PREDICTIVE ANALYSIS FOR MACHINE FAILURE

A Project-II Report

Submitted in partial fulfillment of requirement of the

Degree of

BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE & ENGINEERING

BY
Siddharth Kekre
EN16CS301261

Under the Guidance of
Ms. Ruchi Patel

Department of Computer Science & Engineering
Faculty of Engineering
MEDI-CAPS UNIVERSITY, INDORE- 453331

April 2020

PREDICTIVE ANALYSIS FOR MACHINE FAILURE

A Project-II Report

Submitted in partial fulfillment of requirement of the

Degree of

BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE & ENGINEERING

BY
Siddharth Kekre
EN16CS301261

Under the Guidance of
Ms. Ruchi Patel

Department of Computer Science & Engineering
Faculty of Engineering
MEDI-CAPS UNIVERSITY, INDORE- 453331

April 2020

Report Approval

The project work "**Predictive Analysis for Machine Failure"** is hereby approved as a creditable study of an engineering/computer application subject carried out and presented in a manner satisfactory to warrant its acceptance as prerequisite for the Degree for which it has been submitted.

It is to be understood that by this approval the undersigned do not endorse or approved any statement made, opinion expressed, or conclusion drawn there in; but approve the "Project Report" only for the purpose for which it has been submitted.

Internal Examiner
Name: Ms. Ruchi Patel
Assistant Professor
Medi-Caps University

External Examiner
Name: Dr. Ravi Changle
Mentor and Trainer
Talent Sprint, Tata Consultancy Services

Declaration

I/We hereby declare that the project entitled "**Predictive Analysis for Machine Failure"** submitted in partial fulfilment for the award of the degree of Bachelor of Technology/Master of Computer Applications in Computer Science and Engineering completed under the supervision of **Ms. Ruchi Patel, Assistant Professor, Computer Science,** Faculty of Engineering, Medi-Caps University Indore is an authentic work.

Further, I/we declare that the content of this Project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University for the award of any degree or diploma.

Siddharth Kekre

April, 2020

Certificate

I/We, **Dr. Ravi Changle, Ms. Ruchi Patel** certify that the project entitled "**Predictive Analysis for Machine Faliure"** submitted in partial fulfilment for the award of the degree of Bachelor of Technology/Master of Computer Applications by **Siddharth Kekre** is the record carried out by him/them under my/our guidance and that the work has not formed the basis of award of any other degree elsewhere.

Ms. Ruchi Patel Computer Science Engineering Medi-Caps University, Indore	Dr. Ravi Changle Mentor and Trainer Talent Sprint, TCS

Dr. Suresh Jain
Head of the Department
Computer Science & Engineering
Medi-Caps University, Indore

Acknowledgements

I would like to express my deepest gratitude to Honorable Chancellor, **Shri R C Mittal,** who has provided me with every facility to successfully carry out this project, and my profound indebtedness to **Prof. (Dr.) Sunil K Somani,** Vice Chancellor, Medi-Caps University, whose unfailing support and enthusiasm has always boosted up my morale. I also thank **Prof. (Dr.) D K Panda,** Dean, Faculty of Engineering, Medi-Caps University, for giving me a chance to work on this project. I would also like to thank my Head of the Department **Dr. Suresh Jain** for his continuous encouragement for betterment of the project.

I express my heartfelt gratitude to my **External Guide, Dr. Ravi Changle**, Mentor and Trainer, Talent Sprint, Tata Consultancy Services as well as to my Internal Guide, Ms. Ruchi Patel**,** Assistant Professor, Department of Computer Science Engineering, MU, without whose continuous help and support, this project would ever have reached to the completion.

I would also like to thank to my team at Tata Consultancy Services and Talent Sprint who extended their kind support and help towards the completion of this project.

It is their help and support, due to which we became able to complete the design and technical report.

Without their support this report would not have been possible.

Siddharth Kekre
B.Tech. IV Year
Department of Computer Science & Engineering
Faculty of Engineering
Medi-Caps University, Indore

Abstract

The increase of available data in almost every domain raises the necessity of employing algorithms for automated data analysis. This necessity is highlighted in predictive maintenance, where the ultimate objective is to predict failures of hardware components by continuously observing their status, in order to plan maintenance actions well in advance. These observations are generated by monitoring systems Analysing this history of observations in order to develop predictive models is the main challenge of data driven predictive maintenance.

This report presents the process required to implement a data driven Predictive Maintenance (PdM) not only in the machine decision making, but also in data acquisition and processing. A short review of the different approaches and techniques in maintenance is also given.

This report discusses a preliminary study performed to highlight the potential of applying a classification based random forest (RF) algorithm for predicting the failure of equipment. The RF algorithm is trained on a training dataset and then applied to a test dataset to determine the accuracy of the failure prediction.

Table of Contents

List of Figures

List of Tables

Abbreviations

Sr. No.	Abbreviation	Meaning
1	RMSE	Root Mean Square Error
2	RUL	Remaining Useful Time
3	Scipy	Scientific Python
4	RF	Random Forest
5	SVM	Support Vector Machine

Chapter-1

INTRODUCTION

1.1 Introduction

Machine learning and AI techniques are becoming essential and widely used to improve our daily lives, this is evident from examples of weather prediction, self-driving cars, conversational agents (e.g. Siri) delivered by our smartphones, facial recognition applications and generating credit score to analyse financial risks. One of the promised areas for machine learning is in the domain of machine maintenance.

The identification of faults in complex systems is a difficult task for human operators. Systems are becoming increasingly complex with a rising number of components and interdependencies between the mechanical, electrical and software aspects. During fault situations, this complexity can lead to confusion about the initiating event of a critical event scenario and the location of the fault. A failure at Three Mile Island is an example of how fault detection is challenging. During this event, one pressure release valve failed open during an emergency shutdown and caused a partial meltdown and destruction of a reactor. The damage was extensive because the fault was not identified early enough by the operators due to inadequate control room instrumentation and operator training programs.

In a highly competitive production environment, unscheduled equipment breakdowns cause disruptions in the production capacities. This requires improved response for failure diagnosis and repair times and eventually the capability to proactively handle these failure occurrences for optimized maintenance management. One of the promising approaches to address this challenge is online failure prediction, which requires the current state of a system to be monitored and evaluated to predict the occurrence of failures in the near future. The key contribution from this approach can be divided into methods that revaluate temporal inputs and those that rely on maintenance logs.

Predictive maintenance predicts failure, and the actions could include corrective actions, the replacement of system, or even planned failure. This can lead to major cost savings, higher predictability, and the increased availability of the systems.

1.2 Literature Review

Machine reliability has been the focus of companies for a long time, due to its importance in increasing the overall machine productivity by avoiding machine interruption and catastrophic outcome (Peng, Dong, and Zuo, 2010). Therefore, companies have relied on maintenance strategies to address the issues of machine reliability (Jardine, Lin, and Banjevic, 2006).

According to Munion (2017) Maintenance can be categorised as being reactive or proactive. A reactive approach is used to repair a machine after a failure, whereas a proactive approach tries to prevent the failure from occurring either by scheduling regular maintenance or by using data prediction techniques. Various maintenance strategies have been adopted by companies to manage their maintenance activities; these strategies are Corrective Maintenance, Preventive Maintenance and latest strategy is Predictive Maintenance or sometimes referred to as Conditional Based Maintenance (Zhao et al., 2017). Corrective Maintenance also referred to as Run-to-failure is a reactive approach where repair activities are carried out after a machine fails (Krenek et al., 2016). It is the most straightforward approach to address a failure because it does not require upfront planning or scheduling (Susto et al., 2015). Nonetheless, the unplanned delays and interruption caused by a failure could have severe financial and operational implications to the business (Ibid). In contrast, Preventive Maintenance is a set of defined maintenance activities collected based on manufacturer knowledge, experts, and previous failures, and are performed based on a planned schedule (Krenek et al., 2016). The scheduled maintenance help increase machine reliability and lifetime by regular checks which can detect issues and prevent failures (Susto et al., 2015).

Even though, Preventive Maintenance helps minimise failures still it does not eliminate it. Furthermore, it may lead to inefficient maintenance by carrying unnecessary maintenance that can increase the cost associated with planning, scheduling, and resource to perform the maintenance activities (Ibid). Predictive maintenance or Conditional Based Maintenance (CBM) (Zhao et al., 2017) monitor and collect machine health information which is then used to predict maintenance decisions (Jardine, Lin, and Banjevic, 2006). Predictive maintenance has proven to be more effective than preventive maintenance in optimising the maintenance activities (Susto et al., 2015) by allowing repair activities to be scheduled based on failure predictions which avoid unnecessary maintenance, optimise resources, and reduce machine downtime leading to significant operational savings (Susto et al., 2015).

The research efforts on this field are extensive and a wide variety of alternative fault detection methods and hybrids have been proposed. The contribution of this paper is a methodology for compiling training and testing data sets, using the Functional Failure Identification and Propagation (FFIP) framework, to support a simulation based framework for training and testing alternative methods machine learning based methods for fault detection.

Machine learning studies algorithms enabling computers to learn from data. Although machine learning has been a very active research field since the development of the first artificial neural networks, it has advanced significantly over the last decade because of the increased availability of computing power and the development of new methods with many engineering applications. The abundance of literature on the fault detection and diagnosis domain has motivated researchers to review and categorize the methods, and thus facilitate academic researchers and industrial practitioners.

A three part comparative study of literature related to fault detection classifies the methods into three categories. The "quantitative model-based" methods use models of the system to analytically identify inconsistencies between the actual and the expected behaviour and then decision rules are applied to perform the diagnosis. The "Qualitative models and search strategies" category includes methods based on non-quantitative models of the system, like fault trees and topographic templates created using expert knowledge. The "Process history based methods" do not use any a prior knowledge about the system, but instead rely on large data sets of process data to enable qualitative (e.g. expert systems) or quantitative (e.g. Artificial Neural Networks) methods.

Applications of Artificial Neural Networks have been proposed for safety critical systems. Decision trees are another quantitative method used for fault diagnosis. Fault detection applications based on decision trees have been proposed for photovoltaic arrays , AC transmission lines , power systems and migration paths from fault trees to decision trees for fault detection has been suggested for the International Space Station . Fault detection methods have been proposed, based on data-driven modelling and residual space analysis, Independent Component Analysis, hidden Markov models, optimized fuzzy clustering and Dynamic Case Based Reasoning.

Predictive maintenance systems consist of two main component Diagnostic and Prognostics. Diagnostics is used to detect abnormal behaviour in the machine, while prognostics uses machine learning to predict failure and to calculate Remaining Useful Life. To be able to diagnose and prognoses machine failure, Predictive maintenance systems need to perform the following steps:

1. Data Acquisition: data need to be collected using sensors attached to the machine component.

2. Data Processing: need to handle the processing of data into a format that can be used for analysis

3. Maintenance Decision Making denotes the techniques, e.g. machine learning to drive prediction.

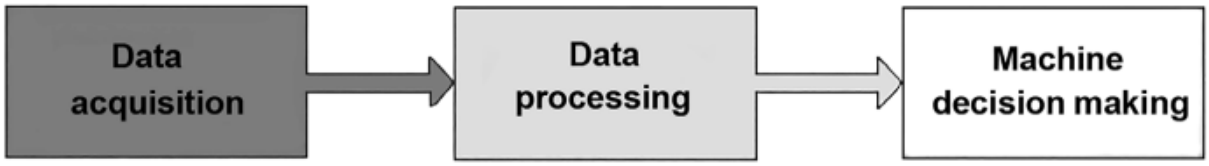

Figure 1.1: Predictive maintenance components

1.3 Objective

Modelling equipment downtime is referred as failure prediction. These models are based on data collected from past failures of a given equipment (or similar ones). **Machine learning is well suited to model current equipment behaviour and its potential breakdowns.** This way, production equipment failures can be anticipated and maintenance can be scheduled before the failure even happens, thus avoiding painful and unnecessary costs.

The intended objectives are:-

- The main objective if this project work is to deploy an Azure machine learning based web service to process predictive analytics for machine failure in factory
- To determine that a piece of equipment is going to break before it actually happens.
- To schedule maintenance ahead of an equipment breakdown in order to prevent downtime.
- To use machine learning algorithms in order to anticipate and predict breakdowns.

1.4 Significance

Predictive Analytics consists of the data processing techniques focusing in solving the problem of predicting future outcome based on analysing previous collected data.

Organizations are increasingly adopting predictive analytics, and adopting these predictive analytics more broadly. Many are now using dozens or even thousands of predictive analytic models. These models are increasingly used in real-time decision making and in operational, production systems.

Manufacturing, Maintenance and Operation Managers can benefit from predictive models. Yet they are not data scientists and may not have the required skills in machine learning nor coding experience to build them from scratch. They collect, in the course of their daily activities, considerable amounts of data. Indeed, most equipment are instrumented with sensors. Therefore, data such as temperature, pressure, moisture, exposition to light, duration of use since the last downtime, are typically collected. Even though often considered as Big Data because they range in the millions of measures over the course of a year for instance, the particular case of failure prediction falls into the Small Data category because it has usually occurred a few dozens or hundreds of times over the past years.

The problem with these old-school approaches is their high cost. Waiting until a component fails means lost production time and revenue. In-person inspections are expensive and can lead to replacing parts unnecessarily, based only on the inspector's best guess. Following the manufacturer's recommended maintenance schedule saves on inspection costs but often results in replacing parts that are still functioning well and could continue to do so.

One solution to decrease the operational cost and to increase the manufacturing system availability is to manage continuously all maintenance activities and to control the degradation to move to predictive maintenance.

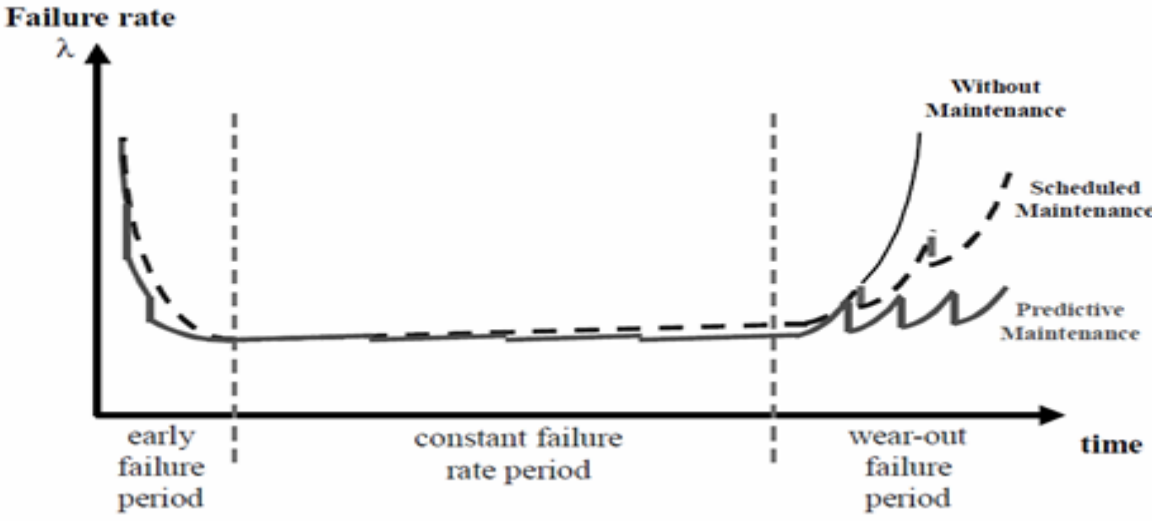

Figure 1.2*: Decreasing of failure rate through predictive maintenance*

1.5 Methodology

The proposed methodology for failure occurrence prediction and the dataset usage proposition are presented in the flowcharts below:-

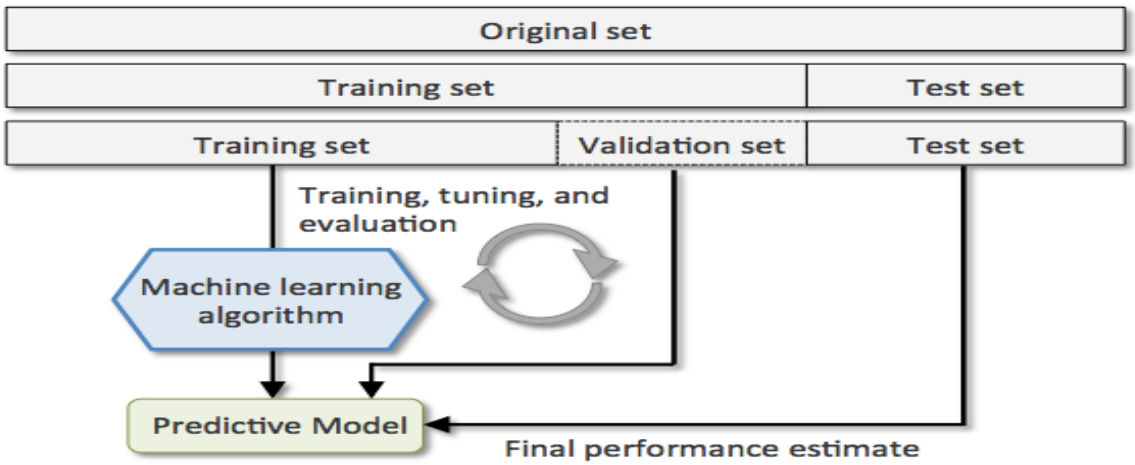

Figure 1.3: Division of dataset

The simulation data set is split in order to obtain separate training and testing data sets. This practice ensures that the system is not trained to remember specific data.

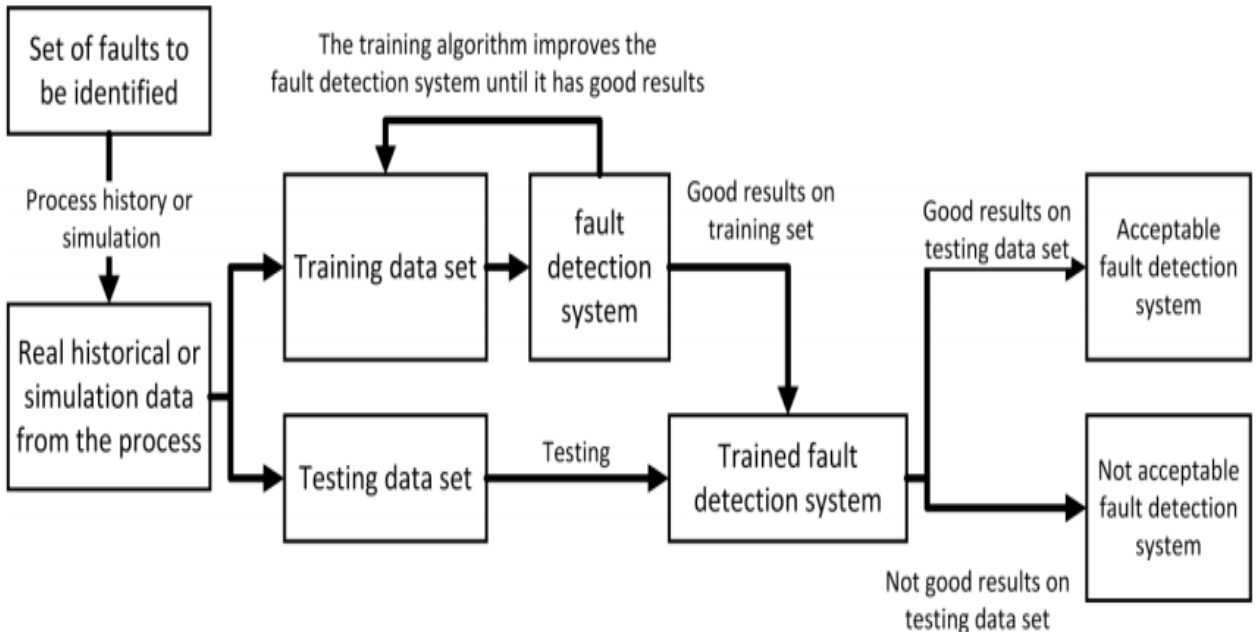

Figure 1.4: Fault detection system

The processing cycle typically involves two phases of processing:

1. Training phase: Learn a model from training data
2. Predicting phase: Deploy the model to production and use that to predict the unknown or future outcome.

Machine learning is a part of Artificial intelligence were machine are empowered to learn by themselves using techniques and tools that allow them to build knowledge from data. The learning process consists of training the model on a dataset with predefined outcomes. This learning method is referred to a supervised, whereas in unsupervised the outcome is not given, and the algorithm tries to predict the best outcome. According to Jar dine, Lin and Banjevic (2006) predicting machine failure are categorised as a classification or regression problem. Classification can be used to detect machine failures modes, while regression techniques can be used in prognosis to calculate machine RUL.

Here we have used classification technique for machine failure detection.

Chapter-2

SYSTEM REQUIREMENT ANALYSIS

2.1 Organization

Machine learning is about extracting knowledge from data in order to create models that can perform tasks effectively.
A typical Machine Learning application consists of the following parts:
• A task and a metric for evaluation. The task comes inherently given a real problem and the metric quantifies effectiveness of a solution.
• A model family that we believe is capable of solving the problem in hand. The selection of the model type depends on several factors, such as the amount of available training data (i.e. size of the given dataset), the complexity of the task and knowledge about its performance on similar problems.
• A dataset on which the best model will be trained in order to solve the task, aiming to the best performance with respect to the evaluation metric.
• A loss function that quantifies the goodness of fit. In contrast to the evaluation metric, the loss function is a differentiable and usually model-specific.
• An optimization algorithm to train the model. The models consist of parameters (usually referred as θ), who's values reflect the performance on the loss function. Thus, an optimization strategy is required in order to select the parameter set that minimizes the loss function.

2.2 Data acquisition

Every failure type detection and predictive maintenance approach begins with the data acquisition part. The acquisition of data is the process of collecting and storing data from a physical process in a system, which is essential for the implementation of a predictive maintenance. Therefore, the decision of the data acquisition technique influences the steps of the work flow. Data acquisition techniques collect information about a real-world system, using different sources.

Data acquisition systems needs to know the type of the data we are dealing with. The two most important types of data for failure type detection and predictive maintenance are data gathered by sensors and data gathered by log files. Here we have chosen sensor data.

In recent years, sensors have become smarter, smaller, easier to implement in existing systems, as well as cheaper and more reliable. A sensor converts physical values into electrical values (voltage, current or resistance). Usually, one sensor measures one mechanical value; for example, the mechanical values of acceleration, pressure, flow, torque and force. With this mechanical value, one can interpret the vibration data, acoustic data, temperature, humidity, weather, altitude, etc.

2.3 Data Description

The dataset taken here for machine failure detection comes from a company that uses many machines to build final products. As production is stopped every time a machine has a failure, management would like to create a predictive model that finds which machine is going to fail next.

As we explored the data, we understood that the company is using 1000 machines. On average, these machines have a failure every 55 weeks. Some of these machines are brand new, others have been running for almost two years. In our dataset, almost 40 % of the machines had a failure in the past two years.

Task type: Binary Classification
Number of columns: 9
Target variable: (broken) Machine actually broken? yes/no.
Weight: Positive class (broken) 40%, Negative class: 60%

The variables are:

- Random
- Machine nbr: from 1 to 1000
- “lifetime” indicates number of weeks since the machine has been used

then we have 3 numeric variables related to

- Temperature
- Pressure
- Moisture

and 2 variables related to

- The team using the machine
- The machine’s provider.

“broken” which is our Goal (Yes or No)

COLUMN NAME	TYPE
Random	Float
Machine nbr	Integer
Lifetime	Integer
pressureInd	Float
temperatureInd	Float
moistureInd	Float
team	Integer
provider	Integer
broken	boolean

Table 2.1 : Dataset description

2.4 Data Preprocessing

Acquired data are susceptible to presenting some missing, inconsistent, and noise values. Data quality has a great impact on the results obtained by data mining techniques. To improve these results, preprocessing methodologies can be applied. Data preprocessing is one of the most critical steps, which deals with the preparation and transformation of the initial dataset. Data preprocessing methods can be divided into three main categories:

• Data cleaning.

• Data transformation.

• Data reduction.

2.4.1. Data Cleaning

Raw data are usually incomplete, noisy, or inconsistent, especially event data, which are manually entered. Errors in data may be caused by many factors including human factors to sensor faults, and detecting and removing those errors improve data quality . Dirty data can cause confusion for the mining procedure, and in general, there is no simple way of cleaning. Some techniques are based on human inspection, which usually is helped by some graphical tool. Mean or median values are typically used to pad unknown values with zeros. In addition to missing data, noisy values are also a problem for data clearance. The work presented by Libralon et al. proposed the use of clustering methods for noise detection. Data outliers can also be detected by clustering techniques, where similar values are organized into groups. Values that are set outside the clusters will be considered as outliers.

Estimate missing values :

If only a reasonable percentage of values are missing, then we can also run simple interpolation methods to fill in those values. However, most common method of dealing with missing values is by filling them in with the mean, median or mode value of the respective feature.

Feature encoding :

Feature encoding is basically performing transformations on the data such that it can be easily accepted as input for machine learning algorithms while still retaining its original meaning. Here we have encoded our target variable which is 'broken' and given it binary values.

Broken(Yes) is encoded as '1' and broken(No) is encoded as '0'.

Identifying and removing outliers:

- Discovering outliers with Boxplot-

  In descriptive statistics, a **box plot** is a method for graphically depicting groups of numerical data through their quartiles. Box plots may also have **lines extending vertically** from the boxes (*whisker*) **indicating variability** outside the upper and lower quartiles, hence the terms box-and-whisker plot and box-and-whisker diagram. **Outliers** may be **plotted** as **individual** points. If there is an outlier it will plotted as point in boxplot but other population will be grouped together and display as boxes.

- Discovering outliers with Scatter Plot-

  A **scatter plot** , is a type of plot or mathematical diagram using Cartesian coordinates to display values for typically two variables for a set of data. The data are displayed as a **collection of points**, each having the value of **one variable** determining the position on the **horizontal** axis and the value of the **other variable** determining the position on the **vertical** axis.

- Discovering outliers with mathematical function(Z-score)-

  The intuition behind Z-score is to describe any data point by finding their relationship with the Standard Deviation and Mean of the group of data points. Z-score is finding the distribution of data where mean is 0 and standard deviation is 1 i.e. normal distribution.

2.4.2 Data Transformation

Data transformation has the aim of obtaining a more appropriate form of the data for one step further in modelling. Transformations can include standardization, where data are scaled to a small range and make different signals comparable. Smoothing is also applied to data to separate the signal and the noise.

2.4.3 Data Reduction

Having a considerable amount of data can be an issue for machine decision making in terms of having a big computational cost. As the number of data increases, the time spent by the hardware will also increase. To maintain the computational cost while the amount of data is sufficient, some methodologies have been developed over the years. The best known one is principal component analysis. This method is based on combining input features linearly to obtain new ones, which are linearly independent of each other and maintain as much of the original information as possible.

2.5 Tools Used in Machine Learning

Tools used in Machine Learning Tools makes machine learning swift and rapid. Machine learning tools provides interface to the machine learning programming language. They provide best practices for process and implementation. Machine learning tools contains platforms which provides capabilities to run a module or project. Examples of platforms of machine learning are:

- Python SciPy subparts such as scikit-learn
- Jupyter Notebook
- VS Code

Machine learning tools contains various libraries which provides all capabilities to complete a project and libraries provides various algorithms. Some of libraries are :

- scikit-learn in Python.
- Pandas
- Numpy
- Matplot
- Seaborn

Chapter-3

PREDICTIVE MODEL CREATION

3.1 Model Selection

Keeping in mind the classification problem, we tried fitting our dataset into various models-

- Support Vector Machine
- Logistic regression
- Decision trees
- Random forest

3.1.1 Support Vector Machine

Support Vector Machine(SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges.
However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes very well.Support Vectors are simply the co-ordinates of individual observation. Support Vector Machine is a frontier which best segregates the two classes (hyper-plane/ line).

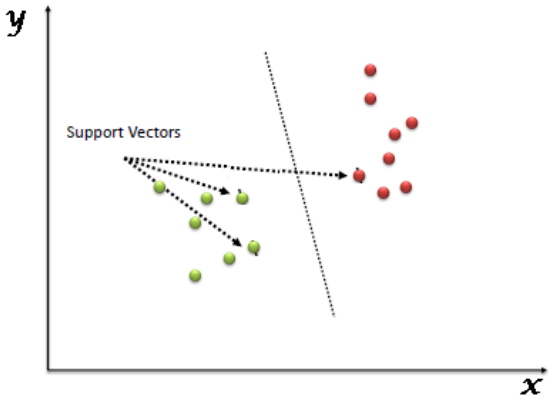

Figure 3.1: Classification using SVM

3.1.2 Logistic Regression

Logistic regression is a statistical method for analysing a dataset in which there are one or more independent variables that determine an outcome. The outcome is measured with a dichotomous variable (in which there are only two possible outcomes). It is used to predict a binary outcome (1 / 0, Yes / No, True / False) given a set of independent variables. To represent binary / categorical outcome, we use dummy variables.

In simple words, it predicts the probability of occurrence of an event by fitting data to a logit function.

In predictive maintenance, tasks like estimating the risk of a system failure, quantifying the condition of a components and predicting missing sensor values can be naturally modeled as regression problems and therefore extensive research and experimentation was performed to select the proper ones for each case

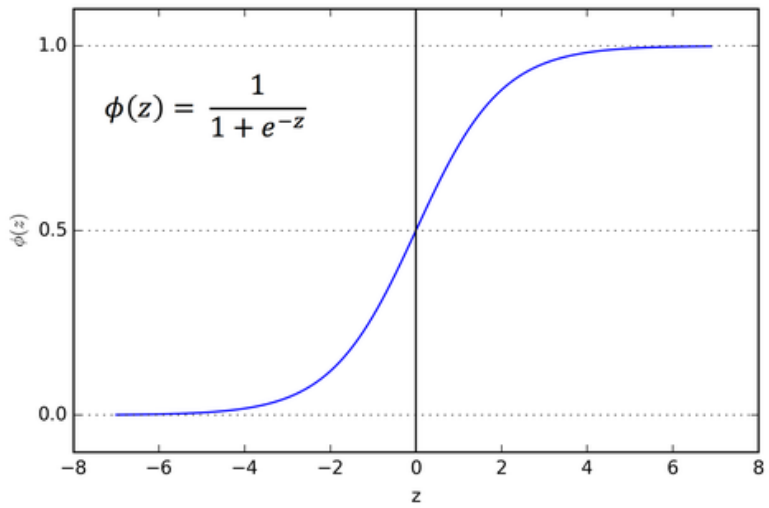

Figure 3.2:Sigmoid activation function

If ‘Z’ goes to infinity, Y (predicted) will become 1 and if ‘Z’ goes to negative infinity, Y (predicted) will become 0.

3.1.3 Decision Trees

A decision tree can be learned as a belief rule based system from a combination of domain expert knowledge and historic data. But all decision-tree based machine learning algorithms mentioned for failure type detection and predictive maintenance follow two steps in creating the decision tree:

• Building a decision tree by learning with the supervised learning technique.

• Pruning the decision tree.

In some algorithms, the both steps are done concurrently. Decision-tree-based machine learning techniques are frequently used for failure type detection and predictive maintenance, but only to classify the state of the real-world system, and not for regressing the residual useful lifetime. The reason is that a decision tree can only have a finite number of leafs, which represent the possible results. Because their number is finite it is not possible to estimate continuous values.

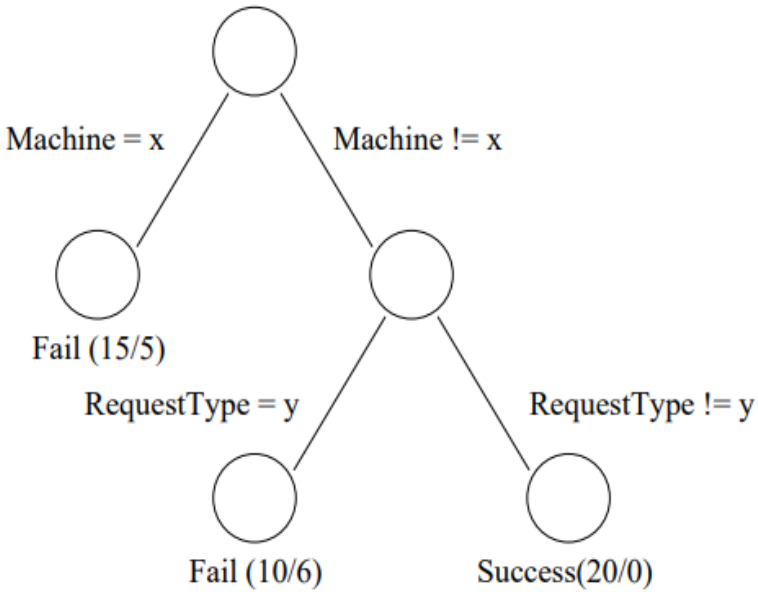

Figure 3.3: An example decision tree for diagnosis

3.1.4 Random Forests

Random forest is an ensemble learning technique that is used both in regression and classification problems. In a regular decision tree, a single decision tree is built. However, in a random forest, many decision trees are built. The number of trees is usually user defined. In an ensemble process, a vote from each decision tree is used in deciding the final class. In this technique, a sample of data with replacement is used for building the decision tree along with the subset of variables. This sampling and subsetting are performed at random. Hence, this technique is called a random forest.

The methodology allows a transition from a time-based to a condition-based maintenance, a reduction of problem complexity and it offers high predictive performance. As the Random Forest approach is free of parametric or distributional assumptions, the method can be applied to a wide range of predictive maintenance problems. This leads to a reduction of tool downtime, maintenance and manpower costs and improves competitiveness in the industry.

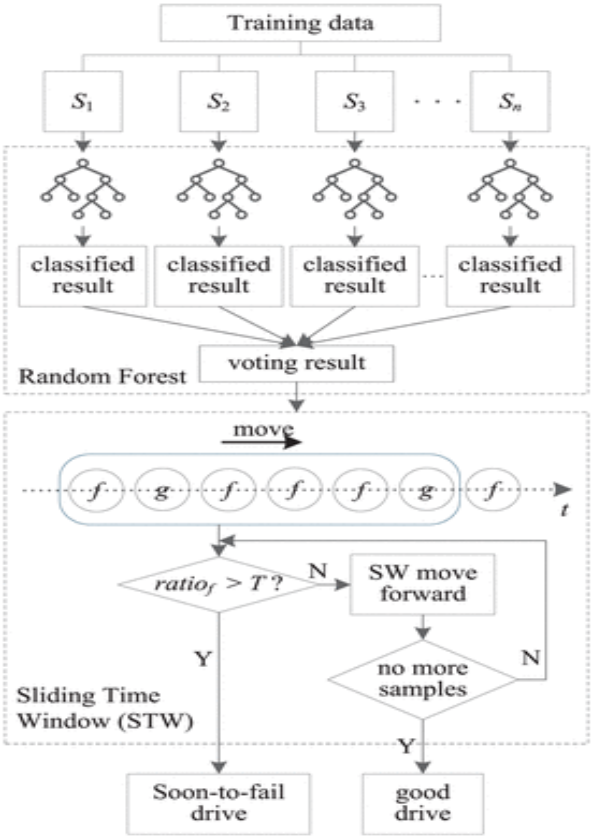

Figure 3.4: RF based failure prediction method

Chapter-4

PREDICTIVE MODEL EVALUATION

4.1 Evaluation Strategy

An evaluation strategy is needed for analysing a failure type detection and predictive maintenance approach. The primary objective of the evaluation step is to measure the performance of the built models using testing data. The evaluation step includes the following activities:

1. Test built models from the Learning and predict step with test.

2. Validate the results based on the provided validation set which contains the calculated result for the test data.

3. Record the testing result of each model using the agreed evaluation measures.

4.2 Evaluation metrics

A good machine learning technique, and especially a failure type detection and predictive maintenance system, should provide accurate and precise estimations. However, it is also important to obtain information from such a system about the reliability of the prediction.

In recent literature, the most commonly used metrics in failure type detection and predictive maintenance are accuracy, precision, mean square error and mean absolute percentage error. This assertion is also valid for failure type detection and predictive maintenance applications.

4.2.1 Model accuracy

Model accuracy in terms of classification models can be defined as the ratio of correctly classified samples to the total number of samples:

$$Accuracy = \frac{\text{number of correct predictions}}{total\ number\ of\ predictions}$$

Generally for classification models one has to create the confusion matrix.

	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

Table 4.1 : Confusion Matrix

True Positive (TP) — A true positive is an outcome where the model *correctly* predicts the positive class.

True Negative (TN)—A true negative is an outcome where the model *correctly* predicts the negative class.

False Positive (FP)—A false positive is an outcome where the model *incorrectly* predicts the positive class.

False Negative (FN)—A false negative is an outcome where the model *incorrectly* predicts the negative class.

For binary classification models, accuracy can be defined as:

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN)$$

4.2.2 Precision and Recall

In a classification task, the precision for a class is the number of true positives (i.e. the number of items correctly labeled as belonging to the positive class) divided by the total number of elements labeled as belonging to the positive class (i.e. the sum of true positives and false positives, which are items incorrectly labeled as belonging to the class).

$$\text{Precision} = TP / (TP + FP)$$

High precision means that an algorithm returned substantially more relevant results than irrelevant ones.

Recall is defined as the number of true positives divided by the total number of elements that actually belong to the positive class (i.e. the sum of true positives and false negatives, which are items which were not labeled as belonging to the positive class but should have been).

$$\text{Recall} = TP / (TP + FP)$$

High recall means that an algorithm returned most of the relevant results.

4.2.3 Root Mean Square Error

RMSE is the square root of the average of prediction square error for the test population represented by equation below, where d represents the error generated from the difference between predicted and true RUL value, and N is the number of units under testing.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} d_i^2}$$

4.2.4 F1 score

The F1 Score is the 2*((precision*recall)/(precision+recall)). It is also called the F Score or the F Measure. Put another way, the F1 score conveys the balance between the precision and the recall.

academic papers where researchers use a higher F1-score as "proof" that their model is better than a model with a lower score. However, a higher F1-score does not necessarily mean a better classifier

4.3 Comparison between approaches

APPROACH	ACCURACY
Support Vector Machine	86%
Logistic Regression	84%
Decision Tree	93.2%
Random Forest	94.21%

Table 4.2 : Comparison of Models

We can clearly see that for the chosen dataset for predictive analysis for machine failure, Random Forests gives the highest accuracy which is 94.21%. Now we will observe random forest approach in more detail emphasizing on its optimisation.

4.4 Hyper-parameter selection for Random Forest

Machine Learning models contain parameters that cannot be optimized during the training phase, such as the regularization term λ in LASSO, C in Support Vector Machines, the number of trees and the maximum depth in random forests.

Parameters that cannot be optimized need to be selected by brute force approaches, such as grid-search. These parameters are known as hyperparameters. In gird-search, one defines the ranges of the hyperparameters in which the search for the best model will be performed. At every step of the search, a combination of the values within the predefined ranges is assigned to

the hyperparameters and a k − f old cross validation is performed. The values of the hyperparameters are closely related to overfitting since they control the complexity of the model. A complex model tends to overfit since it tends to have the capacity to mimic the training set. However, if this complexity is not controlled, the model will perform poorly on the test set.

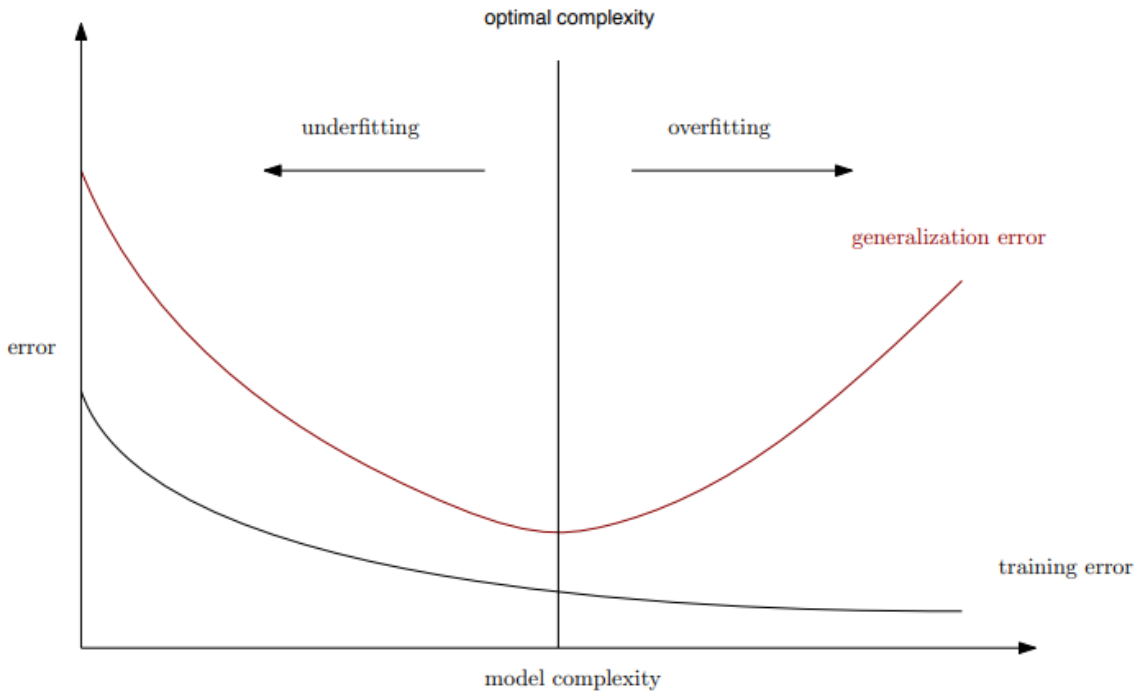

Figure 4.1: Errors and Model Complexity

Consider for example the max_depth parameter in a random forest, which controls the maximum depth which each tree can reach by the iterative splits and thus, the larger the depth the more complex relationships among the features will be used for the split and this may lead to overfitting. In the figure below we present the impact of the max_depth parameter on the train and test errors using our dataset.

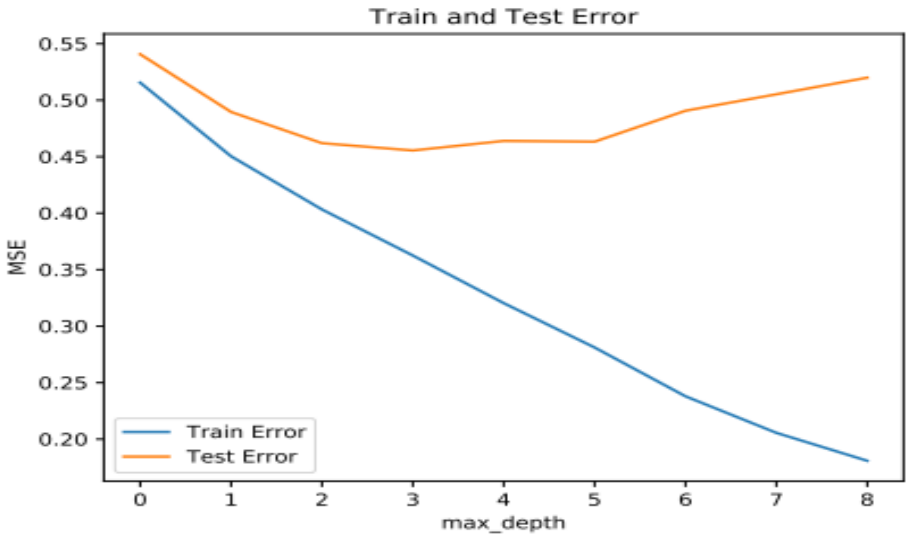

Figure 4.2 : Impact of maximum depth on train and test error

As the figures show, the train error drops by increasing the max-depth whereas the test error has a minimum value when the depth is 4. The figures show that for greater values of the depth the model starts overfitting the data.

4.5 Evaluation metrics for Random Forest

METRICS	SCORE
Accuracy	94.21%
Precision	92.2%
Recall	98.5%
F1-score	95.2%

Table 4.3: Results using Random Forest

	ACTUAL POSITIVE	ACTUAL NEGATIVE
PREDICTED POSITIVE	130	11
PREDICTED NEGATIVE	2	207

Table 4.4: Confusion matrix using Random Forest

Chapter 5
MODEL DEPLOYMENT

5.1 Web Service Deployment

Azure cloud offers software as a service (SaaS), Platform as a Service (PaaS) and Infrastructure as a service (IaaS). These platforms provide multiple integrated cloud services, bundled suites and features including different programming languages, tools, and structures to help enterprises in their cloud journey.

Azure App Service, one of the Azure products, is a fully managed Platform as a Service (PaaS) that provides all the tools and services needed to create reliable and scalable mission-critical Web Apps, Mobile Apps, API Apps, and Logic Apps in a single instance.

5.2 Web Application Deployment

Building/Training a model using various algorithms on a large dataset is one part of the data. But using these model within different application is second part of deploying machine learning in the real world.

To make these models useful, they need to be deployed so that other's can easily access them through an API (application programming interface) to make predictions. This can be done using Flask and Heroku — Flask is a micro web framework that does not require particular tools or libraries to create web applications and Heroku is a cloud platform that can host web applications.

Flask is a Python-based microframework used for developing small scale websites. Flask is very easy to make Restful API's using python. As of now, we have develop a model i.e. `model.pkl` which can predict whether a machine part is broken or not. Now we will design a web application where the user will input all the attribute values and the data will be given the model, based on the training given to the model, the model will predict the what should be the state of the machine i.e. broken or not. Therefore, in order to collect the data we create html form which would contain all the different options to select from each attribute.

Chapter 6
CONCLUSION

6. Conclusion

We presented a novel machine failure prediction method based on the random forest to improve the failure detection accuracy of soon-to-fail machineries. Classification was used for failure prediction of machine parts in a factory. Although the classification approach resulted in perfect prediction accuracy for the dataset used in this preliminary study, such perfect results are not expected when the approach is applied to other equipment failure datasets. More complex approaches of deep learning and neural networks are required for the same.

Chapter 7

FUTURE SCOPE

7. Future Score

1. Applying multiclass classification-
   Rather than just showing Yes/No for the machine failure prediction, the model would predict discrete value taking more than two values (for instance a status of state with values like: On, Risk of breakdown, Down, etc.)
2. Working on time series data and also injecting data from multiple sources into our failure prediction approach.
3. Building a Recommendation System-
   whenever there is a prediction for machine breakdown under the provided conditions, the Web-App would recommend some changes to prevent the same.

Chapter 8
BIBLIOGRAPHY AND REFERECES

8. Bibliography and References

1. ***Failure Diagnosis Using Decision Trees***- *Mike Chen, Alice X. Zheng, Jim Lloyd, Michael I. Jordan, Eric Brewer University of California at Berkeley and eBay Inc.*
2. ***Factor Analysis in Fault Diagnostics Using Random Forest***- *Nagdev Amruthnath* and Tarun Gupta Industrial and Entrepreneurial Engineering, Western Michigan University, Kalamazoo, MI, 49008, USA*
3. ***The Application of Random Forest to Predictive Maintenance*** *Rodney Kizito, Phillip Scruggs, Xueping Li and Reid Kress Industrial and Systems Engineering University of Tennessee, Knoxville, TN 37996, USA*
4. ***Machine Learning Methods for Predicting Failures in Hard Drives: A Multiple-Instance Application*** *Joseph F. Murray, Electrical and Computer Engineering, Jacobs Schools of Engineering University of California, San Diego*