2020 Case Study

Predict Cancer Mortality Rates for US Countries

The problem

Description

Prediction of the “Cancer Mortality Rates” in the countries of United States of America.

Task

Building a Multivariate Ordinary Least Squares Regression Model to Predict “TARGET_deathRate”

Deliverables

A Jupyter Notebook with all the Compiled Code.

Challenges deep-dive

Challenge 1

Exploratory Data Analysis

Data Cleaning and Fixing.

Handling Outliers.

Visualisation.

Challenge 2

Feature Scaling and Selection

MinMax Scaling.

Coarse and Fine Tuning with RFE.

Challenge 3

Model Assessment and Comparison

Adjusted R^2 and AIC, BIC.

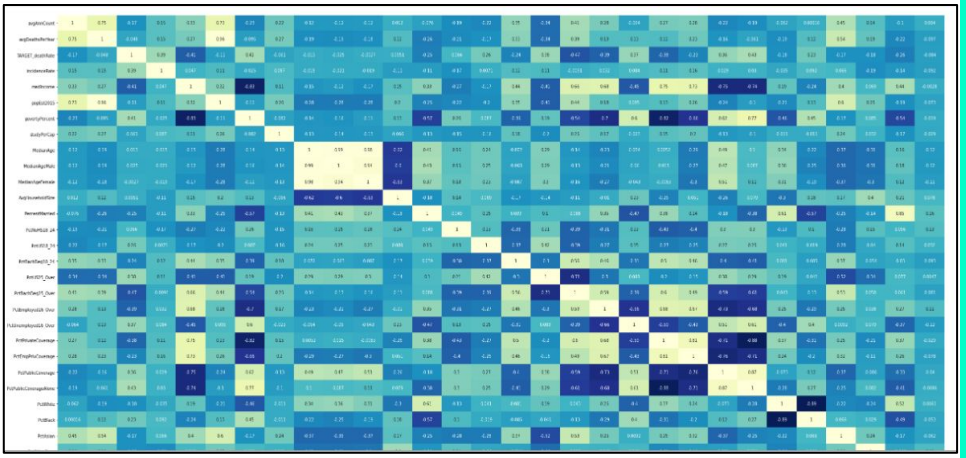
Prediction of the Model.

Implementation

TASK:1

```
df.describe()
```

	avgAnnCount	avgDeathsPerYear	TARGET_deathRate	incidenceRate	medIncome	popEst2015
count	3047.000000	3047.000000	3047.000000	3047.000000	3047.000000	3.047000e+03
mean	606.338544	185.965868	178.664063	448.268586	47063.281917	1.026374e+05
std	1416.356223	504.134286	27.751511	54.560733	12040.090836	3.290592e+05
min	6.000000	3.000000	59.700000	201.300000	22640.000000	8.270000e+02
25%	76.000000	28.000000	161.200000	420.300000	38882.500000	1.168400e+04
50%	171.000000	61.000000	178.100000	453.549422	45207.000000	2.664300e+04
75%	518.000000	149.000000	195.200000	480.850000	52492.000000	6.867100e+04
max	38150.000000	14010.000000	362.800000	1206.900000	125635.000000	1.017029e+07



Exploratory Data Analysis

Fixed the data by removing the Columns with Missing and Irrelevant Values and filling the required ones.

Detecting and handling the Outliers with the Help of Z-Square Method.

Visualisations to support the study and for the Check of the Collinearity.

TASK: 2

Splitting the Data into Training and Testing Sets

```
[ ]  # We specify this so that the train and test data set always have the same rows, respectively
     from sklearn.model_selection import train_test_split
     df_train, df_test = train_test_split(df, train_size = 0.7, test_size = 0.3, random_state = 100)
```

TARGET_deathRate	incidenceRate	medIncome
0.410587	0.528595	0.547126
0.635894	0.559612	0.267096
0.524209	0.542488	0.215817
0.303422	0.586753	0.555591
0.477728	0.602262	0.290254

	Features	VIF
0	const	56.71
4	PercentMarried	7.37
7	PctPrivateCoverage	5.59
10	PctMarriedHouseholds	5.42
8	PctEmpPrivCoverage	4.96
6	PctEmployed16_Over	3.39

```
col = X_train.columns[rfe.support_]
col
```

```
Index(['avgAnnCount', 'avgDeathsPerYear', 'incidenceRate', 'popEst2015',
       'MedianAge', 'AvgHouseholdSize', 'PercentMarried', 'PctBachDeg25_Over',
       'PctEmployed16_Over', 'PctPrivateCoverage', 'PctEmpPrivCoverage',
       'PctPublicCoverage', 'PctPublicCoverageAlone', 'PctOtherRace',
       'PctMarriedHouseholds'],
      dtype='object')
```

Feature Scaling and Selection

Divided the Dataset in 70:30 for Training:Testing dataset.

Scaled the Data in the range of 0 & 1 through MinMax Scaling.

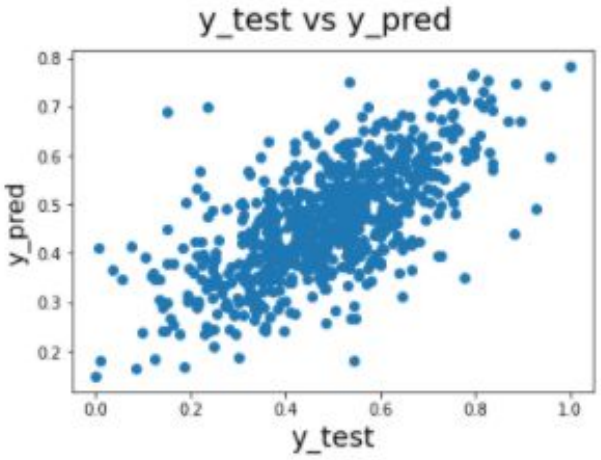
Automated and Manual Feature Selection on the basis of RFE and VIF, keeping the balance between the two for the Training Model.

TASK:3

```
import matplotlib.pyplot as plt

fig = plt.figure()
plt.scatter(y_test,y_pred)
fig.suptitle('y_test vs y_pred', fontsize=20)              # Plot heading
plt.xlabel('y_test', fontsize=18)                          # X-label
plt.ylabel('y_pred', fontsize=16)
```

```
Text(0, 0.5, 'y_pred')
```



Model Assessment and Prediction

Dropping Columns with Higher P-Value or VIF for better and optimised model.

Testing the OLS Regression Model with Testing dataset.

Plotting the Graph between the actual and the predicted data to check how the Model performed.

TASK: 4

OLS Regression Results

Dep. Variable:	TARGET_deathRate	R-squared:	0.468
Model:	OLS	Adj. R-squared:	0.465
Method:	Least Squares	F-statistic:	151.1
Date:	Tue, 18 Aug 2020	Prob (F-statistic):	4.99e-227
Time:	14:16:15	Log-Likelihood:	1276.3
No. Observations:	1726	AIC:	-2531.
Df Residuals:	1715	BIC:	-2471.
Df Model:	10		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	0.6132	0.021	29.193	0.000	0.572	0.654
avgAnnCount	-0.0697	0.022	-3.158	0.002	-0.113	-0.026
incidenceRate	0.3930	0.020	19.510	0.000	0.353	0.432
MedianAge	-0.0987	0.026	-3.848	0.000	-0.149	-0.048
PercentMarried	0.1792	0.048	3.715	0.000	0.085	0.274
PctBachDeg25_Over	-0.2241	0.022	-10.274	0.000	-0.267	-0.181
PctEmployed16_Over	-0.1995	0.031	-6.381	0.000	-0.261	-0.138
PctPrivateCoverage	-0.2732	0.038	-7.220	0.000	-0.347	-0.199
PctEmpPrivCoverage	0.2094	0.036	5.898	0.000	0.140	0.279
PctOtherRace	-0.1366	0.020	-6.974	0.000	-0.175	-0.098
PctMarriedHouseholds	-0.2406	0.044	-5.511	0.000	-0.326	-0.155

Omnibus:	44.868	Durbin-Watson:	2.029
Prob(Omnibus):	0.000	Jarque-Bera (JB):	102.637
Skew:	0.041	Prob(JB):	5.16e-23
Kurtosis:	4.192	Cond. No.	44.0

Final Reports

Having the Final Reports of:

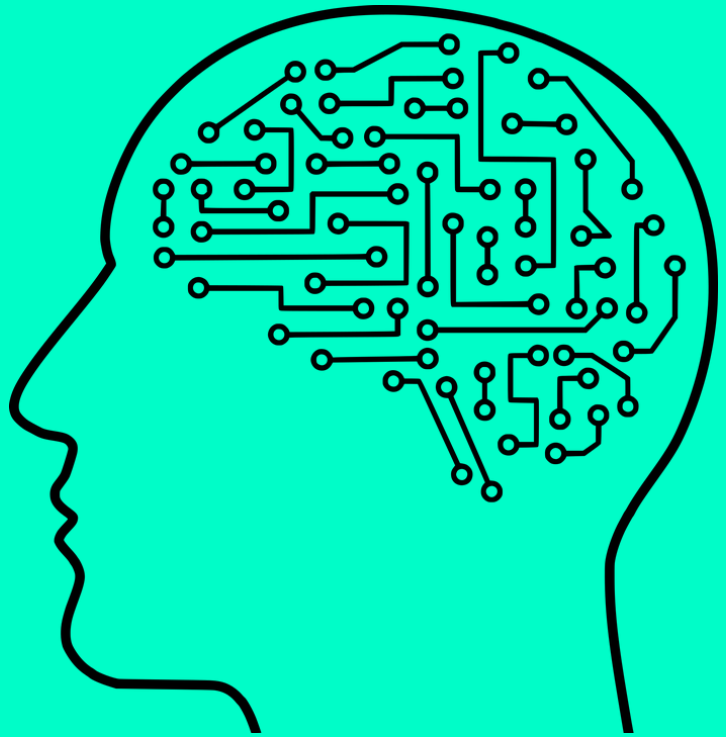
1. The OLS Regression Model
2. The Errors the Model has

```
Mean Absolute Error: 0.09446329318295704
Mean Squared Error: 0.015259141851414757
Root Mean Squared Error: 0.12352789908119849
R2 score:  0.42
```

CHECK ON MODEL

According to me, the Model's:

1. VIFs shows that there is very less collinearity in the model.
2. Parameters are statistically significant.(P>|t|)
3. R^2 value is decent.
4. AIC values are satisfactory showing there are less number of predictor variables.



Thankyou

Shaurya Gulati
18BCS6092
AIML-1
Group-B