**Machine Failure Prediction**

Project Report

DataScience\_JuneBatch \_ Major Project1

Shagun Thakur

# **Introduction**

## Objective:

The objective of this project is to predict machine failure based on sensor data collected from various machines. The dataset comprises sensor readings related to machine operations and includes a binary indicator column indicating machine failure.

## Dataset Description:

The dataset contains the following columns:

* **Footfall:** Number of people or objects passing by the machine.
* **Temp Mode:** The temperature mode or setting of the machine.
* **AQ:** Air Quality index near the machine.
* **USS:** Ultrasonic Sensor data, indicating proximity measurements.
* **CS:** Current Sensor readings, indicating the electric current usage of the machine.
* **VOC:** Volatile organic compounds level detected near the measurements.
* **RP:** Rotational Position or RPM (revolutions per minute) of the machine parts.
* **IP:** Input pressure to the machine.
* **Temperature:** The operating temperature of the machine.
* **Fail:** Binary indicator of machine failure (1 for failure, 0 for no failure)

1. **Data Pre-processing**

## Libraries Used:

* *pandas :* For data manipulation
* *plotly, matplotlib, seaborn :* For data visualization
* *sklearn :* For machine learning algorithms and metrics

## Data Exploration:

* **Data Import:** The dataset was imported successfully using pandas.
* **Initial Exploration :**
* shape : The shape of the dataset was found to be (944, 10) means 944 rows and 10 columns were present in the dataset.
* info ( ): Provided information about data types and non-null counts.
* isnull ( ): Confirmed no missing values were present in the dataset.

## Correlation Analysis:

The correlation between the features and the target variable (fail) was analysed. It was founded out that three of the sensors, namely footfall, tempMode and USS have negative relation with the target variable while others like CS, RP, IP, and Temperature have positive but very small relation with the target value. This means the remaining sensors that are AQ and VOC have the largest impact on the machine failure.

# 

# **Feature Selection and Data Preparation**

## Dependent and Independent Variables:

* **Dependent Variable :** Fail
* **Independent Variable :** All other columns

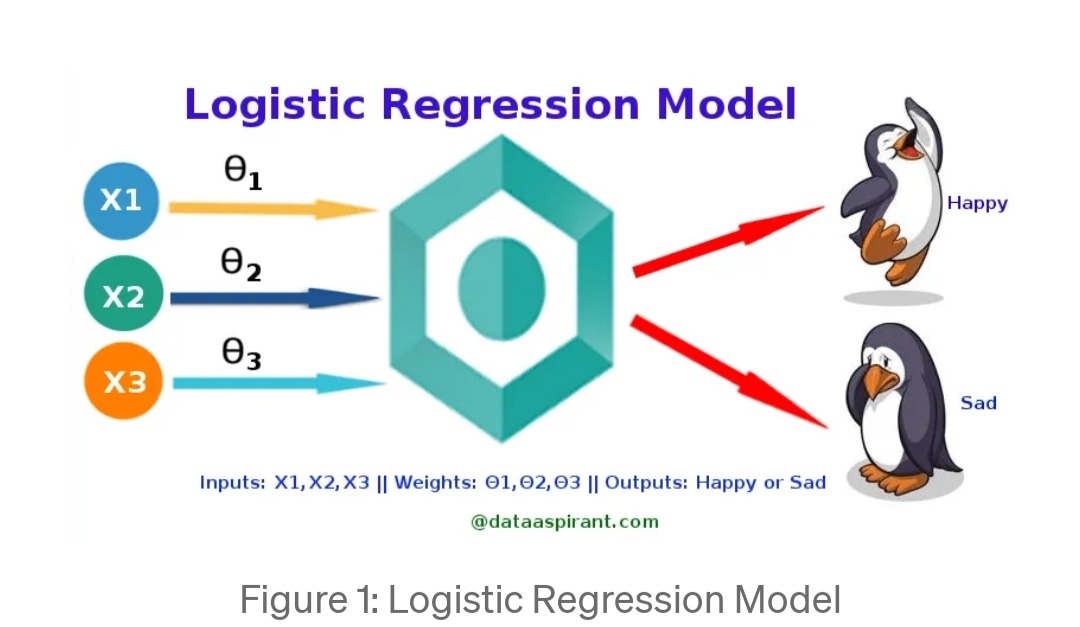
## Data Splitting:

The dataset was split into training and testing sets with an 80 : 20 ratio using train\_test\_split from sklearn.

# **Model Building and Evaluation**

## Model Choice:

A **Logistic Regression** model was choosen for classification, given its suitability for binary outcomes.



**Logistic Regression** is a statistical model used for binary classification problems. It predicts the probability of an input belonging to a specific class. Logistic regression is designed to handle discrete outcomes. It is used when the dependent variable (target) is categorical.

For example,

* To predict whether an email is spam (1) or not (0)
* Whether the tumor is malignant (1) or not (2)

And in our case, whether the machine fails (1) or not (0)

**Simple Logistic Regression**

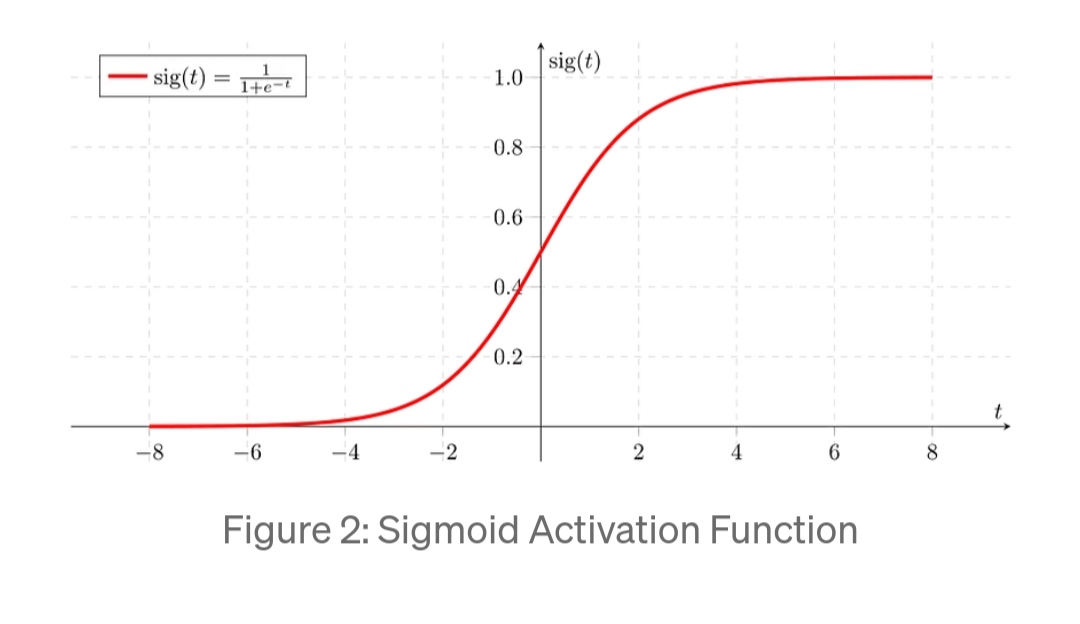
***Model***

Output = 0 or 1

Hypothesis => Z = WX + B

h(x) = sigmoid (Z)

*Sigmoid Function*



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

* The fundamental concept behind logistic regression is the logistic function, also known as the **sigmoid function.** The logistic function maps any real number to a value between 0 and 1. It takes the form:

Sigmoid (z)

The model applies this sigmoid function to a linear combination of the input to obtain a value between 0 and 1. This value represents the probability of the input belonging to a particular class.

## Model Training:

The logistic regression model was trained on the training dataset.

## Model Evaluation:

* **Confusion Matrix:** Theconfusion matrix obtained was: [[88 8], [8 85]]

This matrix indicates:

* True Positives (TP) : 85
* True Negatives (TN): 88
* False Positives (FP): 8
* False Negatives (FN): 8
* **Accuracy:** The accuracy of the model was calculated to be **91%**.
* **Visualization:** The result were visualized using a **heatmap** plotted with the Plotly library, providing a clear representation of the confusion matrix.

# **Results and Insights**

## Model Performance:

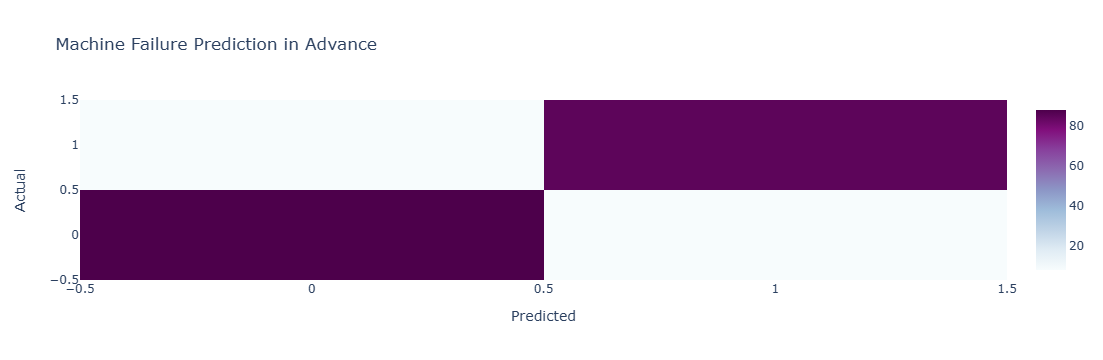
The Logistic Regression model demonstrated strong performance with the accuracy of 91%. This high accuracy indicates that the model is effective in predicting machine failures based on the given features.

## Confusion Matrix Analysis:

The confusion matrix reveals that the model is generally accurate in predicting both failure and non-failure events. The number of false positives and false negatives is relatively low, indicating good model reliability.

## Feature Impact:

The correlation analysis highlighted which features are the most associated with machine failures. Understanding these correlations can help in further refining the model and improving predictive accuracy



# **Conclusion**

The machine failure prediction model, utilizing Logistic Regression, has demonstrated a robust performance in classifying machine failure events. With an achieved accuracy of 91%, the model effectively distinguishes between operational and failure states of the machines based on the provided sensor data. The confusion matrix analysis reveals a good balance between true positives and true negatives, with relatively low false positives and false negative rates, indicating that the model is both reliable and accurate in its predictions.

Overall, the project successfully demonstrates the potential of using sensor data for predictive maintenance and machine failure forecasting. The insights gained from the feature importance and the model performance metrics offer a solid foundation for understanding machine behaviour and planning preventive measure to the minimize downtime and operational disruptions.

**Future Work:**

* Consider experimenting with other classification algorithms such as Gradient Boosting.
* Explore feature engineering techniques to enhance model performance.
* Investigate the impact of additional sensor data or external factors on machine failure.

# **References**

* *pandas* documentation for data manipulation.
* *sklearn* documentation for machine learning algorithms.
* *Plotly* library for data visualization.