INFO 4150E– HW2
Anomaly detection

Domain information

Anomaly detection is one of the applications of Machine learning and is widely used across different domains in industry and consumer / retail space. The idea behind anomaly detection is to be able to predict the potential of an abnormal event happening, much before it happens. This applies to industrial machinery failing to fraud in the consumer industry to healthcare to epidemics. It is a vast subject and through this assignment you should hopefully get an idea of the concepts and glimpse into techniques.

In this HW we work with a simple dataset with some made up features. The idea is to learn to apply anomaly detection. In real life, the formation of the dataset is most critical to the ability to be successful with early predictions.

Some important things to keep in mind while putting together a dataset are listed below:

1. Understand the failure modes of the domain you are building an anomaly detection system for.
2. Understand the different kinds of anomalies they can generate.
3. Study the system to narrow down on the features / variables which either directly measure the magnitude of those anomalies to happen (e.g., high bearing temperature leading to failure) or which give us an indirect signal about the presence of those anomalies (e.g. infectious disease and their spread).
4. Do all the features have to be direct measurements or can we derive some of these features by modeling a scenario?
5. Looking system wide to understand the root causes for failures, which can often originate elsewhere but leads to a failure somewhere else in the system.

Task in this HW

This HW is about building a model to detect and therefore predict anomalies. It will be done by using the anomaly detection algorithm you learned – **both the original model (independent features gaussian type) and the multivariate gaussian model (when there is covariance) will be developed.**

There are two datasets provided – a training dataset and a test dataset.

The model is built with the training set which determines the optimal value of "epsilon" or "eps". As you would have learned from the theory and examples this value is the threshold to determine whether a data point is a normal one or an anomaly. **The training data has no anomalies**. So first pick an "eps" value to satisfy that requirement.

Once you have done this, apply the model on the test set provided, which has 2 data points, and both are anomalies and should be flagged as anomalies. It's very possible that the model may not do this immediately. This therefore means that you must vary the epsilon value till both test set data points are flagged as anomalies.

The goal here is, is to tune the model parameter such that both the test set data points are flagged as anomalies and at the same time the number of data points in the train set which get misclassified as an anomaly (there could be some), are minimized. **This needs to be done for each model independently.**

Datasets

The two datasets are “train.csv” and “test.csv”.
The datasets have been uploaded in eLC.

Guidelines for model building

You are required to build the anomaly detection models first with the original or independent features technique and then the multivariate gaussian technique.

Please do all the development work in the Jupyter Notebook template that is provided. Remember to change that file name by adding your first and last name.

The goal for both methods (original and multivariate) is to make sure the two data points in the test set are classified as anomalies and at the same time minimizing the number of misclassifications in the train set.

It is possible that the original model, assuming independent features method, misclassifies a lot of data points as anomalies despite trying out different threshold values. Please mention your reasons as to why, in your final conclusions. Then work out with the multivariate method and see if that fits better. If it does, explain in your conclusions why you think that is the case.

In both cases plot the contour plots so that we can clearly visualize where the anomalies lie. Plot all the detected anomalies, in train and test data, in red.

Your deliverable will be the Jupyter Notebook. It shall contain all the code, visualization, and your comments / conclusions in a cell at the end of the notebook. Please highlight your observations, assumptions and talk about your results.

Please also write a paragraph on what you learned while doing this project and what some of your challenges were.

The project is due no later than June 29th, 2022, by midnight. My recommendation is to try to submit it earlier and not keep it to the last minute.