Anomaly Detection Algorithm Selection

Simulation of Gas Flow at Easington Langeled

Adam Kaizra

October 28, 2024

Abstract

A brief explanation of the problem and possible challenges, followed by a discussion of a few contender algorithms. I then explain my rational behind selecting Exponential Moving Approach and why Isolation Forest is my backup.

1 Introduction

We have a live one dimensional datastream of floats representing gas flow measured in mcm/day. The data is not normally distributed. We have two simulations, one without anomalies and one with added random anomalies. We want to find an anomaly detection algorithm that is capable of adapting to concept drift and seasonal variations.

Machine learning algorithms are either supervised (trained with labelled data), clean (no anomalies in training data) or unsupervised (no labels, anomalies present). As we simulate the gas flow anomalies we can use either of these 3 settings. We are however assuming that the outages I caught and removed were the only anomalies in the baseline data, which is unlikely and means there are probably anomalies in the baseline. I calculated monthly averages and interpolated the daily results, which should minimise the effect of any anomalies in the 'clean' data.

2 Algorithm Analysis for Gas Flow Monitoring

Algorithms can work on either live or static data. Most algorithms work on static data so batching is required to implement them on live data.

2.1 Promising algorithms and my selection

Isolation Forest algorithm splits the data into a binary tree and then calculates the average depth it takes to get to a data point. Outliers are usually easier to get to. The algorithm is efficient and accurate. However an implementation on live data would require batching to work. It requires periodic retraining and performance can degrade with concept drift in live data.

LODA (Light weight On-line Detector of Anomalies) creates a set of sparse random projections and histograms for each one. Anomalies are detected based on how points deviate from the histogram. Its very efficient but not as accurate as Isolation Forest. It was built specifically for live data and doesn't suffer from concept drift. It works best on high dimensionality data to create multiple projections, so won't work well with our data.

EMA (Exponential Moving Average) is very efficient with O(1) time complexity. It works on live data and can detect single points and extended

durations of anomalies. However it is susceptible to seasonality changes and our data has significant seasonality.

I think there is a fair argument for either of these 3 methods but I believe EMA makes the most sense for our use case. As our data has daily cycles and seasonality I will have to consider ways to deal with this if I want to use EMA. If I can't get EMA to work to an acceptable rate I will use IF and batching instead.

2.2 Algorithms I didn't choose and why

Statistical methods such as Mean Absolute Deviation (MAD) and Z-Score either require or work best on normally distributed data. Our data is not so I won't use these.

Auto Encoder uses a deep neural network to decrease the dimensionality of data and then use that to reconstruct the data. The better it can reconstruct the data the higher likelihood the data is nominal. Our data is single dimensional so I don't think this would work.

Local Outlier Factor compares the local density of a point to its neighbours, won't work well in my simulation as anomalies can last for days (such as outages).

Single-Class SVM (Support Vector Machine) is trained to find an area of nominal data and points falling outside are deemed anomalies. Requires retraining for live data.