# Predictive machine defectiveness using Machine learning

This notebook looks into using various Python-based machine learning and data science libraries in an attempt to build a machine learning model capable of predicting whether or not machine has defective based on given attributes.

We're going to take the following approach:

1. Problem definition
2. Data
3. Evaluation
4. Features
5. Modelling
6. Experimentation

Industry: Manufacturing

Sector: It Help Desk

## 1.Problem Statement

Recent machine failures in

e manufacturing industry are causing fatal accidents, adding complexity, increasing costs, and resulting in significant time losses for companies.

Top of Form

## 2.Problem Definition

## The objective of this machine learning project is to develop a predictive model for identifying potential defects in machines within the manufacturing industry. The goal is to leverage historical attributes and performance data of machines to forecast defects, enabling timely maintenance or replacement to prevent accidents and production disruptions.

.

Top of Form

## 3. Proposed Solution

Establishing a help desk integrated with machine learning and data science to anticipate the likelihood of machinery failures and initiate replacement or repairs proactively, thereby preventing unforeseen operational disruptions.

## 4. Data

The original data came from Cleaveland database from the UCI Machine Learning Repository.

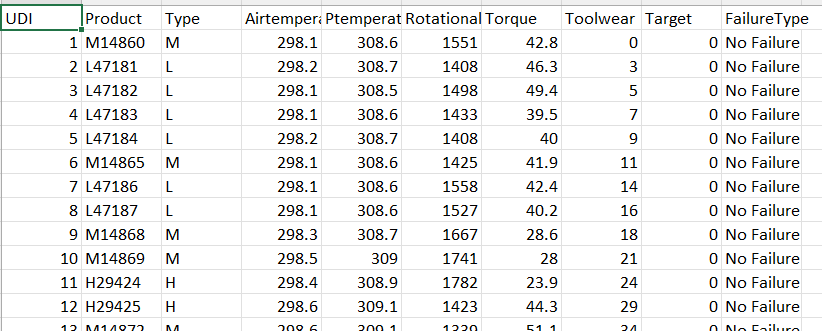
https://archive.ics.uci.edu/ml/datasets/AI4I+2020+Predictive+Maintenance+Dataset

## 5.Dataset

The dataset used for training and testing the model should include historical records of machines, including attributes such as Air temperature, Process temperature, Rotational Speed ,Torque

Additionally, it should contain labels indicating whether the machine experienced a defect within a certain time frame.

Sample Data set



6.Features

This is where you'll get different information about each of the features in your data. You can do this via doing your own research (such as looking at the links above) or by talking to a subject matter expert (someone who knows about the dataset).

**Create data dictionary**

**Variable and Attributes Information**

The dataset consists of 10 000 data points stored as rows with 14 features in columns UID: unique identifier ranging from 1 to 10000 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 air temperature [K]: generated using a random walk process later normalized to a standard deviation of 2 K around 300 K 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. rotational speed [rpm]: calculated from a power of 2860 W, overlaid with a normally distributed noise torque [Nm]: torque values are normally distributed around 40 Nm with a Ïƒ = 10 Nm and no negative values. tool wear [min]: The quality variants H/M/L add 5/3/2 minutes of tool wear to the used tool in the process. and 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

tool wear failure (TWF): the tool will be replaced of fail at a randomly selected tool wear time between 200 and 240 mins (120 times in our dataset). At this point in time, the tool is replaced 69 times, and fails 51 times (randomly assigned).

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 tool rotational speed is below 1380 rpm. This is the case for 115 data points.

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.

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.

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.

Data Cleaning and Preprocessing Techniques

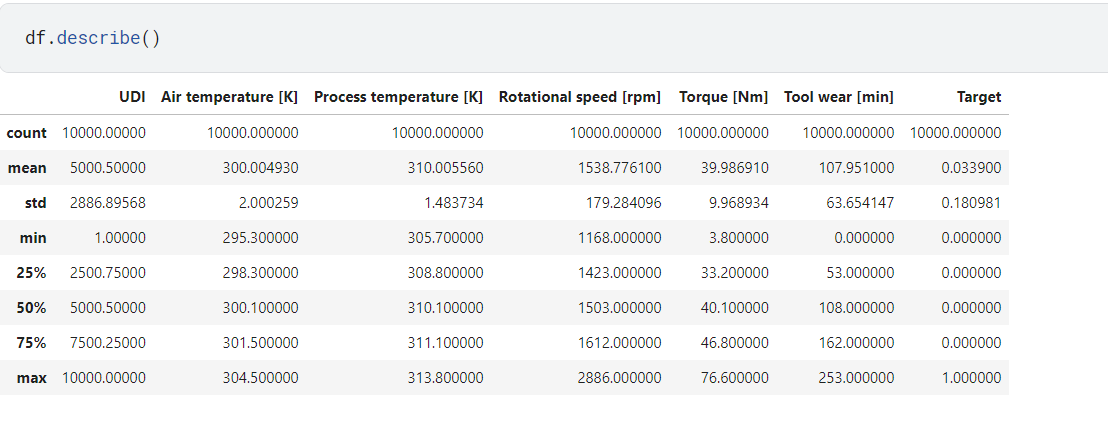
Data preprocessing was performed to ensure the quality and reliability of the dataset. This involved handling missing values, removing duplicates, and addressing inconsistencies or outliers. Additionally, feature encoding, scaling, and normalization techniques were applied as required

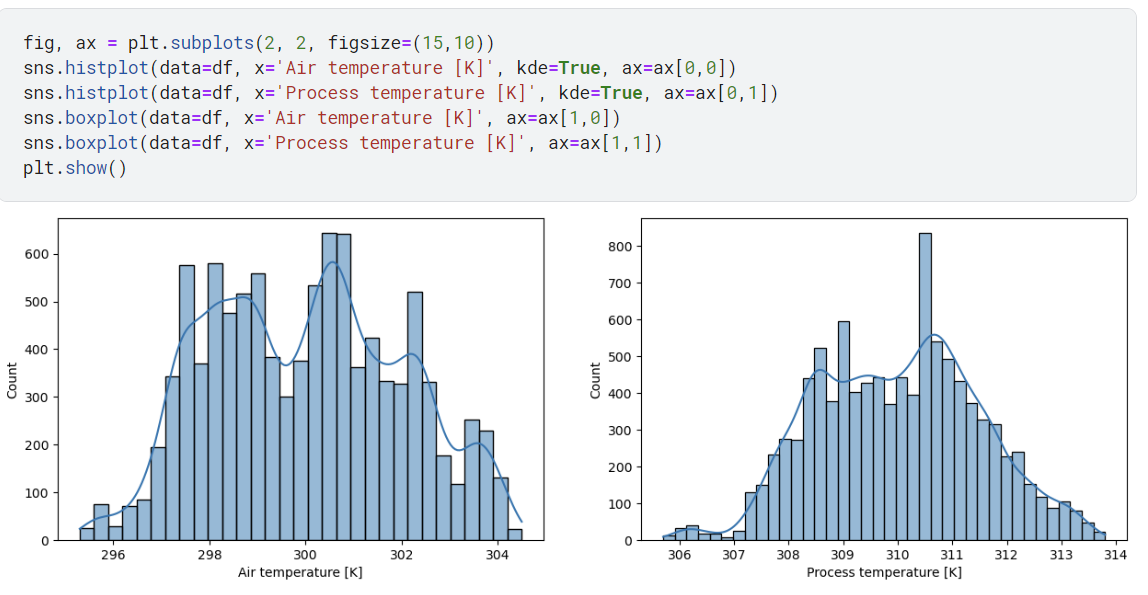
Sample Code

### 

EDA (Exploratory Data Analysis)

Exploratory Data Analysis (EDA) was conducted to gain insights into the dataset. This involved visualizations, statistical summaries, and correlation analysis to understand the relationships between variables and identify any patterns or trends.





### 3. Feature Selection and Engineering

### **Identification of Relevant Features**

Relevant features crucial for predicting the resolution time of problem tickets were discerned based on their potential impact. Variables such as problem type, severity, and the responsible department were considered as significant predictors.

**Feature Engineering Techniques**

To bolster the predictive power of the model, feature engineering techniques were implemented. This process involved the creation of new features and the transformation of existing ones to capture meaningful information. For instance, time-related features were extracted from timestamp variables to enrich the dataset.

**Dimensionality Reduction Methods**

To streamline the dataset, dimensionality reduction techniques, including principal component analysis (PCA) and feature selection algorithms, were employed. These methods effectively reduced the number of features while retaining pertinent information and minimizing noise. Additionally, unnecessary attributes were manually removed using the method:

pythonCopy code

df.drop("Attribute name", axis=1, inplace=True)

This step contributed to a more focused and efficient feature set for model training.

Top of Form

## Conclusion

The developed machine learning model aims to enhance the manufacturing industry's predictive maintenance capabilities, reducing the risk of accidents and improving overall operational efficiency. Continuous monitoring and refinement are essential to maintain the model's accuracy and relevance in a dynamic manufacturing environment.