Failure mode and effects analysis based

Risk assessment using classification techniques

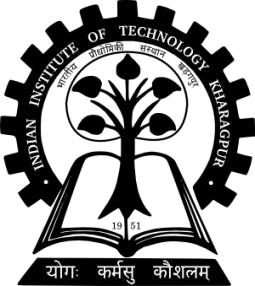
**B. tech Project Report submitted**

**By**

**Koneti Sudheer Kumar**

**(17EC35038)**

Department of electronics and electrical communication engineering



Under the supervision of

Prof. J. Maiti

Department of Industrial and Systems Engineering

Indian Institute of Technology, Kharagpur

**DECLARATION**

I certify that

1. The work contained in this report has been done by me under the guidance of my supervisor.
2. The work has not been submitted to any other Institute for any degree or diploma
3. I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute
4. Whenever I have used materials (data, theoretical analysis, figures, and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the references.

Date: 21/11/2020 Name: Koneti sudheer Kumar.

Place: Kharagpur Roll number: 17EC35038.

**DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING**

**INDIAN INSTITUTE OF TECHNOLOGY, KHARAGPUR**

**KHARAGPUR-721302, INDIA**



**CERTIFICATE**

This is to certify that the project report entitled “**Failure mode and effect analysis based risk assessment using classification techniques**” submitted by **Koneti sudheer Kumar (17EC35038)** to Indian Institute of Technology, Kharagpur towards partial fulfilment of requirements for the award of degree of Dual Bachelor of Technology(Hons.) in Industrial and Systems Engineering is a record of bonafide work carried out by under my supervision and guidance during Autumn Semester 2020-21.

Date: 21/11/2020 Prof J Maiti

Place: Kharagpur Department of Industrial and Systems Engineering,

Indian Institute of Technology Kharagpur

CONTENTS

Page No.

1. Introduction……………………………………… 1

2. Literature Review………………………………... 2

3. Problem Description……………………………... 3

4. Methodology…………………………................... 3-10

4.1. Decision tree Classifier

4.2. Logistic Classifier

4.3. Support Vector Machine Classifier

4.4. Voting Ensemble Classifier

4.5. Trapezoidal Fuzzy Numbers (TrFN)

5. Case Study……………………………………….. 11-13

6. Results and Discussion…………………………... 14-18

7. Conclusion and Future scope……………………. 19

8. References………………………………………... 19

1.INTRODUCTION

There are numerous high-profile examples of product recalls resulting from poorly designed products and/or processes. These failures are debated in the public forum with manufacturers, service providers and suppliers being depicted as incapable of providing a safe product. Failure Mode and Effects Analysis, or FMEA, is a methodology aimed at allowing organizations to anticipate failure during the design stage by identifying all of the possible failures in a design or manufacturing process. Developed in the 1950s, FMEA was one of the earliest structured reliability improvement methods. Today it is still a highly effective method of lowering the possibility of failure.

Failure Mode and Effects Analysis (FMEA) is a structured approach to discovering potential failures that may exist within the design of a product or process. Failure modes are the ways in which a process can fail. Effects are the ways that these failures can lead to waste, defects or harmful outcomes for the customer. Failure Mode and Effects Analysis is designed to identify, prioritize and limit these failure modes. FMEA is not a substitute for good engineering. Rather, it enhances good engineering by applying the knowledge and experience of a Cross Functional Team (CFT) to review the design progress of a product or process by assessing its risk of failure. This analysis has been successfully utilized in a wide range of industries including manufacturing, space research, power generations and automobiles.

An FMEA identifies the opportunities for failure, or “failure modes,” in each step of the process. Each failure mode gets a numeric score that quantifies (a) likelihood that the failure will occur, this is denoted as “o”(occurrence) likelihood that the failure will not be detected, this is denoted by “D” (detection) and (c) the amount of harm or damage the failure mode may cause to a person or to equipment, this is denoted by “S”(severity) .

The product of these three scores is the Risk Priority Number (RPN) for that failure mode. The sum of the RPNs for the failure modes is the overall RPN for the process. These three risk attributes are rated with the numerical scale from 1 to 10 for each failure mode. The RPN is obtained by multiplying these three values which results in RPN ranging from 1 to 1000. This RPN helps in identifying the risk in a failure mode by greater severity rating, higher occurrence and rarer detectability, which in turn may result in the highest risk level.

This FMEA is given in the second section. The third section encompasses research methodology containing the preliminaries related to trapezoidal fuzzy numbers, ensemble techniques in machine learning and case study. The case study further includes explained with the help of a numerical example for failure analysis of rotor brake in an industry. The detailed outcome of the proposed study is discussed in the fourth section. Finally, the last section comprises of the concluding part of the paper along with the limitations of the present work and future scope of study.

2. LITERATURE REVIEW

Considering the fact that three factors of severity (S), occurrence (O) and detection (D) have different weights is important in risk assessments. Logically, the significance of S and O factors are more than D factors for some irreparable systems.

In addition, different combinations of O, S, and D may computationally create the same values of RPN. Many calculation problems in conventional FMEA depend on lack of accurately quantified values and trust just in the experience of Decision makers. However, there are issues and complexity with the Decision makers.

So, there are many improvements made one of them is introducing fuzzy set theory to handle the qualitative and imprecise information the evaluation of the risk attributes by experts are expressed in terms of linguistic variables rather than the exact numbers, Multi-criteria decision making (MCDM) methods are integrated with fuzzy set theory to improve the traditional FMEA (Garcia, P.A., Schirru, R., 2005.). The fuzzy values thus assigned to the linguistic terms are then converted to crisp numbers using some mathematical based techniques.

Recently, so many researchers have done crucial developments in the field of FMEA based risk assessment where the researchers have considered different weights to each of the attributes in calculating RPN and the weights are chosen based on the logistic regression (Mandal and others (2020)).So, we aimed to use machine intelligence in predicting the probability of failure mode and compare the results with previous work which was done using interval-based technique. In this we have used three different classifiers decision tree classifier where the data is classified based on an attribute (Patel, Harsh & Prajapati, Purvi. (2018)) and the other one is the logistic classifier which is a binary classifier and classify the things based on the probabilities of the attributes (Sammut C., Webb G.I. 2011) and the other classifier is the support vector machine classifier which classifies the data based on plane which is drew based on the square error between the points (Zhang Y. (2012)) and we also implemented ensemble technique to improve the accuracies of predicting the failure modes and reporting the probability of each failure mode.

3. PROBLEM DESCRIPTION

In FMEA based risk assessment calculating RPN from three risk attributes Severity(S), Occurrence(O), Detection(D), the score obtained by multiplying these factors are used to failure mode is assessment. We have collected opinions from three experts they have identified some potential failure modes based on their past experiences. These failure modes are then classified based on the risk attributes using different classification techniques and the probabilities of each failure mode are calculated and then predicted whether it is a failure or not.

4. METHODOLGY

We have used three types classification techniques to predict the probabilities of the failure modes they are:

1. Decision tree classifier

2. Logistic model.

3.Support Vector Machine Classifier.

These three models are then combined by voting classifier which uses voting method to combine these three heterogenous models based on the accuracies produced by each model. All these models are explained in detail in coming sections.

4.1. DECISION TREE CLASSIFIER

A Decision Tree is a simple representation for classifying examples. It is a Supervised Machine Learning where the data is continuously split according to a certain parameter.

Decision Tree consists of**:**

1. Nodes: Test for the value of a certain attribute.
2. Edges/ Branch: Correspond to the outcome of a test and connect to the next node or leaf.
3. Leaf nodes: Terminal nodes that predict the outcome (represent class labels or class distribution).

**There are two main types of Decision Trees:**

1. Classification Trees.

2. Regression Trees.

**1. Classification trees**(Yes/No types)**:**

What we’ve seen above is an example of classification tree, where the outcome was a variable like ‘fit’ or ‘unfit’. Here the decision variable is **Categorical/ discrete**.

Such a tree is built through a process known as **binary recursive partitioning.**

**2. Regression trees** (Continuous data types):

Decision trees where the target variable can take **continuous values** (typically real numbers) are called **regression trees**.

Pseudo code for Decision tree classifier:

1. Using the decision algorithm, we start at the tree root and split the data on the feature that results in the **largest information gain (IG)** (reduction in uncertainty towards the final decision).
2. In an iterative process, we can then repeat this splitting procedure at each child node **until the leaves are pure**. This means that the samples at each leaf node all belong to the same class.
3. In practice, we may set a **limit on the depth of the tree to prevent overfitting**. We compromise on purity here somewhat as the final leaves may still have some impurity.

4.2. LOGISTIC CLASSIFICATION MODEL

The logistic classification model (or logit model) is a binary classification model in which the conditional probability of one of the two possible realizations of the output variable is assumed to be equal to a linear combination of the input variables, transformed by the logistic function.

Consequently, we are trying to **classify**the variable as **one**value or **zero**value, which means **logistic regression is a classification algorithm.** The outcomes/predictions of Logistic Regression are always between 0 and 1. They can't be less than 0 and can't be greater than 1. we need to define a function for logistic regression as well. This function g is known as "**sigmoid function**", or "**logistic function".** The classification algorithm got its name because of the term logistic function. As a trivial fact, both sigmoid and logistic function means the same. When we combine the model equation as well as the g(z) equation, we get an expression like this.

Using this function, we need to select the right values for the parameter **theta** and find the value of . This expression will help us in classifying the values present in the dataset.

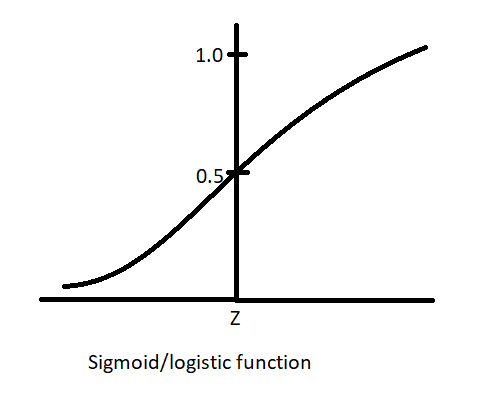


FIGURE-1 Graphical representation of Sigmoid Function

Since the value needs to be strictly between 0 and 1, the function starts at **0** and goes all the way up to **1**. In between, it rises up to **0.5** and then goes flat up to 1. This basically means that the function **asymptotes at 0 and 1.**

For logistic regression the cost function is defined as:

With optimizations in place we can rewrite the cost function as

To minimize the cost function, we have to run gradient decent

We have to update the values of the theta by

Until the convergence is observed.

4.3 SUPPORT VECTOR MACHINE CLASSIFIER

The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space (N — the number of features) that distinctly classifies the data points.

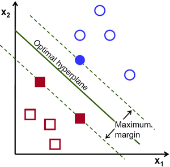


FIGURE-2 Possible hyperplanes

To separate the two classes of data points, there are many possible hyperplanes that could be chosen. Our objective is to find a plane that has the maximum margin, i.e. the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence.

Hyperplanes and Support Vectors:

Hyperplanes are decision boundaries that help classify the data points. Data points falling on either side of the hyperplane can be attributed to different classes. Also, the dimension of the hyperplane depends upon the number of features. If the number of input features is 2, then the hyperplane is just a line. If the number of input features is 3, then the hyperplane becomes a two-dimensional plane. Support vectors are data points that are closer to the hyperplane and influence the position and orientation of the hyperplane. Using these support vectors, we maximize the margin of the classifier. Deleting the support vectors will change the position of the hyperplane. These are the points that help us build our SVM.

## Cost Function and Gradient Updates:

In the SVM algorithm, we are looking to maximize the margin between the data points and the hyperplane. The loss function that helps maximize the margin is hinge loss.

Hinge loss function (function on left can be represented as a function on the right)

The cost is 0 if the predicted value and the actual value are of the same sign. If they are not, we then calculate the loss value. We also add a regularization parameter lambda the cost function. The objective of the regularization parameter is to balance the margin maximization and loss. After adding the regularization parameter, the cost functions look as below.

Now that we have the loss function, we take partial derivatives with respect to the weights to find the gradients. Using the gradients, we can update our weights

When there is no misclassification, i.e. our model correctly predicts the class of our data point, we only have to update the gradient from the regularization parameter.

When there is a misclassification, i.e. our model makes a mistake on the prediction of the class of our data point, we include the loss along with the regularization parameter to perform gradient update.

Now these three classifiers are combined to build a voting classifier.

4.4 VOTING ENSEMBLE CLASSIFIER

Ensemble methods is a machine learning technique that combines several base models in order to produce one optimal predictive model.

Ensemble methods can be divided into two groups:

1. *sequential* ensemble methods where the base learners are generated sequentially (e.g. AdaBoost). The basic motivation of sequential methods is toexploit the dependence between the base learners**.** The overall performance can be boosted by weighing previously mislabelled examples with higher weight.
2. *parallel* ensemble methods where the base learners are generated in parallel.  
   The basic motivation of parallel methods is to exploit independence between the base learners since the error can be reduced dramatically by averaging.

Most ensemble methods use a single base learning algorithm to produce homogeneous base learners, i.e. learners of the same type, leading to homogeneous ensembles. There are also some methods that use heterogeneous learners, i.e. learners of different types, leading to heterogeneous ensembles. In order for ensemble methods to be more accurate than any of its individual members, the base learners have to be as accurate as possible and as diverse as possible.

In this we used voting ensemble the Voting Ensemble estimates multiple base models and uses voting to combine the individual predictions to arrive at the final ones. However, the key difference lies in the base estimators. Models such as Voting Ensemble (and Stacking Ensemble) do not require the base models to be homogenous. In other words, we can train different base learners, for example, a Decision Tree and a Logistic Regression, and then use the Voting Ensemble to combine the results.

Hard: the final class prediction is made by a majority vote — the estimator chooses the class prediction that occurs most frequently among the base models.

Soft: the final class prediction is made based on the average probability calculated using all the base model predictions. For example, if model 1 predicts the positive class with 70% probability, model 2 with 90% probability, then the Voting Ensemble will calculate that there is an 80% chance of the observation belonging to the positive class and choose the positive class as the prediction.

Additionally, we can use custom weights to calculate the weighted average. This is apt for cases in which we put more trust in some models, but still want to consider the ones we trust less. One thing to bear in mind is that in order to use soft voting, all the base models must have the predict proba method. Soft voting can result in better performance than hard voting (but not necessarily), as by averaging the probabilities it “gives more weight” to the confident votes.

The above-mentioned voting schemes are only valid for a classification problem (both binary and multi-class). In scikit-learn there is a separate voting estimator for regression problems (Voting Regressor), which uses the average of the base estimators’ predictions as the final prediction.

The Voting Ensemble is a useful technique, which comes especially handy when a single model shows some kind of bias. It is also possible that the Voting Ensemble results in a better overall score than the best of the base estimators, as it aggregates the predictions of multiples models and tries to cover for potential weaknesses of the individual models.

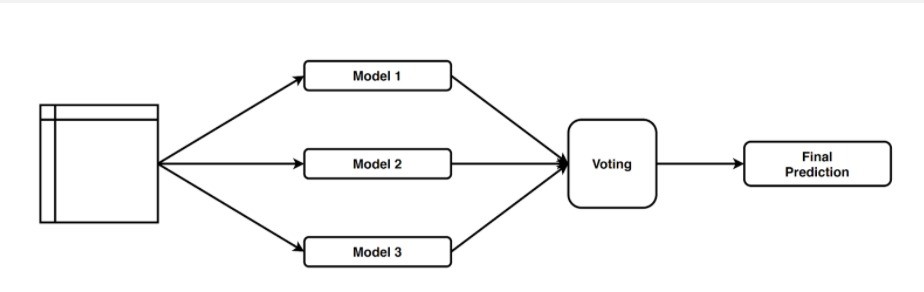


FIGURE-3 Flow chart of Voting Ensemble classifier

4.5. TRAPEZOIDAL FUZZY NUMBERS

A fuzzy number is a quantity whose value is imprecise, rather than exact as is the case with “ordinary” (single- valued) numbers. Any fuzzy number can be thought of as a function whose domain is a specified set (usually the set of real numbers, and whose range is the span of non-negative real numbers between, and including, 0 and 1000. Each numerical value in the domain is assigned a specific “grade of membership” where 0 represents the smallest possible grade, and 1000 is the largest possible grade.

In the fuzzy theory, triangular and trapezoidal fuzzy numbers are used extensively, whose membership functions are, respectively, defined by

For brevity, triangular and trapezoidal fuzzy numbers are often denoted as A= (a1, a2, a3, a4) and B= (a1, a2, a3, a4) and Clearly, triangular fuzzy numbers are special cases of trapezoidal fuzzy numbers with a2=a3.

The detailed methodology is shown in the flow chart below

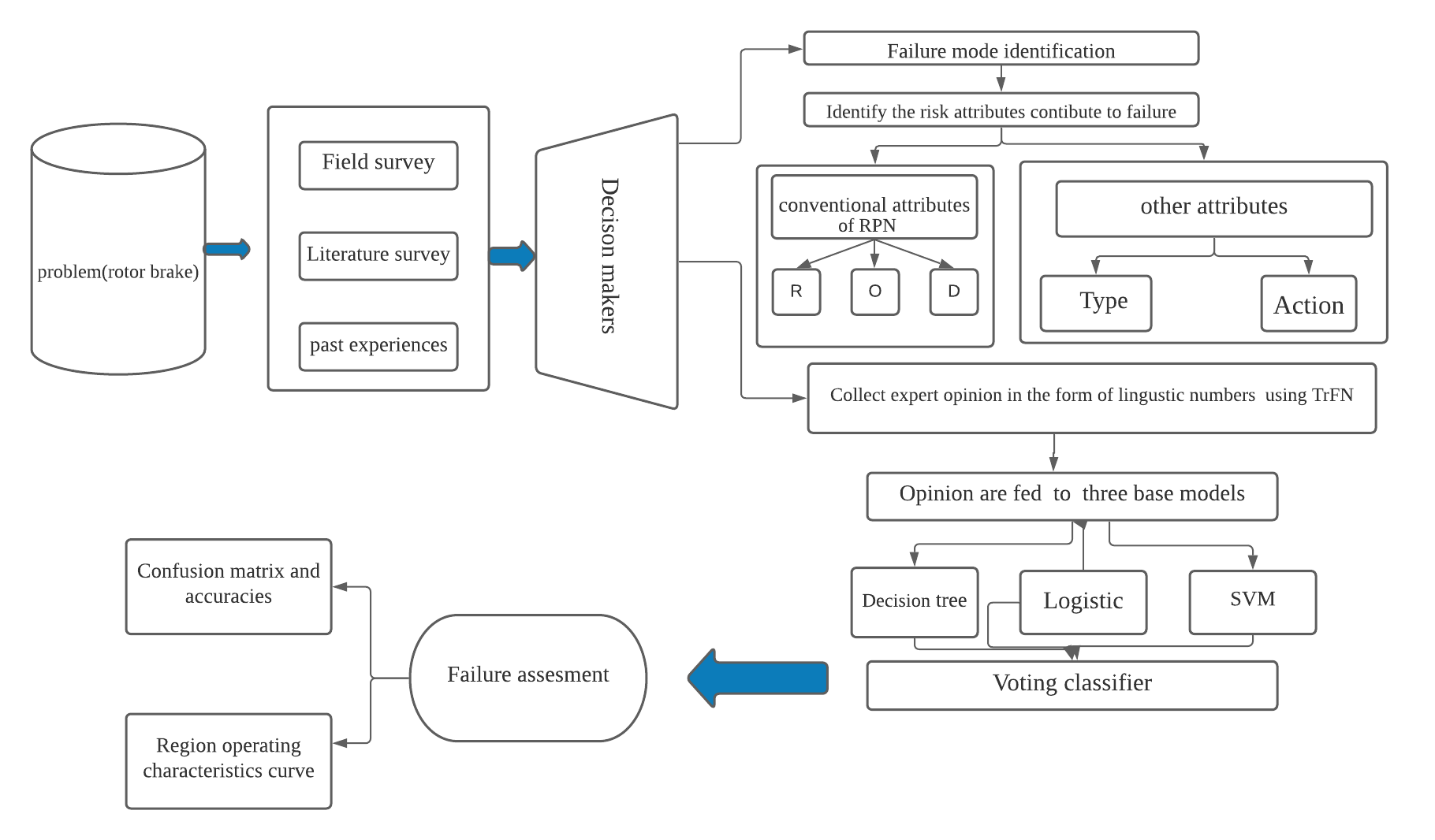


FIGURE-4 A Pictorial Representation of the work

5.CASE STUDY

In an automotive industry, while checking the flaws of a vehicle which leads to an accident of the vehicle is failure in the brake system. The Brake rotors are an import­ ant component in the braking system that stops your vehicle. Brake rotors (they're also called brake discs) are what your vehicle's brake pads clamp down on to stop the wheels from spinning. The entire system is assessed using FMEA techniques in industry so we have collected the data from three decision makers. We have divided the entire process into three stages

*Stage 1*: Data acquisition.

*Stage 2:* Converting linguistic terms to numerical values using trapezoidal fuzzy numbers.

*Stage 3:* Feeding values into three base model and building ensemble classifier.

Stage 1: Data acquisition:

We have collected the data from a team of three decision makers (DM1, DM2, DM3) who have knowledge on the rotor brake system and We have asked them to list all the potential failure modes and We have suggested them not to give any weightage to the failure attributes and they are asked to give the rating according to the TrFN in linguistic terms. The evaluation criteria for severity is listed in the table 1 and also the evaluation criteria for both the occurrence and the detection were also depicted in the tables 2 and 3. respectively. In conventional RPN only these three attributes are taken but as mentioned earlier that in practical situations I also have some other components which effect the system.

TABLE 1 TABLE 2

|  |  |  |
| --- | --- | --- |
| Linguistic variable | Failure severity% | TrFNs |
| Very Poor ( VP) | 15% | (0,0,0.1,0.2) |
| Poor (P) | 30% | (0.1,0.2,0.2,0.3) |
| Medium Poor (MP) | 45% | (0.2,0.3,0.4,0.5) |
| Medium (M) | 50% | (0.4,0.5,0.5,0.6) |
| Medium Good (MG) | 65% | (0.5,0.6,0.7,0.8) |
| Good (G) | 80% | (0.7,0.8,0.8,0.9) |
| Very Good (VG) | 100% | (0.8,0.9,1.0,1.0) |

FMEA “Severity” evaluation criteria FMEA “Occurrence” evaluation criteria

|  |  |  |
| --- | --- | --- |
| Linguistic variable | Mean time failure | TrFNs |
| Very Poor (VP) | 3 years | (0,0,0.1,0.2) |
| Poor (P) | 2 years | (0.1,0.2,0.2,0.3) |
| Medium Poor (MP) | 1.5 years | (0.2,0.3,0.4,0.5) |
| Medium (M) | 1 year | (0.4,0.5,0.5,0.6) |
| Medium Good (MG) | 6 months | (0.5,0.6,0.7,0.8) |
| Good (G) | 3 months | (0.7,0.8,0.8,0.9) |
| Very Good (VG) | 1 month | (0.8,0.9,1.0,1.0) |

So, we have considered other two attributes one is the type of component and action. Totally 67 components are identified by the experts and they have categorized the components into basic, secondary, consumables. The score is shown in the table (2). Another important attribute that was taken into consideration is the type of action to be taken for that failure mode. I have to either repair /replace the component. this was decided by the decision makers with their experience because any false decision can make an organization loss. The action suggested as 0 for repair and 1 for replace it is listed the table (5).

TABLE 3 TABLE 4

FMEA “Detection” evaluation criteria Component evaluation criteria

|  |  |  |
| --- | --- | --- |
| Linguistic variable | Failure detection% | TPFNs |
| Very Poor ( VP) | 100% | (0,0,0.1,0.2) |
| Poor (P) | 80% | (0.1,0.2,0.2,0.3) |
| Medium Poor ( MP) | 65% | (0.2,0.3,0.4,0.5) |
| Medium (M) | 50% | (0.4,0.5,0.5,0.6) |
| Medium Good (MG) | 45% | (0.5,0.6,0.7,0.8) |
| Good (G) | 30% | (0.7,0.8,0.8,0.9) |
| Very Good (VG) | 15% | (0.8,0.9,1.0,1.0) |

|  |  |
| --- | --- |
| Type of component | Score |
| Basic | 1 |
| Secondary | 2 |
| Consumable | 3 |

|  |  |
| --- | --- |
| Type of action | Score |
| Replace | 1 |
| Repair | 0 |

TABLE-5

Action evaluation criteria

*stage 2:* Converting Linguistic terms using trapezoidal fuzzy numbers.

The decision makers are asked to express their opinion in the form linguistic variables and then we have rated them using trapezoidal fuzzy numbers and then we have converted them to single numerical value which is called a crisp number. I have different ways of converting TPFN’s to crisp numbers one is by taking centroid of all the values, we have taken average of all the values and resulting number was replaced in the place of Severity(S), Occurrence (O), detect

Stage 3: Feeding values into three base model and building ensemble classifier.

We have used voting ensemble classifier which uses three base models for voting as mentioned above we have used three base model’s decision tree, logistic and support vector machine. These three models are the averaged and given weightage according to the voting. The data acquired from stage 2 is portioned to training data and test data and then it is fed to three base models which is used to predict probabilities of the failure modes and the accuracies of the testing data and training data. Roc curve of every model is shown later. These three base models are voted and their means are averaged and final probabilities are calculated and roc curve and accuracies for the voting classifiers are also calculated.

TABLE 6

Tabulation of some Failure modes for illustration purpose

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| FAILURE MODES | COMPONENTS | OCCURANCE | SEVERITY | DETECTION | ITEM TYPE | ACTION | FALIURE |
| FM1 | Lever | MP | P | VH | 1 | 0 | 1 |
| FM2 | Stud | MP | P | H | 2 | 0 | 1 |
| FM3 | Knob | MH | H | H | 1 | 1 | 0 |
| FM4 | Spring Ring (1) | VH | p | VH | 1 | 2 | 1 |
| FM5 | Pawl | H | M | P | 1 | 1 | 0 |
| FM6 | Pin (1) | MH | P | P | 1 | 0 | 0 |
| FM7 | Bush (1) | M | P | M | 2 | 1 | 1 |
| FM8 | Special Washer | H | M | VP | 1 | 2 | 0 |
| FM9 | Plunger | M | H | MP | 1 | 0 | 1 |
| FM10 | Spring (1) | VH | p | VH | 1 | 2 | 1 |

6. RESULTS AND DISCUSSION

CONFUSION MATRIX AND ACCURACIES:

Confusion matrix is a performance measurement for machine learning classification problems where output can be two or more classes. The figure 5 below is a different combination of predicted and actual values. It is extremely useful for measuring Recall, Precision, Specificity, Accuracy and most importantly AUC-ROC Curve.

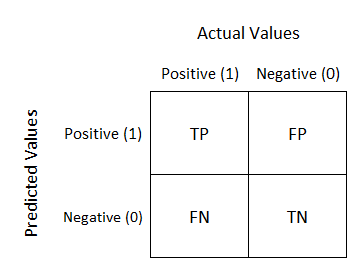


FIGURE-5 Confusion matrix

Here TP, FP, FN, TN are the elements of the confusion matrix.

TP-True Positives- refers to the number of observations correctly assigned to the positive class.

FP-False Positives- the number of observations correctly assigned to the negative class.

FN-False Negatives- the number of observations assigned by the model to the positive class, which in reality belong to the negative class.

TN-True Negatives- the number of observations assigned by the model to the negative class, which in reality belong to the positive class.

The confusion matrix for all the base model were calculated the confusion matrix for the both training and test data of decision tree is listed in table (7), table (8) respectively. With the help of confusion matrix, the accuracy for the model is calculated and for the training data it is observed to be 94.9% and for the testing data it is 77.5%next base model is the logistic classifier the confusion matrix for both the testing and training data were tabulated in table (9), table (10). the accuracy was calculated and it turns out for the training data 69.6% and for the testing data 65%.

The next base model is SVM the confusion matrix is shown in table (11) and table (12). the accuracies for the training data and testing data are 78.7% and 72.5% respectively. For the final voting ensemble model, the confusion matrix for the training and testing data is shown in table (13) and table (14) respectively. The final accuracy is calculated to be 93% for the testing data and 82% for the training data.

TABLE 7 TABLE 8

|  |  |  |
| --- | --- | --- |
|  | Actual |  |
| Predicted | 0 | 1 |
| 0 | 16 | 5 |
| 1 | 4 | 15 |

Confusion matrix of training data of decision tree Confusion matrix of test data of decision tree

|  |  |  |
| --- | --- | --- |
|  | Actual |  |
| Predicted | 0 | 1 |
| 0 | 65 | 2 |
| 1 | 6 | 85 |

TABLE-9 TABLE-10

Confusion matrix of training data of logit model Confusion matrix of test data of logit model

|  |  |  |
| --- | --- | --- |
|  | Actual |  |
| Predicted | 0 | 1 |
| 0 | 33 | 34 |
| 1 | 14 | 77 |

|  |  |  |
| --- | --- | --- |
|  | Actual |  |
| Predicted | 0 | 1 |
| 0 | 8 | 13 |
| 1 | 1 | 18 |

TABLE 11 TABLE 12

Confusion matrix of training data of SVM Classifier Confusion matrix of test data of SVM classifier

|  |  |  |
| --- | --- | --- |
|  | Actual |  |
| Predicted | 0 | 1 |
| 0 | 15 | 6 |
| 1 | 5 | 14 |

|  |  |  |
| --- | --- | --- |
|  | Actual |  |
| Predicted | 0 | 1 |
| 0 | 52 | 15 |
| 1 | 19 | 72 |

TABLE-13 TABLE-14

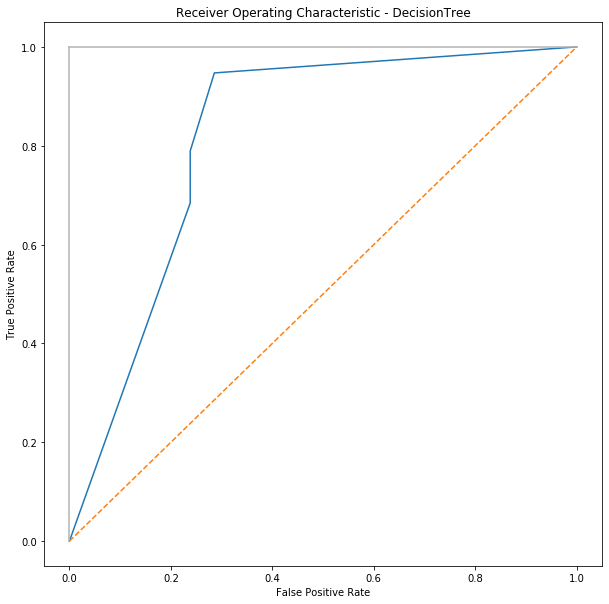
Confusion matrix of training data of voting classifier Confusion matrix of test data of voting classifier

|  |  |  |
| --- | --- | --- |
|  | Actual |  |
| Predicted | 0 | 1 |
| 0 | 61 | 6 |
| 1 | 5 | 86 |

|  |  |  |
| --- | --- | --- |
|  | Actual |  |
| Predicted | 0 | 1 |
| 0 | 15 | 6 |
| 1 | 1 | 18 |

RECEIVER OPERATING CHARACTERISTIC CURVE.

A receiver operating characteristic curve, or ROC curve, is a graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied. The method was originally developed for operators of military radar receivers, which is why it is so named. The ROC curve is created by plotting the True positive rate (TPR) against the False positive rate (FPR) at various threshold settings. The true-positive rate is also known as sensitivity, recall or probability of detectionin machine learning. The false-positive rate is also known as probability of false alarm and can be calculated as (1 − specificity). It can also be thought of as a plot of the power as a function of the Type 1 error of the decision rule (when the performance is calculated from just a sample of the population, it can be thought of as estimators of these quantities). The ROC curve is thus the sensitivity or recall as a function of fallout. Roc curves for all the three base model were shown in the following figures.

 FIGURE-6 Receiver Operating Characteristics Curve of Decision Tree

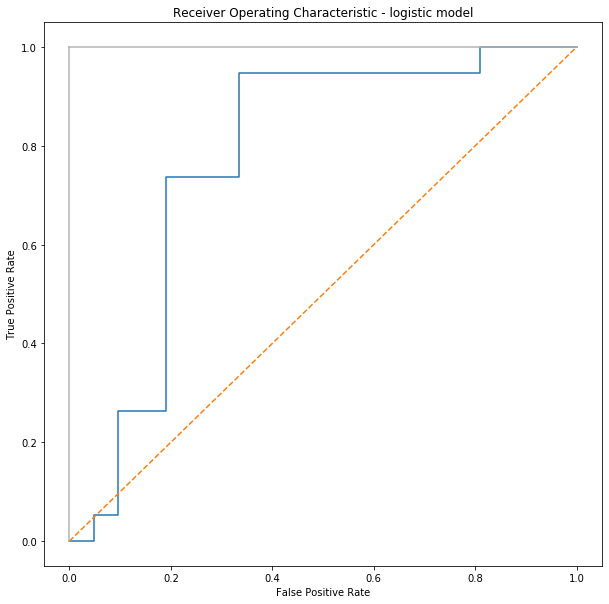


FIGURE-7 Receiver Operating Characteristics curve of logit model

6.1. COMPARISON

I have compared the results with the past works done using the interval number based logistic regression where they have converted interval values to crisp number and then the values are fed into logit model and accuracy was mentioned to be 81.98% and the accuracy for this model was calculated to be 93.3% .ROC curve for previous work was shown in the (Pushparenu Bhattacharjee, Vidyut Dey, U.K. Mandal,2019).ROC curve for the voting classifier is shown below.

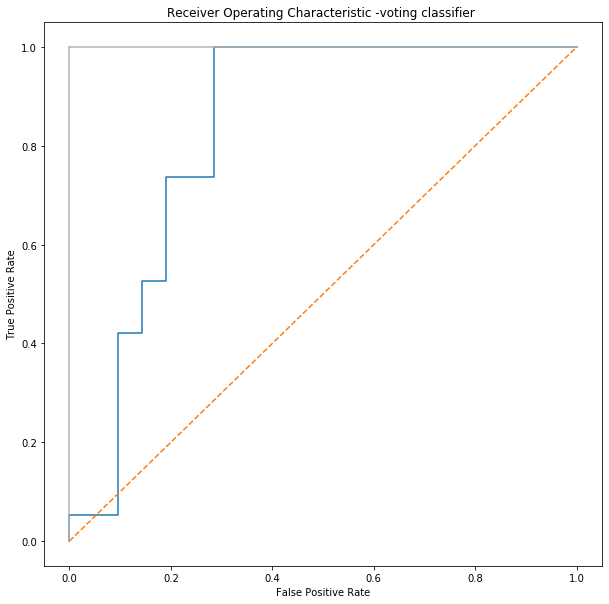


FIGURE-6 Receiver operating characteristics curve of voting ensemble classifier

TABLE 15

Predicted Probabilities of some failure modes

|  |  |  |
| --- | --- | --- |
| FAILURE MODES | FALIURE | PROBABILITIES |
| FM34 | 1 | 0.843364 |
| FM4 | 0 | 0.372403 |
| FM18 | 0 | 0.269641 |
| FM11 | 0 | 0.1882 |
| FM35 | 0 | 0.1882 |
| FM51 | 1 | 0.846751 |
| FM27 | 1 | 0.824708 |
| FM59 | 1 | 0.764331 |
| FM182 | 0 | 0.29981 |

7.CONCLUSION AND FUTURE SCOPE

We limited our study to predict the probabilities using much advanced technique of classifiers we collected opinions from a team of three decision makers on the potential failure modes we have used trapezoidal fuzzy numbers to avoid the linguistic words. Converted the linguistic terms to numerical values and the values are taken classified into training and testing data then these are fed to three different models and then weightage has been given according to accuracies and the resultant classifier is taken to predict the failure modes.

The accuracies for all the three models were calculated to be 94%,69%and 75%.The accuracy of the final classifier resulted to 93.3%.The f-score for the resultant classifier is calculated to be 0.837.The Individual probabilities are shown in the table where for the class 1 the probability is almost above 0.6 and for the class 0 the probability is below 0.5 so we can conclude that our threshold limit can be 0.6 for classifying the failure modes.

But in future research works we can include or consider the weights of the different failure modes .It may lead to rank the failure modes and we can prioritize our investments and work in getting right things done .Thus I eliminated the conventional approach of FMEA in calculating RPN by multiplying the three risk attributes (Severity, Occurrence, Detection.).Also I have not considered other risk attributes which may also effect a particular component that may be considered

REFERENCES

Can, G.F., 2018. An intuitionistic approach based on failure mode and effect analysis for prioritizing corrective and preventive strategies. Hum. Factors Ergon. Manuf. Serv.Ind. 28 (3), 130–147.

Franceschini, F., Galetto, M., 2001. A new approach for evaluation of risk priorities of failure modes in FMEA. Int. J. Prod. Res. 39 (13), 2991–3002.

Garcia, P.A., Schirru, R., 2005. A fuzzy data envelopment analysis approach for FMEA.

Prog. Nucl. Energy 46 (3–4), 359–373.

Zhang Y. (2012) Support Vector Machine Classification Algorithm and Its Application. In: Liu C., Wang L., Yang A. (eds) Information Computing and Applications. ICICA 2012. Communications in Computer and Information Science, vol 308. Springer, Berlin, Heidelberg.

Patel, Harsh & Prajapati, Purvi. (2018). Study and Analysis of Decision Tree Based Classification Algorithms. International Journal of Computer Sciences and Engineering. 6. 74-78. 10.26438/ijcse/v6i10.7478.

(2011) Logistic Regression. In: Sammut C., Webb G.I. (eds) Encyclopedia of Machine Learning. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-30164-8\_493.