Immagine che contiene testo, Carattere, logo, emblema

Descrizione generata automaticamente

Data Mining and Machine Learning

**Machine Failure Classifier**

Francesco De Vita

Index

[1. Introduction 3](#_Toc159085692)

[2. Dataset 3](#_Toc159085693)

[3. Data Pre-Processing 5](#_Toc159085694)

[3.1 Attribute Creation 5](#_Toc159085695)

[3.2 Train-Test split 6](#_Toc159085696)

[3.3 Outlier Detection 6](#_Toc159085697)

[3.4 Feature Selection 7](#_Toc159085698)

[3.5 Normalization 7](#_Toc159085699)

[4. Evaluation 8](#_Toc159085700)

[KNN 8](#_Toc159085701)

[Decision Tree 9](#_Toc159085702)

[RandomForest 9](#_Toc159085703)

[Bayesian Classifier 10](#_Toc159085704)

[4.1 Results 10](#_Toc159085705)

[5. Comparison with other Studies 11](#_Toc159085706)

[6. User Interface 13](#_Toc159085707)

[7. Future Improvements 14](#_Toc159085708)

# 1. Introduction

This project aims to apply data analysis and machine learning techniques for fault prediction in industrial production, using a real dataset from a metal component manufacturing process. The dataset contains information on various physical and chemical parameters of the process, such as temperature, rotation speed, torque, tool wear and machine failure types. The goal of the project is to train a machine learning model to detect damage, classifying it into different types of failures.

A graphical user interface application was also developed to test the trained model.

# 2. Dataset

The dataset we are using is called ‘ai4i2020’ and comes from an industrial production process of metal components.   
The dataset contains 10 000 rows and 14 columns, each of which represents a feature of the process, such as air temperature, rotational speed, torque, tool wear and machine failure types. Our goal is to use these features to predict failures and classify them into different types, such as tool wear, overheating, overvoltage, overload and random failures. The dataset is synthetic, but reflects the real data that is encountered in industry.

1. UID: unique identifier ranging from 1 to 10000
2. product ID: consisting of a letter L, M, or H for low (50% of all products), medium (30%) and high (20%) as product quality variants and a variant-specific serial number
3. type: just the product type L, M or H from column 2
4. air temperature [K]: generated using a random walk process later normalized to a standard deviation of 2 K around 300 K
5. process temperature [K]: generated using a random walk process normalized to a standard deviation of 1 K, added to the air temperature plus 10 K.
6. rotational speed [rpm]: calculated from a power of 2860 W, overlaid with a normally distributed noise
7. torque [Nm]: torque values are normally distributed around 40 Nm with a SD = 10 Nm and no negative values.
8. tool wear [min]: The quality variants H/M/L add 5/3/2 minutes of tool wear to the used tool in the process.
9. a 'machine failure' label that indicates, whether the machine has failed in this particular datapoint for any of the following failure modes are true.

The machine failure consists of five independent failure modes

1. tool wear failure (TWF): the tool will be replaced of fail at a randomly selected tool wear time between 200 - 240 mins (120 times in our dataset). At this point in time, the tool is replaced 69 times, and fails 51 times (randomly assigned).
2. heat dissipation failure (HDF): heat dissipation causes a process failure, if the difference between air and process temperature is below 8.6 K and the tools rotational speed is below 1380 rpm. This is the case for 115 data points.
3. power failure (PWF): the product of torque and rotational speed (in rad/s) equals the power required for the process. If this power is below 3500 W or above 9000 W, the process fails, which is the case 95 times in our dataset.
4. overstrain failure (OSF): if the product of tool wear and torque exceeds 11,000 minNm for the L product variant (12,000 M, 13,000 H), the process fails due to overstrain. This is true for 98 datapoints.
5. random failures (RNF): each process has a chance of 0,1 % to fail regardless of its process parameters. This is the case for only 5 datapoints, less than could be expected for 10,000 datapoints in our dataset.  
   If at least one of the above failure modes is true, the process fails and the 'machine failure' label is set to 1. It is therefore not transparent to the machine learning method, which of the failure modes has caused the process to fail.

Immagine che contiene testo, schermata, linea, Carattere

Descrizione generata automaticamente

1 Dataset Sample

# 3. Data Pre-Processing

## 3.1 Attribute Creation

We have created attributes to capture some information that will help us in the prediction, these 3 elements are the power, tool\_torque and delta\_temperature

power: is the power absorbed by the machine during the process, calculated as the product of the rotational speed and the torque. Usefull to predict power failure (PWF).

tool\_torque: Is the product between the input torque and the output torque. Usefull to predict overstrain failure (OSF).

delta\_temperature: is the temperature difference between the air and the metal component, calculated as the air temperature minus the process temperature. Usefull to predict heat dissipation failure (HDF).



2 Attribute Creation

The features of this dataset that represent the fails have been grouped into a single feature. The rows that have more failures will be duplicated and each duplicate will have a different failure.

The dataset is very unbalanced, because in the field of predictive maintenance it is difficult to find data that reflect the failure of a machine, about 9600 rows belong to the class ‘nf’ no failure, the remaining ones are distributed with a very low number of samples in the other 5 classes. For this reason it will be necessary to perform an oversampling with SMOTE

Immagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamenteImmagine che contiene schermata, testo, Rettangolo, Diagramma

Descrizione generata automaticamente

3 Class Distribution

## 3.2 Train-Test split

The first step after removing the null values and encode the features the dataset was divided into training and test sets. This prevented any information from the preprocessing steps from leaking into the test set. The test set size was set to 0.3, meaning it contained 30% of the data. The Stratify option was used to keep the same proportion of classes in both sets.

Than we apply the SMOTE Oversampling to the Training set

Immagine che contiene testo, schermata, schermo, Rettangolo

Descrizione generata automaticamente

4 SMOTE Oversampling

## 3.3 Outlier Detection

Looking at the distributions and boxplots for each attribute for each class, we can state that there are no outliers, values that deviate considerably from the mean or the median.

## 3.4 Feature Selection

This operation consists of selecting the most relevant features for the classification model, eliminating those that do not provide useful information or that are correlated with each other. To do this, the mutual information criterion was used, in particular, the “mutual\_info\_classif” method was applied, which calculates the mutual information between each feature and the class variable, returning a value between 0 and 1.

Immagine che contiene testo, calligrafia, Carattere

Descrizione generata automaticamente

5 Mutual Information

I decided to eliminate the features that had a mutual information very close to zero, the features eliminated are air\_temperature, process\_temperature and type.

## 3.5 Normalization

Since the data do not have any outliers, we can apply a MinMax normalization to the features, in order to reduce the disparity between the values of some of them.



# 4. Evaluation

To evaluate the performance of the classifiers, the accuracy and F1-Score metrics are used. The F1-Score is especially important because the dataset has a strong imbalance between the classes, and the accuracy alone is not a reliable indicator of the quality of the classifier.

The Classes from 0 to 5 in the images represents resepectively ['hdf' , 'nf', 'osf', 'pwf', 'rnf', 'twf']

### KNN

The best k value for KNN is 3 because it’s the right trade between Accuracy and F1-Score

Immagine che contiene testo, schermata, linea, Diagramma

Descrizione generata automaticamenteImmagine che contiene testo, linea, diagramma, Diagramma

Descrizione generata automaticamente

Immagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamente

Immagine che contiene testo, schermata, diagramma, Rettangolo

Descrizione generata automaticamente

### Decision Tree

Immagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamente

Immagine che contiene testo, schermata, diagramma, Rettangolo

Descrizione generata automaticamente

### RandomForest

Immagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamente

Immagine che contiene testo, schermata, diagramma, Rettangolo

Descrizione generata automaticamente

### Bayesian Classifier

Immagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamente

Immagine che contiene testo, schermata, diagramma, schermo

Descrizione generata automaticamente

## 4.1 Results

From the results of the classification report and the confusion matrix, it emerges that the most performing classifier is the Random Forest, which achieved the highest values in all the metrics. It was predictable that all the models would encounter difficulties in classifying class 4 and 5 (‘rnf’ and ‘twf’), as they are not failures determined by the attributes, but random events or dependent on the wear of the machine (rnf is a sudden failure of a machine, while twf is a failure caused by the failure to renew a worn-out machine).

The misclassification of some datapoints could be attributed to the fact that they were classified with more than one error in the initial dataset.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | hdf | nf | osf | pwf | rnf | twf |
| KNN | 0.50 | 0.95 | 0.62 | 0.55 | 0.02 | 0.10 |
| Decision Tree | 0.88 | 0.98 | 0.75 | 0.80 | 0.00 | 0.08 |
| Random Forest | 0.87 | 0.99 | 0.79 | 0.81 | 0.00 | 0.09 |
| Bayesian Classifier | 0.54 | 0.72 | 0.49 | 0.19 | 0.01 | 0.09 |

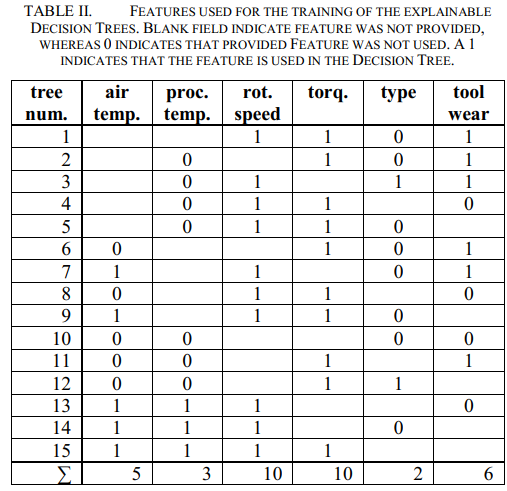
#### F1-Score for each Class

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | KNN | Decision Tree | Random Forest | Bayesian Classifier |
| F1-Score | 0.93 | 0.97 | 0.98 | 0.71 |

#### Average F1-Score

# 5. Comparison with other Studies

Another study examines this classification problem using a Bagged Decision Tree model. It trained 15 trees, the features of each tree were randomly selected from the 6 available in the dataset. Each tree uses 4 features out of 6, following the pattern of Table II. The selection criterion was to maximize the sum of the predictive importance of the features (the higher the importance, the more likely the choice).



The only way to compare this study is through the F1-Score, because the evaluation of the classifier was done by the authors of the paper on a sample of 20 Samples.

Immagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamente

The results are quite similar to our model but in this study they used a simpler model, than as the principle of Occam’s razor suggests, it is preferable to use simpler models with the same accuracy, because more complex models can tend to overfit better to the training data.

Another thing we can notice is that the classifier proposed in the paper struggles to recognize TWF failures, as tuple are label in a “random” way.

Immagine che contiene testo, ricevuta, numero, Parallelo

Descrizione generata automaticamente

Immagine che contiene testo, numero, Parallelo, documento

Descrizione generata automaticamenteA problem that can be observed in the previous Table is that many samples do not have an explanation associated, because the decision trees fail to classify them correctly. To overcome this problem, a normalization technique based on the Z-Score was applied, which takes into account the standard deviation and the expected value of the variables. In this way, the two attributes that have the highest Z-Score in absolute value were selected, and used as an explanation for the samples. This solution does not guarantee a high accuracy of the explanations, but it can be helpful to help experts with some hints to identify the cause of the problem.

Immagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamente

# 6. User Interface

To show the practical use of the model, we created a simple user interface for the application, using the Gradio framework. In this way, you can easily interact with the model, entering the input data and observing the class predicted by the model in real time.

Immagine che contiene testo, schermata, numero, ricevuta

Descrizione generata automaticamente

7. Future Improvements

Moreover, as a possible improvement, it would be interesting to add explanations generated by the model for each sample, based on the most relevant factors. These explanations could help to understand the reasoning of the model and to evaluate its reliability.

To make the model more effective, you could think of applying it to real industrial systems, which need to monitor the state of the machines and prevent failures. In this way, the model could provide the operators with useful information for the maintenance and diagnosis of the problems, improving the safety and efficiency of the machines. You could also integrate the model with other sources of data, such as sensors to acquire new data and make the model more robust and reliable, thus increasing its predictive ability and its accuracy.