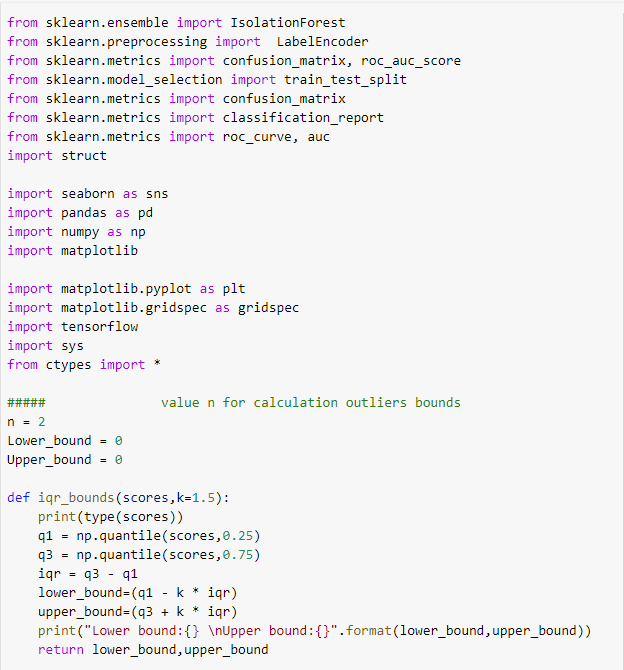
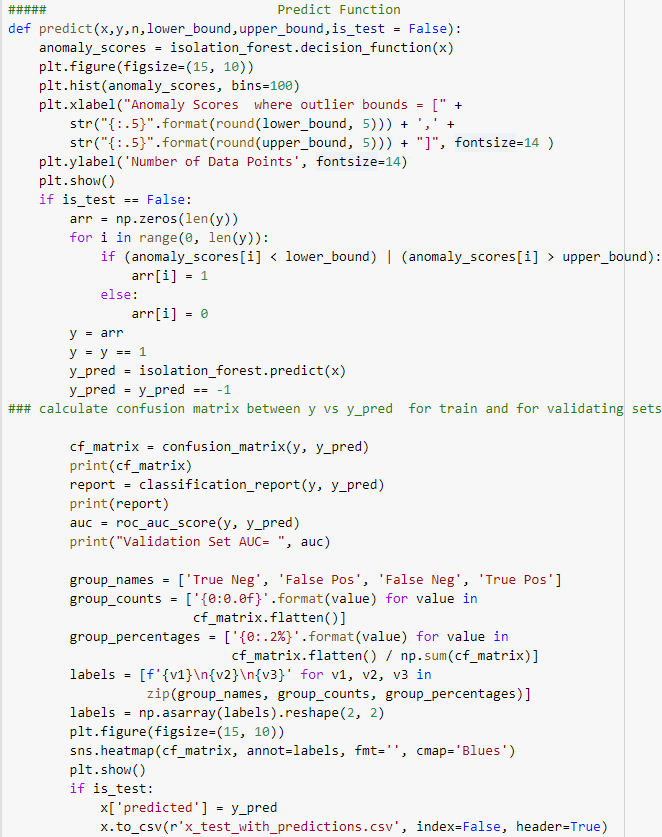
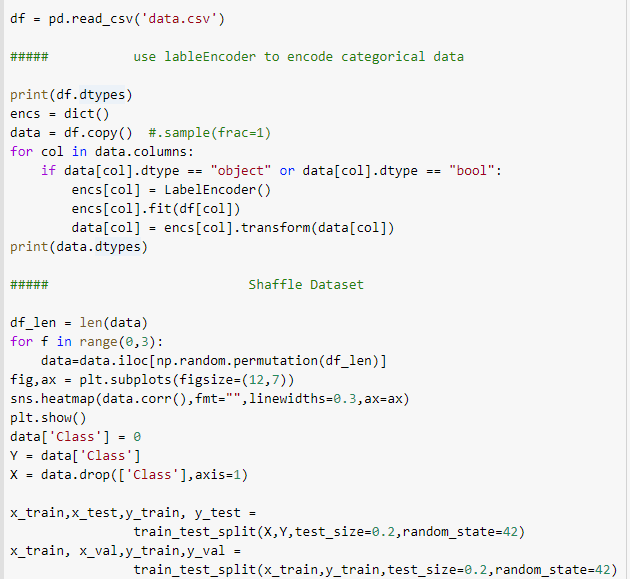
Performed by Inna Williams

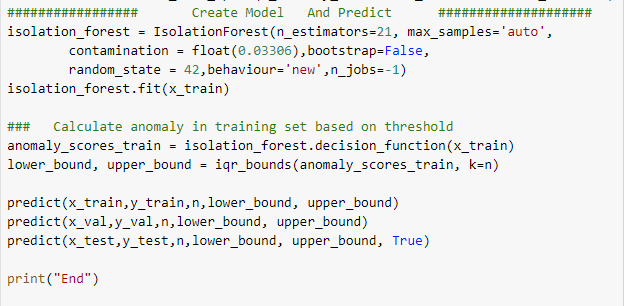
10/25/2020

Code

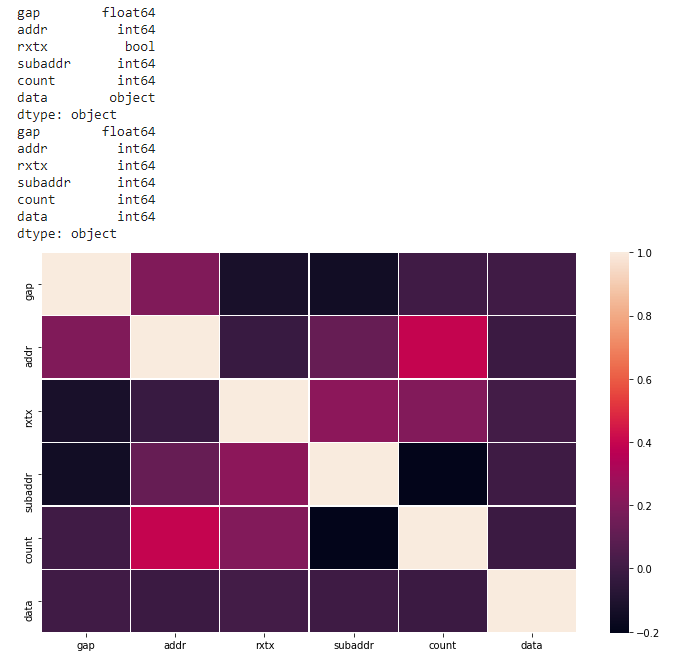








**Visualization Output and Conclusions**



For detecting outliers boundary the following formula used:

Outliers =

Lower Bound = Quantile1 - 2 \* (InterQuantileRange)

UpperBound = Quantile3 + 2 \* (InterQuantileRange)

Algorithm used for detecting Anomalies is Isolation Forest with python library Sklearn.

This algorithm is not biased to outliers and therefore can be used for this purpose.

There is also other algorithms that is good for this is Autoencoders Neural network.

But I will use today the Isolation forest to detect outliers.

Categorical values encoded using sklearn.preprocessing LabelEncoder

The values that were encoded are hexadecimal data values and bool values.

At the end program outputs the test file with a column name predicted added to the end of that testing set. True if the sample is outlier and False if it is not.

Originally the “Class” label set to = 0

Then the anomaly score is calculated first using the training set x\_train.

Low Bound and Upper bounds are used to reset the labels to True if it is outside of bounds

And set to False if it is in bounds for x\_train and then using the bounds found in training set

Setting the labels for the validation set.

Then Confusion Matrix, Classification report and AUC score were calculated for both

x\_train and x\_val. To avoid underfitting and overfitting the tuning has been performed using

n\_estimators=21 and contamination = float(0.03306). Also statistics were compared in training and validating datasets.

If Statistical Metrics are low in both training and validating sets -> indicates undefitting

If Statistical Metrics are high in training set and low in validation set -> indicates overfitting

If Statistical Metrics are very close to each other -> indicates the best choice for the model.

You can open the file and view the outlines by searching for a True value in the file

X\_test\_with\_predictions.

When we open this file with notepad++

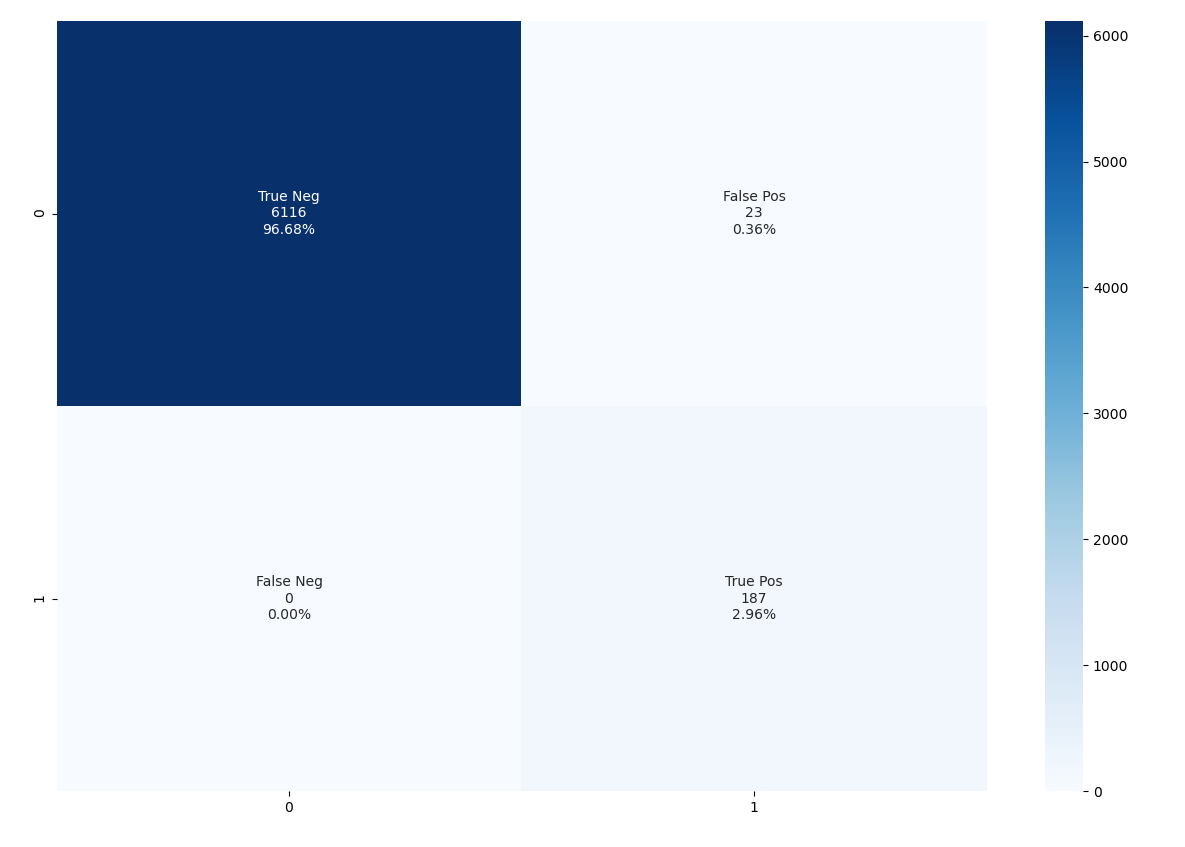
We can see that the file has 1979 sample

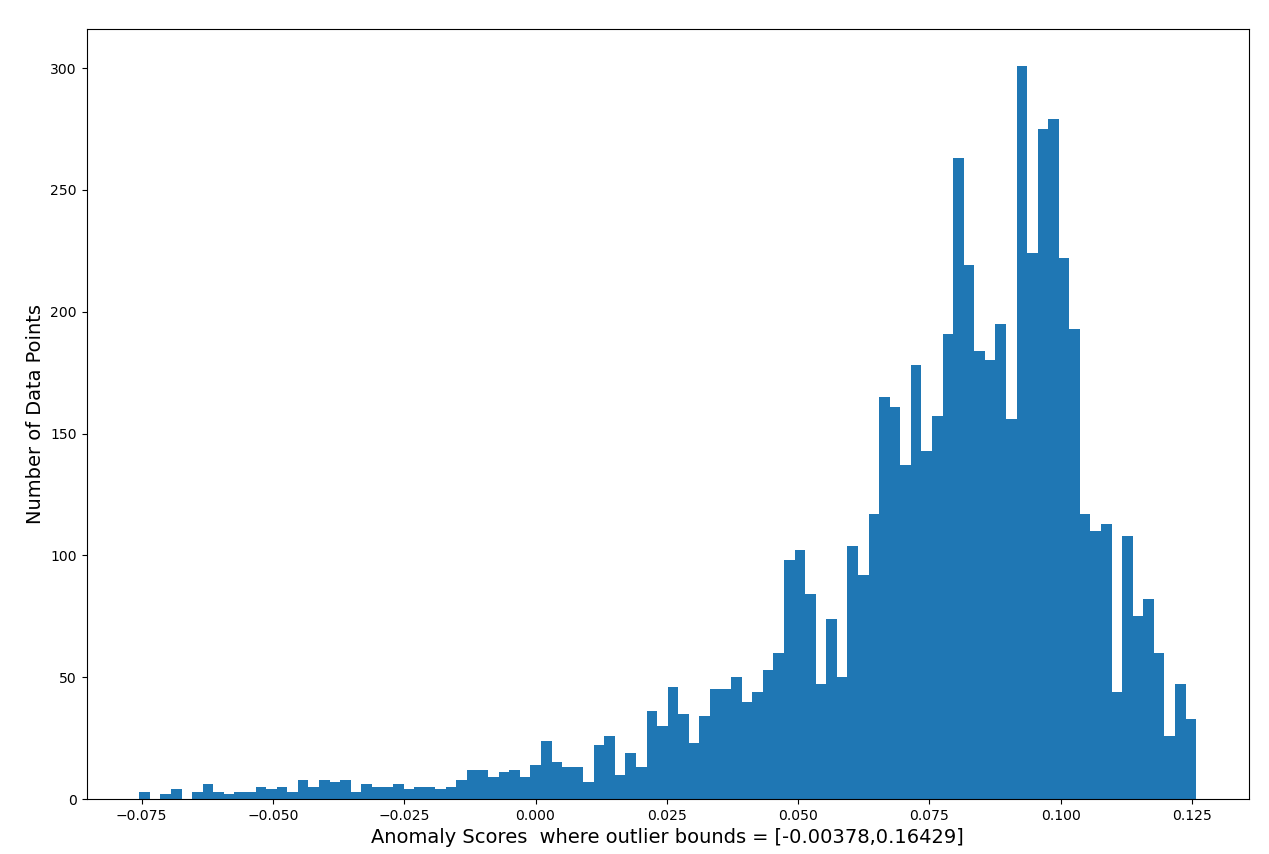
If we search for the True values we can find 61 values

(61 / 1979) = 0.0308 ~ contamination rate that we set in the model = 0.03306

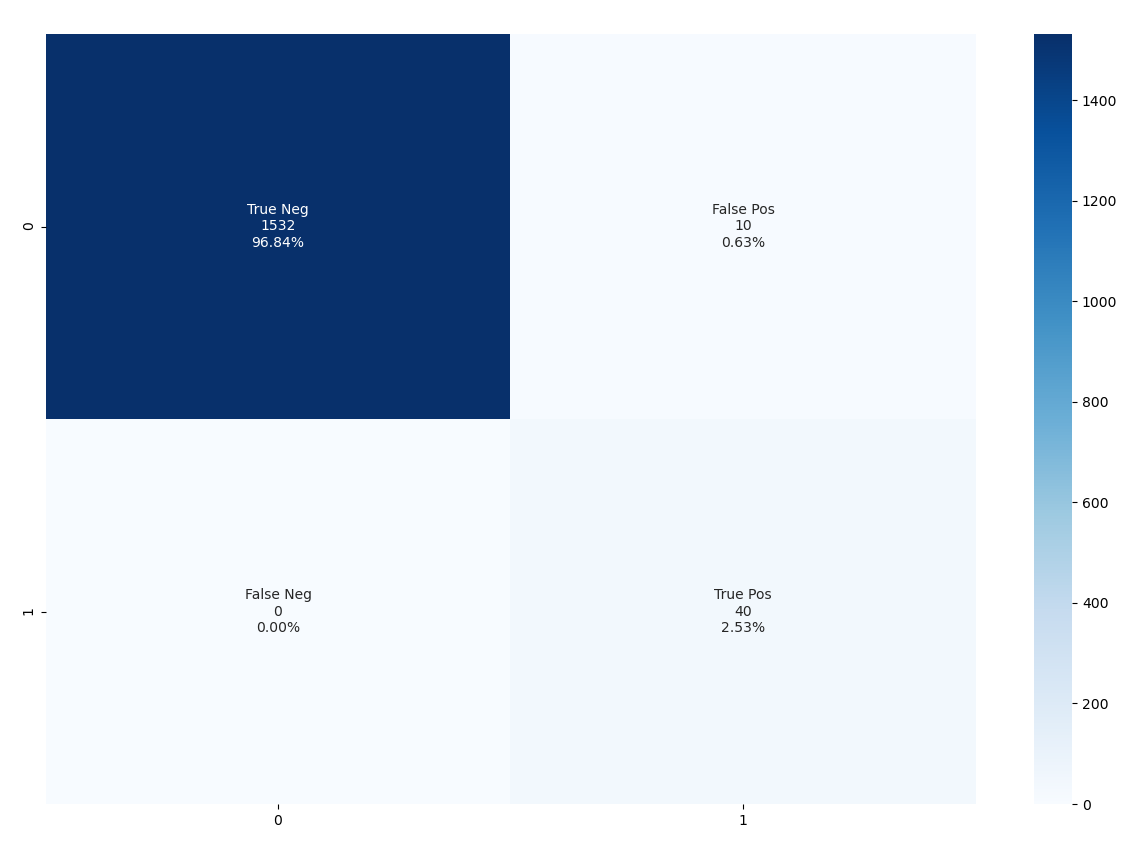
**Training Set**

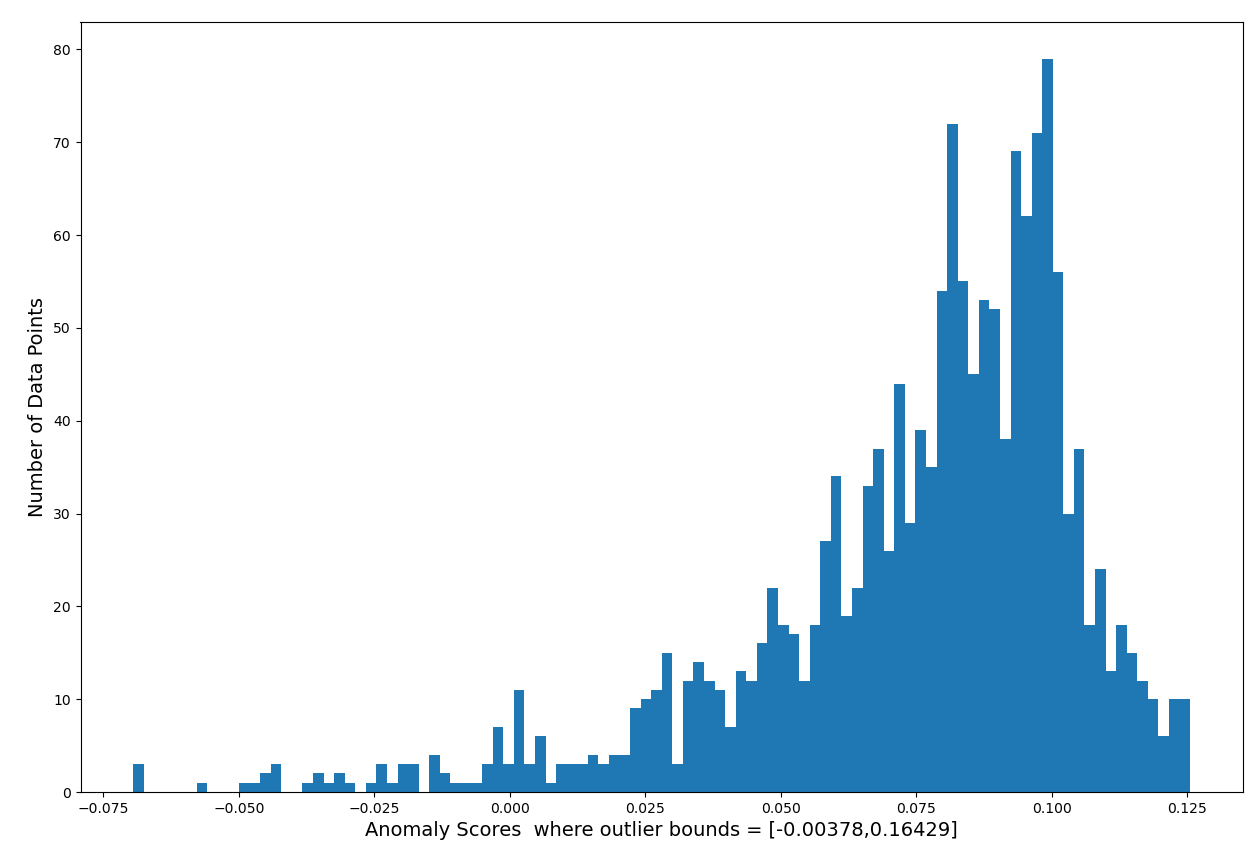
****

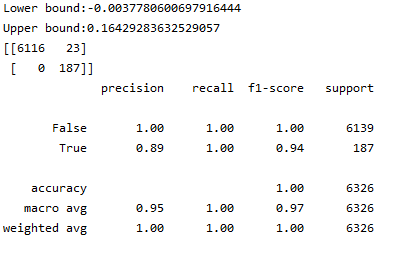




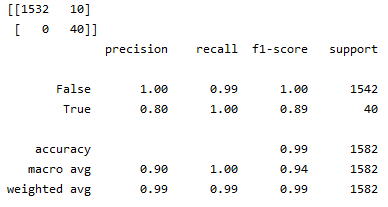
**Validation Set**









****

****

**This is a High Recall Problem. -> must have high value in order not to miss the False Negatives. To have a value of Recall the highest possible is very important for this business problem.**

**The above output for training and validation set shows**

**That Recall = 100% in both sets, for True that is represent the anomaly**

**The above output shows that the model has Precision of 89% in training set and**

**80% for validation set for True(represents the anomalies). This value represents how confident we are about the True Positives. It is ok. It should not be very low.**

**Accuracy values in training and validation sets are very close hat is very important for**

**Underfitting and Overfitting.**

**AUC Score also close to each other on both training and validation sets and ~ 99%**

**Predicted values for the x\_test is in x\_test\_with\_predictions**

**In column predictions.**