CAR PRICE PREDICTION PROJECT

Submitted by:

AKRITI KAKKAR

ACKNOWLEDGMENT

I have referred to datatrained course material, fliprobo use case documentation and python documentation for this project.

INTRODUCTION

- Business Problem Framing
- With the covid 19 impact in the market, we have seen lot of changes in the car market. Now some cars are in demand hence making them costly and some are not in demand hence cheaper. One of our clients works with small traders, who sell used cars. With the change in market due to covid 19 impact, our client is facing problems with their previous car price valuation machine learning models. So, they are looking for new machine learning models from new data. We have to make car price valuation model.
- Conceptual Background of the Domain Problem
- **Name**
- Of all the 5895 used cars records there are 348 unique cars Maruti Suzuki wagon R has appeared maximum number of times that is 294 other cars have appeared 5601 times. Total cumulative price of Rolls Royce Ghost is 85 million, which is the highest price. Total cumulative price of Skoda others is less than 20 million second highest cumulative price is for Tata Nexon EV which is 83 million. Third highest cumulative price for Toyota Fortuner which is 82 million. Cars such as Tata indica EV2 , ,Mini Cooper Mercedes Lexus , Toyota Cynos, Audi Q5 etcetera are priced less than 20 million. There are four cars in the range 60 million to 80 million Mahindra xuv 500, Toyota innova, Maruti Suzuki wagon R, Maruti Suzuki eeco and Maruti Suzuki swift. Price of the most frequently occurring car Maruti Suzuki Wagon R is in the range 20 million to 40 million.
- **Place**
- Across all 5895 records, there are 3152 unique localities the highest priced locality is bandra West which is a locality at Mumbai Maharashtra it has posted its cars sale prices in the range of 40 million to 50 million. The second highest priced locality is Ernakulam which is in Kerela which has posted cumulative price more than 40 million and is a place at kerela.The most famous locality is preet vihar, Delhi which has appeared 39 times and has priced its cars below 20 million. Evidently, bandra West, Mumbai is the most expensive locality for looking out for car purchases and Preet Vihar, Delhi, offers high quantum cars at affordable prices.
- **Date Of Ad**
- Across all 5895 records most ads were posted today, number being 3398 advertisements. There are 233 unique dates of which the highest price was witnessed today which is more than 1.8 billion. Evidently high number of posts is a reason for high prices on the day.

Owner

Across all 5895 records, there are three distinct owners: first owner, second owner and third owner. First owner depicts that the car has passed through only one owner. Second owner depicts that the car has passed through two owners and 3rd owner depicts that car has passed

through three owners second owner appears maximum time, that is, 2595 times. Highest price is kept by first owners which is 1.4 billion cumulatively; second owners have posted prices up to 1.2 billion cumulatively and 3rd owners have posted prices up to 0.5 billion cumulatively. Evidently, if the car has passed through only one owner prices tend to be higher than if the car has passed through two honors and similarly if the car has passed through three owners the prices tend to be the lowest.

Fuel

Across all 5895 records there are five distinct fuel types used, namely, petrol; diesel; CNG and hybrid; electric and LPG. Diesel has appeared the most number of times that is 2000. Price covered by diesel cars is 50.1% of total price. Price covered by LPG cars is 6.8% of total price evidently diesel covers the highest prices and LPG covers the lowest prices. Then the second highest prices are covered by petrol which is 19.7% of total prices. CNG and hybrid cover 11.9% of total prices and electric covers 11.5% of total prices.

- Review of Literature

  Follow the links to the dashboards:

  https://docs.google.com/spreadsheets/d/1ICG7P-lWXkZFW6PpOjMSu5ZsqYHJsgdOP-FHAC7V3IE/edit?usp=sharing

  https://docs.google.com/spreadsheets/d/1O9pG3a0bnJanHIvkMeijokVPdsS9oa9ECDEdrYGur5s/edit?usp=sharing

  https://docs.google.com/spreadsheets/d/1dxkFwBEMFj4V-ue8K0Ry8M_hiNnTpYjlLBfMWfjV9Lk/edit?usp=sharing

  https://docs.google.com/spreadsheets/d/1En1uQfUc-krCrQrEpF9RqyRwcIfMW6gTGkDb9GY0ye4/edit?usp=sharing

  https://docs.google.com/spreadsheets/d/1U6Z-WzSb_l_zOj1wYy4OiNmuDcMTJ0COgD8NthjtWcM/edit?usp=sharing

  https://docs.google.com/spreadsheets/d/1xNxEGWUglN7ABJl0plvmK3AH_ee53idqxHIJRngWgyw/edit?usp=sharing

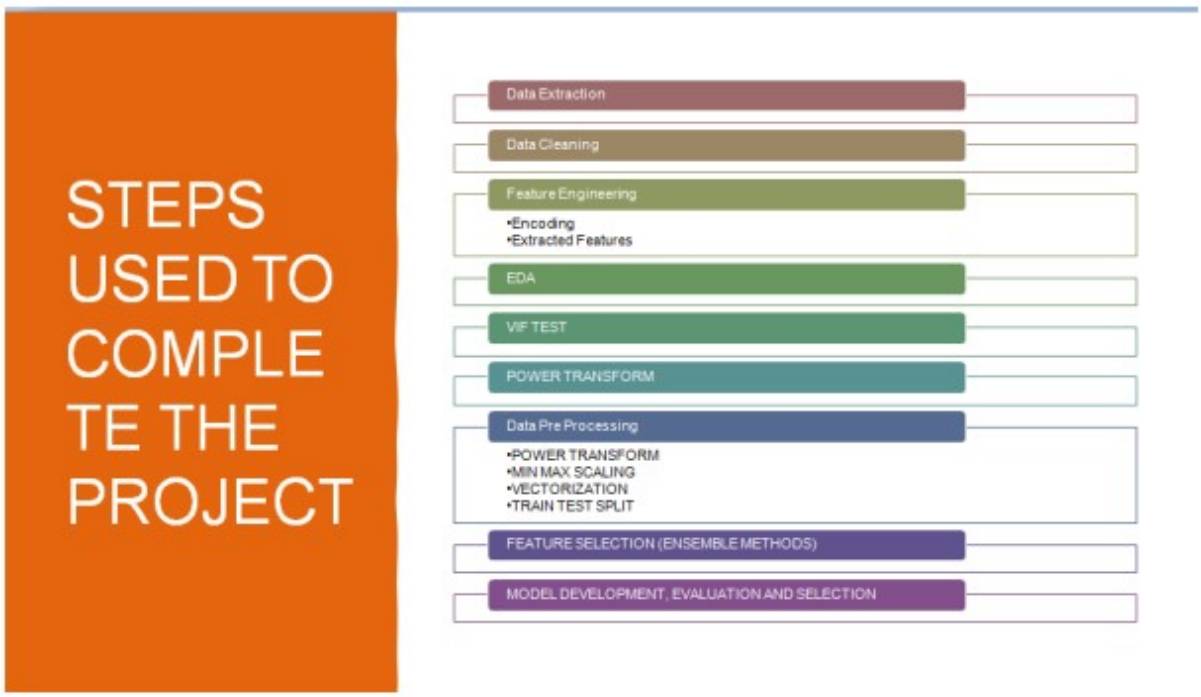
  https://docs.google.com/spreadsheets/d/1jubSGc7DWYaIOcUcfDdkVEozdyvTyIf6WgU9t-Zy4_0/edit?usp=sharing

- Motivation for the Problem Undertaken

  OBJECTIVE: To help the companies increase their overall revenue, profits, improving their strategies and focusing on changing trends in cars sales and purchases caused due to covid19.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem



1. Data Extraction: Using read_ csv function of pandas library to read the data in tabulated format and analyze it.

2. Data Cleaning for missing values detection and its handiling.

3. Feature Engineering for encoding object format data and deriving more features.

4. EDA For data visualization and biasness detection:

• HEAD VIEW OF DATA

• TAIL VIEW OF DATA

• SAMPLE VIEW OF DATA

• GROUPBY EXPLORATION

• DESCRIPTIVE STATISTICS

• SCATTER PLOTS

• CORRELATION ANALYSIS

• BOX PLOTS EXPLORATION

• DESCRIPTIVE STATISTICS

• DISTRIBUTION PLOTS

5. VIF Test for multicollinearity reduction.

6. Power Transformation for standard scaling and outliers transformation.

7. Data PreProcessing for data transformation, scaling and vectorization.

8. Feature Selection (Ensemble Methods): ANOVA Test, p value, ftest, constant threshold filter to classify features based on relevance and biasness and select the most relevant features.

9. Model Development, Evaluation And Selection (Ensemble Methods and Grid Search CV) to do best hyper parameter tuning and develop low bias and low variance with right fit and minimal difference between test metrics and train metrics.

- Data Sources and their formats

Data Collection Source: WWW.OLX.IN

Data Collection Technique: Web Scraping

Data Collection Technical Tools: Python+selenium

Data Collection Code Technical Tool: Jupyer Notebook NBFormat

Data Collection Source Code File: data_script.ipynb

Data Collection Output: cars_data.zip

Data Format Used In Present Code: cars_data.csv

- Data Preprocessing Done
- • Data Pre Processing
- • POWER TRANSFORM
- • MIN MAX SCALING
- • VECTORIZATION
- • TRAIN TEST SPLIT
- The data was further used for feature selection.
- Assumption made:
- • Acceptable Skewness Range Is +/-0.65
- • Acceptable VIF Score Is Than 6
- • Acceptable P Value Is Less Than 0.05
- • Variance Threshold Is 0.01

- Data Inputs- Logic- Output Relationships

- **Data Inputs are:**

```
Index(['Fuel_encoded', 'Date Of Ad_encoded', 'km_driven_pct_change',
       'Place_encoded_pct_change'],
      dtype='object')
```

- **Data Type: Float64**

Relation with label:

Weak Negative to Strong Negative Relationship Is Found With Following Features:

Fuel_encoded -0.072542

Place_encoded_pct_change -0.007783

Weak Positive To Strong Positive Relationship Is Found With Following Features:

km_driven_pct_change 0.002610

Date Of Ad_encoded 0.034009

- State the set of assumptions (if any) related to the problem under consideration

  • Acceptable Skewness Range Is +/-0.65

  • Acceptable VIF Score Is Than 6

  • Acceptable P Value Is Less Than 0.05

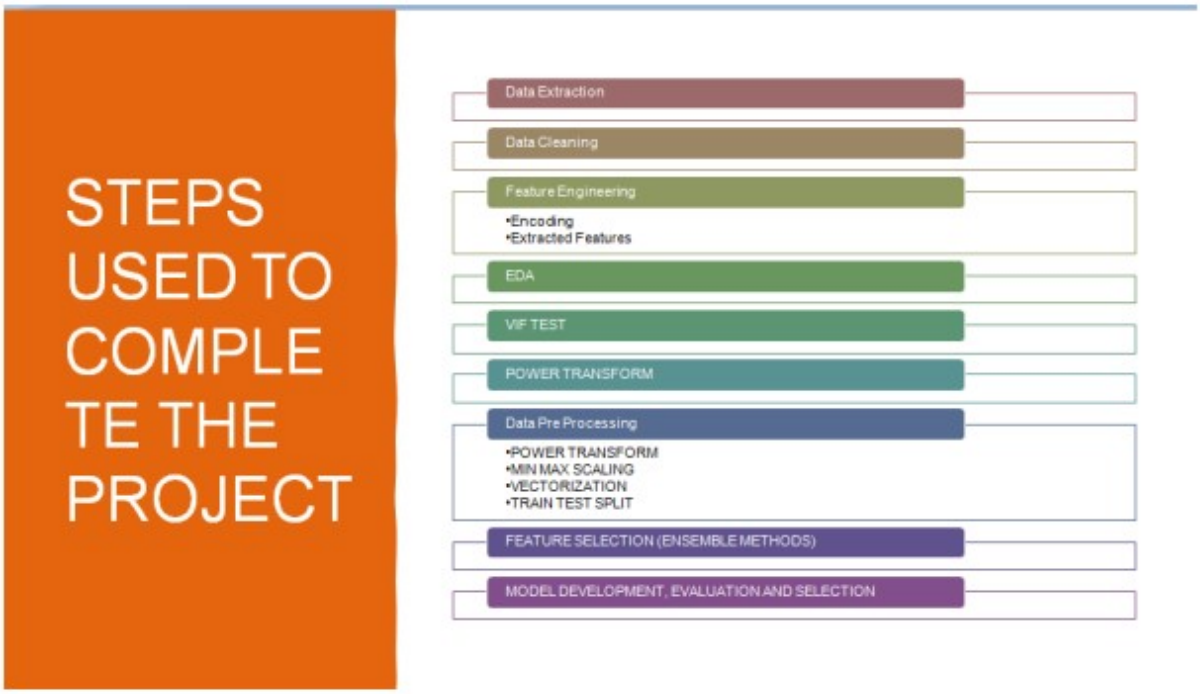
  • Variance Threshold Is 0.01

- Hardware and Software Requirements and Tools Used

Installation Of Anaconda Library.
Required Installations: • Pandas (Within environment) • Numpy (Within environment) • Seaborn (Within environment) • Matplotlib (Within environment) • Cufflinks • Plotly Express • Sklearn

Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)



1. Data Extraction: Using read_ csv function of pandas library to read the data in tabulated format and analyze it.

2. Data Cleaning for missing values detection and its handiling.

3. Feature Engineering for encoding object format data and deriving more features.

4. EDA For data visualization and biasness detection:

• HEAD VIEW OF DATA

• TAIL VIEW OF DATA

• SAMPLE VIEW OF DATA

• GROUPBY EXPLORATION

• DESCRIPTIVE STATISTICS

• SCATTER PLOTS

• CORRELATION ANALYSIS

• BOX PLOTS EXPLORATION

• DESCRIPTIVE STATISTICS

• DISTRIBUTION PLOTS

5. VIF Test for multicollinearity reduction.

6. Power Transformation for standard scaling and outliers transformation.

7. Data PreProcessing for data transformation, scaling and vectorization.

8. Feature Selection (Ensemble Methods): ANOVA Test, p value, ftest, constant threshold filter to classify features based on relevance and biasness and select the most relevant features.

9. Model Development, Evaluation And Selection (Ensemble Methods and Grid Search CV) to do best hyper parameter tuning and develop low bias and low variance with right fit and minimal difference between test metrics and train metrics:

• Total Models = 7

• Selection Reasoning Of Models 1 to 7

• The goal of ensemble methods is to combine the predictions of several base estimators built with a given learning algorithm in order to improve generalizability / robustness over a single estimator. • Two families of ensemble methods are usually distinguished: • In averaging methods, the driving principle is to build several estimators independently and then to average their predictions. On average, the combined estimator is usually better than any of the single base estimator because its variance is reduced. • Examples: Bagging methods, Forests of randomized trees, etcetera • By contrast, in boosting methods, base estimators are built sequentially and one tries to reduce the bias of the combined estimator. The motivation is to combine several weak models to produce a powerful ensemble. • Examples: AdaBoost, Gradient Tree Boosting,etcetera • The use case assigned revolves around uneven label points hence there is high probality of not achieving a good fit. Hence, I have tried different ensembele techniques that can lower variance and bias and help achieve good scores. (Just as a reminder, I have already applied variance threshold of 0.01 to ensure that risk of models is low). • The theories in the above two cells explain why I have chosen Model 1, Model 2, Model 3, Model 4, Model 5, Model 6 and Model7

- # Testing of Identified Approaches (Algorithms)

Total Models = 7

• Model 1: Random Forest Regressor With Grid Search CV Hyper Parameter Tuning

• Model 2: Random Forest Regressor With Default Hyper Parameter Tuning

• Model 3: Ada Boost Regressor And Random Forest Regressor With Grid Search CV Hyper Parameter Tuning and Ada Boost Boosting

• Model 4: Extra Trees Regressor With Grid Search CV Hyper Parameter Tuning

• Model 5: Linear Regression With Intuitional Hyper Parameter Tuning

• Model 6: Huber Regressor With Default Hyper Parameter Tuning

• Model 7: Ada Boost Regressor With Huber Regressor As Base Estimator

- Run and Evaluate selected models
- Follow the link to the dashboard:

https://docs.google.com/spreadsheets/d/1XcZbjH6G9gHMIyHY5mwnpnsUL27qi6XZJ-a1AbjkIGc/edit?usp=sharing

- Key Metrics for success in solving problem under consideration

- 1. Power Transform: To remove outliers from extremely spread out data.
- 2. VIF Scores: To reduce multicollinearity from a highly biased dataset.
- 3. Ensemble Methods: To remove over fitting in a complex dataset and finding maximum explanatory power .

- Visualizations
- Follow the links to the dashboards:
- https://docs.google.com/spreadsheets/d/1ICG7P-lWXkZFW6PpOjMSu5ZsqYHJsgdOP-FHAC7V3IE/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1O9pG3a0bnJanHIvkMeijokVPdsS9oa9ECDEdrYGur5s/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1dxkFwBEMFj4V-ue8K0Ry8M_hiNnTpYjlLBfMWfjV9Lk/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1En1uQfUc-krCrQrEpF9RqyRwcIfMW6gTGkDb9GY0ye4/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1U6Z-WzSb_l_zOj1wYy4OiNmuDcMTJ0COgD8NthjtWcM/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1xNxEGWUglN7ABJl0plvmK3AH_ee53idqxHIJRngWgyw/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1jubSGc7DWYaIOcUcfDdkVEozdyvTyIf6WgU9t-Zy4_0/edit?usp=sharing

- Interpretation of the Results

Based On EDA done above in two parts, I will do ftest and pvalue test on these seemingly weak indicators based primarily on skewness, kurtosis and multicollinearity:

Year Of Purchase Pct Change (Multicollinearity & Kurtosis)

Most of the biased features have strong explanatory power in terms of correlation with feature, skewness or kurtosis and hence can be filtered in ensemble method of feature selection by p value and constant variance threshold.

ANOVA Test On Selected Features

Ftest score should be greater than 1 and p value should be less than 0.05, to determine to keep these features for further analysis

Feature 1: Year Of Purchase Pct Change

In [50]:
```
from scipy.stats import f_oneway
f,p=f_oneway(data['Year_Of_Purchase_pct_change'], data['Price'])
f,p
```

Out[50]: (1561.7593192183163, 7.24e-321)

Error Removal And Data Handling

Based on above analysis:

1. There are many outliers in the data.

2. Strong multicolliearity features are important for prediction because there f test and p value are acceptable. This means that the amount of multicollineaity is insignificant and removing the feature will impact the model much.

3. Extereme leptokurtic and right skewed features are also relatively significant based on f test and p test.

Hence, as a solution, feature scaling will do a better job in explaining the dependent variable than removing whole columns.

Outliers Transformation With Power Transform

In [57]:
```
from sklearn.preprocessing import power_transform
x_array=power_transform(x, method='yeo-johnson')
x_frame=pd.DataFrame(x_array, columns=x.columns)
x_frame
```

```
C:\Users\Lenovo\anaconda3\newinstall\lib\site-packages\numpy\core\_methods.py:232: RuntimeWarning: overflow encountered in multiply
  x = um.multiply(x, x, out=x)
C:\Users\Lenovo\anaconda3\newinstall\lib\site-packages\numpy\core\_methods.py:243: RuntimeWarning: overflow encountered in reduce
  ret = umr_sum(x, axis, dtype, out, keepdims=keepdims, where=where)
```

Out[57]:

	Year_Of_Purchase	km_driven	Name_encoded	Place_encoded	Date Of Ad_encoded	Owner_encoded	Fuel_encoded	Year_Of_Purchase_pct_change	km_driven_pct_change	Name_encod
0	0.770168	-0.498082	-1.205853	-1.742197	0.089679	-1.167946	1.322265	0.603028	-1.356411	
1	1.676154	-1.349400	-0.749061	-0.586395	0.377852	-1.167946	1.322265	0.603028	-1.356411	
2	0.989009	-0.623526	0.081552	-0.497903	-0.725514	-1.167946	1.322265	-0.449675	1.121047	
3	1.212878	-1.349400	-1.247820	0.979841	0.691004	-1.167946	1.322265	0.152365	-1.236876	
4	0.770168	-0.404772	0.293731	0.803252	0.691004	-1.167946	1.322265	-0.299281	1.294787	
...	...	...	...	...	...	...	...	...	...	
5890	-0.629920	0.249650	1.153523	-0.475783	-1.840721	1.483167	1.322265	0.304142	-0.254288	
5891	-1.978864	-0.882035	0.021912	-1.204169	-1.840721	1.483167	1.322265	-1.212161	-1.314423	
5892	-0.443311	0.281119	-1.688904	0.248167	-1.840721	1.483167	1.322265	1.360903	1.206393	
5893	-0.443311	0.262294	0.069649	-0.414679	-1.840721	1.483167	1.322265	0.001926	0.025253	
5894	-0.629920	0.067394	-0.736070	-1.117188	-1.840721	1.483167	1.322265	-0.149197	-0.168337	

5895 rows × 14 columns

MinMax Scaler Transformation And Variance Inflation Factor

In [60]:
```
import sklearn
from sklearn.preprocessing import MinMaxScaler
from statsmodels.stats.outliers_influence import variance_inflation_factor
import warnings
warnings.filterwarnings('ignore')
scaler=MinMaxScaler([0,1])
X_scaled=scaler.fit_transform(x_frame)
X_scaled_frame=pd.DataFrame(X_scaled, columns=x_frame.columns)
X_scaled_frame
```

Out[60]:

	Year_Of_Purchase	km_driven	Name_encoded	Place_encoded	Date Of Ad_encoded	Owner_encoded	Fuel_encoded	Year_Of_Purchase_pct_change	km_driven_pct_change	Name_encod
0	0.884225	0.235560	0.280400	0.124003	0.757678	0.0	1.0	0.524835	0.174021	
1	0.975782	0.134291	0.383258	0.432077	0.868334	0.0	1.0	0.524835	0.174021	
2	0.906341	0.220638	0.570290	0.455664	0.444651	0.0	1.0	0.496495	0.683501	
3	0.928965	0.134291	0.270950	0.849550	0.988582	0.0	1.0	0.512702	0.198603	
4	0.884225	0.246660	0.618068	0.802481	0.988582	0.0	1.0	0.500544	0.719229	
...	...	...	...	...	...	...	...	...	...	
5890	0.742735	0.324507	0.811670	0.461560	0.016421	1.0	1.0	0.516788	0.400668	
5891	0.606413	0.189887	0.556861	0.267412	0.016421	1.0	1.0	0.475968	0.182656	
5892	0.761593	0.328251	0.171630	0.654525	0.016421	1.0	1.0	0.545238	0.701052	
5893	0.761593	0.326011	0.567610	0.477847	0.016421	1.0	1.0	0.508652	0.458155	
5894	0.742735	0.302827	0.386183	0.290597	0.016421	1.0	1.0	0.504584	0.418344	

5895 rows × 14 columns

In [67]:

```
vif=pd.DataFrame()
vif['vif']=[variance_inflation_factor(X_scaled, w) for w in range(X_scaled.shape[1])]
vif['Features']=x.columns
vif.sort_values(by='vif')
```

Out[67]:

	vif	Features
1	2.778909	Fuel_encoded
0	3.414603	Date Of Ad_encoded
3	4.013556	Place_encoded_pct_change
2	4.499351	km_driven_pct_change

Now, all the features, seem to have lower multi collinearity, is below 6. Now, I will apply p values feature selection, to deciede which features to include.

CONCLUSION

- Key Findings and Conclusions of the Study
- Follow the links to the dashboards:
- https://docs.google.com/spreadsheets/d/1ICG7P-lWXkZFW6PpOjMSu5ZsqYHJsgdOP-FHAC7V3IE/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1O9pG3a0bnJanHIvkMeijokVPdsS9oa9ECDEdrYGur5s/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1dxkFwBEMFj4V-ue8K0Ry8M_hiNnTpYjlLBfMWfjV9Lk/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1En1uQfUc-krCrQrEpF9RqyRwcIfMW6gTGkDb9GY0ye4/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1U6Z-WzSb_l_zOj1wYy4OiNmuDcMTJ0COgD8NthjtWcM/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1xNxEGWUglN7ABJl0plvmK3AH_ee53idqxHIJRngWgyw/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1jubSGc7DWYaIOcUcfDdkVEozdyvTyIf6WgU9t-Zy4_0/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1XcZbjH6G9gHMIyHY5mwnpnsUL27qi6XZJ-a1AbjkIGc/edit?usp=sharing
-
- Learning Outcomes of the Study in respect of Data Science

- Follow the links to the dashboards:
- https://docs.google.com/spreadsheets/d/1ICG7P-lWXkZFW6PpOjMSu5ZsqYHJsgdOP-FHAC7V3IE/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1O9pG3a0bnJanHIvkMeijokVPdsS9oa9ECDEdrYGur5s/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1dxkFwBEMFj4V-ue8K0Ry8M_hiNnTpYjlLBfMWfjV9Lk/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1En1uQfUc-krCrQrEpF9RqyRwcIfMW6gTGkDb9GY0ye4/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1U6Z-WzSb_l_zOj1wYy4OiNmuDcMTJ0COgD8NthjtWcM/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1xNxEGWUglN7ABJl0plvmK3AH_ee53idqxHIJRngWgyw/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1jubSGc7DWYaIOcUcfDdkVEozdyvTyIf6WgU9t-Zy4_0/edit?usp=sharing
- https://docs.google.com/spreadsheets/d/1XcZbjH6G9gHMIyHY5mwnpnsUL27qi6XZJ-a1AbjkIGc/edit?usp=sharing
-

- ## Limitations of this work and Scope for Future Work

  Further optimatization can be obtained by applying deep learning solutions. Since, it requires very high RAM capacity, it could not be displayed in jupyter notebook... I would like to update Google Colab Notebook for future projects, if acceptable... That can help me to submit a completely optimized model.