Anomaly Detection Engine for Medical Technology Equipment

By Mahadev Vasireddy, Naveen Kumar, Vivek Hinduja, Sree Vishnu

We would like to start with thanking Indian School Of Business and Cyient Insights for providing us an opportunity to work on this challenging and educative project. The project was interesting from the academic viewpoint that it touched various aspects like data visualization, feature engineering, predictive model building and handling the huge data volume. It challenged us at every step moving forward making the whole journey exciting and overcoming each obstacle a cherishable experience.

Project Description

Monitoring of capital intensive health equipment is required (a) to reduce the equipment downtime, (b) increase the revenue generated from equipment usage, and (c) ensure the safety of operations as per regulations and auditory requirements.

The problems of interest are:

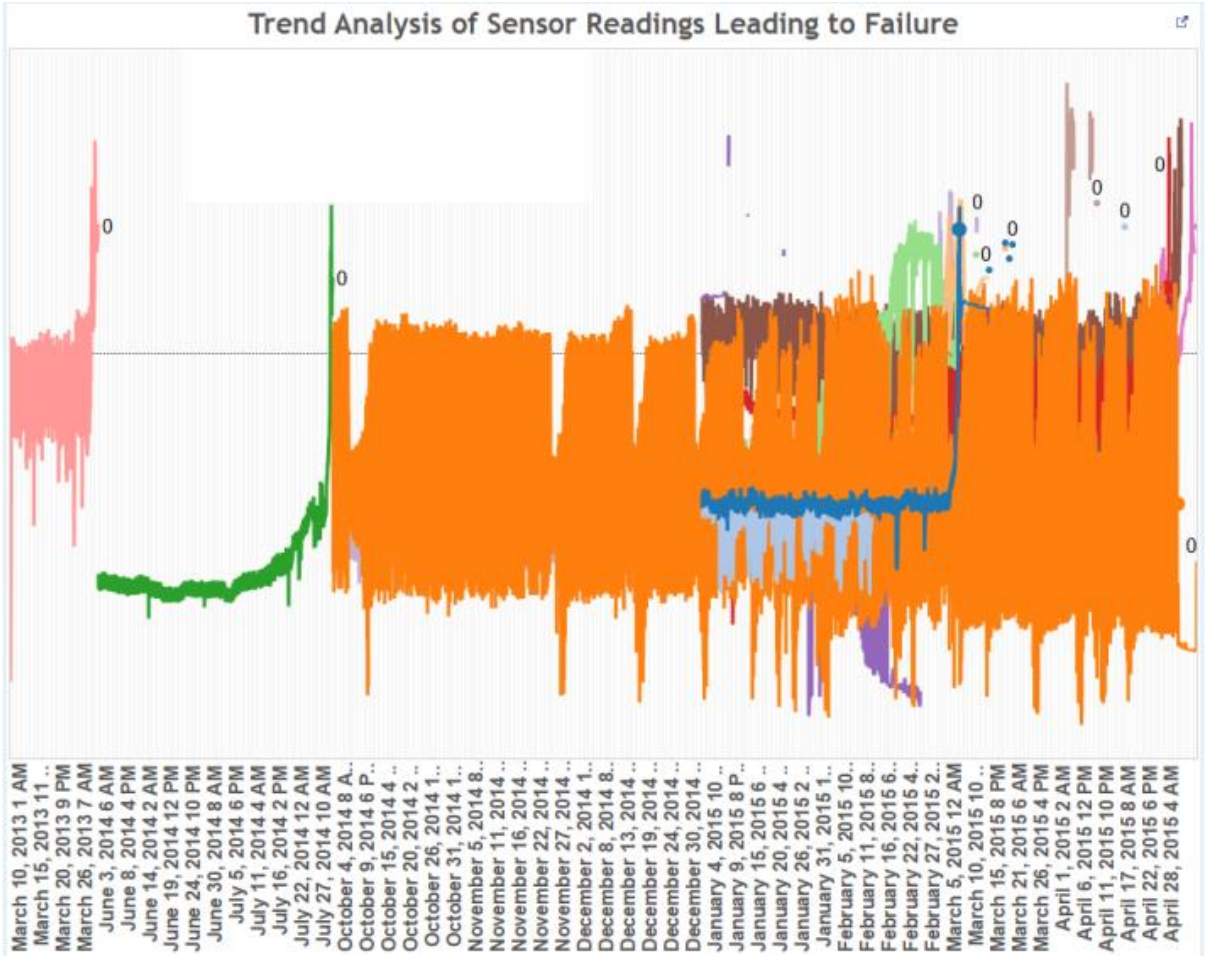
1. To be able to "identify the abnormal (or) anomaly state of equipment operations from the normal operating conditions" based on the continuous stream of data sent from the sensors embedded on these equipment.
2. Identifying patterns/trends and correlations between abnormal event(s) occurrence and operating conditions as captured by the sensors in real time.
3. Ability to categorize the data streams to: Normal, Alarm, Critical zones in (near) real time.
4. Device intelligent algorithm(s) for an early abnormal detection using historical sensors data and streaming data.

Dataset Description

Data for five machine failure types has been provided in five different datasets, where each data set has data pertaining to a specific failure type. The five failures are **1st Stage Leak**, **2nd Stage Oil Return**, **2nd Stage Leak**, **1st Stage Compressor Failure, 2nd Stage Compressor Failure.** The total size of these datasets is around 1 GB.

Column(s) Name	Column Description
ProdID	Machine Identifier
Date_Time	Time when the sensor readings are captured
TC1,TC2,TC3,TC4,TC6,TC7,TC9,TC10	Temperature readings captured by the eight sensors. Each is a different dimension.
Mains_Voltage	The Voltage reading of the machine
State	State of the machine
FailureType	Failure Type of that dataset
Failure_Instant	Time when the machine failed
Time.Format	Time format used
Date_Time_New_24	Date time in 24 Hr Format
Date_Time_New_12	Date time in 12 Hr format
DbF	No of days from this instant to the point where the machine failed

Exploratory Data Analysis



The Zero label above represents the point of failure, the X axis represents the time duration of the machine and Y-axis marks the sensor readings. The different colour coding represents the various machines. As the machine approaches the point of failure it is marked by abrupt changes in its temperature readings. The conclusion drawn out of it is these abrupt changes can be identified through detection of outliers and hence to crack this problem relevant outlier analysis needs to be done and we are treating this as a case of "**Time-series and Multidimensional streaming data outlier detection**".

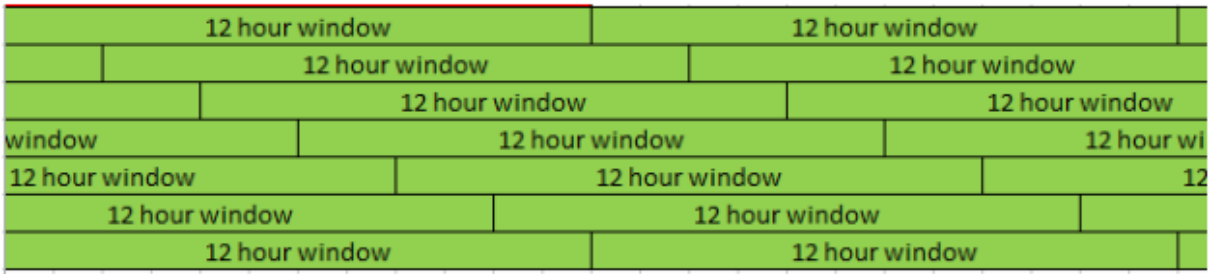
Feature Engineering

We considered the ratios of the existing dimensions, the mean temperatures across various sensors, the rolling window features and finally built around 400 + features. Further, the features have been created keeping in mind the correlations across time and sensors. We did a Z-scoring of all the attributes across various sensors and time as so as to make a level playing field during analysis of the data. Features on Z-score such as Mean, SD, and min & max across moving time windows.

Windows of varying length:

<table>
<tr><td colspan="6">24 hour window</td></tr>
<tr><td colspan="3">12 hour window</td><td colspan="3">12 hour window</td></tr>
<tr><td colspan="2">8 hour window</td><td colspan="2">8 hour window</td><td colspan="2">8 hour window</td></tr>
<tr><td colspan="2">6 hour window</td><td colspan="2">6 hour window</td><td colspan="2">6 hour window</td></tr>
<tr><td>4 hour Window</td><td>4 hour Window</td><td>4 hour Window</td><td>4 hour Window</td><td>4 hour Window</td><td>4 hour Window</td></tr>
</table>

Windows with staggered start and end time:



Density based outlier detection

With multivariate data set we cannot expect to see clusters of data points of regular shape. To discover clusters with arbitrary shape and outliers, we have used density-based clustering methods. We have calculated the Outlier Factor at each point in the data. Outlier Factor is a measure of density in the area of the point. Based on the Knee in the variation of the Density plot against time to failure, a threshold for classification is decided.

Models tried

Naïve Bayes Classification: We used a 2 class classification to predict the transition from working mode to 1 day to failure mode.

Survival Analysis: We used the time frame from the first anomaly to the machine failure for all the machines and built a basic Survival function to estimate the probability of the machine failing on that day.

Accelerated Time to Failure Models: We wanted to quantify the effect outliers had on the time to failure. Accelerated Time to Failure models can be used to see the effect of intermediate stages have on the overall life of the machine.

Markov Chain Analysis: We wanted to see the state transition between working and failing of the machine. We wanted to use multiple states like working, deteriorating (outliers) and failure. We have used only a 2 state transition matrix.

Multi Class Classifiers using 4 states – Normal, Alarm, Critical and Failure: Logistic regression, SVM and Decision Tree and Random Forest.

Survival Vs Markov Chain Report – Interesting Analogies:

Failure Type 1: Markov chain tells us that if I am in state 1 i.e. functional state it will take me 33 days to state 0 i.e. failure state. The Accelerated Failure time (parametric method) states that 10 percentile have failure within the first 35 days.

The Kaplan Meier curve shows the first failure on 27th day and 2nd failure on 31st day. The 90th percentile in the Accelerated Failure Type model shows 90th percentile surviving up to 153 days and the Kaplan Meier curve shows the probability of surviving 152 days 0.08.

Conclusion

Of all the approaches we tried above, Naïve Bayes comes closest to the objective of predicting a failure with confidence or a score. Although Naïve Bayes has been applied only on one fault-type (due to computational constraints) we are fairly confident that this technique can be extended to other fault types. Other approaches we tried are SVM, Decision Trees and Multinomial Logistic regression.

Although the results weren't as good as Naïve Bayes approach, we believe these methods are worth exploring further to improvise their respective accuracies.

Learnings

1. Don't jump to a model – Let the model develop on its own from the data. Understanding the problem and preparing the data for model building can take up to 70% of your total project time, and it is also the most critical part of the whole project. Do not jump to conclusions.
2. Visualize – Nothing works like a visual representation of the data. Visualize the data and let the picture tell you the story of the data.
3. Team members will have different opinions – Team does not have to agree on everything. If someone wants to try something new or different from the agreed upon direction, let them try it after they have completed their assigned task.
4. Never stop learning – Learning the breadth of the subject is as important as identifying one area of interest to you and deep diving into that area of interest. There are a ton of opportunities to continue learning.