SAJAL ROYCHOWDHURY | DDA1610082 | March 13, 2017

Capstone Project-Predictive Maintenance

Post Graduation diploma in data analysis-2016-2017



Problem Description

A major problem faced by businesses in asset-heavy industries such as manufacturing is the significant costs that are associated with delays in the production process due to mechanical problems. Most of these businesses are interested in predicting these problems in advance so that they can proactively prevent the problems before they occur which will reduce the costly impact caused by downtime.

# Data Sources

Common data sources for predictive maintenance problems are

1. Failure history: The failure history of a machine or component within the machine.
2. Maintenance history: The repair history of a machine, e.g. error codes, previous maintenance activities or component replacements.
3. Machine conditions and usage: The operating conditions of a machine e.g. data collected from sensors.
4. Machine features: The features of a machine, e.g. engine size, make and model, location.
5. Operator features: The features of the operator, e.g. gender, past experience.

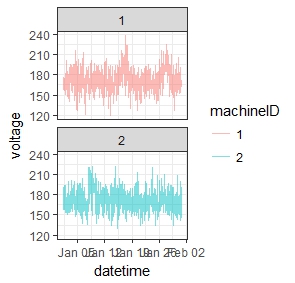
The data for this example comes from 4 different sources which are real-time telemetry data collected from machines, error messages, historical maintenance records that include failures and machine information such as type and age.

# Methodology

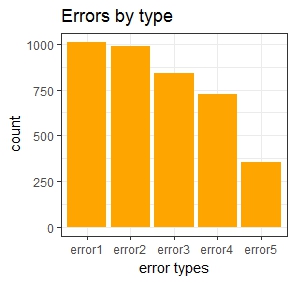
1. Loading the data files and Exploratory Data Analysis

The Data files namely Telemetry, Errors, Maintenance, Machines, Failures were loaded on to R Studio IDE, changes to the time dimensions and identification of the null values and the method of replacement was determined.

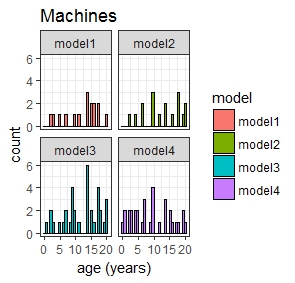
Exploratory Data Analysis such as mean, quantiles, mode, sum etc. and graphical representations to find the inter relation between the variables was also performed. Some of the sample plots are included here:



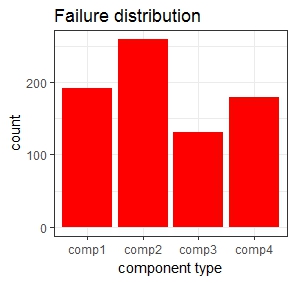
**Fig.1** Sample time plot on Volt for different machine ID



**Fig.2** Types of errors



**Fig.3**. Machine age w.r.t. modes



**Fig.4**. Failure Distribution across components

1. Feature Engineering

The time lag features of the data, rolling mean and the standard deviation was calculated. The time lag featuring was done because the original data was mapped into hourly intervals and hence was collated into a 24-hour period. The same procedure was repeated for all the data files, to create features that best describe a machines’ health condition at a given point in time.

1. Label Construction

The prediction problem for this example scenario is to estimate the probability that a machine will fail in the near future due to a failure of a certain component. More specifically, the goal is to compute the probability that a machine will fail in the next 24 hours due to a certain component failure (component 1,2,3 or 4). In the following, labelling is done by labelling all the feature records that fall into the 24 hours’ window before a failure due to component 1, component 2, component 3 and component 4 as comp1, comp2, comp3 and comp4 respectively.

1. Division of Training and Test Data and creation of Training formula

The data is divided into 8 months of train data and 4 months of test data. 3 sets of such data are created, dividing on the basis of time stamps.

The training formula is based on removing the labelled features and deriving the formula on Gradient Boosted Modelling which is used for predicting the failures of the components.

1. Gradient Boosted Algorithm

The idea of boosting came out of the idea of whether a weak learner can be modified to become better. The statistical framework cast boosting as a numerical optimization problem where the objective is to minimize the loss of the model by adding weak learner using a gradient descent like procedure. This is called as Gradient Boosting.

This mainly evolves 3 elements: a loss function which is to be optimized, a weak learner to make predictions and an additive model to add weak learners to minimize the loss function.

Decision trees are used as weak learners in gradient boosting.

1. Validation and Evaluation

The model thus defined is validated an evaluated using a custom script for designing a Confusion Matrix also recall function which actually tests the accuracy and definition of kappa and precision rate. We have defined the CI as 0.95 i.e. we can confidently predict the results 95 % times.

Confusion Matrices for the 3 models are defined here:

|  |
| --- |
| Predicted |
| Actual comp1 comp2 comp3 comp4 none |
| comp1 484 21 2 14 7 |
| comp2 3 835 9 41 2 |
| comp3 22 7 383 4 0 |
| comp4 10 15 4 554 1 |
| none 14 0 0 5 120315 |
| Predicted | | |
| Actual comp1 comp2 comp3 comp4 none | | |
| comp1 378 15 0 8 7 | | |
| comp2 1 693 8 17 3 | | |
| comp3 8 8 301 2 1 | | |
| comp4 16 9 6 418 1 | | |
| none 11 0 0 5 95982 | | |

|  |
| --- |
| Predicted |
| Actual comp1 comp2 comp3 comp4 none |
| comp1 284 12 0 8 2 |
| comp2 4 555 0 9 2 |
| comp3 8 0 213 3 0 |
| comp4 8 15 5 293 1 |
| none 8 0 0 5 72437 |

The results are summarized in the tabular format for better readability.

| Sr. No | failure | model1 Recall | model2 Recall | model3 Recall |
| --- | --- | --- | --- | --- |
| 1 | comp1 | 0.9166 | 0.9264 | 0.9281 |
| 2 | comp2 | 0.9382 | 0.9598 | 0.9736 |
| 3 | comp3 | 0.9206 | 0.9406 | 0.9508 |
| 4 | comp4 | 0.9486 | 0.9288 | 0.9099 |
| 5 | none | 0.9998 | 0.9998 | 0.9998 |

Looking at the data we can safely conclude that prediction for the model is above 90 %.

# Conclusion

For predicting the failures and doing predictive maintenance on the industrial machines we have used Gradient Bosting model for doing the same. The predictability of the algorithm i.e. the accuracy can be improved by iterating the algorithm via Cross Validation, with the cost of increasing the computational power.