Classification Problem

Title: To predict value from the dataset that is categorical in nature.

Binary Classification:

Binary Classification refers to predicting one of the two classes/columns.
To predict the ORIGINALSALPRICE of the car on from the dataset.

Import Libraries

```
# Import numpy as np
import numpy as np
# Import pandas as pd
import pandas as pd
# Import matplotlib.pyplot as plt
import matplotlib.pyplot as plt
```

Collecting Data

Reading in the CarsEU.csv file into a pandas dataframe.

```
df = pd.read_csv("C:/Users/Satwika/Desktop/Desktop/Cars451.csv")
```

```
df.head()
```

	id	Manufacturer	Model	Type	Min.Price	Max.Price	MPG.City	MPG.Highway	AirBags	Passengers	Length	Wheelbase	Width	Turn.circle	Rear.seat.room	Luggage.room	Weight	Origin	Make
0	1	Acura	Integra	Small	12.9	15.8	25	31	None	5	177	102	68	37	26.5	?	2705	non-USA	Acura Integra
1	2	Acura	Legend	Midsize	29.2	33.9	18.7	26	Driver & Passenger	5	195	115	71	38	30.0	15.0	3560	non-USA	Acura Legend
2	3	Audi	90	Compact	25.9	29.1	20.3	26	Driver only	5	180	102	67	37	28.0	14.0	3375	non-USA	Audi 90
3	4	Audi	100	Midsize	30.8	37.7	18.3	25	Both	6	193	106	70	37	31.0	17.0	3405	non-USA	Audi 100
4	5	BMW	535i	Midsize	23.7	30.0	18.2	25	Driver only	4	186	109	69	35	27.0	13.0	3640	non-USA	BMW 535i

5 rows × 25 columns

Missing Data Treatment

Checking the data by removing NaN values and unnecessary columns from the dataset.

```
df.isna().sum(axis = 0)
```

```
id                  0
Manufacturer        0
Model               0
Type                0
Min.Price           0
Max.Price           0
MPG.City            0
MPG.Highway         0
AirBags             0
Passengers          0
Length              0
Wheelbase           0
Width               0
Turn.circle         0
Rear.seat.room      0
Luggage.room        0
Weight              0
Origin              0
Make                0
dtype: int64
```

```
df.isna().sum(axis = 1).sort_values
```

```
id  Manufacturer  Model  Type  Min.Price  Price  Max.Price  MPG.City  MPG.Highway  AirBags  Passengers  Length  Wheelbase  Width  Turn.circle  Rear.seat.room  Luggage.room  Weight  Origin  Make
0   False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
1   False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
2   False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
3   False         False  False False  False  False      False     False        True     False       False   False      False  False  False        False           False         False   False   False
4   False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
...
90  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
91  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         True    False   False
92  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
93  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
94  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
Length: 95, dtype: bool
```

90 rows × 25 columns

```
df.isna().sum(axis = 0).sort_values(ascending = False)
```

```
id                  0
Manufacturer        0
Model               0
Type                0
Min.Price           0
Max.Price           0
MPG.City            0
MPG.Highway         0
AirBags             0
Passengers          0
Length              0
Wheelbase           0
Width               0
Turn.circle         0
Rear.seat.room      0
Luggage.room        0
Weight              0
Origin              0
Make                0
dtype: int64
```

```
df.isna().sum(axis = 1).sort_values(ascending = False)
```

```
id  Manufacturer  Model  Type  Min.Price  Price  Max.Price  MPG.City  MPG.Highway  AirBags  Passengers  Length  Wheelbase  Width  Turn.circle  Rear.seat.room  Luggage.room  Weight  Origin  Make
0   False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
1   False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
2   False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
3   False         False  False False  False  False      False     False        True     False       False   False      False  False  False        False           False         False   False   False
4   False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
...
90  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
91  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         True    False   False
92  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
93  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
94  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
Length: 95, dtype: bool
```

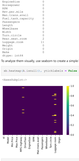
90 rows × 25 columns

```
df.isna().sum(axis = 1).sort_values(ascending = False)
```

```
id                  0
Manufacturer        0
Model               0
Type                0
Min.Price           0
Max.Price           0
MPG.City            0
MPG.Highway         0
AirBags             0
Passengers          0
Length              0
Wheelbase           0
Width               0
Turn.circle         0
Rear.seat.room      0
Luggage.room        0
Weight              0
Origin              0
Make                0
dtype: int64
```

To analyze them visually, use seaborn to create a simple heatmap to see where is the data missing.

```
sns.heatmap(df.isnull(), yticklabels=False, cbar = False)
```




```
# Import seaborn as sns
import seaborn as sns
%matplotlib inline
```

```
df.isna().sum()
```

```
id                  0
Manufacturer        0
Model               0
Type                0
Min.Price           0
Max.Price           0
MPG.City            0
MPG.Highway         0
AirBags             0
Passengers          0
Length              0
Wheelbase           0
Width               0
Turn.circle         0
Rear.seat.room      0
Luggage.room        0
Weight              0
Origin              0
Make                0
dtype: int64
```

```
df.isna().sum(axis = 1).sort_values(ascending = False)
```

```
id  Manufacturer  Model  Type  Min.Price  Price  Max.Price  MPG.City  MPG.Highway  AirBags  Passengers  Length  Wheelbase  Width  Turn.circle  Rear.seat.room  Luggage.room  Weight  Origin  Make
0   False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
1   False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
2   False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
3   False         False  False False  False  False      False     False        True     False       False   False      False  False  False        False           False         False   False   False
4   False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
...
90  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
91  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         True    False   False
92  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
93  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
94  False         False  False False  False  False      False     False        False    False       False   False      False  False  False        False           False         False   False   False
Length: 95, dtype: bool
```

```
sns.heatmap(df.isnull(), yticklabels=False, cbar = False)
```

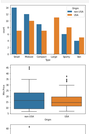
Dropping Unnecessary Colums:

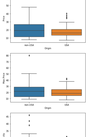
```
df = df[df.columns[:-1]]
df = df.drop(columns=["id","Make","Model","Manufacturer","Origin"],axis=0)
```

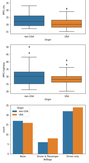
Analyzing Data

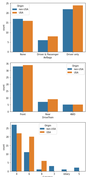
Creating different plots to check the relationship between variables. Simply explore the data by using various columns.

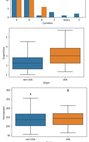
```
for c in df.columns:
    if df[c].dtype == "object":
        pd.crosstab(df[c],df[c]).plot(
            kind="bar",
            stacked=True,figsize=(8,5),
            grid=False)
    else:
        sns.boxplot(df[c],df.Origin,N=5)
        plt.show()
```

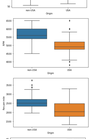


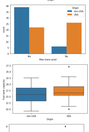


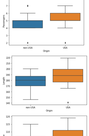


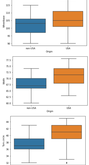


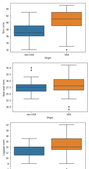


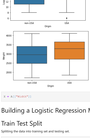


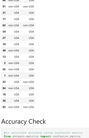












```
df = df[df[df.columns[-1]] != False]
df.head()
```

Building a Logistic Regression Model

Train Test Split

Spliting the data into training set and testing set.

```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(df.iloc[:,0:-1],df.iloc[:,-1:],random_state=1)
```

Logistic Regression

Logistic Regression is used as a mean classification problems. It is used to predict categorical variable.

```
from sklearn.linear_model import LogisticRegression
lr = LogisticRegression()
lr.fit(X_train,y_train.values.ravel())
print lr model parameter values:
```

```
print
```

```
model=lr.fit(X_train,y_train.values.ravel())
print("Intercept: ",model.intercept_)
print("Coefficients: ",model.coef_)
```

```
Intercept:  [ 0.]
Coefficients:  [[ 0.00000000 -0.00000000 -0.00000000 -0.00000000 -0.00000000
  -
```