World Happiness Report

Submitted by:

Swetha Joshi

PROJECT FLOW:

1. PROBLEM STATEMENT
2. DATA ANALYSIS
3. EDA CONCLUSIONS
4. DATA PRE-PROCESSING PIPELINE
5. MODEL BUILDING
6. CONCLUDING REMARKS

1. PROBLEM STATEMENT

The World Happiness Report is a landmark survey of the state of Global Happiness. The first report was published in 2012, the second in 2013, the third in 2015 and the fourth in 2016 update. The World Happiness 2017, which ranks 155 countries by their happiness levels, was released at the United Nations at an event celebrating International Day of Happiness on March 20th. The reports continue to gain global recognition as governments, organizations and civil society increasingly use happiness indicators to inform their policy-making decisions. Leading experts across fields - economics, psychology, survey analysis, national statistics, health, public policy and more - describe how measurements of Well-being can be used effectively to assess the progress of nations. The reports review the state of happiness in the world today and show how the new science of happiness explains personal and national variations in happiness.

What are the residuals?

The residuals or unexplained components, differ for each country, reflecting the extent to which the six variables either over - or under-explain average life evaluations. These residuals have an average value of approximately zero over the whole set of countries.

What do the columns succeeding the Happiness Score (like Family, Generosity, etc.) describe?

The following columns: GDP per Capita, Family, Life Expectancy, Freedom, Generosity, Trust Government Corruption describe the extent to which these factors contribute in evaluating the happiness in each country.

The Dystopia Residual metric actually is the Dystopic Happiness Score (1.85) + the Residual value or the unexplained value for each country.

If you add all these factors up, you get the happiness score so it might be un-reliable to model them to predict Happiness Scores.

2. DATA ANALYSIS

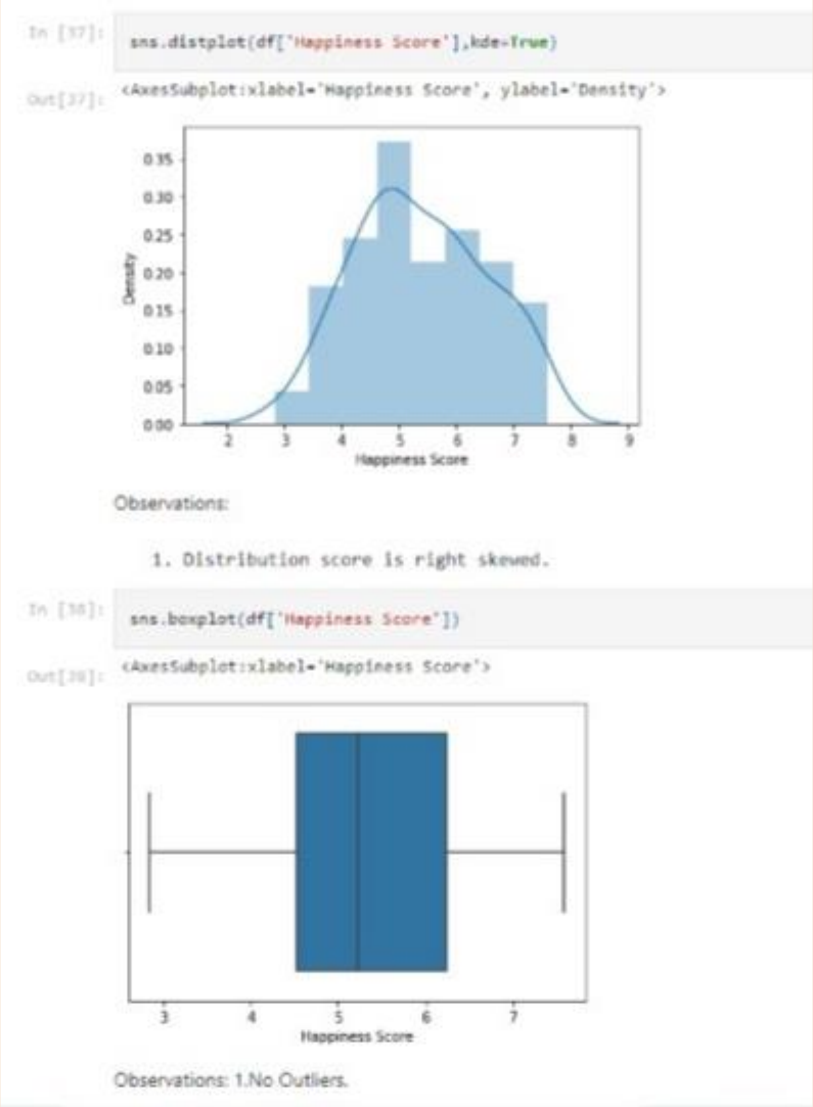
We have analysed this dataset in three stages

1.UNIVARIATE ANALYSIS:

In this analysis we have taken each column one by one and we thoroughly analysed it, depending upon the data it holds we used suitable technique to extract maximum information out of it.

In the below example of Happiness Score column analysis through distplot showing how the Happiness Scores are being distributed. We can see that Happiness Score is slightly right skewed.

Univariate Analysis mainly infers that analysis of only one variable.



In [37]:

```
sns.distplot(df['Happiness Score'],kde=True)
```

Out[37]:

```
<AxesSubplot:xlabel='Happiness Score', ylabel='Density'>
```

Observations:

1. Distribution score is right skewed.

In [38]:

```
sns.boxplot(df['Happiness Score'])
```

Out[38]:

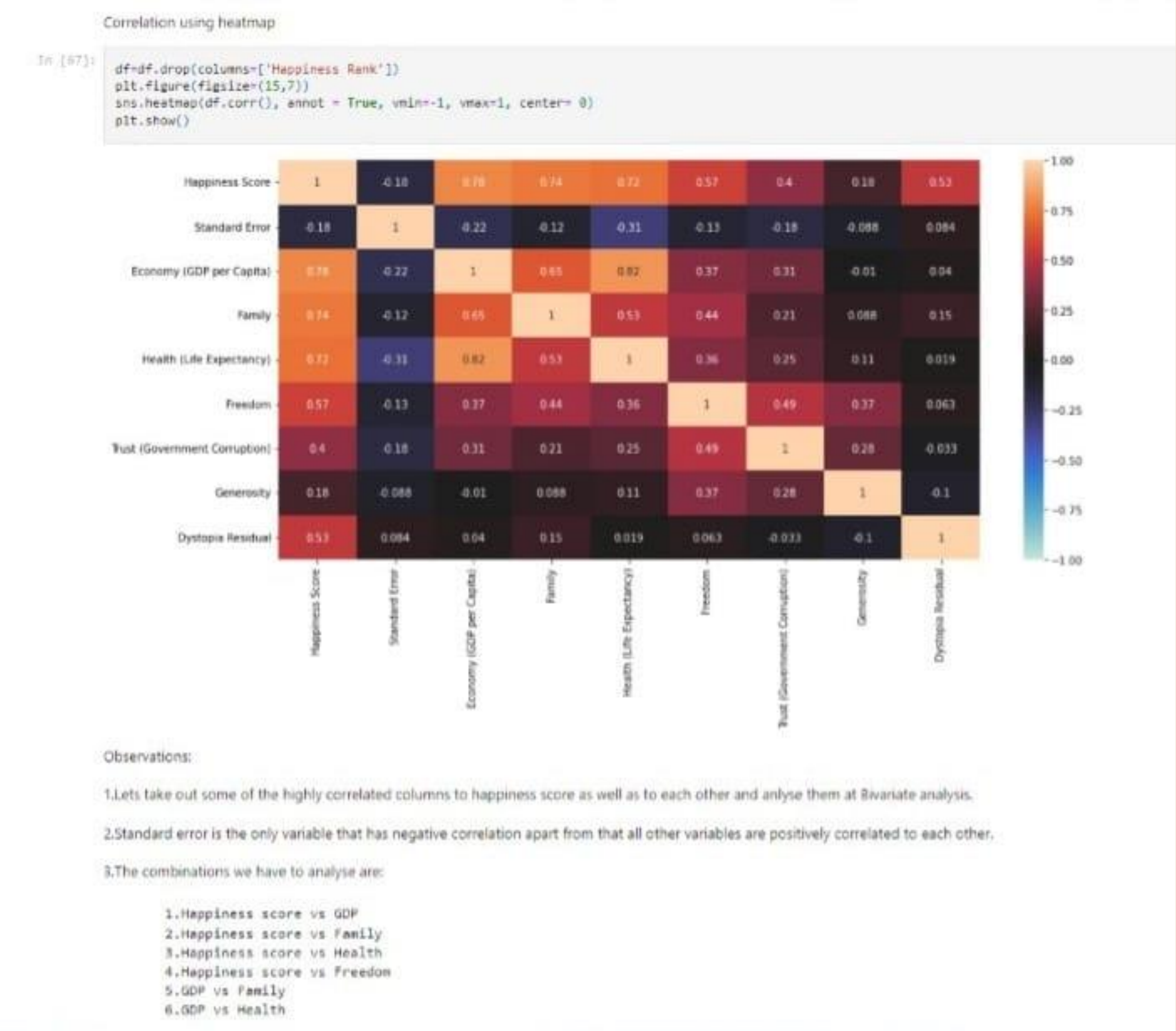
```
<AxesSubplot:xlabel='Happiness Score'>
```

Observations: 1.No Outliers.

We also got to know that there are no outliers in Happiness Score column through boxplot.

2.MULTIVARIATE ANALYSIS:

In case of Multivariate Analysis, we analyse more than two variable or multiple variables simultaneously. We will get to know the relationship between different variables, by doing so we can extract maximum information which are hidden.



Correlation using heatmap

```
In [67]: df=df.drop(columns=['Happiness Rank'])
         plt.figure(figsize=(15,7))
         sns.heatmap(df.corr(), annot = True, vmin=-1, vmax=1, center= 0)
         plt.show()
```

	Happiness Score	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom	Trust (Government Corruption)	Generosity	Dystopia Residual
Happiness Score	1	-0.18	0.78	0.74	0.72	0.57	0.4	0.18	0.53
Standard Error	-0.18	1	-0.22	-0.12	-0.31	-0.13	-0.18	-0.088	0.084
Economy (GDP per Capita)	0.78	-0.22	1	0.65	0.82	0.37	0.31	-0.01	0.04
Family	0.74	-0.12	0.65	1	0.53	0.44	0.21	0.088	0.15
Health (Life Expectancy)	0.72	-0.31	0.82	0.53	1	0.36	0.25	0.11	0.019
Freedom	0.57	-0.13	0.37	0.44	0.36	1	0.49	0.37	0.063
Trust (Government Corruption)	0.4	-0.18	0.31	0.21	0.25	0.49	1	0.28	-0.033
Generosity	0.18	-0.088	-0.01	0.088	0.11	0.37	0.28	1	-0.1
Dystopia Residual	0.53	0.084	0.04	0.15	0.019	0.063	-0.033	-0.1	1

Observations:

1.Lets take out some of the highly correlated columns to happiness score as well as to each other and anlyse them at Bivariate analysis.

2.Standard error is the only variable that has negative correlation apart from that all other variables are positively correlated to each other.

3.The combinations we have to analyse are:

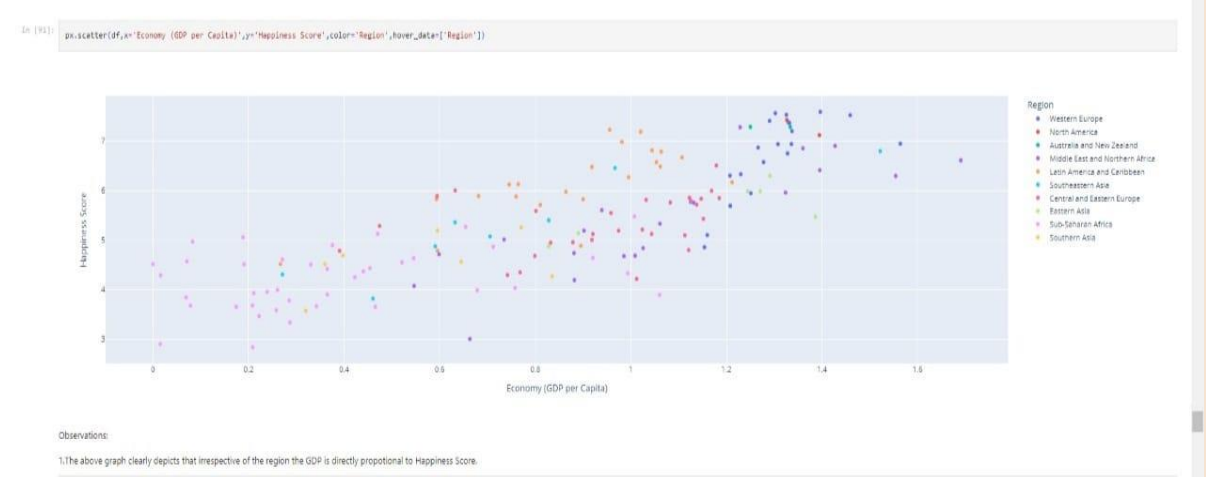
```
1.Happiness score vs GDP
2.Happiness score vs Family
3.Happiness score vs Health
4.Happiness score vs Freedom
5.GDP vs Family
6.GDP vs Health
```

The above picture is the example of multivariate analysis, the plot drawn in the picture is corrplot depicting the relationship between all the elements simultaneously.

We have different plots for carrying out the multivariate analysis, Heatmap and corrplot are the major ones through which we will get to know how each variable is relates to each other. Grouping the variable and analysing them is also a type of multivariate analysis, from which we can extract maximum information because we can group the categories like yes or no and we can compare the features. From heatmap we will get to know whether a variable is positively correlated or negatively correlated.

3.BIVARIATE ANALYSIS:

From multivariate analysis we will get to know which are those independent variables which are significantly correlated to the response variable we can analyse those pairs in the bi variate analysis.



```
In [91]: px.scatter(df,x='Economy (GDP per Capita)',y='Happiness Score',color='Region',hover_data=['Region'])
```

Observations:

1.The above graph clearly depicts that irrespective of the region the GDP is directly propotional to Happiness Score.

Above is an example of bi variate analysis in which we are analysing Economy (GDP per capita income) and Happiness Score.

From bi variate analysis we will get know how an independent variable is related to response variable, and we can prescribe required precautions to be effective.

3. EDA CONCLUSIONS

1. Western Europe, Australia, North America, Latin America and Caribbean has got significantly high Happiness score.

2.There are countries Israel in the northern Africa which has got good Happiness Score. 3.Sub Sahara Africa has got countries which has low Happiness scores.

3. The Country's GDP is directly proportional to Happiness Score.

4. Health is the major criteria for the Happiness Score irrespective of the region.

5. Family is the major criteria for the Happiness Score.

6. Freedom is also an important criterion for Happiness Score.

4. DATA PRE-PROCESSING PIPELINE

Cleaning and Preparing a precise input data:

1. Cleaning the dataset: This phase includes finding null values or unexpected characters in the dataset and treating them, we can treat them by

various techniques like replacing, deleting or by removing the rows.

2. Removing the Outliers: This phase includes removing of the outliers, there are various methods for removing the outliers. i) zscore method, ii) IQR method and iii) Standard Deviation Method

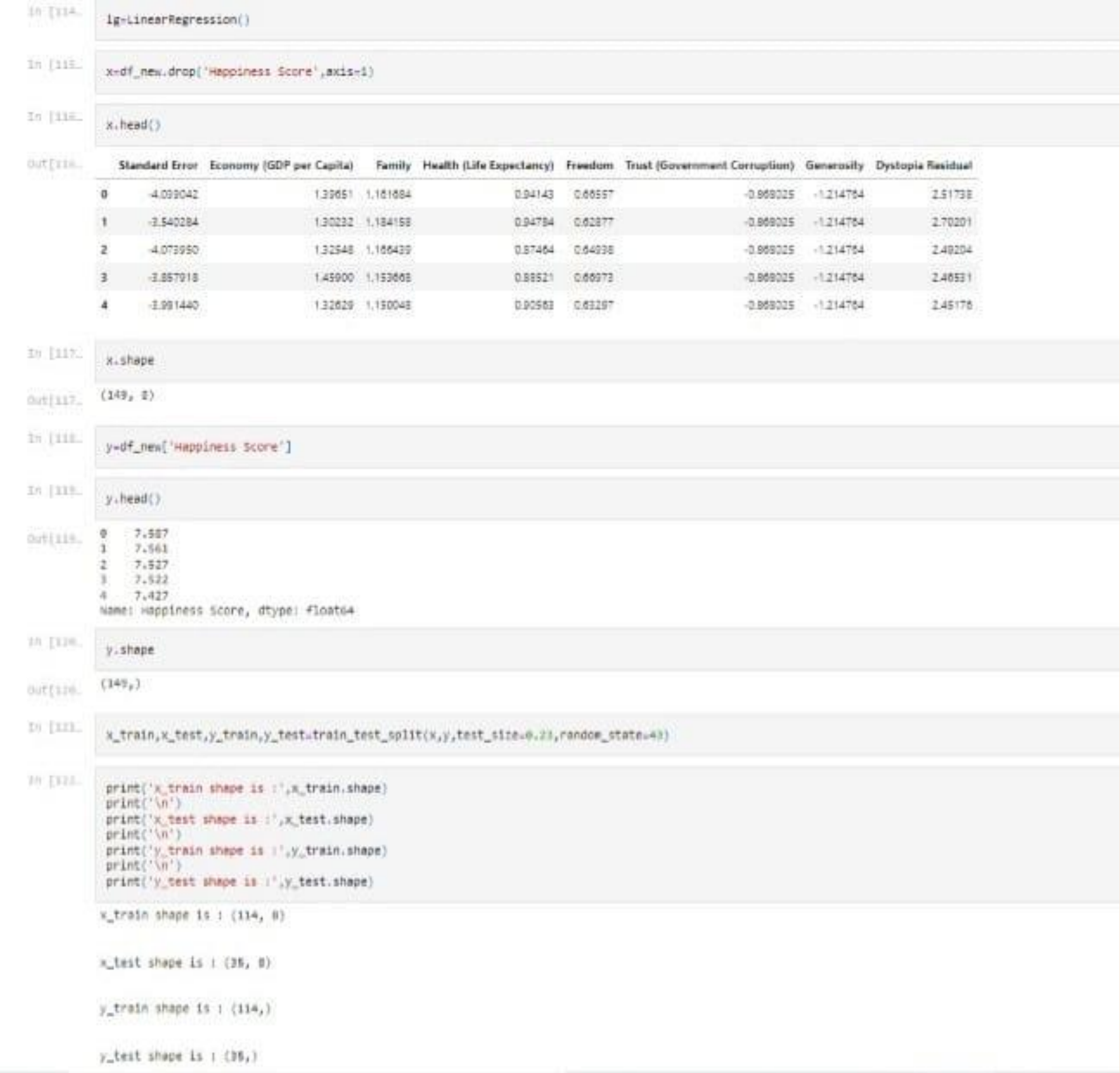
We can use convenient method depending upon the situation, we have to ensure that the data loss should not be more than 7%.

3. Removal of Skewness: Except categorical columns we have to remove the skewness of all the columns.
4. Scaling the independent variables: We have to scale the independent variables if required so that the model will adapt high accuracy.

5.MODEL BUILDING

We have ample number of models for analysing and building a solution for the classification type of problem. In this data set we are going to look at the best accuracy score and f1 score of each model and select the best model based on the result.

1. Linear Regression:



```
In [114…  lg=LinearRegression()
```

```
In [115…  x=df_new.drop('Happiness Score',axis=1)
```

```
In [116…  x.head()
```

Out[116…

	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom	Trust (Government Corruption)	Generosity	Dystopia Residual
0	-4.099042	1.39651	1.161684	0.94143	0.66557	-0.869025	-1.214764	2.51738
1	-3.540284	1.30232	1.184158	0.94784	0.62877	-0.869025	-1.214764	2.70201
2	-4.073950	1.32548	1.166439	0.87464	0.64938	-0.869025	-1.214764	2.49204
3	-3.857918	1.45900	1.153668	0.88521	0.66973	-0.869025	-1.214764	2.46531
4	-3.991440	1.32629	1.150048	0.90563	0.63297	-0.869025	-1.214764	2.45176

```
In [117…  x.shape

Out[117…  (149, 8)
```

```
In [118…  y=df_new['Happiness Score']
```

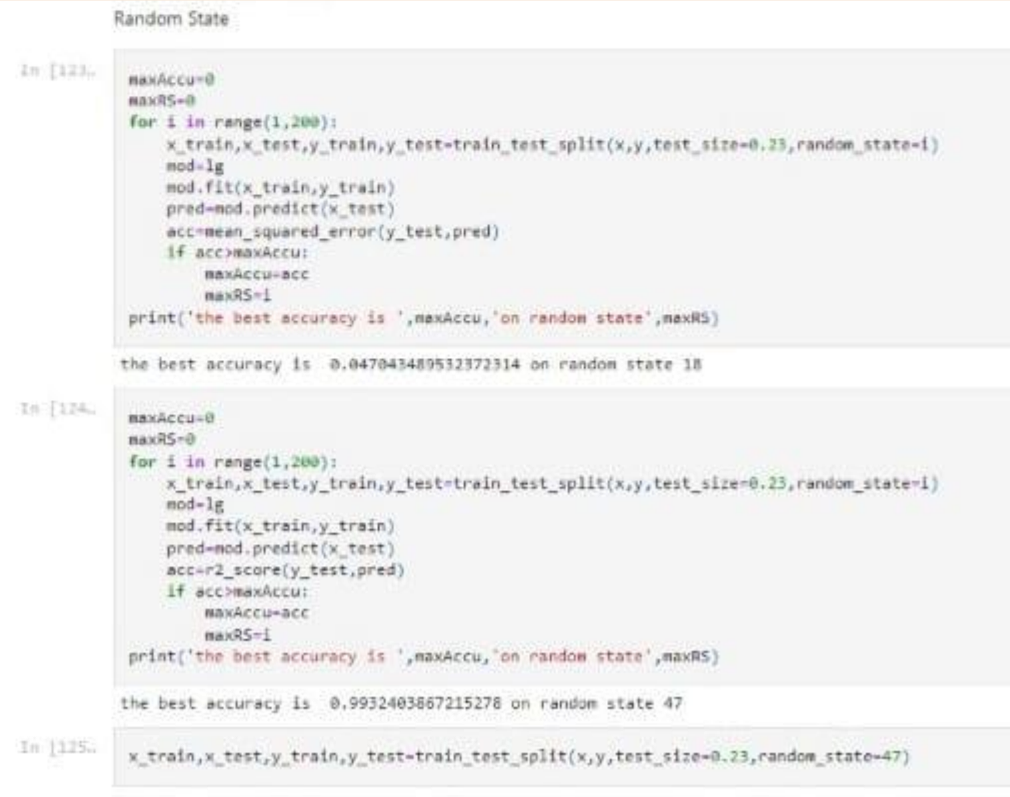
```
In [119…  y.head()

Out[119…  0    7.587
          1    7.561
          2    7.527
          3    7.522
          4    7.427
          Name: Happiness Score, dtype: float64
```

```
In [120…  y.shape

Out[120…  (149,)
```

```
In [121…  x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.23,random_state=43)
```

```
In [122…  print('x_train shape is :',x_train.shape)
          print('\n')
          print('x_test shape is :',x_test.shape)
          print('\n')
          print('y_train shape is :',y_train.shape)
          print('\n')
          print('y_test shape is :',y_test.shape)

          x_train shape is : (114, 8)


          x_test shape is : (35, 8)


          y_train shape is : (114,)


          y_test shape is : (35,)
```

2. Random State:



Random State

```
maxAccu=0
maxRS=0
for i in range(1,200):
    x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.23,random_state=i)
    mod=lg
    mod.fit(x_train,y_train)
    pred=mod.predict(x_test)
    acc=mean_squared_error(y_test,pred)
    if acc>maxAccu:
        maxAccu=acc
        maxRS=i
print('the best accuracy is ',maxAccu,'on random state',maxRS)
```

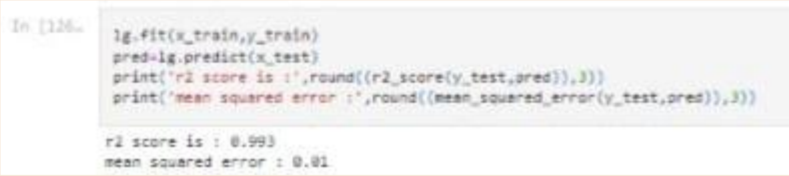
```
the best accuracy is  0.047043489532372314 on random state 18
```

```
maxAccu=0
maxRS=0
for i in range(1,200):
    x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.23,random_state=i)
    mod=lg
    mod.fit(x_train,y_train)
    pred=mod.predict(x_test)
    acc=r2_score(y_test,pred)
    if acc>maxAccu:
        maxAccu=acc
        maxRS=i
print('the best accuracy is ',maxAccu,'on random state',maxRS)
```

```
the best accuracy is  0.9932403867215278 on random state 47
```

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.23,random_state=47)
```

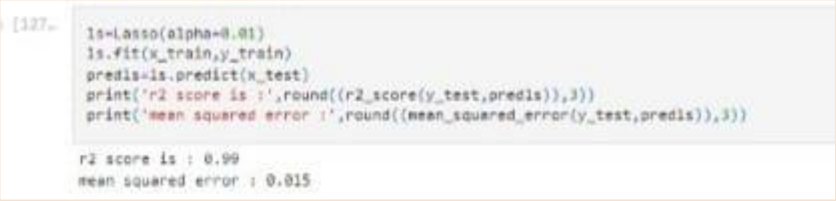
3. lg.fit:



```
lg.fit(x_train,y_train)
pred=lg.predict(x_test)
print('r2 score is :',round((r2_score(y_test,pred)),3))
print('mean squared error :',round((mean_squared_error(y_test,pred)),3))
```

```
r2 score is : 0.993
mean squared error : 0.01
```

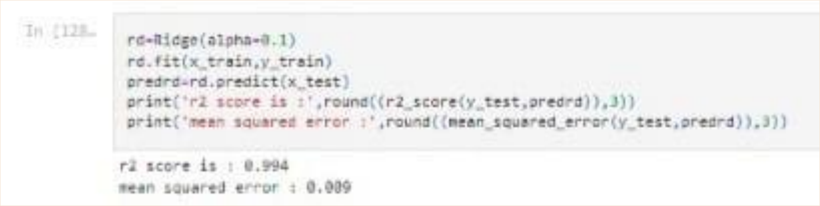
4. Lasso:



```
[127…
ls=Lasso(alpha=0.01)
ls.fit(x_train,y_train)
predls=ls.predict(x_test)
print('r2 score is :',round((r2_score(y_test,predls)),3))
print('mean squared error :',round((mean_squared_error(y_test,predls)),3))
```

```
r2 score is : 0.99
mean squared error : 0.015
```

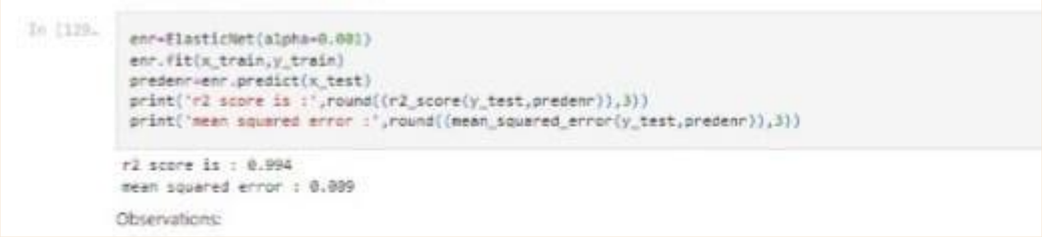
5. Ridge:



```
In [128…
rd=Ridge(alpha=0.1)
rd.fit(x_train,y_train)
predrd=rd.predict(x_test)
print('r2 score is :',round((r2_score(y_test,predrd)),3))
print('mean squared error :',round((mean_squared_error(y_test,predrd)),3))
```

```
r2 score is : 0.994
mean squared error : 0.009
```

6. ElasticNet:



```
In [129…
enr=ElasticNet(alpha=0.001)
enr.fit(x_train,y_train)
predenr=enr.predict(x_test)
print('r2 score is :',round((r2_score(y_test,predenr)),3))
print('mean squared error :',round((mean_squared_error(y_test,predenr)),3))
```

```
r2 score is : 0.994
mean squared error : 0.009
```

Observations:

6.CONCLUSION REMARKS:

The model has been built for best accuracy optimising all loop holes in the dataset. The ElasticNet (enr) model has best r2_score and least mean_squared_error so, we will take enr model and save.

Thank you!