Machine learning course (Coursera)

# **Outlier detection using python**

To solve the issues like anomaly detection in millions of records by hand is next to impossible. Rule-based logic do not scale well beyond basic scenarios.

In such cases machine learning gives us some tools that can make this task easier by automating analysis, combining data from different sources in ways

that are very hard to do in typical programming, and modelling complex patterns in data.

There are lot of methods for anomaly detection using supervised as well as unsupervised learning.

**Supervised** –

In this case labelled data is provided .So we try to tune the model to fit to the labelled data(output data that we know we get when we give some input.) and the features provided.

**Unsupervised** –

Used in all those cases where we don't have much information on labelled data i.e. we don't have labels.

Anomaly detection is best suited for Unsupervised algorithms as we don't have examples where anomalies have occured in the past i.e. lack of labelled data.

One way to detect anomalies is using SVM (Support Vector Machines).They can be used both as a supervised as unsupervised learning algorithm.

**One Class SVM**

We are going to use One Class SVM which tells the points in the current dimensional space is an outlier or not i.e it whether belongs to outlier or anomaly (one) class. It does that by representing the points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to one category or another, based on which side of the gap they fall on.

**Hyper-parameters** –

To divide the values we need to give conditions like which kernel to select in order to control the variance and distance between each category .So it takes decision of dividing into categories based on the hyper-parameters that are supplied to it.

* nu is the proportion of outliers we expect in our data.The number of outliers or class members you expect should be supplied.
* kernel is the kernel type to be used. The default is rbf (RBF - radial basis function) which

is a Gaussian kernel and changes the variance of the dimensional space.

More details are given in the coursera’s Machine learning course by Andrew ng.

**Implementation**



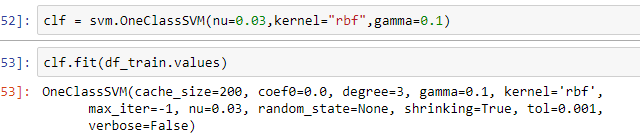
**Figure shows :** There are 9 labels detected earlier.

In the figure we have label data which can be used to check the **nu** hyper-parameter.

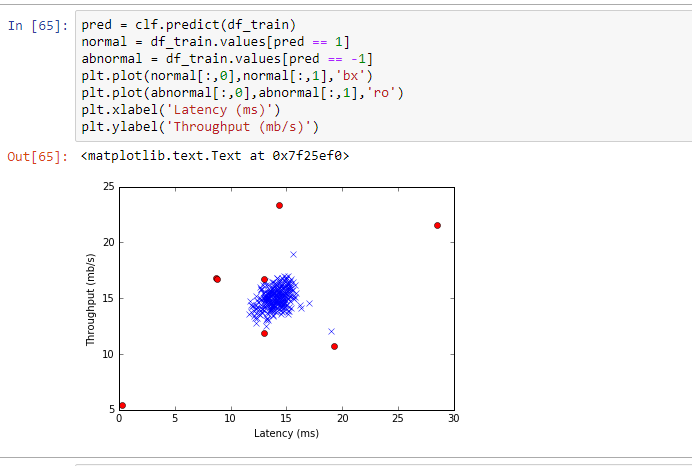
Here mu=Total anomalous labels/Total labels= 9.0/297 0.03

We train the model by calling the fit function from scikit-learn's svm.OneClassSVM and supply nu, kernel.

Training is basically dividing the data points between two classes one’s that are near to SVM curve and other that are far away from it.



Now lets see how it predicts / categorises the values . We will supply the new data.



**Result**  –

So our given data has Latency and throughput on the resp axis .The values with extremely high throughput or very high latency are considered to be anomalous .This is predicted successfully by our model.