PREDICTING AND VISUALIZING PLAYERS IMPROVEMENT IN NBA LEAGUE

Datta Bezawada	Sameer Pashikanti	Prudhvi Teja	Pathipati Sricharan
dv395@snu.edu.in	sp674@snu.edu.in	pt146@snu.edu.in	sp383@snu.edu.in
1910110128	1910110339	1910110291	191011047

Abstract— NBA (National Basketball Academy) is the most popular basketball league in the world. With teams competing heavily in order to succeed, the most essential factor in determining whether team wins the title is the performance of its players. The compensation of players is mostly determined by their previous performance. Player performance, on the other hand, varies from season to season. This paper focuses on visualizing the attributes that impact players improvement and predicting players improvement for next season. We built various regression and classification models in order to achieve the task and compared their performance.

1. INTRODUCTION

NBA, the best basketball league, has millions of fans around the world cheering their favorite player/team. Team franchises eye on acquiring players that help them win championship. They mostly look out for players who gradually improved over seasons. Player performance, varies from season to season and their pay is mostly determined by their previous season performance. Every year, a few of players progress considerably from the previous year. Those players add a lot of value to the teams they play for, both competitively and financially. The NBA recognizes their importance by awarding the Most Improved Player (MIP) Award to the player who has improved the most over the previous season. As a result, franchises benefit from being able to precisely estimate if and how much a player will progress in the coming season.

Hence, we aim to contribute the franchises on depicting the factors that influence a player's success and predict their performance for the next season so that franchises can pick players who tend to improve based on their previous performance and fantasy league players can know who to watch for the coming season.

2. LITERATURE REVIEW

The aims of this study were to identify variables which may potentially influence player performance, and to implement a statistical model to study their relative contribution in order to explain two outcomes: points and win score. We used all the possible variables affecting player performance creating a comprehensive database. We dealt with a balanced study design with repeated measurements given that each player was observed the same number of games, and therefore the player was considered as a random effect. We carried out mixed models to quantify the variability in points and win score among players. Minutes played, the usage percentage and

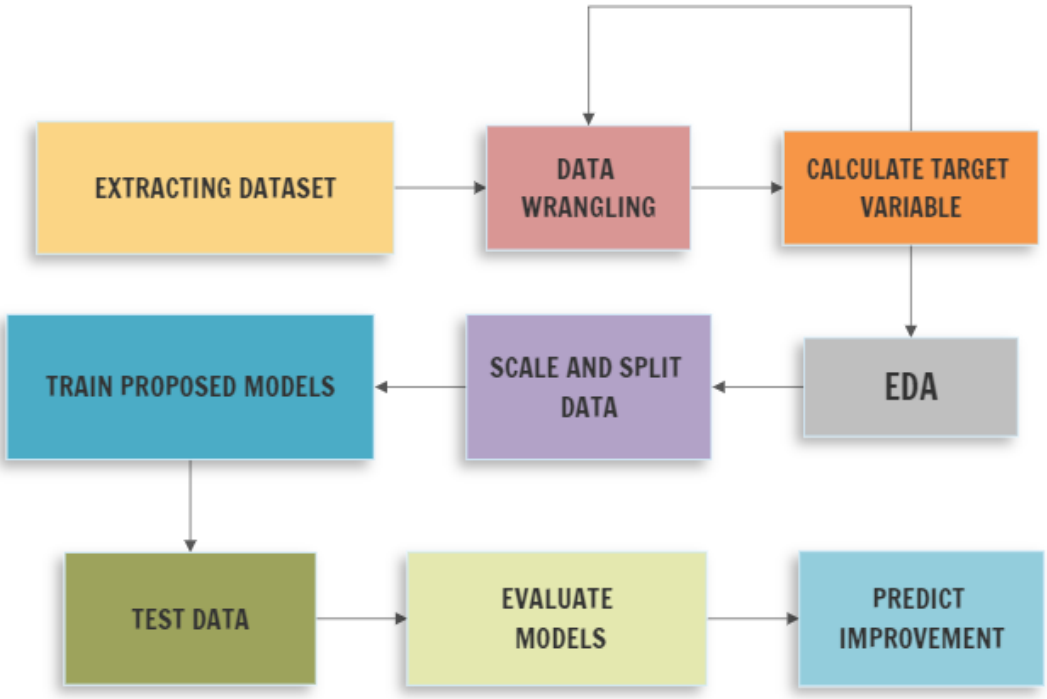
the difference of quality between teams were the main factors for variations in points made and win score. The interaction between player position and age was important in win score. We encourage managers and coaches of sports teams to choose appropriate methods according to their aims.

We modelled player performance in basketball using a mixed model approach based on the approach in Application of Machine Learning on NBA Data Sets and other papers. We considered the use of covariates to model random effects variability, taking into account the strategies described to give an adequate and parsimonious model. Therefore, it is possible to know which variables are potentially useful in explaining random effects variation and which random effects may have their variability explained by covariates. We calculated the win-loss classification based on the paper by Kevin Wheeler.

There is no evidence that season period influenced points and win score. Considering that the recent study of Sampaio, Drikwater and Leite (2010) did not find evidence for the effect of season period on game related statistics, we acknowledge that more support to our result would be desirable. It is true that the method and variables used by Sampaio, Drikwater and Leite (2010) were different, but again further research should deepen into this issue in order to obtain clearer evidence.

We constructed different models for this, which include Random Forest and also Gradient Boost model which we taken in reference to the paper Analysis of NBA Players Using Random Forest and XGBoost models by Maram Shikh Oughali, Mariah Bahloul and Sahar A. El_Rahman.

3. PROPOSED MODEL



1. Extracting dataset - The dataset to be used for prediction purpose is extracted and loaded into a data frame for accessing it easily furthermore.
2. Data wrangling - Cleaning and handling the dataset. Checking and removing null values, and verifying the datatype of each variable.
3. Calculating target variable - Required target variable is calculated on basis of other important feature variables in the dataset. Further, data wrangling is to be performed on these calculated values to later merge it with main dataset.
4. EDA - Perform exploratory data analysis to visualize the relationship between the variables.
5. Scale and split data - Scaling data for normalizing the values and splitting it into training and testing data with default size of 75% and 25%.
6. Train proposed models - Training the proposed models required to predict the improvement with the training data splitted earlier.
7. Test data - Using the test set to identify true positives, true negatives, false positives, and false negatives.
8. Evaluate models - Evaluating the models on basis of various metrics like recall, Precision, F-Score, and Accuracy.
9. Predict improvement - Predicting improvement by using the most accurate model.

4. DATASET

The dataset used is acquired from kaggle -"NBA Draft value" that contains information and stats of players such as age, games played, true shooting percentage, win share percentage, block percentage, steal percentage, Attempt rate, assist percentage, picks etc, right from 1978 to 2018 with 13378 samples.

5. METHODOLOGY

The dataset did not include player improvement year over year, thus it had to be computed. So, we defined a function that calculates our target variable by subtracting win share of a player between two successive years. Out of a few indicators, win shares was picked as the most interpretable. After all, we play basketball to win. The normal distribution of calculated player improvement was centered around 0, with most values falling between -5 and 5.

Improvement (year) = WS (year+1) - WS (year)

The relationship between the target variable and other independent variables is then determined. We visualize how factors such as age, games played, win share and previous year improvement impact player's improvement in forthcoming season by means of multiple histograms.

To forecast player improvement, two types of models can be used: regression and classification. Regression models provides more information about how much a player has improved, whereas

classification models focus on how likely a player can improve. Although the basic methods for regression and classification models are identical, certain audiences may favor one over the other. An NBA franchise, for example, may be more interested in the amount of progress (regression models), whereas a common NBA fan may find the categorization model findings more understandable. As a result, we used both regression and classification modelling in this project. The most suitable parameters for these models were chosen using grid search for the models to perform better.

Regression models -

To predict our target variable using regression, we used MinMaxScaler to scale our values between 0 to 1, since most of the feature variables are in form of percentage values, these values are divided into training and testing sets and are fit into our regression models. For this project, we built linear regression, ridge regression, SVM, random forest, and gradient boost models. For regression models, root mean squared error (RMSE) is used as the tuning and evaluation metric.

Linear regression - It attempts to model the relationship between two variables by fitting a linear equation to observed data. One variable is considered to be an explanatory variable, and the other is considered to be a dependent variable.

A linear regression line has an equation of the form,

$$Y = a + bX$$

Where, X is the explanatory variable

Y is the dependent variable.

The slope of the line is ***b***, and ***a*** is the intercept

Ridge Regression - It is a technique for analyzing multiple regression data that suffer from multicollinearity. When multicollinearity occurs, least squares estimates are unbiased, but their variances are large so they may be far from the true value. By adding a degree of bias to the regression estimates, ridge regression reduces the standard errors.

The cost function for ridge regression

$$Min\ (||Y - X\ (theta)\ ||^2 + \lambda||theta||^2)$$

Support Vector Machines (SVM) – It is a supervised machine learning algorithm used for both classification and regression. Though we say regression problems as well its best suited for classification. The objective of SVM algorithm is to find a hyperplane in an N-dimensional space that distinctly classifies the data points.

We know that *max [f(x)]* can also be written as *min [1/f(x)]*, it is common practice to minimize a cost function for optimization problems; therefore, we can invert the function.

$$\operatorname{argmin}(w^*, b^*) \frac{\|w\|}{2} \text{ such that } y_i\left(\vec{w}.\vec{X}+b\right) \geq 1$$

To make a soft margin equation we add 2 more terms to this equation which is **zeta** and multiply that by a **hyperparameter 'c'**

$$\operatorname{argmin}\left(w^*, b^*\right) \frac{\|w\|}{2} + c\sum_{i=1}^{n}\zeta_i$$

Random forest - Random Forest is a ***Supervised Machine Learning Algorithm*** that is ***used widely in Classification and Regression problems***. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.

When using the Random Forest Algorithm to solve regression problems, you are using the mean squared error (MSE) to how your data branches from each node.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(fi - yi)^2$$

This formula calculates the distance of each node from the predicted actual value, helping to decide which branch is the better decision for your forest.

When performing Random Forests based on classification data, you should know that you are often using the Gini index, or the formula used to decide how nodes on a decision tree branch.

$$Gini = 1 - \sum_{i=1}^{C}(p_i)^2$$

You can also use entropy to determine how nodes branch in a decision tree.

$$Entropy = \sum_{i=1}^{C} -p_i * \log_2(p_i)$$

Gradient Boost Model - **Gradient boosting** is a machine learning technique used in regression and classification tasks, among others. It gives a prediction model in the form of an ensemble of weak prediction models, which are typically decision trees. When a decision tree is the weak learner, the resulting algorithm is called gradient-boosted trees; it usually outperforms random forest. A gradient-boosted trees model is built in a stage-wise fashion as in other boosting methods, but it generalizes the other methods by allowing optimization of an arbitrary differentiable loss function.
The loss function is given below:

$$L = \frac{1}{n}\sum_{i=0}^{n}\left(y_i - \gamma_i\right)^2$$

Classification Models-

The application of classification task was quite simple. We splitted the samples into two categories (improvement>=0 or <0). In each class, there were almost the same amount of samples. We refined and constructed a logistic regression, as well as SVM, random forest, gradient boost models. The accuracy score metric was chosen because it more accurately assesses each model's performance.

Logistic regression - Logistic Regression is a classification technique used in machine learning. It uses a logistic function to model the dependent variable. The dependent variable is dichotomous in nature, i.e. there could only be two possible classes.

For logistic regression, the cost function is given by the equation:

$$\text{Cost}(h_\Theta(x), Y(\text{actual})) = -\log(h_\Theta(x)) \text{ if } y=1$$
$$-\log(1-h_\Theta(x)) \text{ if } y=0$$

Random forest - Random Forest is a robust machine learning algorithm that can be used for a variety of tasks including regression and classification. It is an ensemble method, meaning that a random forest model is made up of a large number of small decision trees, called estimators, which each produce their own predictions.

The sum of the feature's importance value on each tree is calculated and divided by the total number of trees:

$$RFfi_i = \frac{\sum_{j \in all\ trees} normfi_{ij}}{T}$$

RFfi sub(i)= the importance of feature i calculated from all trees in the Random Forest model
normfi sub(ij)= the normalized feature importance for i in tree j
T = total number of trees

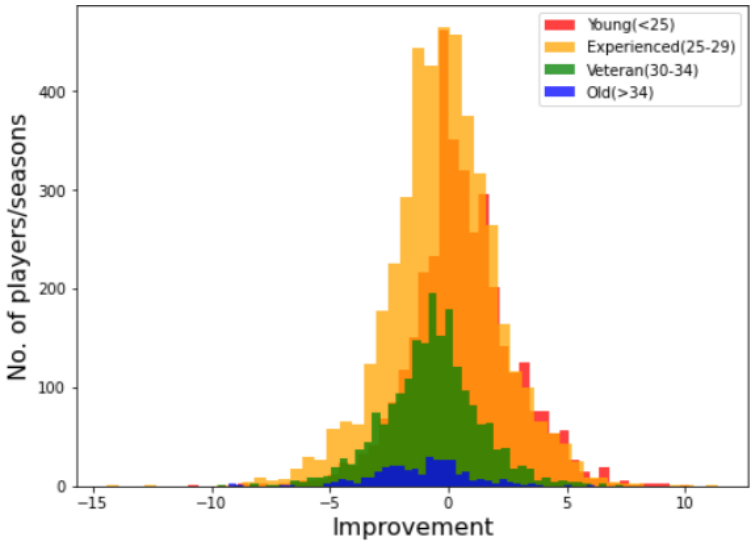
Gradient Boost Model - **Gradient boosting** is a machine learning technique used in regression and classification tasks, among others. It gives a prediction model in the form of an ensemble of weak prediction models, which are typically decision trees. When a decision tree is the weak learner, the resulting algorithm is called gradient-boosted trees.
The loss function for the classification problem is given below:

$$L = -\sum_{i=1}^{n} y_i \log(p) + (1-p)\log(1-p)$$

5. RESULTS AND FINDINGS

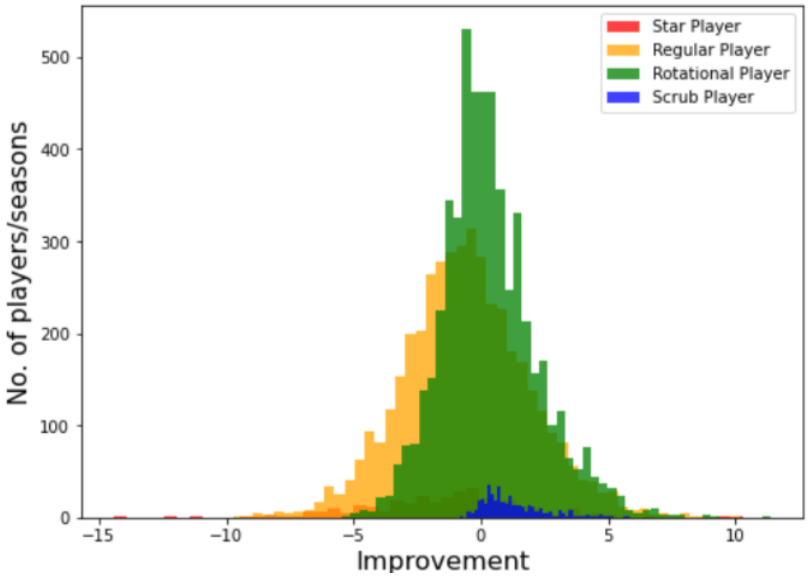
Relationship between improvement and age -

The idea that younger players are more likely to progress than older players is commonly held, and our results backs this up. It demonstrates that younger players make more progress than older players. However, the differences in age group means were statistically significant, but they were minor. Also, there was a significant difference in improvement between participants of the same age.



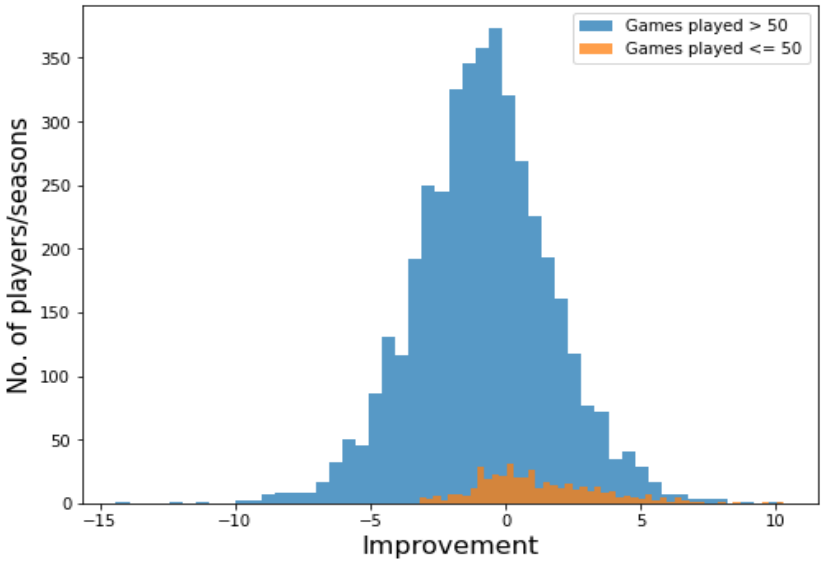
Relationship between improvement and player type -

The players who are already stars have little room for improvement, whereas a poor player can still improve. This theory was supported by our findings. We observed negative relation between a player's total skill and his improvement the next season, using 4 win share per 48 minutes (WS/48) as a measure of overall ability. Star players (WS/48 > 0.2), regular players (WS/48 between 0.1 and 0.2), rotational players (WS/48 between 0 and 0.1), and "scrubs" (WS/48 below 0) mean improvement were significantly different from each other.



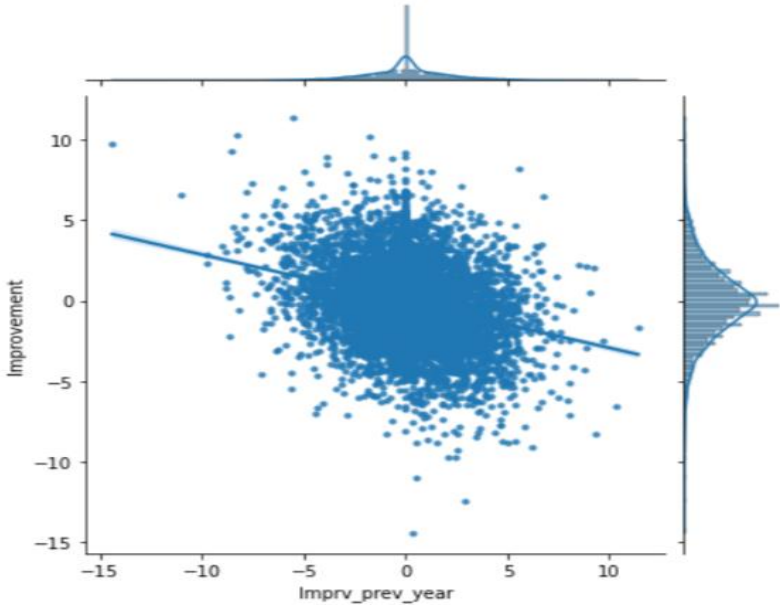
Relationship between improvement and games played -

We discovered a negative relationship between a player's improvement and the number of games played. If a good player missed a lot of games, it was most likely due to injury, which hampered his performance. Next season, he could revert to his previous form and so improve. Players who had played fewer than 50 games had a better chance of improving than those who had played more than 50 games.



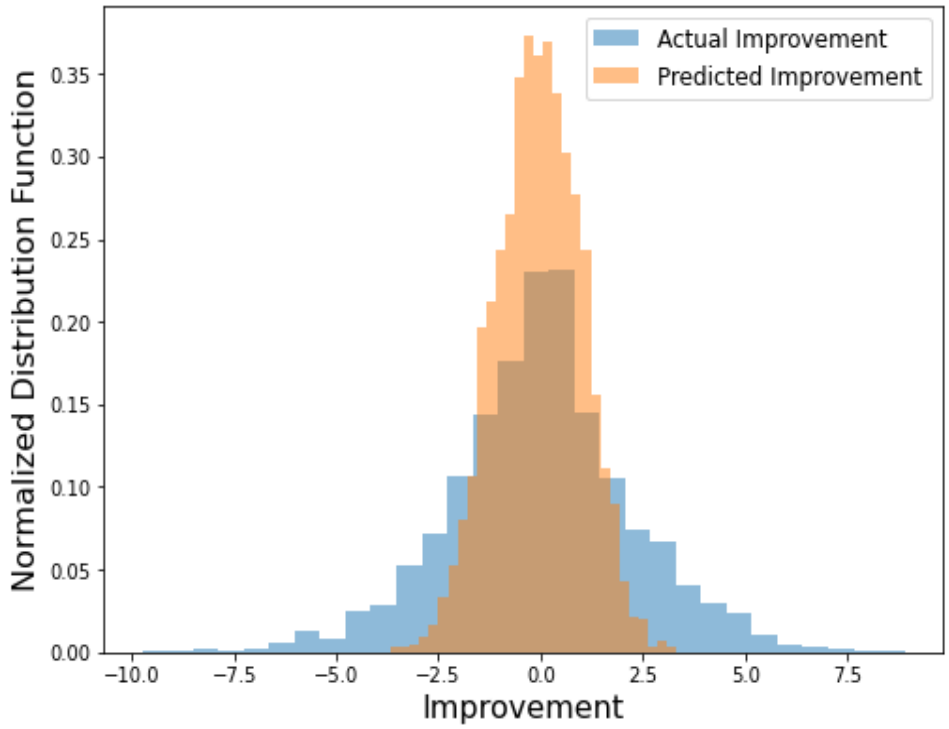
Relationship between improvement and last year's improvement -

As younger players' improvement may develop continually for a few years and older players may fall for a few years, we postulated that a player's improvement may be connected with his past improvement. It was shown that there was a negative correlation between improvement and previous improvement. To put it another way, rather than consistently improving or declining, a player will "regress to the mean" more often than not.



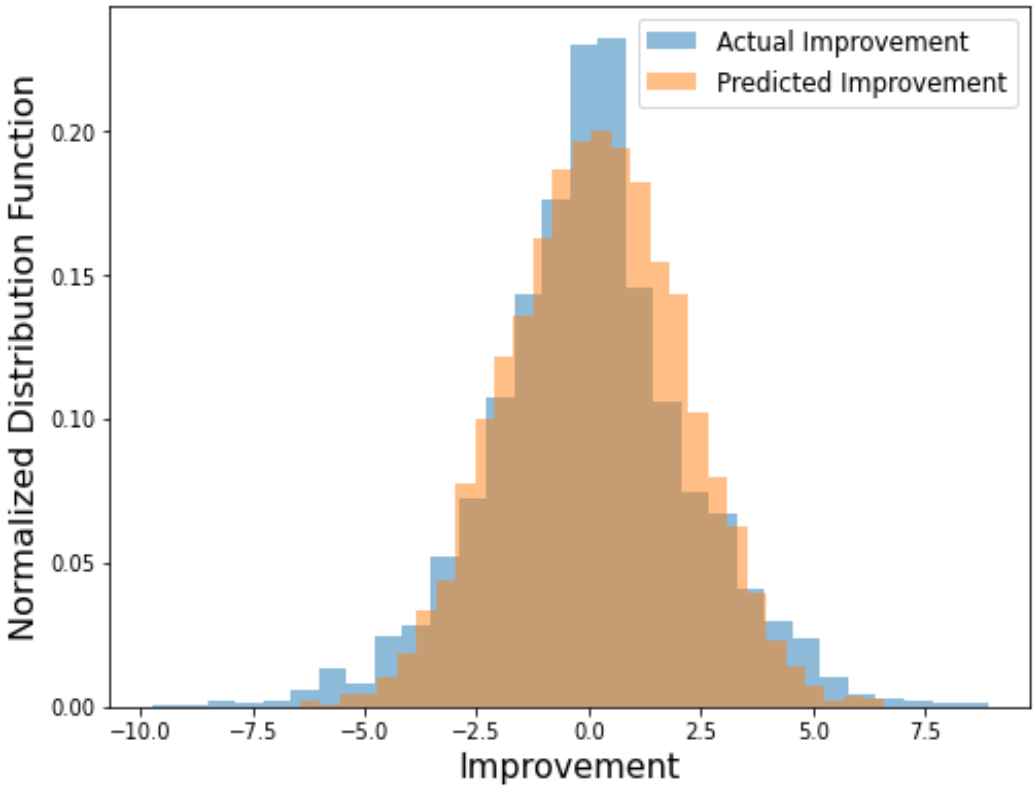
Results of Regression Models -

The proposed regression models were implemented and improvement was predicted. However, the predicted improvement was varying a lot from the actual improvement as seen in below figure.



This might occur due to the uneven distribution of player improvement because players with little improvement were more common than players with a big improvement. Therefore, the models tried to prioritize minimizing errors on players with little improvement when RMSE was used as the evaluation metric. We tried using many scalers to scale values in a way that values with larger improvement were given priority.

Hence, we assigned weights to values by dividing them by the number of samples with similar target values. These weighted when fit into the regression model predicts data which is not much varying from actual improvement.



Hence, we re-built our proposed regression models by employing the innovative approach of varying sample weights. Hyperparameters were set using the same metric and cross validation for each model. SVM had the best performance among all models.

The performance of these models are as follows -

	Linear Regression	Ridge Regression	SVM Regressor	Random forest Regression	Gradient boost Regression
RSME	2.697022	2.706236	2.680481	2.689723	2.702503

It appears that SVM was the best model based on the weighted squared error metric.

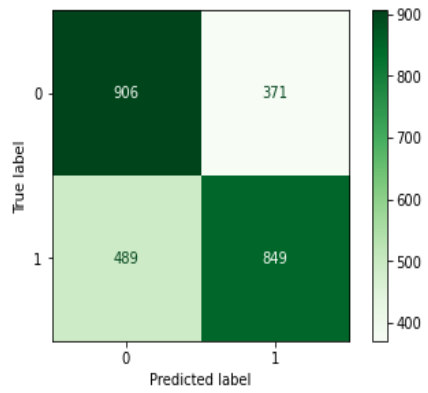
Results of Classification Models -

Upon implementation of our classification models, we observed Logistic regression performing more accurately than other models on our data. Along with accuracy, f1 score and recall score were also evaluated to check the performance of the models.

The performance of these models are as follows –

Logistic regression:

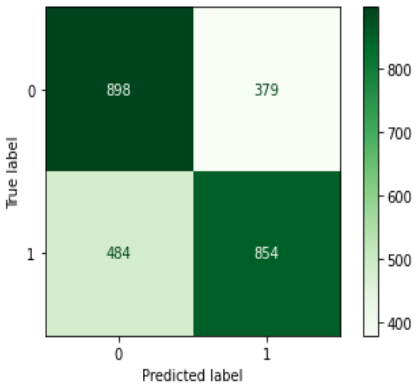
Confusion Matrix of Logistic regression model :-



	precision	recall	f1-score	support
0	0.65	0.71	0.68	1277
1	0.70	0.63	0.66	1338
accuracy			0.67	2615
macro avg	0.67	0.67	0.67	2615
weighted avg	0.67	0.67	0.67	2615

SVM Classifier:

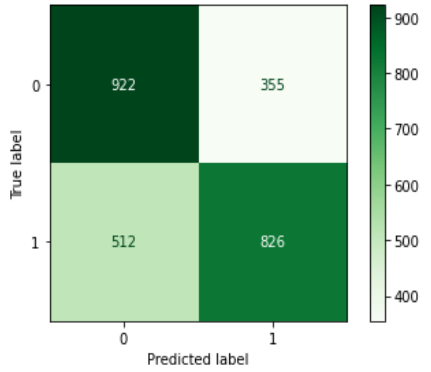
Confusion Matrix of SVM classifier model :-



	precision	recall	f1-score	support
0	0.65	0.70	0.68	1277
1	0.69	0.64	0.66	1338
accuracy			0.67	2615
macro avg	0.67	0.67	0.67	2615
weighted avg	0.67	0.67	0.67	2615

Random Forest:

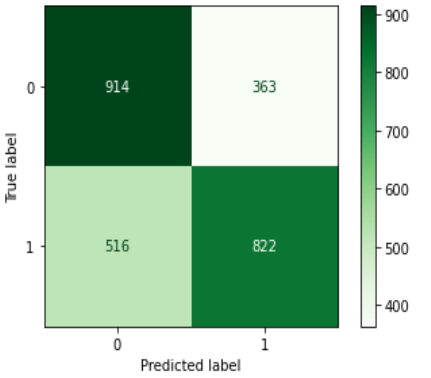
Confusion Matrix of Random forest model :-



	precision	recall	f1-score	support
0	0.64	0.72	0.68	1277
1	0.70	0.62	0.66	1338
accuracy			0.67	2615
macro avg	0.67	0.67	0.67	2615
weighted avg	0.67	0.67	0.67	2615

Gradient boost Classifier:

Confusion Matrix of Gradient boosting classifier model :-

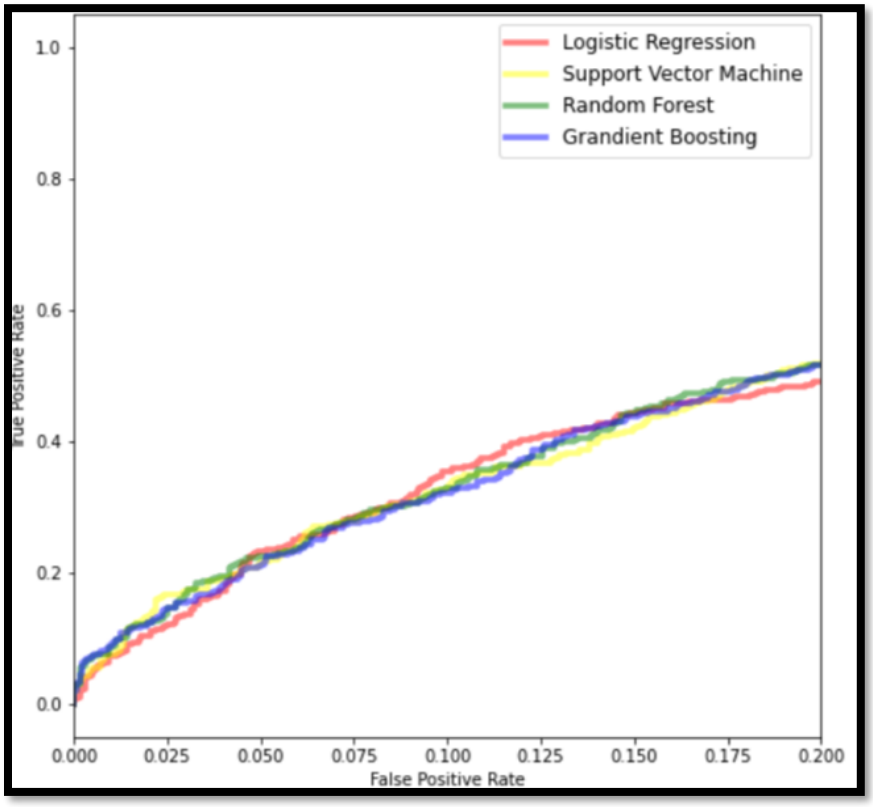


	precision	recall	f1-score	support
0	0.64	0.72	0.68	1277
1	0.69	0.61	0.65	1338
accuracy			0.66	2615
macro avg	0.67	0.67	0.66	2615
weighted avg	0.67	0.66	0.66	2615

	Logistic Regression	SVM	Random Forest	Gradient Boost
Accuracy	0.67112	0.66806	0.66845	0.66386
No. of True Positives	849	854	826	822
No. of False Positives	371	379	355	363
No. of False Negatives	489	484	512	516
No. of True Negatives	906	898	922	914

ROC Curve -

We also evaluated the models using their ROC curves based on their true and false positive rates. Since, all the models perform at almost same accuracy without varying much, Logistic regression model have a slight upper hand over other models.



7. CONCLUSION AND LIMITATIONS

In this study, we studied and visualized at the relationship between NBA players' performance and biographic data, as well as their improvement. The most essential features that determine a player's improvement next season are age, win share, minutes/games played, and last season's improvement. To predict whether and how much a player will improve or deteriorate, we developed both regression and classification models. In a variety of ways, these models can be quite beneficial to NBA franchises. In regression models, SVM had the best performance among all models with weighted mean squared error of around 2.68. On the other hand, in the classification task, logistic regression model with a accuracy of around 67% gives better performance in categorizing whether a player will tend to improve or not. However, the accuracy of the models are not so impressive and can be improved in further developments. This could be due to a lack of suitable data scaling, as players with the least improvement were more prevalent than players with the greatest progress, which should have been prioritized first. Furthermore, the defined best parameter function demonstrates a high level of complexity, as grid search takes longer to find the best suited parameters for the models to perform better.

8. REFERENCES

[1] Alberto Arteta Albert, Luis Fernando de Mingo López, Kristopher Allbright and Nuria Gómez Blas , "*A Hybrid Machine Learning Model for Predicting USA NBA All-Stars*", College of Arts and Sciences, Troy University, 129-A MSCX, 600 University Avenue, Troy, AL 36082.

[2] Kevin Wheeler, "*Predicting NBA Player Performance*".

[3] Jingru Wang and Qishi Fan, "*Application of Machine Learning on NBA Data Sets*", 2021 J. Phys.: Conf. Ser. 1802.

[4] M. S. Oughali, M. Bahloul and S. A. El Rahman, "*Analysis of NBA Players and Shot Prediction Using Random Forest and XGBoost Models*," 2019 International Conference on Computer and Information Sciences (ICCIS)*, 2019*, pp. 1-5, doi: 10.1109/ICCISci.2019.8716412.

[5] Casals, Martí & Martinez, Jose. (2013), "Modelling player performance in basketball through mixed models" in International Journal of Performance Analysis in Sports. 13. 64-82.