**Context**:

In the early 2000s, Billy Beane and Paul DePodesta worked for the Oakland Athletics. While there, they literally changed the game of baseball. This data set contains a set of variables that Beane and DePodesta focused heavily on.

They determined that stats like on-base percentage (OBP) and slugging percentage (SLG) were very important when it came to scoring runs, however they were largely undervalued by most scouts at the time. Since these players weren't being looked at by other teams, they could recruit these players on a small budget.

Columns in the dataset:

* Team
* League
* Year
* Runs Scored (RS)
* Runs Allowed (RA)
* Wins (W)
* On-Base Percentage (OBP)
* Slugging Percentage (SLG)
* Batting Average (BA)
* Playoffs (binary)
* RankSeason
* RankPlayoffs
* Games Played (G)
* Opponent On-Base Percentage (OOBP)
* Opponent Slugging Percentage (OSLG)

Objective:

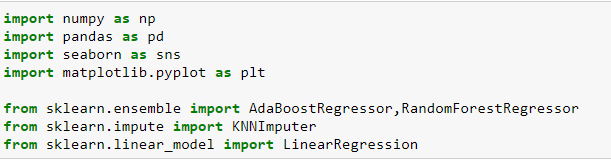
The objective here is to predict the number of wins by a team for the year 2002 by analyzing the game stats for the years from 1962-2001.

*Note: The number of wins recorded by the team in the year 2002 was 103. The idea is to replicate the data analysis done in the year 2001 with the latest ML models and check if we can get the prediction close to 103. Hence, I will only be using the 4 features that were used in 2001 for this task.*

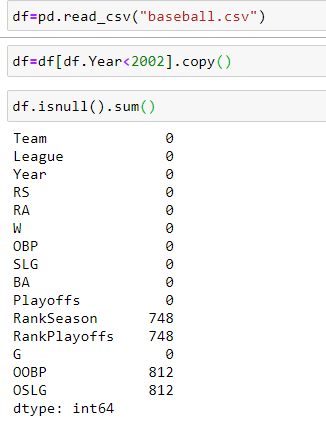
*If you like, you could use all the features in the dataset and train different models.*

**STEPS**:

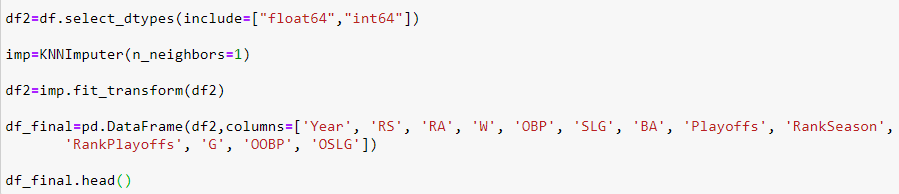
1. Import the necessary libraries



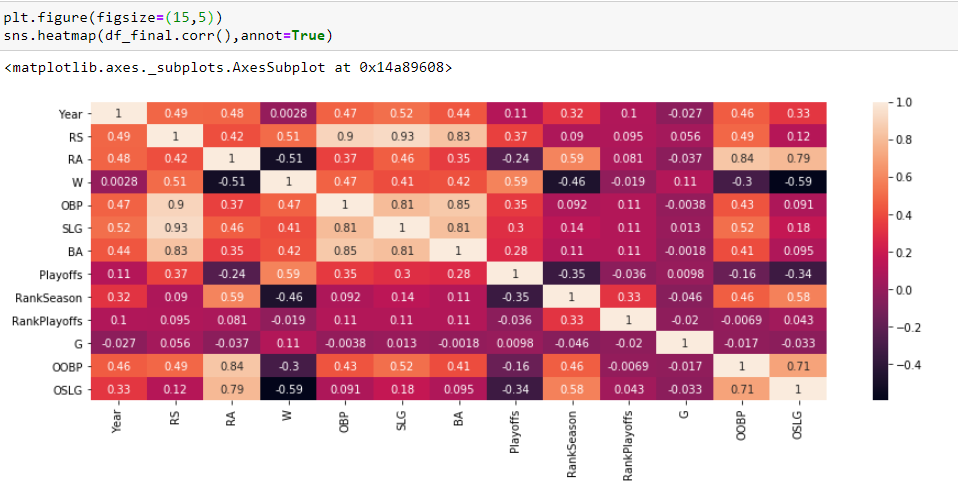
1. Import the dataset, extract the data for the years 1962-2001 and look for null values



1. Excluding the categorical variables to impute the missing values by using KNNImputer



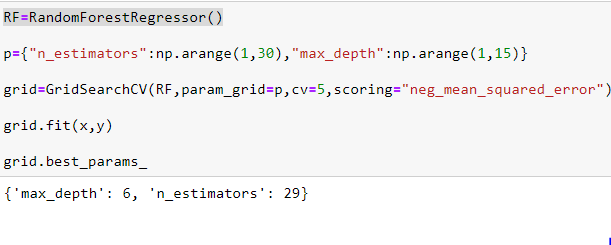
1. Checking the correlation of the independent variables with the target variable “W”



7. Selecting the independent and dependent variables ( Here, I am including the 4 features used for analysis in the year 2001)



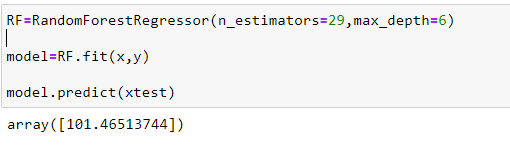
8. Hyperparameter tuning using GridSearchCV



9. Creating a DataFrame for the test record



10. Training a Random Forest model and predicting the value for the test record



Let’s check if we can get a better prediction by Boosting. I will be using the AdaboostRegressor with the base\_estimator=RandomForest.

11. Hyperparameter tuning for the AdaboostRegressor, training the ensemble model and predicting the number of wins.

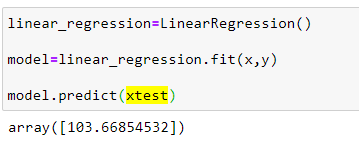


**The prediction for the boosted RandomForest is 101.998 (which is very close to the actual value (103))**

Now let us build a Linear Regression model and check if we can get a better prediction.

**STEPS:**

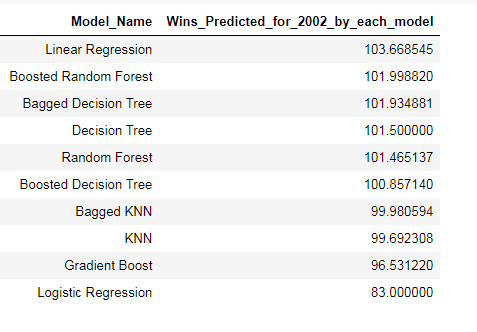
1. Building a linear regression model and looking at the prediction



**Now this is in fact a better model than the ensemble model. The prediction varies from the actual by only 0.668.**

I have also checked the values predicted by other models and by far Linear Regression has given the best prediction.

Below is a snapshot of the values predicted by the different models:



Link to the Dataset and Problem Description : <https://www.kaggle.com/wduckett/moneyball-mlb-stats-19622012>