 0.001, batch size of 64 and Neuron set of 44-130-1 is used and normalization increased the neural network model accuracy by more than 2%. Training set accuracy increased to 64% and test set accuracy is now approximately 62% after 100 epochs. Figure 13 shows ROC curve for Logistic regression model and Figure 14 shows Neural net accuracy plot.

Figure - 13

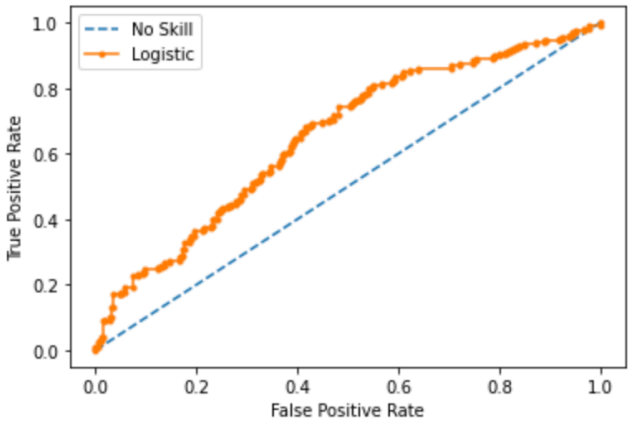
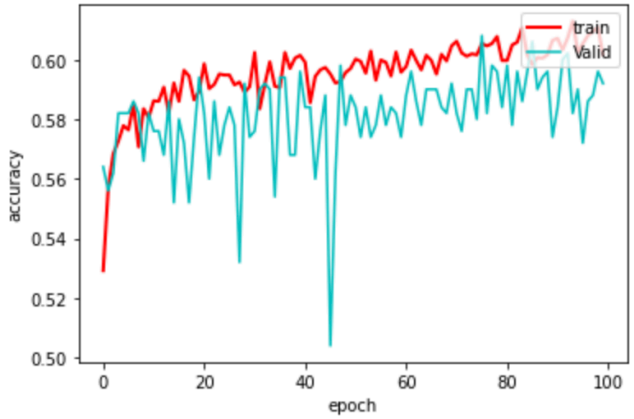
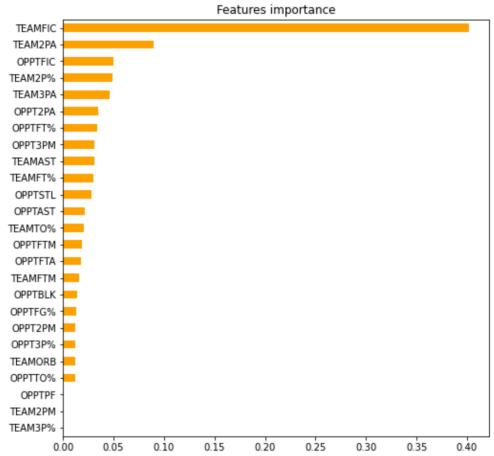


Figure - 14



Third approach was more analytical, one important analysis we did was PCA analysis on the normalised data to analyse which component among the features in dataset is contributing most to the model accuracy, for this part we used Decision tree classier model from SKlearn library which have a feature importance function which can be used to find out the importance of various features in the mode **[21]**. Figure 15 shows the feature important of our model in part B.

Figure - 15



Another analysis tool called as Principal component Analysis is used to find the Explained Variance of each individual features and also Cumulative Explained Variance of each component cumulative with each component above in the Table 4 shown below.

Table 4

No.	Explained Variance	Cumulative Explained Variance	No.	Explained Variance	Cumulative Explained Variance	No.	Explained Variance	Cumulative Explained Variance	No.	Explained Variance	Cumulative Explained Variance
1	0.172388	0.172388	12	0.028480	0.762965	23	0.010736	0.967154	34	0.000085	0.999884
2	0.114555	0.286943	13	0.026995	0.789959	24	0.008542	0.975696	35	0.000080	0.999963
3	0.089625	0.376568	14	0.024746	0.814705	25	0.007689	0.983385	36	0.000032	0.999995
4	0.065819	0.442387	15	0.024091	0.838796	26	0.006815	0.990199	37	0.000003	0.999999
5	0.061475	0.503862	16	0.023185	0.861981	27	0.005508	0.995707	38	0.000001	1.000000
6	0.046904	0.550765	17	0.020405	0.882386	28	0.002412	0.998120	39	0.000000	1.000000
7	0.040643	0.591408	18	0.019960	0.902346	29	0.000728	0.998848	40	0.000000	1.000000
8	0.039981	0.631389	19	0.015527	0.917873	30	0.000384	0.999232	41	0.000000	1.000000
9	0.036595	0.667983	20	0.014543	0.932416	31	0.000204	0.999436	42	0.000000	1.000000
10	0.034114	0.702098	21	0.013062	0.945478	32	0.000192	0.999628	43	0.000000	1.000000
11	0.032387	0.734484	22	0.010940	0.956418	33	0.000170	0.999798	44	0.000000	1.000000

Main advantage of this analysis is we can reduce the dimensionality of our data if we have too many numbers of features which will help in memory computation. Right

now we do not have much columns but we can include multiple new columns from RAW data and then use Dimensionality Reduction to get a Model which can be highly efficient and computationally efficient as well. From Table 4 we can observer that on 24th Row the Cumulative Variance is ~98%. This implies that we can reduce the total feature of the model from 44 to 24 with just 2% loss of information.

After reduction of dimensionality of data from 44 to 24 features, data was looking like shown in Figure 16.

Figure - 16

```
array([[-1.69338568e-01,  2.45497991e-02, -4.48525064e-01,
         7.86557248e-02, -1.98594199e-01, -2.22923955e-01,
         2.50604645e-01,  1.87233970e-01,  7.64004436e-02,
        -3.74209222e-01,  9.74643823e-02,  2.16037763e-01,
        -2.77911376e-02,  3.98030934e-01, -9.89292789e-03,
         3.34023846e-01,  4.44393252e-01, -6.41701392e-02,
        -1.54748880e-01,  2.84896108e-01,  1.43710707e-01,
        -2.42421832e-01,  5.57537032e-02,  1.60828820e-01],
       [-8.69357242e-01, -1.44967287e-02,  7.14155386e-02,
         1.64893478e-01,  1.68786851e-01,  1.69851009e-01,
         2.29050704e-01, -8.74556566e-03,  9.54286030e-02,
        -2.98058911e-01,  2.73250505e-01,  4.98114425e-01,
         1.05534603e-01,  2.45273048e-01,  2.40525546e-01,
        -2.79589277e-01, -1.02311269e-02,  2.30848262e-01,
         5.25086402e-02, -1.48246848e-01, -4.72348616e-03,
         6.83368099e-02,  1.52066123e-01,  9.67499197e-02],
```

Logistic regression model and neural net is implemented with almost similar configuration for dimensionality reduction data and there were no improvement in the model accuracy. ROC and accuracy plot is shown below in Figure 17 and 18 respectively.

Figure - 17

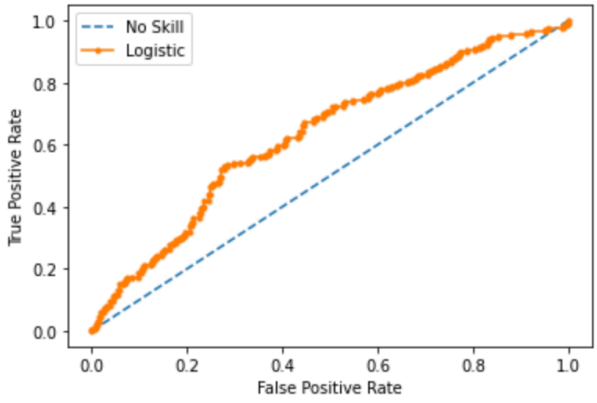
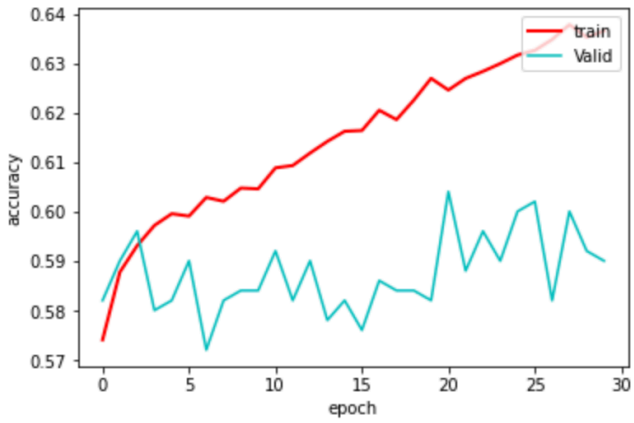


Figure - 18



This method did not proved to be most significant in our study but it's a very important tool for data modelling for higher dimensionality data where computation efficiency is a big constrain.

6. Experiment Results

Experiment results follows same approach as implementation part.

Part A - Naive Approach

Naive approach results do not matter most as the model is insignificant but I have still added it. Figure 19 shows the accuracy plot of all three models implemented in part A along with table 5 showing exact accuracy for models.

Figure - 19

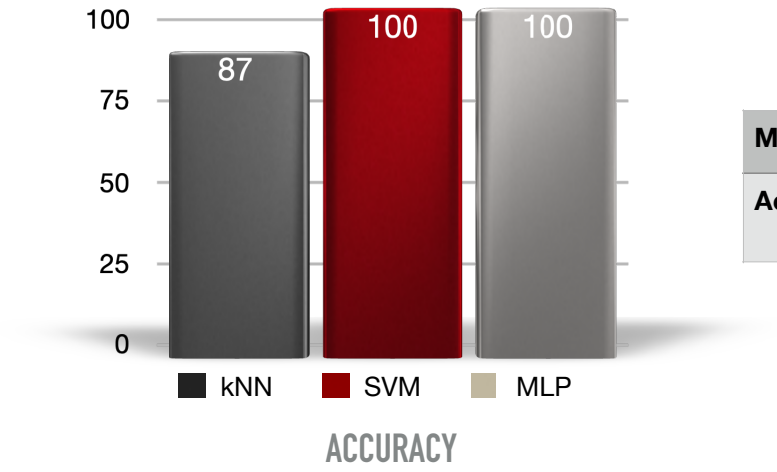


Table 5

Models	kNN	SVM	MLP
Accuracy	87.25	99.6	99.6

ACCURACY

Part B - Experienced Learning

In this section we have used 2 different type of Logistic regression Models with neural net. Accuracy of all three models are plotted in figure 20 shown below along with table 6 showing exact values.

Figure - 20

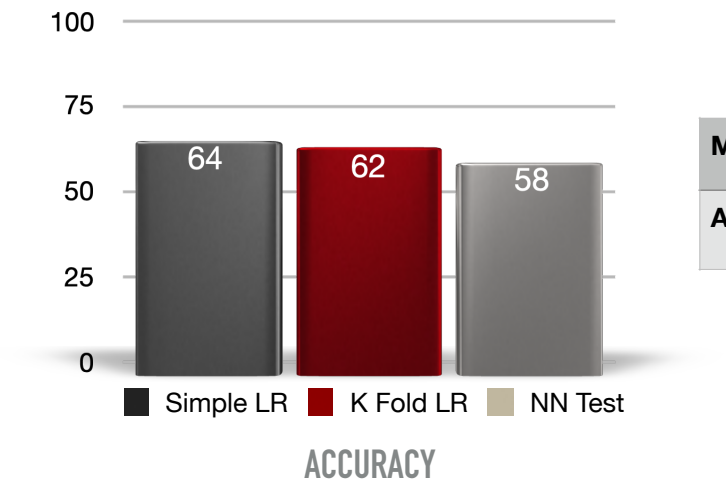


Table 6

Models	Simple LR	K Fold LR	NN
Accuracy	63.8	62.1	57.88

ACCURACY

Above graph shows that Logistic Regression model are more accurate than neural net models for sports prediction. Logistic regression with K fold have lesser accuracy as compare to simple logistic regression models, we have talked about the reason for the same in last segment. Neural network is at 57.88% on test set and train model have 58.08% accuracy which is not too bad as compare to the standard model accuracy for sports prediction problem.

Part C - Model Tuning

In this section we used various techniques to increase the accuracy of the Models. We divided this section into 3 different approaches as we talked about in last section so we will plot the accuracy curve for all three approaches along with the basic Mean value data on one plot shown below in figure 21 along with table 7.

Figure - 21

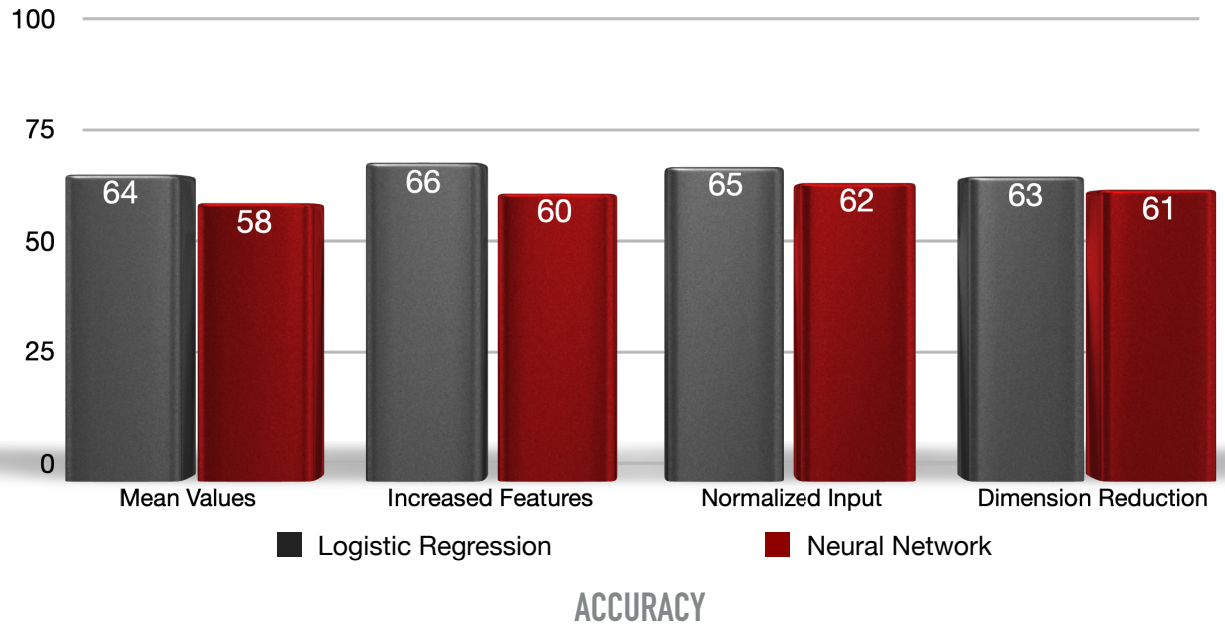
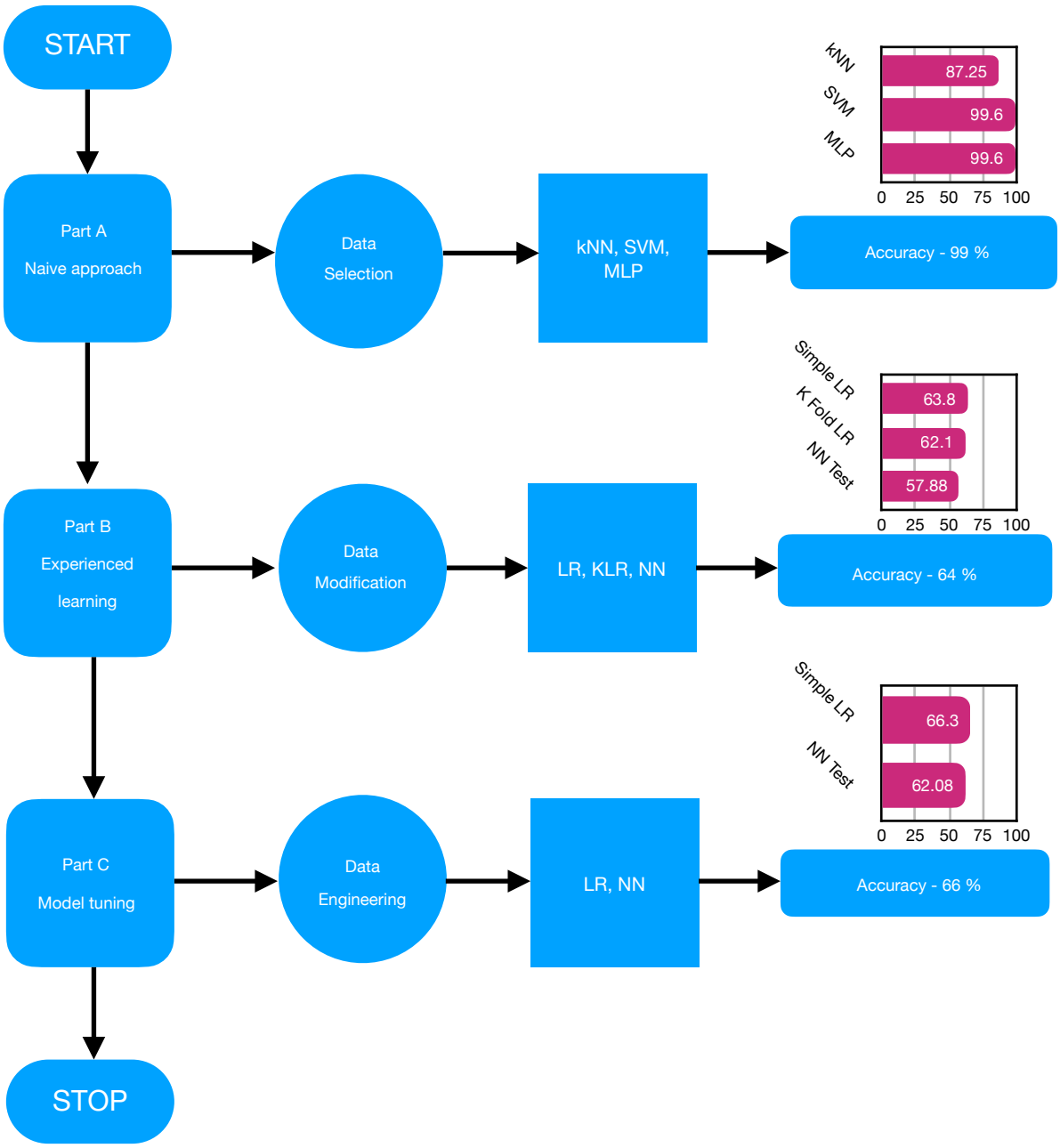


Table 7

Models	Mean Value	Feature Engineering	Normalized input	Dimentionality Reduction
Logistic Regression	63.8	66.3	65.4	63.4
Neural Network	57.88	59.86	62.08	60.68

7. Conclusion

Experiment phase can be displayed as a flowchart for a better understanding of the project workflow with the best accuracy plotted in small graphs above accuracy.



In this work, I proposed a Neural network approach along logistic Regression model to predict the winner of a NBA game. It is discovered that if we use Box score stats for the same we need to remodel the data with average of last few games played by both the teams as directly using the box score stats may give high accuracy model but the model will not be able to forecast future predictions. After remodelling the data as average data from previous 5 games we got an accuracy of approximately 60% on both the models which is not so good as prediction so we tried increasing the accuracy with the help of other machine learning optimization techniques and we reached a mark slightly more than 66% with Logistic regression model. This was the highest accuracy attained by our model but betting website analysts have maximum accuracy of around 68% which is higher that our model. Basketball is a game where there are so many variable in action along with the teams playing on court, players injury, their contract negotiation and management changes which can affect the team performance and its not only these factors, there are unlimited number of such factors. I believe there is still so much room of improvement in the model but it can only be possible if we can somehow include all these unpredictable constrains as a feature in basketball game statistics and record those variables in terms of some mathematical value so we can use those to create a machine learning model which can give a much better accuracy. Betting experts are very well trained professional and they are familiar with every activity going on in the NBA and they can use those stats for prediction in real time. I believe that there are multiple ongoing researches happening in this field and researchers are going to figure out ways to achieve way better accuracies in term of sports prediction than current stat of art models, it's just matter of time. For my project, to the best of my knowledge, I gained a maximum accuracy which is around 66.3% and it is not so bad with respect to time provided for designing the project.

Acknowledgment

I would like to sincerely thank my instructor Professor Ivan Bajic for his awesome teaching, continuous support and help throughout the duration of the project.

Appendix :

List of Figures :

FIGURE

List of Tables:

Tables

Logistic Regression Model Experiment Readings:

Logistic Regression

Model	Input	Accuracy
34 Features in dataset		
Logistic Regression with K fold (10 folds)	Data with Mean value (last 5 games)	0.623
	Scaled data using MinMaxScalar (0,1)	0.622
	PCA Dimentionality reduction data (15)	0.608
Logistic Regression simple	Data with Mean value (last 5 games)	0.622
	Scaled data using MinMaxScalar (0,1)	0.619
	PCA Dimentionality reduction data (15)	0.597
44 Features in dataset		
Logistic Regression simple	Data with Mean value (last 5 games)	0.663
	Scaled data using MinMaxScalar (0,1)	0.654
	PCA Dimentionality reduction data (24)	0.634

Neural Network Experiment Readings:

NN1

Model	Train/ Val/ Test	L.Rate	Epoch	Batch	Input	Hidden1	Hidden 2	Output	Train Accuracy	Test Accuracy
34 Features in dataset										
Binary Cross Entropy Loss										
1	(6355x26), (500x26), (501x26)	0.001	1000	32	26	10000+L2(0.1)	No	1+L2(0.1)+Sigmoid	54.97	49.90
2	(6355x26), (500x26), (501x26)	0.001	500	32	26	1000+L2(0.1)	No	1+L2(0.1)+Sigmoid	49.8	49.3
3	(6355x26), (500x26), (501x26)	0.001	500	128	26	1000+L2(0.1)	No	1+L2(0.1)+Softmax	49.86	49.30
4	(6355x26), (500x26), (501x26)	0.01	500	64	26	1000+L2(0.1)+ Relu	No	1+L2(0.1)+Sigmoid	49.61	49.30
5	(6355x26), (500x26), (501x26)	0.01	300	64	26	500+ Relu	No	1+Sigmoid	58.44	53.29
6	(6355x26), (500x26), (501x26)	0.01	300	64	26	100+Relu	No	1+Sigmoid	57.89	59.08
7	(6355x26), (500x26), (501x26)	0.01	300	64	26	130+Relu	No	1+Sigmoid	58.58	59.48
Hinge Loss										
1	(6355x26), (500x26), (501x26)	0.01	300	64	26	130+Relu	No	1+Sigmoid	58.7	58.28
2	(6355x26), (500x26), (501x26)	0.01	300	64	26	100+Relu	No	1+Sigmoid	58.08	58.08
3	(6355x26), (500x26), (501x26)	0.01	300	64	26	100+Relu	30+ Relu	1+Sigmoid	58.42	57.29
Adding More Features to the Data										
Hinge Loss										
1	(6355x44), (500x44), (501x44)	0.01	300	64	44	100+Relu	No	1+Sigmoid	59.7	60.88
Binary Cross Entropy										
1	(6355x44), (500x44), (501x44)	0.01	300	64	44	100+Relu	No	1+Sigmoid	50.1	50.7

NN2

Model	Train/ Val/ Test	L.Rate	Epoch	Batch	Input	Hidden1	Hidden 2	Output	Train Accuracy	Test Accuracy
44 Features in Dataset										
Mean data	(6355x44) ,(500x44), (501x44)	0.001	300	64	44	130+ Relu	No	1+Sigmoid	61.01	61.28
Normalized Data	(6355x44) ,(500x44), (501x44)	0.001	300	64	44	130+ Relu	No	1+Sigmoid	64.56	62.67
Dimension - 24	(6355x44) ,(500x44), (501x44)	0.001	20	64	44	130+ Relu	No	1+Sigmoid	62.47	60.28
Binary Cross Entropy										
1	(6355x44) ,(500x44), (501x44)	0.01	300	64	44	100+Relu	No	1+Sigmoid	50.1	50.7
2	(6355x44) ,(500x44), (501x44)	0.01	300	64	44	100+Relu	No	1+Sigmoid	50.1	50.7
3	(6355x44) ,(500x44), (501x44)	0.01	300	64	44	100+Relu	30+ Relu	1+Sigmoid	57.1	56.2
With Scaled data [0,1] + Binary Cross Entropy										
1	(6355x44) ,(500x44), (501x44)	0.01	100	32	44	10000+Relu	No	1+Sigmoid	59.9	57.49
2	(6355x44) ,(500x44), (501x44)	0.001	300	32	44	10000+Relu	No	1+Sigmoid	62.3	61.88
3	(6355x44) ,(500x44), (501x44)	0.001	200	32	44	10000+Relu	No	1+Sigmoid	61.7	60.28
4	(6355x44) ,(500x44), (501x44)	0.001	200	32	44	1000+Relu	No	1+Sigmoid	62.25	61.48
5	(6355x44) ,(500x44), (501x44)	0.001	145	128	44	1000+Relu	No	1+Sigmoid	60.66	58.28
6	(6355x44) ,(500x44), (501x44)	0.0001	200	128	44	1000+Relu	No	1+Sigmoid	61.16	61.68
7	(6355x44) ,(500x44), (501x44)	0.0001	200	128	44	1000+Relu	500 + Relu	1+Sigmoid	70.57	56.29
8	(6355x44) ,(500x44), (501x44)	0.0001	100	128	44	1000+Relu	No	1+Sigmoid	59.8	61.08
9	(6355x44) ,(500x44), (501x44)	0.00001	100	128	44	1000+Relu	No	1+Sigmoid	58.19	57.88
10	(6355x44) ,(500x44), (501x44)	0.001	300	64	44	1000+Relu	No	1+Sigmoid	63.25	61.48

Bibliography :

[1] Shlomo Sprung (2021) "*NBA TV Ratings On TNT, ESPN, ABC Up 34% From Last Year, Per Nielsen*". Retrieved from [https://www.forbes.com/sites/shlomosprung/2021/01/21/nba-tv-ratings-on-tnt-espn-abc-up-34-from-last-year-per-nielsen/?sh=156dadd24ddc]

[2] Ed. Dixon (2021). "*NBA All-Star Game viewership falls 18% to 5.94m*". Retrieved from [https://www.sportspromedia.com/news/nba-all-star-game-2021-tv-ratings-viewership-tnt-tbs]

[3] Adrian Wojnarowski and Zach Lowe (2021). "*NBA revenue for 2019-20 season dropped 10% to $8.3 billion, sources say"*. Retrieved from [https://abcnews.go.com/Sports/nba-revenue-2019-20-season-dropped-10-83/story?id=73886875]

[4] NBA Player Salaries for season 2020-2021 (2021). Retrieved from [http://www.espn.com/nba/salaries]

[5] NBA Wiki 2021. Retrieved from [https://en.wikipedia.org/wiki/National_Basketball_Association]

[6] Sports line experts (2021). Retrieved from [https://www.sportsline.com/experts/]

[7] Renato Amorim Torres (2013). "*Prediction of NBA games based on Machine Learning Methods"*. Retrieved from [https://homepages.cae.wisc.edu/~ece539/fall13/project/AmorimTorres_rpt.pdf]

[8] M.Beckler, H.Wang, and M.Papamichael (2009). "*NBA Oracle"*. Retrieved from [https://www.m-beckler.org/coursework/2008-2009/10701_report.pdf]

[9] Grant Avalon, Batuhan Balci and Jesus Guzman (2016). "*Various Machine Learning Approaches to Predicting NBA Score Margins"*. Retrieved from [http://cs229.stanford.edu/proj2016/report/Avalon_balci_guzman_various_ml_approaches_NBA_Scores_report.pdf]

[10] M.C. Purucker (1996). "*Neural network quarterbacking"*. Retrieved from [https://ieeexplore.ieee.org/abstract/document/535226/]

[11] Alan McCabe; Jarrod Trevathan (2008). "*Artificial Intelligence in Sports Prediction"*. Retrieved from [https://ieeexplore.ieee.org/abstract/document/4492661]

[12] Davoodi, E. Khanteymoori, A.R. (2010). "*Horse racing prediction using Artificial Neural Networks"*. Retrieved from [https://www.researchgate.net/profile/Alireza-Khanteymoori/publication/228847950_Horse_racing_prediction_using_artificial_neural_networks/links/53fc54590cf2dca8ffff0d-f8/Horse-racing-prediction-using-artificial-neural-networks.pdf]

[13] Adam Maszczyk (2014). "*Application of Neural and Regression Models in Sports Results Prediction*". Retrieved from [https://www.sciencedirect.com/science/article/pii/S1877042814017790]

[14] Paul Rossotti (2019). "*NBA Enhanced Box Score and Standings (2012 - 2018)*". Retrieved from [https://www.kaggle.com/pablote/nba-enhanced-stats]

[15] ESPN (2021). Box Score. Retrieved from [https://www.espn.in/nba/boxscore/_/gameId/401307587]

[16] Wikipedia (2021). "*Linear regression*". Retrieved from [https://en.wikipedia.org/wiki/Linear_regression]

[17] Wikipedia (2021). "*Logistic regression*". Retrieved from [https://en.wikipedia.org/wiki/Logistic_regression]

[18] Jason Brownlee (2018). "*How to Use ROC Curves and Precision-Recall Curves for Classification in Python*". Retrieved from [https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-classification-in-python/]

[19] Feature engineering technique (2021). Retrieved from [https://basketball.realgm.com/info/glossary]

[20] Min max scalar (2021). Retrieved from [https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html]

[21] Feature importance (2021). Retrieved from [https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html]