



Baseball Case Study

Problem Statement :

This dataset utilizes data from 2014 Major League Baseball seasons in order to develop an algorithm that predicts the number of wins for a given team in the 2015 season based on several different indicators of success. There are 16 different features that will be used as the inputs to the machine learning and the output will be a value that represents the number of wins.

- [Kundan Patil](#)



This article contains the following subtopics

- 1. Problem Definition**
- 2. Data Analysis**
- 3. EDA Concluding Remarks**
- 4. Pre-processing Pipeline**
- 5. Building ML Models**
- 6. Concluding Remarks**



Problem Definition

To forecast the number of wins for a given team for upcoming 2015 season by analysing past records of previous year(2014) for a given team.

Choosing the Type of Machine Learning

Because the target variable is continuous,
we need to create a supervised ML Regression model,
as stated in the problem statement.

```
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import r2_score, mean_squared_error, mean_absolute_error
from sklearn.linear_model import RANSACRegressor
from sklearn.linear_model import Ridge, Lasso, ElasticNet
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import SGDRegressor
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import GridSearchCV
```

Import the necessary libraries

```
] : 1 #importing data
    2 df = pd.read_csv(r"C:\Users\Kundan Patil\DS0522\Evaluation Phase- Batch DS05
```

```
] : 1 df.head()
```

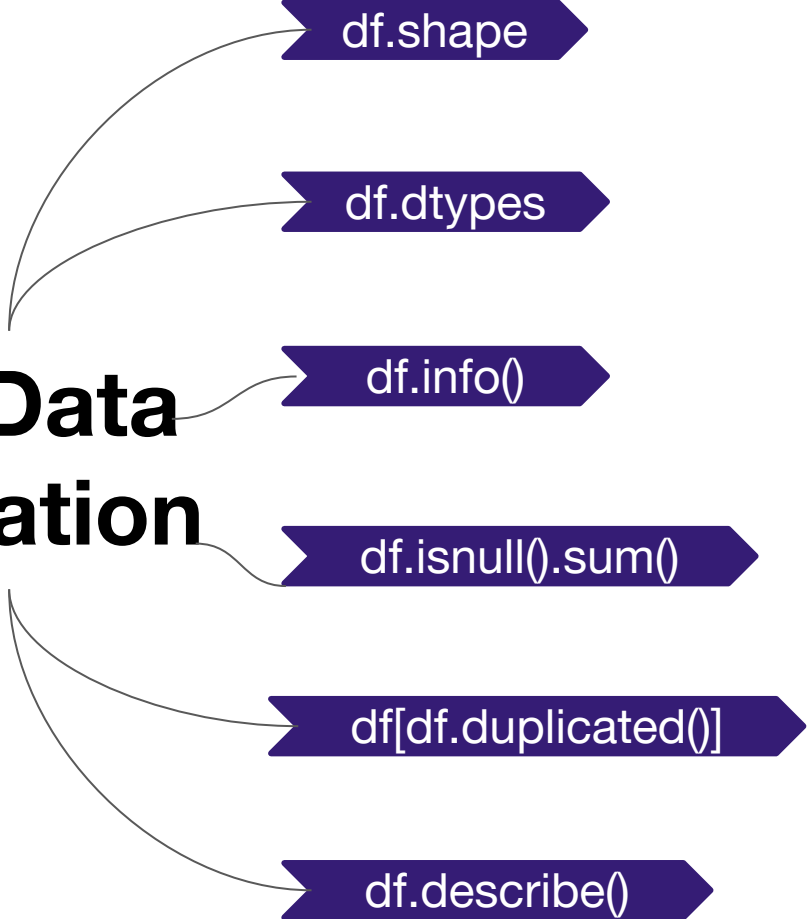
	W	R	AB	H	2B	3B	HR	BB	SO	SB	RA	ER	ERA	CG	SHO	SV	E
0	95	724	5575	1497	300	42	139	383	973	104	641	601	3.73	2	8	56	88
1	83	696	5467	1349	277	44	156	439	1264	70	700	653	4.07	2	12	45	88
2	81	669	5439	1395	303	29	141	533	1157	55	55	55	55	55	55	55	55

Basic dataset information

Description of the variables

- R: Runs
- AB: At Bats
- H: Hits
- 2B: Doubles
- 3B: Triples
- HR: Homeruns
- BB: Walks
- SO: Strikeouts
- SB: Stolen Bases
- RA: Runs Allowed
- ER: Earned Runs
- ERA: Earned Run Average (ERA)
- CG: Complete Game
- SHO: Shutout
- SV: Save
- E: Errors
- W: Win (Target variable)

Basic Data Exploration



```
graph LR; A[Basic Data Exploration] --> B[df.shape]; A --> C[df.dtypes]; A --> D[df.info()]; A --> E[df.isnull().sum()]; A --> F[df[df.duplicated()]]; A --> G[df.describe()];
```

`df.shape`

`df.dtypes`

`df.info()`

`df.isnull().sum()`

`df[df.duplicated()]`

`df.describe()`

The basic data exploration results were found to be satisfactory

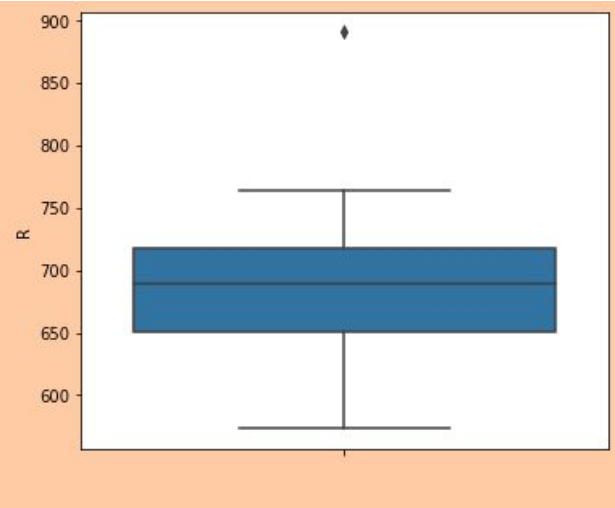
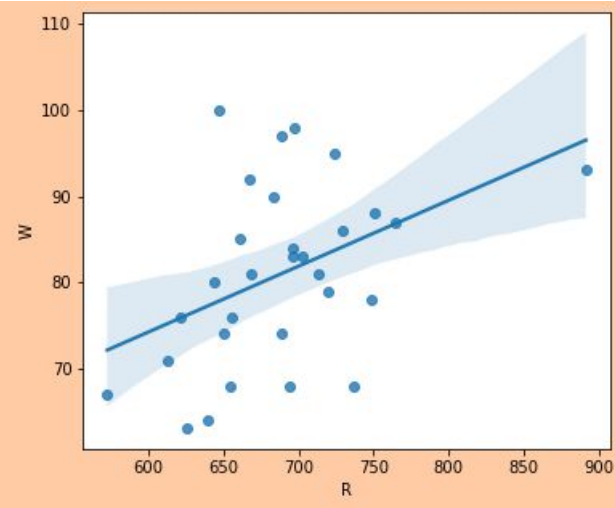
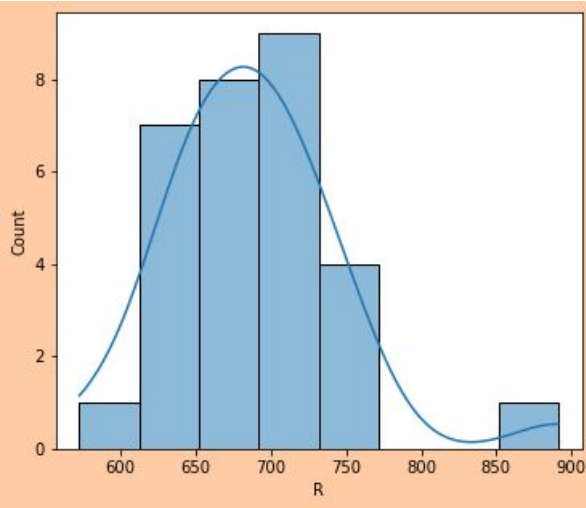


Visualize distribution of all the Continuous Predictor variables

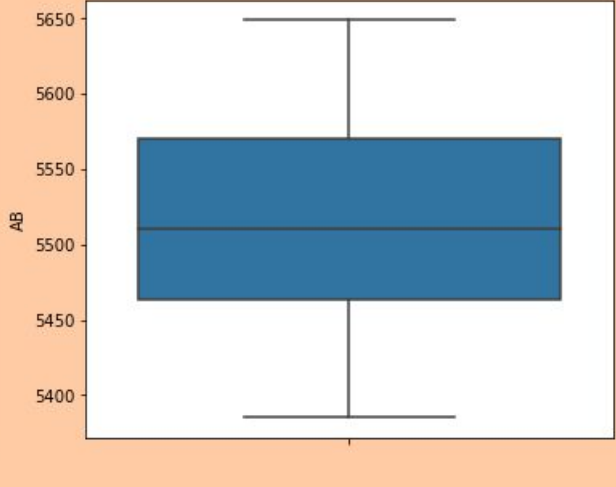
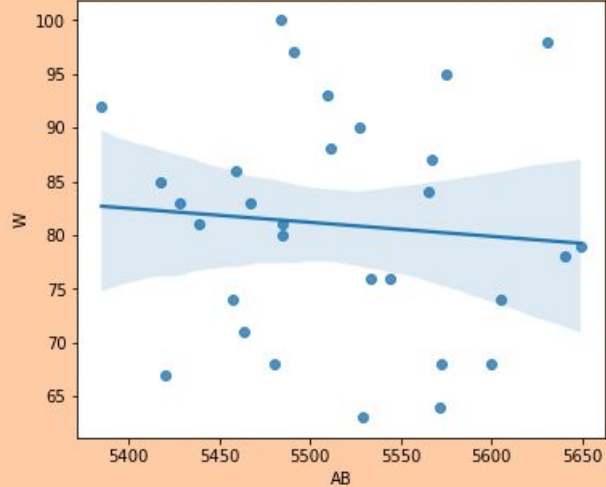
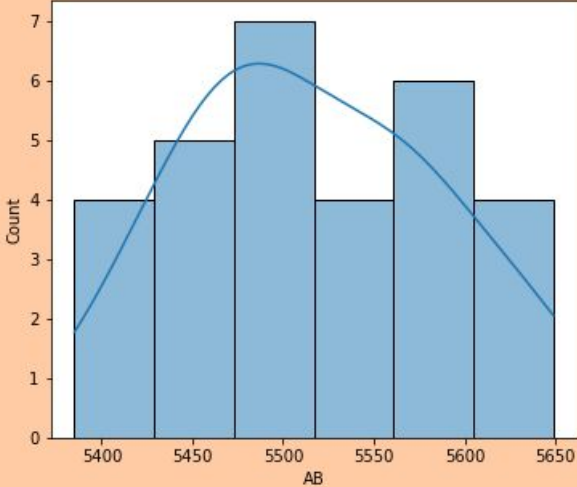
Using

- Histogram,
- box plot,
- regplot

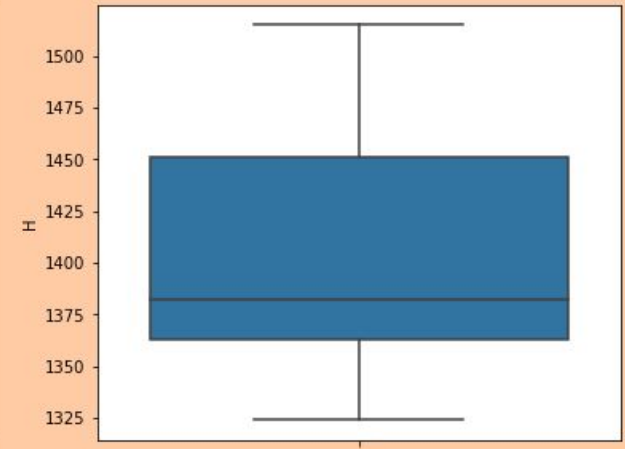
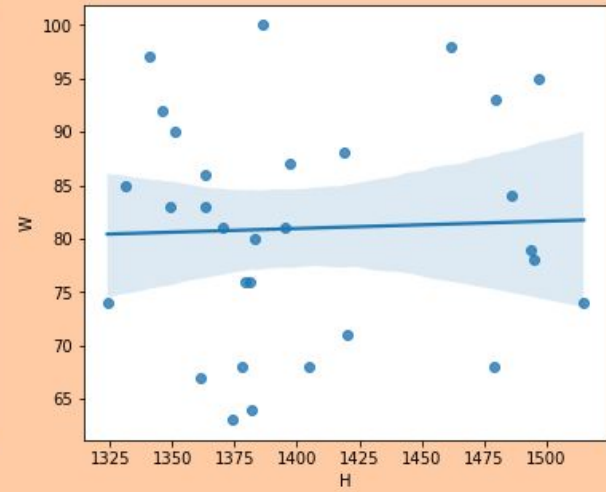
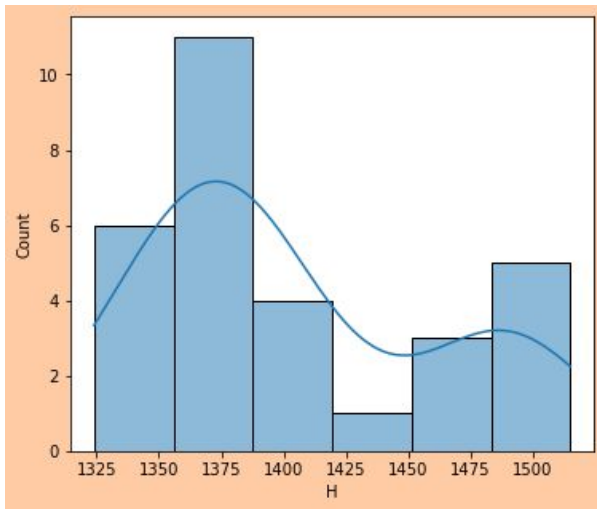
● R: Runs



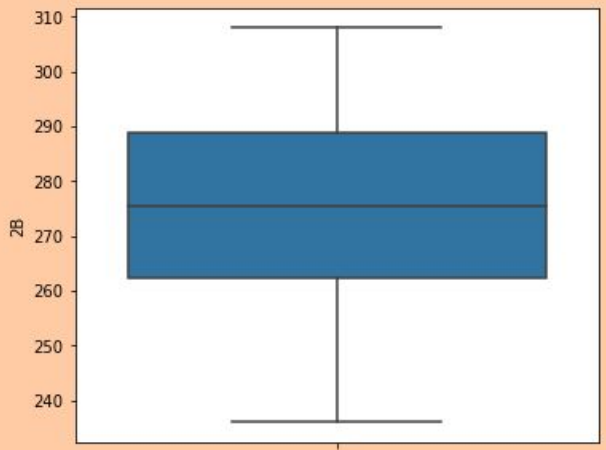
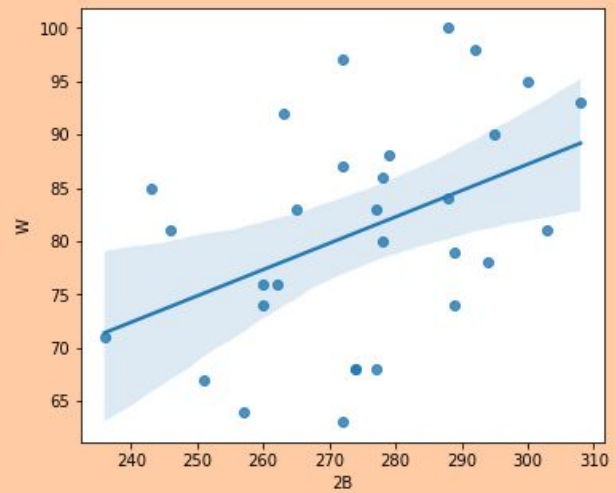
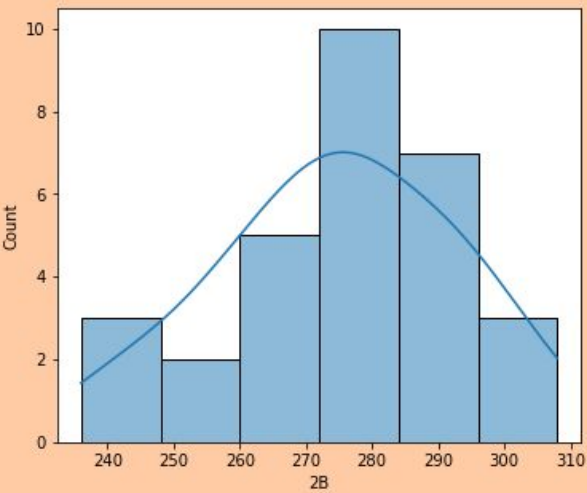
● AB: At Bats



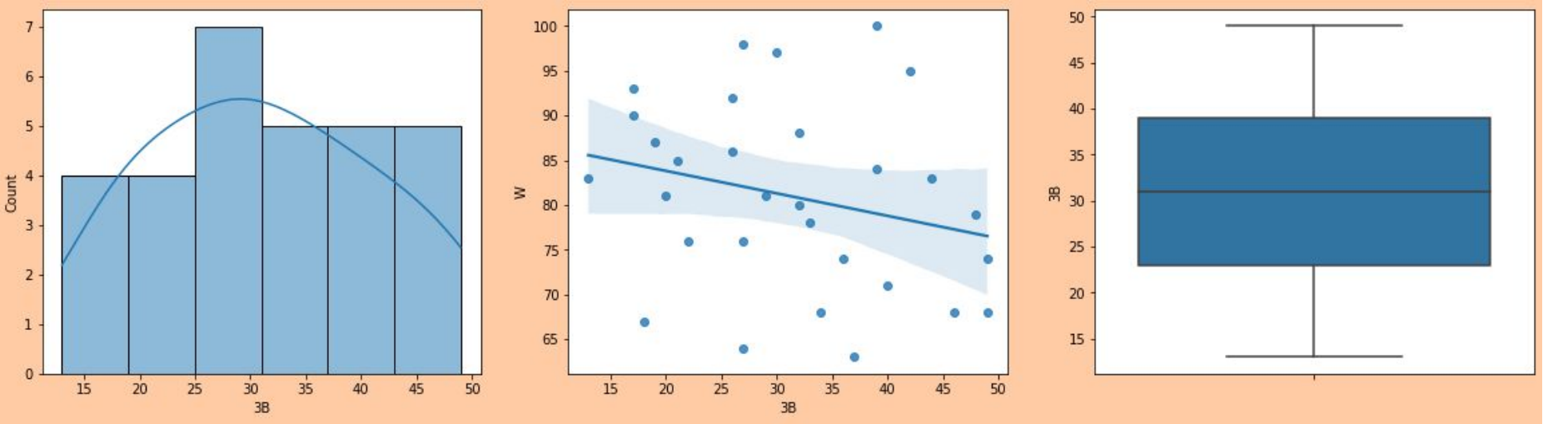
● H: Hits



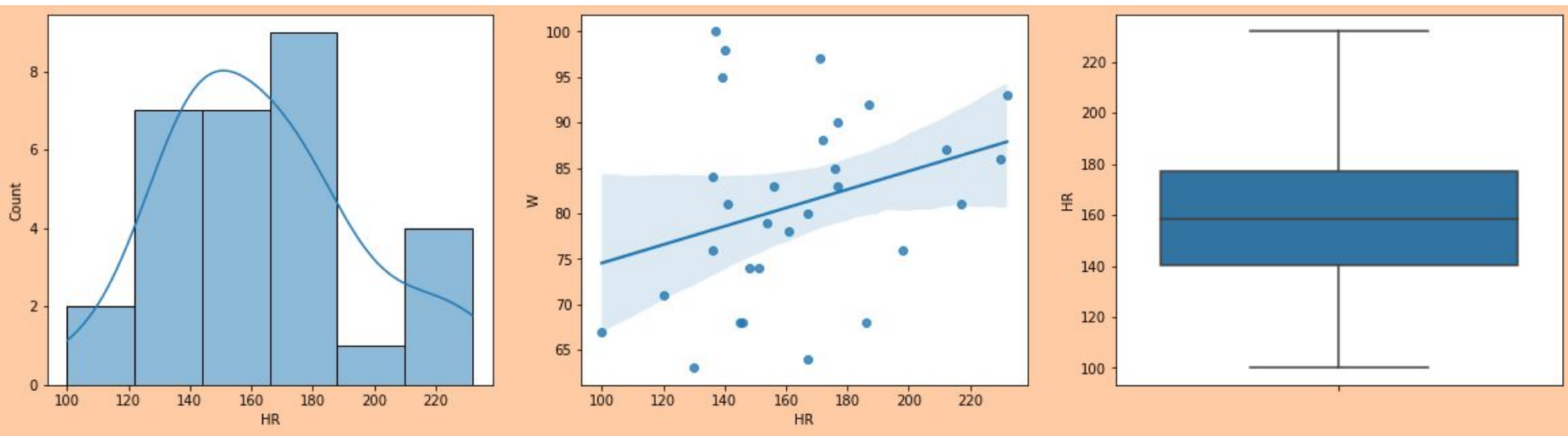
● 2B: Doubles



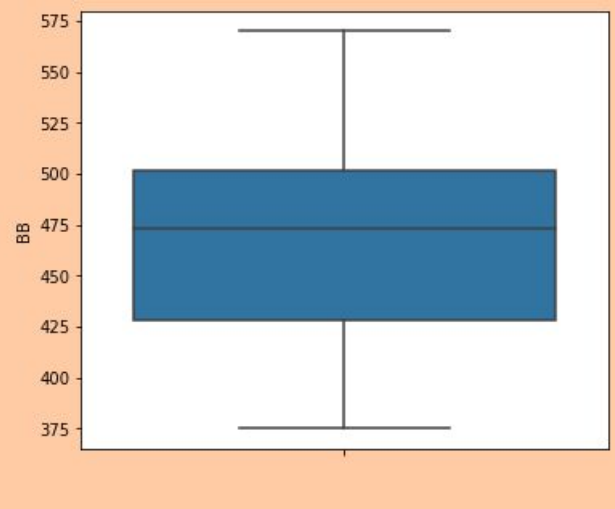
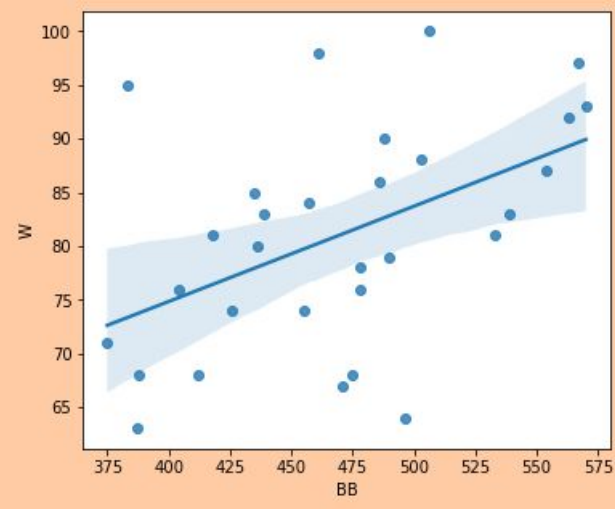
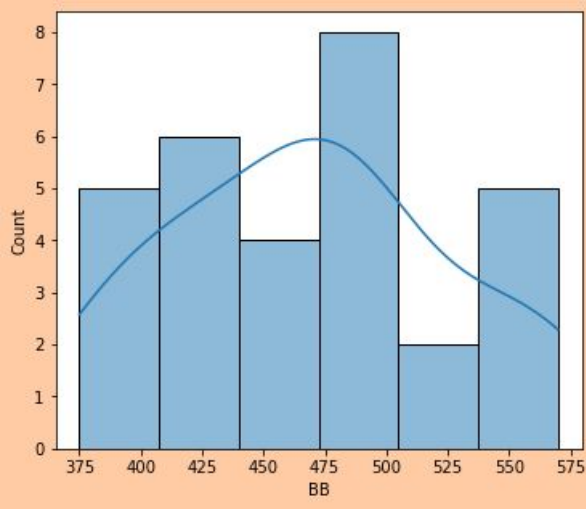
● 3B: Triples



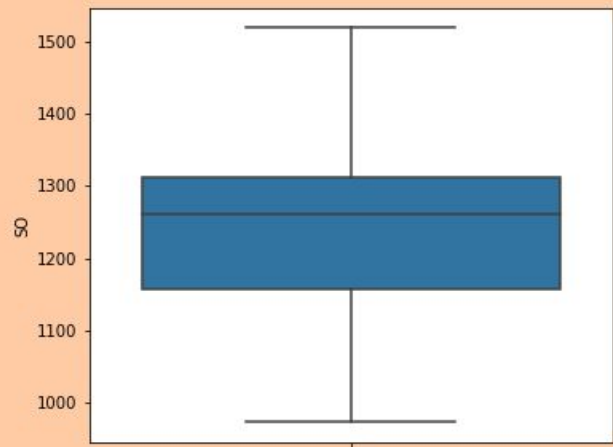
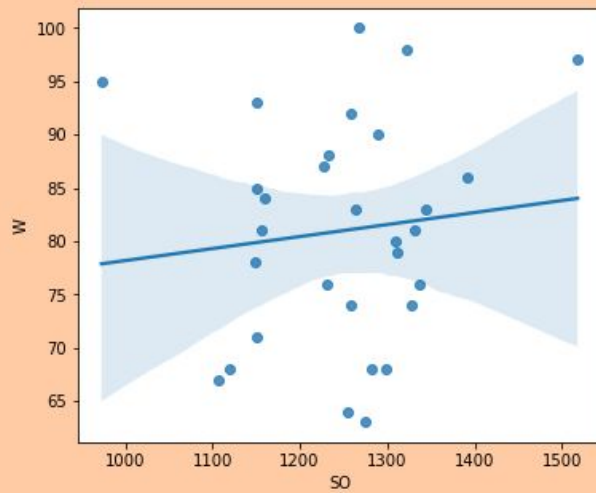
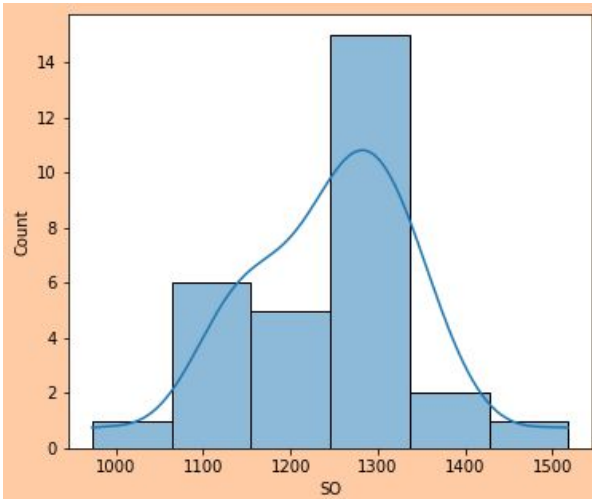
● HR: Homeruns



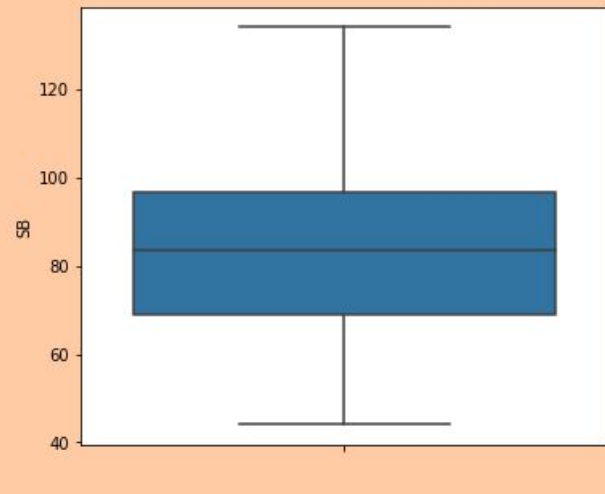
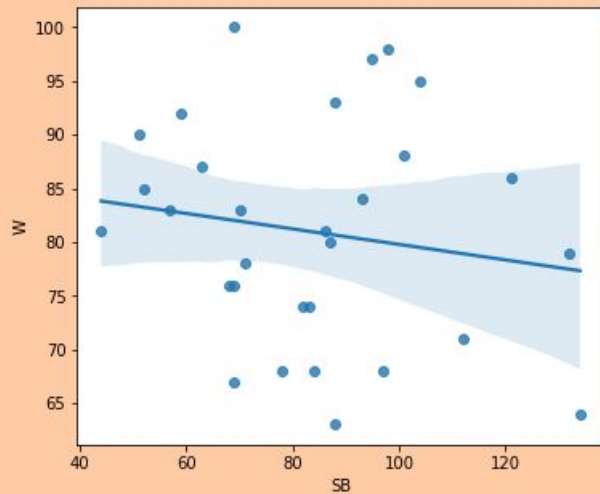
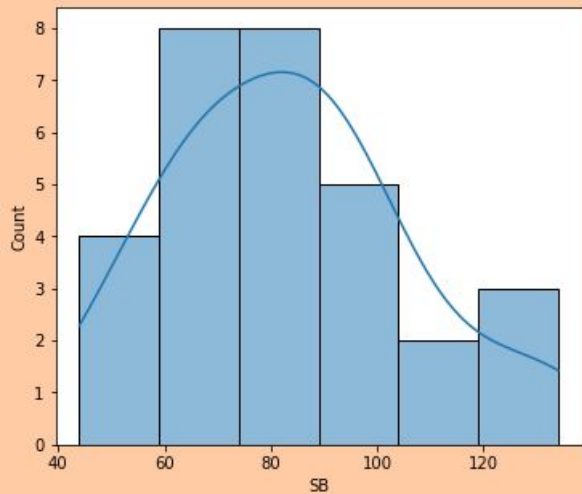
● BB: Walks



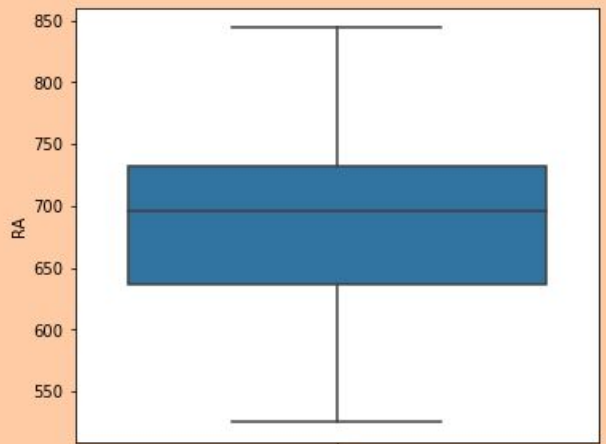
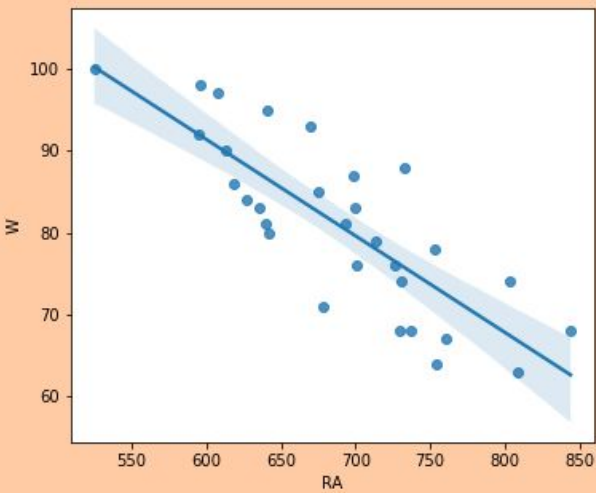
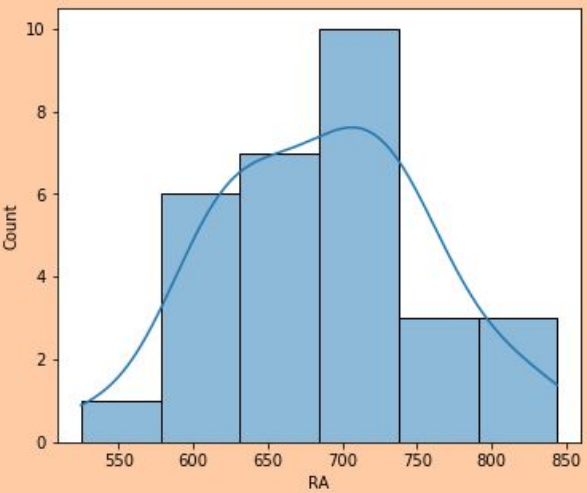
● SO: Strikeouts



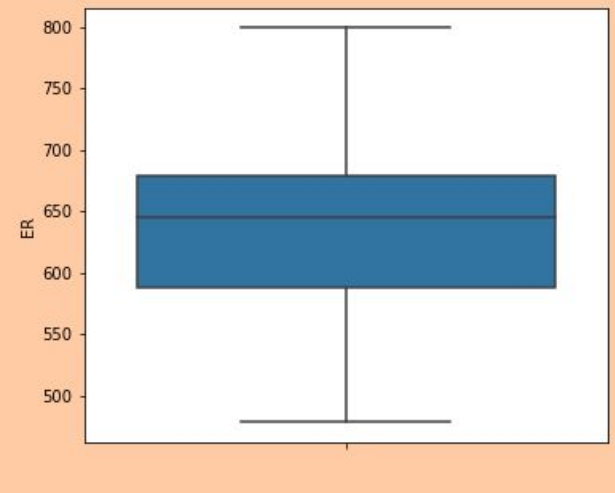
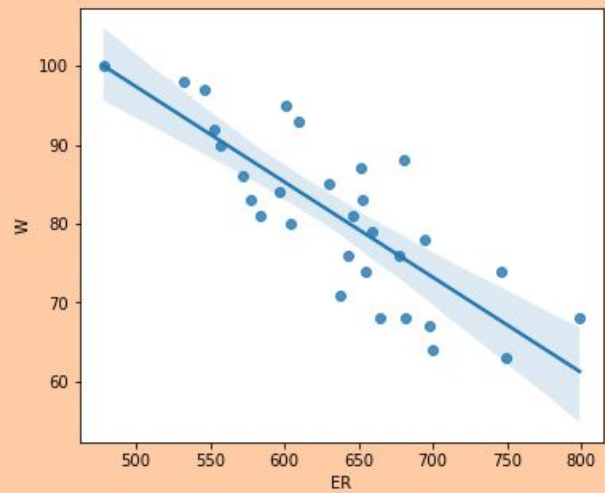
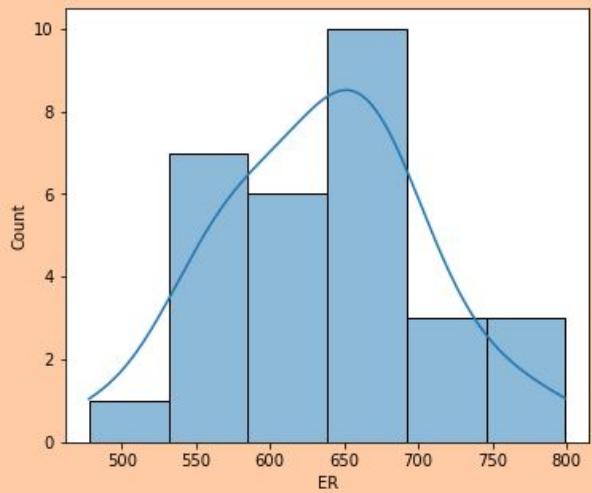
● SB: Stolen Bases



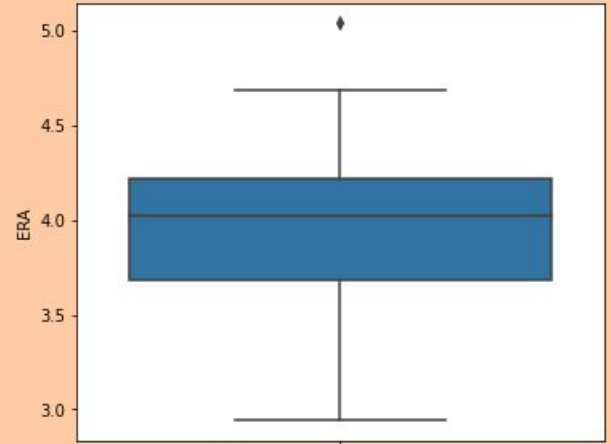
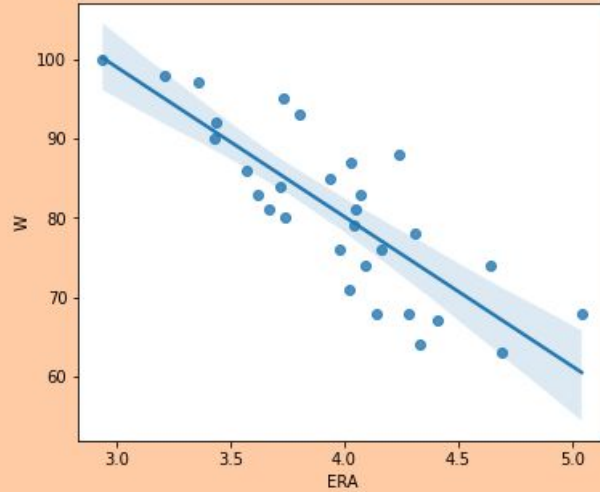
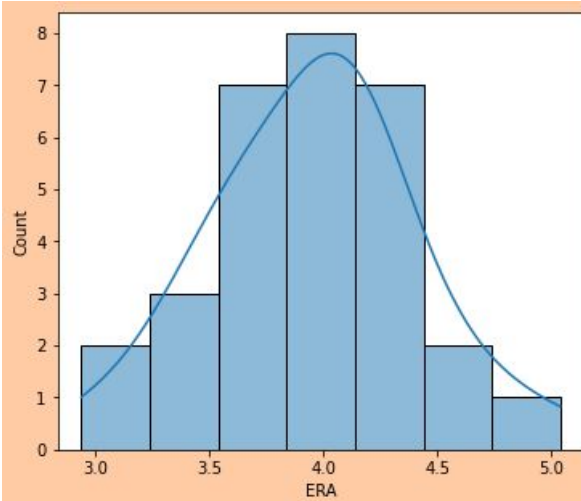
● RA: Runs Allowed



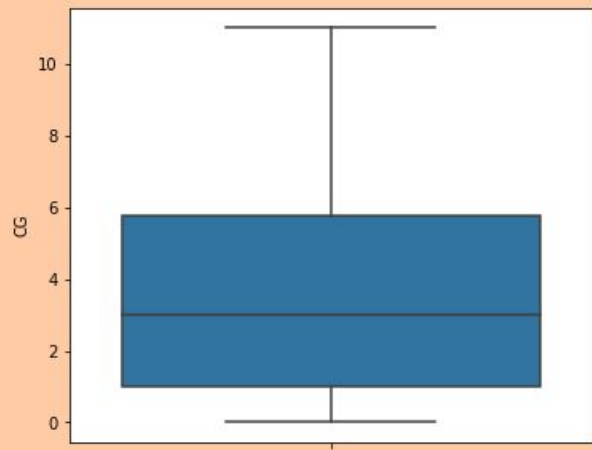
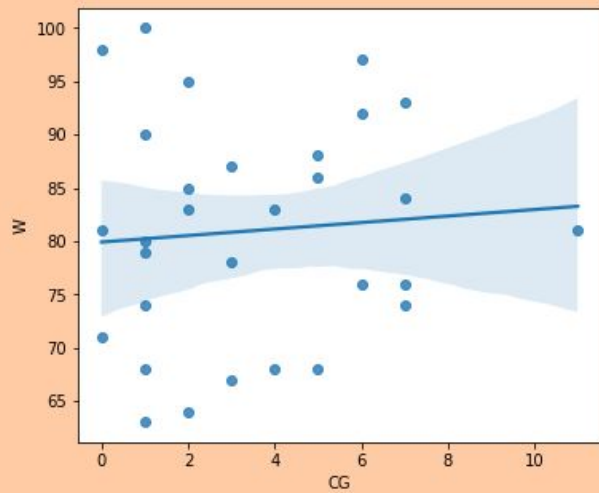
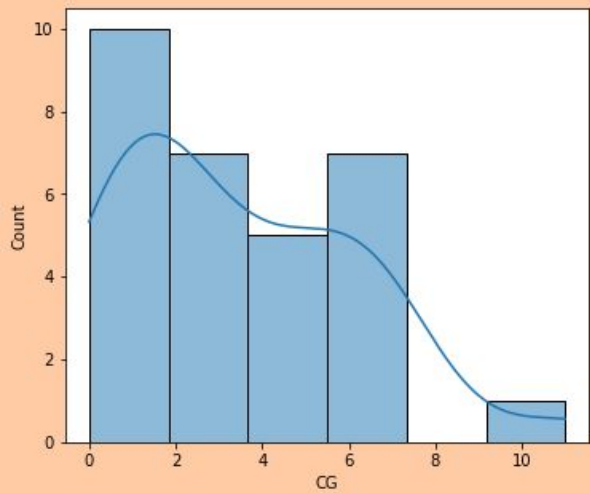
● ER: Earned Runs



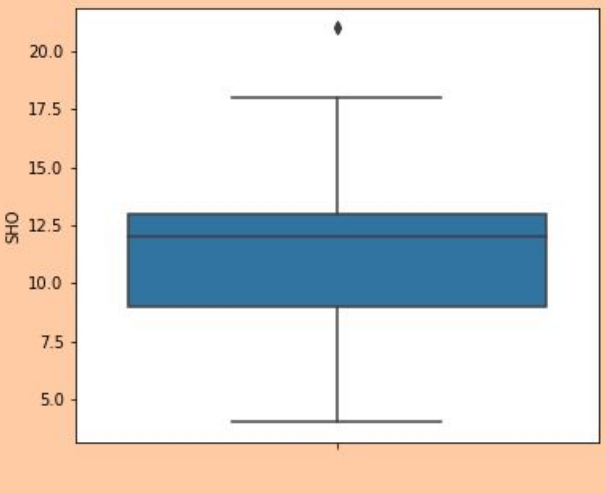
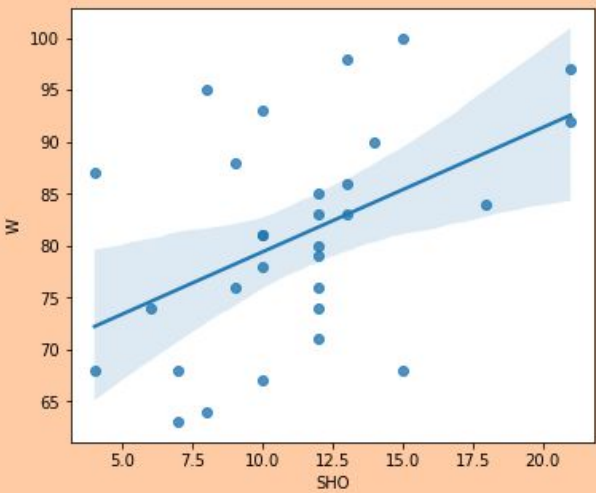
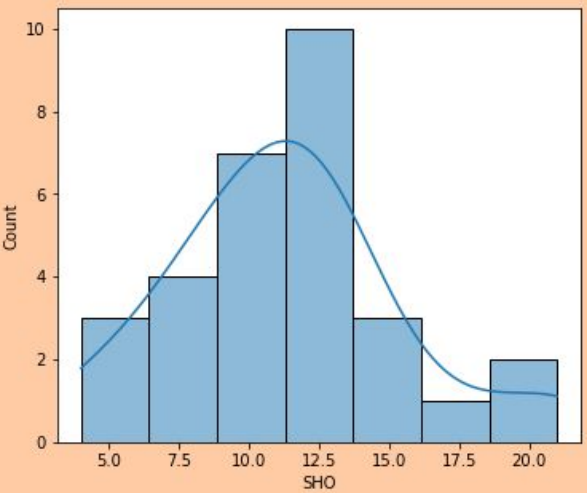
● ERA: Earned Run Average (ERA)



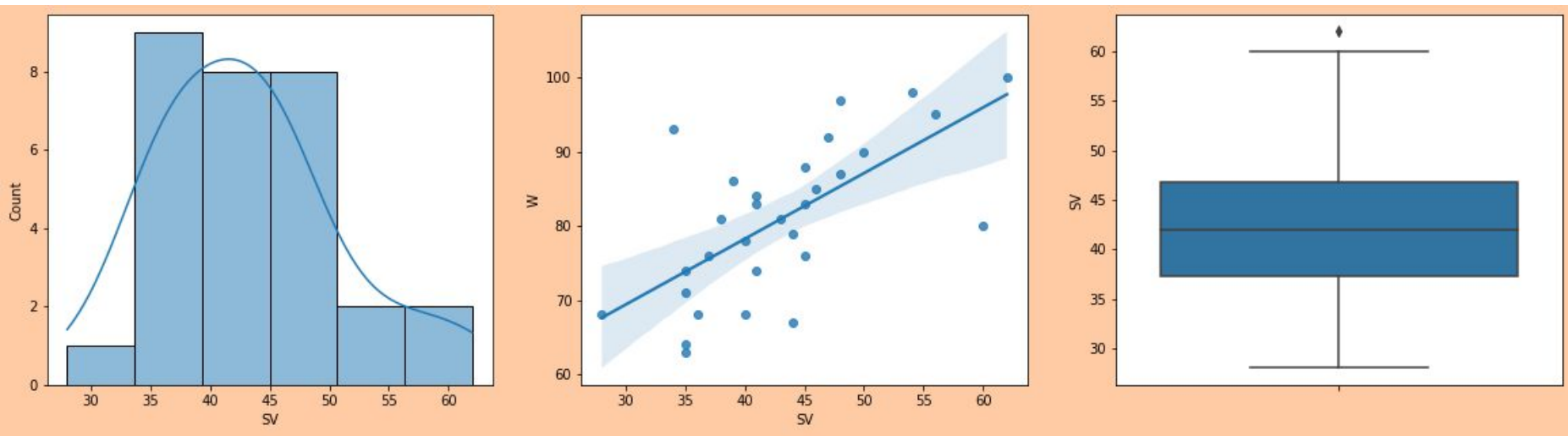
● CG: Shutouts



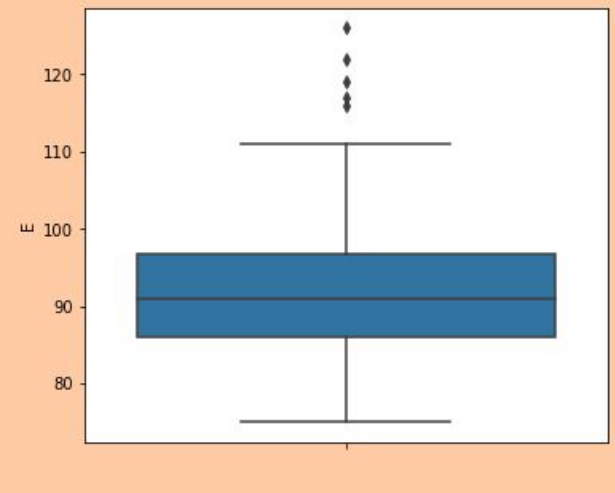
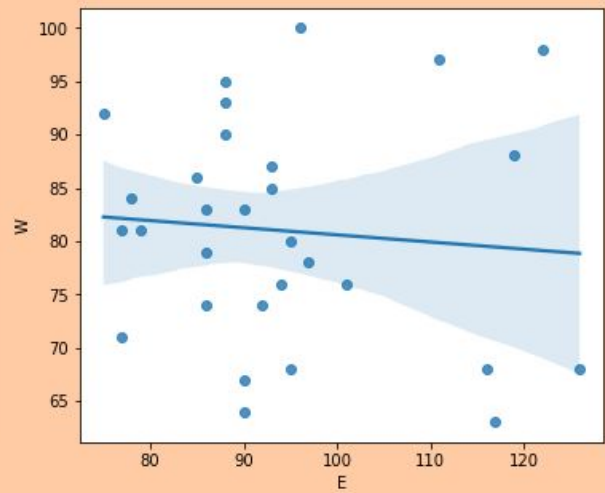
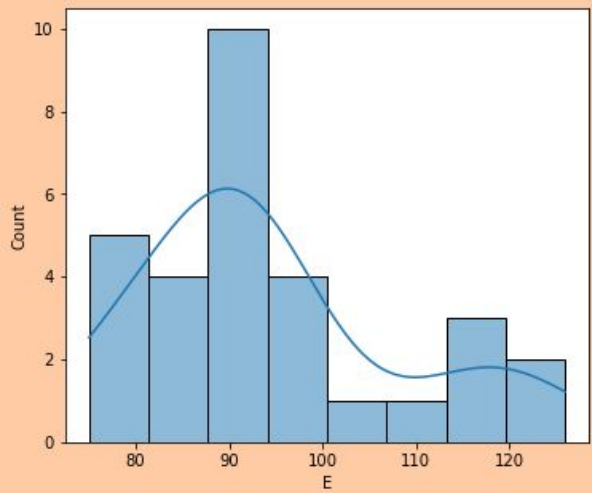
● SHO: Shutout



● SV: Save



● E: Errors



Histogram Interpretation

- Histograms depict the data distribution for a single continuous variable. The X-axis shows the range of values and Y-axis represent the number of values in that range
- The ideal outcome for histogram is a bell curve or slightly skewed bell curve. If there is too much skewness, then outlier treatment should be done and the column should be re-examined, if that also does not solve the problem then only reject the column.

Scatter plot interpretation

What should you look for in scatter plots?

Trend. You should investigate whether or not there is a discernible trend. There could be three possibilities.

Increasing Trend :

This indicates that both variables are positively correlated. In simpler terms, they are directly proportional to each other; as one value rises, the other rises as well. This is great news for ML!

Decreasing Trend:

This indicates that both variables are negatively correlated. In simpler terms, they are inversely proportional; as one value rises, the other falls. This is also beneficial to ML!

No Trend:

There is no discernible increasing or decreasing trend. This indicates that there is no relationship between the variables. As a result, the predictor cannot be used for ML.

Now that the numerical data has been analysed, it is time to tell some stories

- The number of home runs, runs, doubles, shutouts, saves and walks are all strongly positively correlated.
- Stolen bases, runs allowed, and earned runs are all highly negative linearly correlated.
- The other features have less linear correlation with no of Wins

Outlier treatment

- Outliers are extreme values in data that are far from the majority of the values. They can be seen as tails in the histogram.
- Why should outliers be treated?
Outliers bias the training of machine learning models. As the algorithm tries to fit the extreme value, it deviates from the majority of the data.
- There are below two options to treat outliers in the data.
Option-1: Delete the outlier Records. Only if there are just few rows lost.
Option-2: Impute the outlier values with a logical formula



How to remove Outliers..?

The outliers can be removed using the methods listed below.

- Z-score method
- IQR method



Standardization

It is a technique for transforming a data set into a normal distribution. Various methods can be used to remove skewness.

- 1) Log-transformation
- 2) Square root transformation
- 3) Cube root transformation
- 4) Reciprocal transformation
- 5) Box-cox transformation
- 6) Power transformation
- 7) Square
- 8) Cube

Normalization The process of normalization consists of scaling the data into a certain range of values

1

Min-Max-Scaler

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

2

Mean Normalization

$$x' = \frac{x - x_{mean}}{x_{max} - x_{min}}$$

3

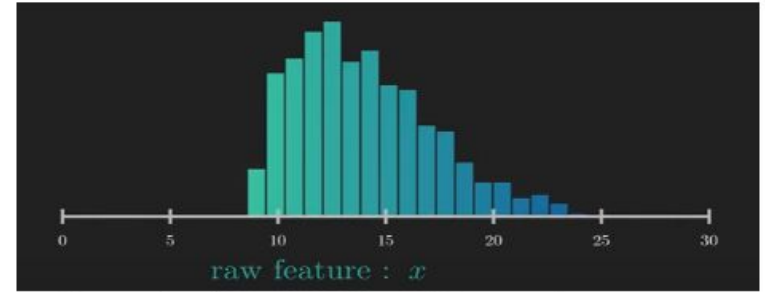
Standard scaling(z-score)

$$z = \frac{x - \mu}{\sigma}$$

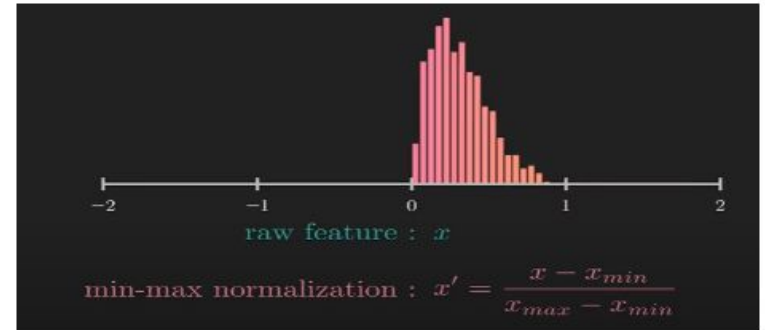
- **Min-Max-Scaler**

it converter the dataset into the range of 0 to 1

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$



Data before scaling Range is **10 to 25**



Data after scaling range is **0 to 1**

- **Mean Normalization**

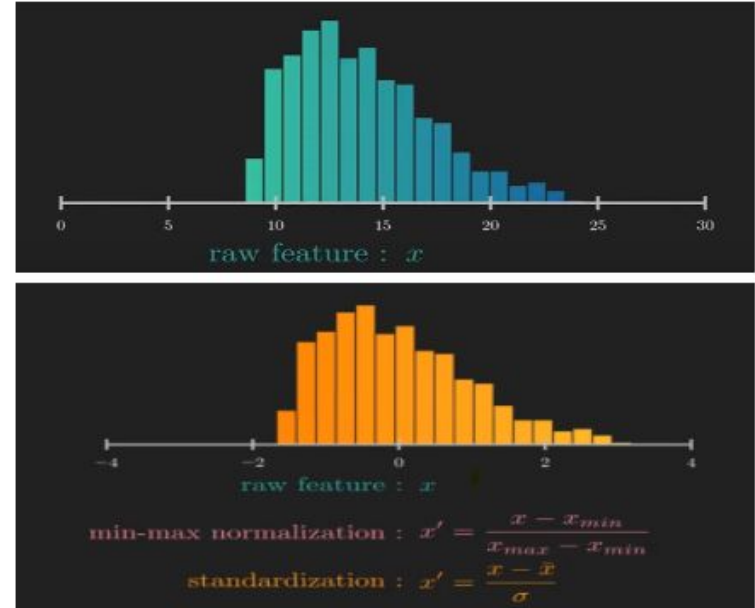
it converts the data set into the range of -1 to 1
and the mean is 0.

$$x' = \frac{x - x_{mean}}{x_{max} - x_{min}}$$

- **Standard scaling(z-score)**

its converts data set as mean=0, standard deviation as 1.

$$z = \frac{x - \mu}{\sigma}$$

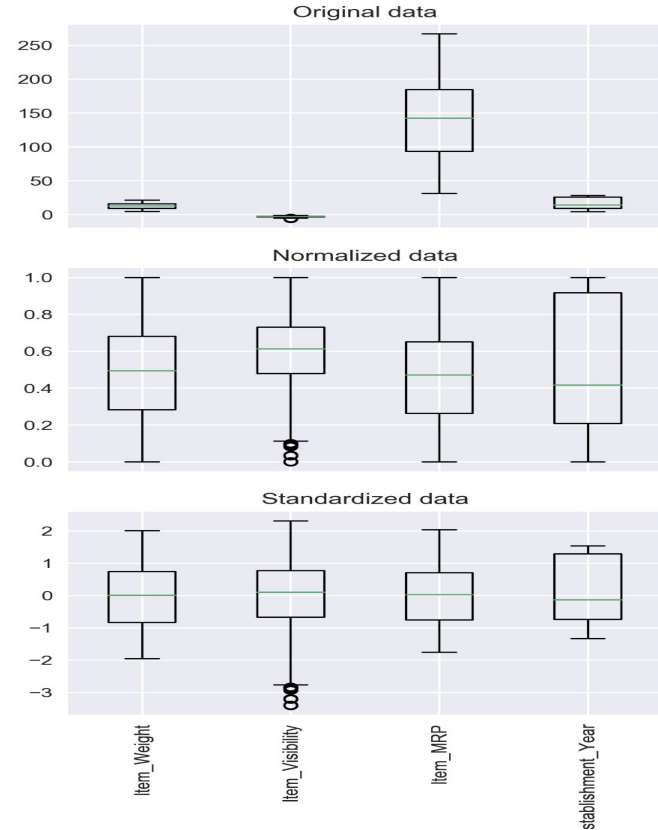


Data after scaling range is **mean=0 SD = 1**

Note:- there is a huge misconception among beginners that is if we apply Standard scaling(z-score) then it makes the distribution normal distribution But this is not true

Normalization vs Standardization

- There is no any thumb rule to use of Standardization or Normalization for special ML algorithms.
- But mostly Standardization is used for clustering analyses, Principal Component Analysis (PCA).
- Normalization prefer for Image processing because pixel intensity is between 0 to 255, the neural network algorithm requires data in scale 0-1, K-Nearest Neighbors




Feature Selection Techniques

This is the time to select the best columns (Features) that are relevant to the Target.


This can be accomplished directly by measuring correlation values or using ANOVA/Chi-Square tests.

To get a better sense of the data,

It is always helpful to visualise the relationship between the Target variable and each of the predictors.



The following techniques can be used to determine the relationship between two variables



1) Visual exploration of relationship between variables

1. Continuous Vs Continuous ----> Scatter Plot
2. Categorical Vs Continuous ----> Box Plot
3. Categorical Vs Categorical ----> Grouped Bar Plots

2) Statistical evaluation of the strength of a relationship between variables

1. Continuous Vs Continuous ----> Correlation matrix
2. Categorical Vs Continuous ----> ANOVA test--->SelectKBest
3. Categorical Vs Categorical ---> Chi-Square test--->SelectPercentile

Target variable
is Continuous,
hence following
scenarios will
be present

Pearson correlation coefficient / Formula :

$$r = \frac{\sum (x_i - \bar{x}) (y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

r = correlation coefficient

x_i = values of the x-variable in a sample

\bar{x} = mean of the values of the x-variable

y_i = values of the y-variable in a sample

\bar{y} = mean of the values of the y-variable

This value can only be computed between two numerical columns.

1

The scatter plot will show a downward trend if the correlation is between $[-1,0]$.

2

Correlation between $(0,1)$ denotes a direct proportional relationship. The scatter plot will indicate an upward trend.

3

Correlation $\{0\}$ means No relationship. The scatter plot will show no discernible trend.

4

If the magnitude of the correlation between two variables is greater than 0.5, It denotes a favourable relationship.

5

We observed that the relationships between features and label to determine which columns are relevant to the target variable

Variance Inflation Factor (VIF)

VIF is a measure of regression analysis multicollinearity.

Whenever there is a correlation between multiple independent variables in a multiple regression model, it is considered multicollinearity.

Due to multicollinearity, the variance inflation factor can be used to estimate the amount by which the variance of a regression coefficient is inflated.

$$VIF\ x_i = \frac{1}{Tolerance} = \frac{1}{1 - R_i^2}$$

Machine Learning

Splitting the data into Training
and Testing sample

Model Prediction

Model Selection

Model Evaluation Metrics

Model Training

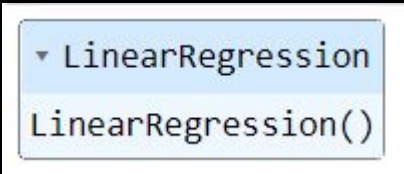
Splitting the data into Training and Testing sample

```
x_train,x_test,y_train,y_test=train_test_split(sca  
led_feature,l,test_size=0.3,random_state=30)
```

Model Selection & Model Training

```
lrm = LinearRegression()
```

```
lrm.fit(x_train, y_train)
```



```
▼ LinearRegression  
LinearRegression()
```

Model Prediction

- `y_pred = lrm.predict(x_test)`
- `y_pred`
- `print ('Chance of Win is',lrm.predict (scalar.transform ([[650,260,148,426,4.09,6,41]]])))`

Model Evaluation Metrics

- 1) Residuals
- 2) Mean absolute error
- 3) Mean Square error
- 4) Root Mean Square Error
- 5) R-squared /Adjusted R squared

Regularization

When we use regression models to train some data, there is a good chance that the model will overfit the given training data set. Regularization helps sort this overfitting problem by restricting the degrees of freedom of a given equation i.e. simply reducing the number of degrees of a polynomial function by reducing their corresponding weights.

In a linear equation, we do not want huge weights/coefficients as a small change in weight can make a large difference for the dependent variable (Y). So, regularization constraints the weights of such features to avoid overfitting.

To regularize the model, a Shrinkage penalty is added to the cost function. Let's see different types of regularizations in regression:

Type of Regularization

- **Lasso / L1 form**

LASSO regression penalizes the model based on the sum of magnitude of the coefficients. The regularization term is given by

$$\text{Regularization} = \lambda * \sum |\beta_i|$$

Where λ is shrinkage factor

- **Ridge /L2 form**

Ridge regression penalizes the model based on the sum of squares of magnitude of the coefficients. The regularization term is given by

$$\text{Regularization} = \lambda * \sum |\beta_i|^2$$

Where λ is shrinkage factor

Difference between Ridge and Lasso

Ridge regression shrinks the coefficients for those predictors which contribute very less in the model but have huge weights, very close to zero. But it never makes them exactly zero.

Thus, the final model will still contain all those predictors, though with less weights. This doesn't help in interpreting the model very well.

This is where Lasso regression differs with Ridge regression. In Lasso, the L1 penalty does reduce some coefficients exactly to zero when we use a sufficiently large tuning parameter λ .

So, in addition to regularizing, lasso also performs feature selection.

Why use Regularization?

- Regularization helps to reduce the variance of the model, without a substantial increase in the bias. If there is variance in the model that means that the model won't fit well for dataset different than training data.
- The tuning parameter λ controls this bias and variance tradeoff. When the value of λ is increased up to a certain limit, it reduces the variance without losing any important properties in the data. But after a certain limit, the model will start losing some important properties which will increase the bias in the data. Thus, the selection of good value of λ is the key.
- The value of λ is selected using cross-validation methods. A set of λ is selected and cross-validation error is calculated for each value of λ and that value of λ is selected for which the cross-validation error is minimum.
- It is a technique that are used to **calibrate** machine learning models
- Overfitting is prevented by this technique by adding extra information to it.



Concluding Remarks

- `lasso_reg.score`
- `ridge_model.score`
- `regression.score`

So, we have seen by using different type of regularization, we still are getting the same r^2 score. That means our OLS model has been well trained over the training data and there is no overfitting.

Contact me if you have ANY QUESTIONS

Contact No: +91 8485010139 / 7773916816

Email Id: patilkundan.1718@gmail.com

[LinkedIn](#)

[github](#)