

# Data Science Take Home Exercise

## Content

Data Science Take Home Exercise .....	1
1. Exploratory Questions.....	2
1.1 What proportion of customers with a credit score below 0.2 made a claim in the last year? .....	2
1.2 Produce a maximum of 3 plots to help the client understand their data better. Explain what the plots show/any insight they provide the client about their data.....	4
2. Modelling .....	6
2.1 Perform any necessary data pre-processing in order to build a model. Briefly discuss any assumptions being made about the data and explain any pre-processing decisions. ....	6
2.2 Select and fit one proof of concept model to predict the outcome column from the customer data. State the reasons for the choice of model used.....	6
Problem identification 1: Supervised or unsupervised.....	6
Problem Identification 2: Regression or Classification (only if PI-1 is Supervised).....	6
Method 1: Using the Machine Learning algorithm which supports Classification: .....	6
Method 2: Using Machine Learning Algorithm with Feature Selection- Recursive Feature Elimination .....	7
Method 2: Using Machine Learning Algorithm with Feature Selection- Select K best.....	8
2.3 Validate your model using appropriate metrics, explain the choice of metric and state how you would expect it to perform on the rest of the client's data. ....	8
2.4 In light of the above questions the client is keen to know if they have collected the right data for their business needs. Make a maximum of 5 recommendations to the client for future data collection.....	10
Libraries.....	10

## 1. Exploratory Questions

1.1 What proportion of customers with a credit score below 0.2 made a claim in the last year?

1) To give solution to the above question, I have pre-processed the data to fill the null values.

2) In the dataset, three columns have null values.

```
id          0
age         0
gender      0
race        0
driving_experience  0
education   0
income      0
credit_score 982
vehicle_ownership  0
vehicle_year  0
married     0
children    0
postal_code  0
annual_mileage 957
vehicle_type  0
speeding_violations  0
DUIs        0
past_accidents  0
outcome     0
telematic_score  0
monthly_mileage 957
dtype: int64
```

*Fig1: Null value check for the dataset*

3) To fill the null values, I have used the "mean".

4) Now the dataset is free from null values.

5) Calculated the proportion of the customer who made claim below 0.2 credit score in the year. In the dataset there is no separate column of the year. Only vehicle year with "after 2015", "before 2015". I have considered the last year as "after 2015".

Fig2: Proportion of the

```
In [99]: #Filtering the outcome and vehicle _year column based on the credit score below 0.2
scoreBelow=dataset[dataset["credit_scores"]<=0.2][["outcome", "vehicle_year"]]

In [83]: #Len of the score below 0.2
len(scoreBelow)

Out[83]: 98

In [84]: #Filtering the dataset for the last year(As per the dataset only vehicle year is present.So calculated using the same.)
claim=scoreBelow[scoreBelow["vehicle_year"]=="after 2015"]["outcome"]

In [85]: print("The total count of the credit score:",len(dataset))
The total count of the credit score: 10000

In [86]: print("The Total count of below 0.2:",len(claim))
The Total count of below 0.2: 11

In [87]: claimedTrue=claim.value_counts()[1]

In [88]: print("The claim for below 0.2:",claimedTrue)
The claim for below 0.2: 1

In [89]: percentage=(claimedTrue/len(dataset))*100

In [90]: print(percentage,"% of the customer claimed with credit score below 0.2 in the last year ")
0.01 % of the customer claimed with credit score below 0.2 in the last year

In [91]: print("{}: {} proportion of customers with a credit score below 0.2 made a claim in the last year".format(claimedTrue,len(dataset)))
1:10000 proportion of customers with a credit score below 0.2 made a claim in the last year
```

*customer who made claim below 0.2 in the last year.*

Percentage: 0.01% claim made for the credit score below 0.2 in the last year.

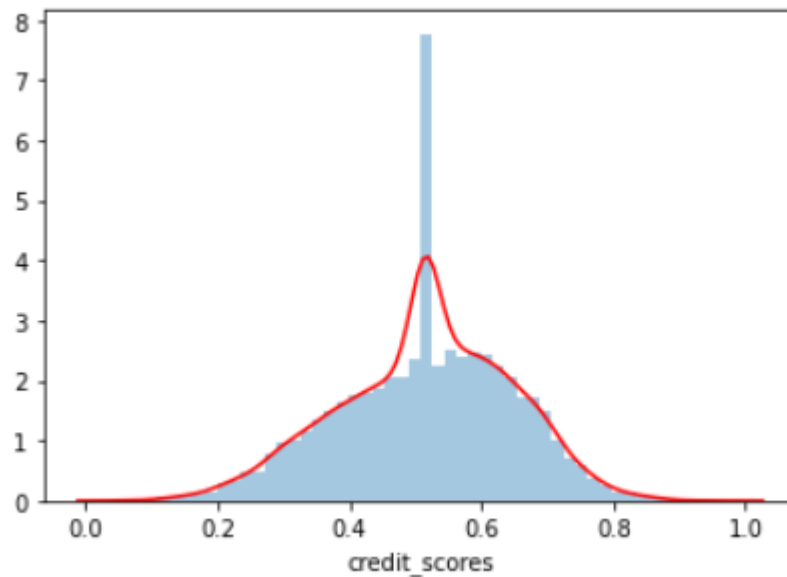
1:10000 proportion claimed for the credit score below 0.2 in the last year.

Note: I have crossed the filled with null values did not change the proportion.

1.2 Produce a maximum of 3 plots to help the client understand their data better.  
Explain what the plots show/any insight they provide the client about their data.

1) Density plot: As per requirement, density plot for credit\_score is reasonable, can see the various range of the values.

Fig3:

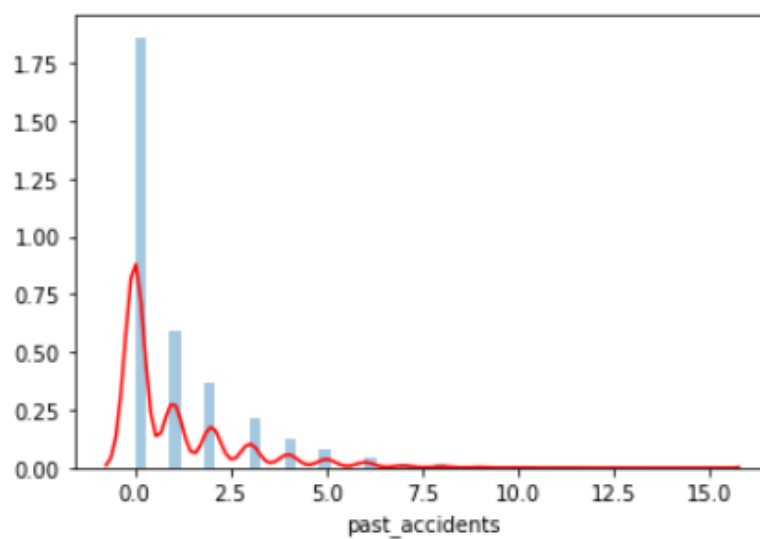


*Fig3: Density plot for credit score*

The graph explains the majority range is between 0.4 - 0.6 only. Majority of the customer credit score lies between 0.4-0.6.

2) Density plot: "past\_accidents" talks about customer driving pattern.

```
<AxesSubplot:xlabel='past_accidents'>
```



*Fig4: Density plot for the past\_accidents*

The above graph tells that majority of the customer don't have past accident history (0).

### 3)Box Plot: Credit\_score

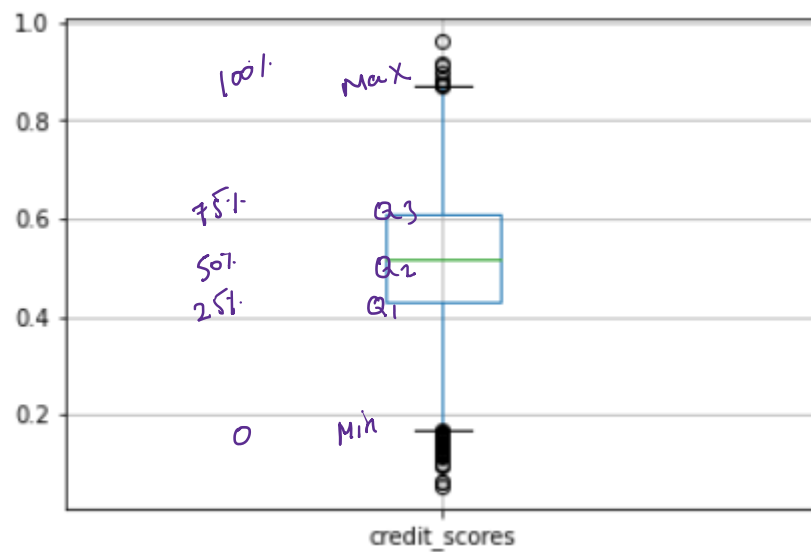


Fig5: Box plot: Credit score.

Box plot tells that percentile of the column, here have plotted for the single variable to see clear value.

From down 1st horizontal line min -->min of the between 0.2

2nd horizontal line q1-->25% of the data (person values) lies between 0.4.2

3rd horizontal line q2-->50% of the data (person values) lies between 0.5

4th horizontal line q3-->75% of the data (person values) lies between 0.6

5th horizontal line q4-->100% of the data (person values) lies between 0.8

Kindly refer the "1.1 and 1.2 Exploratory Analysis.ipynb"

## 2. Modelling

2.1 Perform any necessary data pre-processing in order to build a model. Briefly discuss any assumptions being made about the data and explain any pre-processing decisions.

1) Filled the null values with "mean"

Why? – with null values cannot able to perform any operation. So, filling the null values is more important.

2) Converting the string variable to numerical variable. String variables are nominal data so using "One hot encoder" converted in to the numerical variable.

Before the conversion: 21 columns

After the conversion: 26 columns

3) Now, the dataset is full of numerical columns so that we can perform predictive modelling.

Kindly refer "2.1.Data pre-processing.ipynb"

2.2 Select and fit one proof of concept model to predict the outcome column from the customer data. State the reasons for the choice of model used.

1) To predict "outcome" column, predictive modelling is build using Machine learning algorithms.

Problem identification 1: Supervised or unsupervised

The given problem statement is to predict the "outcome" column. When the requirement is clear. The problem statement comes under the Supervised Learning.

Problem Identification 2: Regression or Classification (only if PI-1 is Supervised)

The problem statement is supervised learning, so have to find this is Classification or Regression.

The predictive variable is "True" or "False". This falls under the Classification problem statement.

The problem statement is Supervised Learning and Classification.

Method 1: Using the Machine Learning algorithm which supports Classification:

Training set: 7500

Test set: 2500

Unique count of Outcome column.

```
False    6867
True      3133
```

The above statement tells that the dataset is imbalanced. For imbalanced dataset, Evaluation metrics is F1-Measure.

F1-measure gives separate prediction confidence of each class.

Accuracy Calculated for Test set.

Compared with KNN, Navie Bayes, support vector machine with linear kernel, support vector machine with non-linear kernel, Decision Tree, Random Forest.

Support vector machine with non linear gives the best result.

Kindly refer Jupyter Notebook for the mentioned algorithms.

1)Support Vector Machine with kernel non linear(RBF)

```
: obj.SVM_nonlinear(X_train, X_test, y_train, y_test)
[[1528  174]
 [ 231  567]]
The Accuracy is: 0.838
precision    recall  f1-score   support

   False     0.87     0.90     0.88     1702
   True      0.77     0.71     0.74      798

 accuracy          0.84     2500
 macro avg         0.82     2500
weighted avg         0.84     2500
```

Fig6: Evaluation Metrics

The accuracy of the model is 0.838

F1-score:

False: 0.88

True: 0.74

Kindly refer "2.2 - Method 1 using Machine Learning Algorithm.ipynb"

Method 2: Using Machine Learning Algorithm with Feature Selection- Recursive Feature Elimination

Recursive Feature Elimination, when feature n=5

Logistic regression with Logistic gives the best accuracy even for the 5 features.

result - DataFrame


Index	Logistic	SVMI	SVMnl	KNN	Navie	Decision	Random
Logistic	0.8248	0.8108	0.8248	0.7884	0.7332	0.8248	0.8248
Random	0.7496	0.7424	0.7548	0.7244	0.6768	0.6624	0.7232
DecisionTree	0.7716	0.7572	0.774	0.7452	0.75	0.6868	0.742

kindly refer: "2.2 - Method 2- using feature with Machine Learning Algorithm-RFE.ipynb"

Method 2: Using Machine Learning Algorithm with Feature Selection- Select K best

Select K best n=15,

Support vector machine with non-linear, estimated with chi square gives the best accuracy.

 result - DataFrame

Index	Logistic	SVMl	SVMnl	KNN	Navie	Decision	Random
ChiSquare	0.804	0.7908	0.8064	0.7808	0.7304	0.778	0.7924

Note: I have crossed checked with the various feature number.

Kindly refer:" 2.2 - Method 2- using feature with Machine Learning Algorithm-select Kbest.ipynb"

2.3 Validate your model using appropriate metrics, explain the choice of metric and state how you would expect it to perform on the rest of the client's data.

Based on the Three set of processes,

- 1)Using Machine Learning Algorithm
- 2)Using Feature selection with Machine Learning Algorithm-Recursive Feature Elimination
- 3) Using Feature selection with Machine Learning Algorithm-Select K best

Method 2: Using Machine Learning Algorithm with Feature Selection- Recursive Feature Elimination gives the best result for the problem statement even for 5 features.

: feature\_name

:

	vehicle_ownership	driving_experience_10-19y	driving_experience_20-29y	driving_experience_30y+	vehicle_year_before 2015
0	True	0	0	0	0
1	False	0	0	0	1
2	True	0	0	0	1
3	True	0	0	0	1
4	True	1	0	0	1
...	...	...	...	...	...
9995	True	1	0	0	1
9996	True	1	0	0	0
9997	True	0	0	0	1
9998	False	1	0	0	1
9999	True	0	0	0	1

10000 rows × 5 columns

Recursive Feature Elimination work with estimator. The model used Logistic Estimator. It selected 5 Features based on the user input. Selected 5 features applied to logistic classification algorithm.



The accuracy:0.824

Note: Method 1, give 0.83 but features are 26, but method 2 gives 0.82 for the features 5

```
5]: print(report)
```

	precision	recall	f1-score	support
False	0.90	0.83	0.87	1702
True	0.69	0.81	0.75	798
accuracy			0.82	2500
macro avg	0.80	0.82	0.81	2500
weighted avg	0.84	0.82	0.83	2500

*Fig: Report for the best model*

Evaluation metrics:

Dataset: Imbalanced dataset

Metrics: F1-Score (Because of imbalanced dataset)

Additional Metrics: Accuracy(Overall performance)

Note: If we convert imbalanced dataset into balanced dataset, reality of the data will be vanished. Because dataset has 10000 entry, but "outcome" column has "False: 6000+", True:"3000+".

Based on this if we convert 3000+ for both TRUE and FALSE , half the data for False will be vanished.

To protect the reality of the dataset, performed with imbalanced dataset with F1 metric score.

```
In [29]: vehicle_ownership=int(input("Enter the vehicle ownship 0 or 1:"))
driving_experience_19=int(input("Enter the experience 0-19 0 or 1:"))
driving_experience_20=int(input("Enter the driving experience 20-20y 0 or 1:"))
driving_experience_30=int(input("Enter the driving experience 30y 0 or 1:"))
vehicle_year_before=int(input("Enter the vehicle year 0 or 1:"))
```

```
Enter the vehicle ownship 0 or 1:1
Enter the experience 0-19 0 or 1:0
Enter the driving experience 20-20y 0 or 1:0
Enter the driving experience 30y 0 or 1:1
Enter the vehicle year 0 or 1:1
```

```
In [ ]:
```

```
In [30]: classifier.predict([[vehicle_ownership,driving_experience_19,driving_experience_20,driving_experience_30,vehicle_year_before]])
```

```
Out[30]: array([False])
```

*Fig: Prediction based on the model*

Kindly refer:" 2.3 Final model.ipynb"

2.4 In light of the above questions the client is keen to know if they have collected the right data for their business needs. Make a maximum of 5 recommendations to the client for future data collection.

Features to be added in the data collection.

- 1)claimed year
- 2)Age-single value (int), but in existing range (object)
- 3)Driving experience-single value(int), but in existing range(object)
- 4)accident venue
- 5) Traffic rules violation

#### Libraries

numpy==1.17.4

pandas==0.25.4

scikitlearn==0.21.3

seaborn==0.9.0

python==3.7.6