PROJECT PROPOSAL

1.Title of the Project :

Predictive Maintenance: Machine Failure Prediction.

2. Brief on the project :

The project aims to develop a predictive maintenance model for anticipating machine failures in an industrial setting. Predictive maintenance involves using historical data, sensor readings, and machine logs to predict when equipment is likely to fail, allowing proactive maintenance to be performed, thereby reducing downtime and maintenance costs. The motivation behind this project lies in the potential to optimize operational efficiency, reduce unplanned downtime, and minimize production losses. Previous work in predictive maintenance has shown promising results, indicating the feasibility and benefits of implementing such systems.

3. Deliverables of the project :

- **List of Questions:**
  - When is machine likely to fail?
  - How accurate can we predict machine failure?

- **Model Details:**
  - Develop machine learning models based on historical data.
  - Explore various algorithms such as Random Forest , Logistic Regression , SVM , Gradient Boost , KNN , Naïve Bayes , Decision Tree.

- **Resources :**
  - **Data set Source :**
    The dataset for this project is provided by The IOT Academy. This dataset includes historical data , sensor readings and machine logs from industrial machinery enabling the development of the predictive maintenance model.

  - **Software :**
    Jupyter Notebook will be used in this project in accordance with Python programming language utilizing python libraries numpy , Pandas , Scikit Learn , Matplotlib , Sns , for data processing , modeling , evaluation.

4. Individual Details :

- **Name** : Ankush Kumar
- **Email** : mr.ankush2020@gmail.com

5. Milestones:

- **Define the Problem**:

Developing a predictive maintenance model for industrial machinery to forecast equipment failures based on historical data, sensor readings, and machine logs. The objective is to proactively identify potential failures, enabling businesses to schedule maintenance activities during planned downtimes, minimize disruptions to production processes, and reduce maintenance costs. Key challenges include identifying relevant patterns in the data indicative of machine health degradation and accurately predicting failures with sufficient lead time for timely maintenance interventions. The ultimate goal is to improve operational efficiency, minimize unplanned downtime, and extend the lifespan of industrial machinery, thereby enhancing overall productivity and profitability.

- **Understanding the Business Problem** :

    - Predictive Maintenance Need: Businesses require a predictive maintenance solution to anticipate equipment failures before they occur.

    - Proactive Maintenance Approach: Predictive maintenance involves forecasting failures based on historical data, sensor readings, and machine logs

    - Minimize Disruptions: Proactively identifying potential failures allows for scheduling maintenance during planned downtimes, minimizing disruptions to production processes.

    - Cost Reduction: By scheduling maintenance activities strategically, businesses can reduce maintenance costs associated with reactive repairs and unplanned downtime.

    - Timely Interventions: The model must accurately predict failures with sufficient lead time to allow for timely maintenance interventions, avoiding costly breakdowns.

    - Operational Efficiency: Developing a robust predictive maintenance model aims to optimize operational efficiency by extending machinery lifespan and maximizing asset utilization.

    - Overall Business Impact: Implementing an effective predictive maintenance solution enhances productivity, reduces maintenance costs, and improves profitability for the business.

- **Data Acquisition** :
    - Specification: The IOT Academy outlines the project's data requirements and objectives.
    - Collection: Relevant data sources, including sensor readings, machine logs, and historical maintenance records, are gathered.
    - Quality Assurance: Quality checks ensure the accuracy, integrity, and completeness of the acquired data.
    - Privacy and Compliance: Data privacy regulations and compliance standards are adhered to, with measures in place to protect sensitive information.
    - Documentation: Detailed documentation of the acquisition process, including metadata, is maintained for transparency and reproducibility.
    - Access and Sharing: Access to the data is provided to project stakeholders as needed, with data sharing agreements established where applicable.
    - Security Measures: Robust security measures safeguard the data against unauthorized access and cyber threats.
    - Governance: Data governance policies ensure ethical and legal standards are followed in acquiring, managing, and utilizing the data.

➢ Choosing the Python Platform

In selecting the Python platform for the project, several factors contribute to its suitability for developing a predictive maintenance model. Here's a brief overview:

- Versatility: Python offers a wide range of libraries and frameworks tailored for data analysis, machine learning, and predictive modeling, making it well-suited for this project's requirements.

- Community Support: Python boasts a large and active community of developers and data scientists, providing access to extensive resources, tutorials, and community forums for assistance and collaboration.

- Data Science Ecosystem: Python's ecosystem includes popular libraries such as Pandas for data manipulation, NumPy for numerical computations, and Scikit-learn for machine learning, facilitating seamless integration and workflow for data analysis and modeling tasks.

- Visualization Capabilities: Python libraries like Matplotlib and Seaborn enable the creation of insightful visualizations, essential for exploratory data analysis and communicating results effectively.

- Ease of Learning: Python's syntax is intuitive and beginner-friendly, making it accessible to individuals with varying levels of programming experience. This facilitates rapid prototyping and experimentation in developing predictive maintenance models.

- Scalability: Python's scalability is enhanced by its compatibility with distributed computing frameworks like Apache Spark and Dask, enabling the handling of large-scale data sets and parallel processing for efficient model training and evaluation.

- **Explore and pre-process data :**
  - .head( ) function is used to display the top 5 records.

  - .shape method is used to check the number of rows and columns.(10000 * 9)

  - .columns method used to view the names of features present in the data frame.

    Features present: 'UDI', 'Product ID', 'Type', 'Air temperature [K]', 'Process temperature [K]', 'Rotational speed [rpm]', 'Torque [Nm]', 'Tool wear [min]', 'Target'],

  - .duplicated( ).sum( ) used to check the duplicate values present in the features . (observed to be none)

  - .info( ) is used to check the data types of features present in the data. (out of total 9 features present 7 are numerical features while 2 are categorical features.

  - .null.sum( ) used to check the null values present in the features . (observed to be none)

  - .nunique( ) used to check for the unique elements in all the features.

  - .drop( ) used to drop irrelevant features 'UDI', 'Product ID', since they have unique values for each machine.

  - .rename( ) used to rename the remaining features as :
    'Air temperature [K]': 'Air_temperature',
    'Process temperature [K]':'Process_temperature',
    'Rotational speed [rpm]':'Rotational_speed',
    'Torque [Nm]': 'Torque',

'Tool wear [min]':'Tool_wear'

- Splitting the data into two separate data frames for better analysis as: num_column (for numerical features) and cat_column (for object features)

➢ **Create Features :** No new Feature was created .

➢ **EDA :**

❖ **num_ column :**

- .describe( ) used to display the descriptive statistics of the numerical features.
- .corr( ) used to check co relation between the independent numerical features with the target variable.
- Visualizing the co relation with the help of heat map.
- .valuecounts( ) shows that target feature has class imbalance features which needs to be taken care of.
- Visualizing the class imbalance problems using a pie chart.

❖ **num_ column(uni-variate analysis) :**

(a) Air_temperature :

- Using box plot to visualize outliers present (detected none)
- Using KDE plot to visualize skewness (it was normal skewed)

(b) Process_temperature :

- Using box plot to visualize outliers present (detected none)
- Using KDE plot to visualize skewness (it was normal skewed)

(c)Torque :

- Using box plot to visualize outliers present (detected)
    - Using IQR method to find the outliers boundaries.

        - Replacing the outliers with median value(median is less sensitive to outliers compared to the mean, making it a robust measure of central tendency.)
        - Using box plot to visualize outliers again after replacing with median.(detected none)
    - Using KDE plot to visualize skewness after treating the outliers. (it was normal skewed)

(c) Tool_wear:

- Using box plot to visualize outliers present (detected none)
- Using KDE plot to visualize skewness (it was normal skewed)

(d) Rotational_speed

- Using box plot to visualize outliers present (detected)
    - Using IQR method to find the outliers boundaries.
    - Applying logarithmic transformation to reduce outliers impact since data was right skewed
    - Using box plot to visualize outliers again after replacing with median.(detected none)
- Using KDE plot to visualize skewness after treating the outliers. (it was normal skewed)

❖ Using pairplot to visualize the correlation of features with the target feature

❖ Cat_column :

(a) Type :

- Value_counts( ) used to display the total count of types of failures occurred in the machines while operating.
- Visulaizing the total count of types of machine failure occurred.

- **Feature Engineering :**
  - Using labelencoder ( )to convert categorical feature ‘Type’ to numerical variable’
  - Concatenating the num_data and cat_data together to create combined_data.
  - Creating X (the target variable is dropped) and y datasets (target variable is included)
  - Using MinMaxScaler( ) to standardize the data X dataset on the scale of 0 to 1.
  - Splitting the data into X_train,X_test,y_train,y_test using train_test_split.
  - Apply SMOTE for over-sampling the minority class on X_train and y_train as X_resampled and Y_resampled.

- **Model Development :**
  - Logistic regression :
    X_resampled and Y_resampled are passed to Logistic regressor classifier ( )
  - Decision Tree :
    X_resampled and Y_resampled are passed to Decision Tree classifier ( )
  - Random Forest :
    X_resampled and Y_resampled are passed to Random Forest classifier ( )
  - SVM :
    X_resampled and Y_resampled are passed to SVM classifier ( ) using rbf kernel amd linear kernel.
  - Naïve Bayes :
    X_resampled and Y_resampled are passed to Naïve Bayes classifier ()
  - Gradient Boost :
    X_resampled and Y_resampled are passed to Gradient Boost classifier ( )

- **Model Evaluation** :
  - Training and Testing accuracies for all the models created.
    - Logistic Regression : Training Accuracy: 0.7943
      Testing Accuracy: 0.7917
    - Decision Tree : Training Accuracy: 0.1
      Testing Accuracy: 0.9523
    - Random Forest : Training Accuracy: 0.1
      Testing Accuracy: 0.9687
    - SVM(RBF) : Training Accuracy: 0.9380
      Testing Accuracy: 0.9070
    - SVM (Linear) : Training Accuracy: 0.7943
      Testing Accuracy: 0.8073
    - Naïve Bayes : Training Accuracy: 0.8277
      Testing Accuracy: 0.8177
    - Gradient Boost : Training Accuracy: 0.9454
      Testing Accuracy: 0.9320
  - Model Selection :
    - based on the training accuracy and testing accuracy of the above model Random Forest algorithms performs the best among all .
    - it tends to overfit due to 0.1 accuracy in training and 0.9687 in testing to
    - To overcome this problem hyper parameter tuning is performed using Grid SearchCV and Randomized SearchCV
      - Grid Search CV Optimization : Training Accuracy: 0.9269
        Testing Accuracy: 0.9013
        It can be noticed that accuracy dropped drastically usin Grid Search CV Optimization.
      - Randomized Search CV : Training Accuracy: 0.9748
        Testing Accuracy: 0.9500
        It can be noticed that it gives better accuracy.

- **Final Model Selection** : Random Forest Classifier with Randomized parameters with a ROC AUC value of 90 signifies a strong and reliable predictive model.

  - Model Name (with hyper parameters):
    rf_model =RandomForestClassifier(n_estimators=600,max_depth=None,min_samples_leaf=10,n_jobs=-1,random_state=42)

- **Report Writing**:

  Predictive maintenance is revolutionizing the industrial sector by leveraging data analytics and machine learning to anticipate equipment failures before they occur. By proactively identifying potential issues, predictive maintenance strategies aim to minimize downtime, optimize maintenance schedules, and enhance overall operational efficiency.

- Key Components of Predictive Maintenance:
  - Data Acquisition:mGathering data from sensors, machine logs, and historical maintenance records.
  - Ensuring data integrity and completeness for accurate analysis.

- Exploratory Data Analysis (EDA):
  - Analyzing data distributions, correlations, and patterns.
  - Identifying key features indicative of equipment health degradation.

- Model Development:
  - Selecting appropriate machine learning algorithms (e.g., Random Forest, Gradient Boosting) for predictive modeling.
  - Training models on historical data to predict equipment failures.

- Model Evaluation:

  - Assessing model performance using metrics such as ROC AUC, accuracy, and precision-recall curves.
  - Fine-tuning models and selecting the best-performing ones for deployment.

- Benefits of Predictive Maintenance:
    - Reduced Downtime: By predicting failures in advance, organizations can schedule maintenance during planned downtimes, minimizing disruptions to production processes.
    - Cost Savings: Proactive maintenance reduces repair costs associated with unplanned breakdowns and extends the lifespan of industrial machinery.
    - Improved Safety: Identifying potential equipment failures helps mitigate safety risks to workers and prevents workplace accidents.
    - Enhanced Productivity: Optimized maintenance schedules and reduced downtime contribute to increased operational efficiency and productivity.

- Case Studies:

    - Manufacturing Industry: Predictive maintenance models have been successfully deployed in manufacturing plants to forecast equipment failures and optimize maintenance schedules, leading to significant cost savings and efficiency gains.
    - Energy Sector: Utility companies use predictive maintenance to monitor power generation equipment and predict potential failures, ensuring reliable electricity supply and reducing outage durations.

- Conclusion:
    - Predictive maintenance is a game-changer for industries seeking to maximize efficiency and reduce operational costs. By harnessing the power of data analytics and machine learning, organizations can proactively manage their assets, minimize downtime, and achieve sustainable growth in today's competitive landscape.

➢ Project submission:

- Objective:
    - The objective of our project is to develop and deploy a predictive maintenance model for industrial machinery to forecast equipment failures, minimize downtime, and optimize maintenance schedules.
- Key Components:
    - Data Acquisition: Acquired data from sensors, machine logs, and historical maintenance records to build a comprehensive dataset.
    - Exploratory Data Analysis (EDA): Analyzed data distributions, correlations, and patterns to identify important features indicative of equipment health degradation.
    - Model Development: Trained machine learning models (e.g., Random Forest, Gradient Boosting) to predict equipment failures based on historical data.
    - Model Evaluation: Evaluated model performance using metrics such as ROC AUC, accuracy, and precision-recall curves to ensure effectiveness and reliability.
    - Deployment and Monitoring: Implemented the model into production environments, continuously monitoring its performance and making adjustments as necessary.

- ❖ Benefits:

- Reduced Downtime: Proactive maintenance scheduling minimizes disruptions to production processes, leading to increased uptime and productivity.

- Cost Savings: Early identification of potential failures reduces repair costs associated with unplanned breakdowns and extends the lifespan of machinery.

- Improved Safety: Mitigating safety risks through timely maintenance interventions ensures a safer working environment for personnel.

- Enhanced Efficiency: Optimized maintenance schedules and reduced downtime contribute to improved operational efficiency and resource utilization.

- ❖ Conclusion:
    - Our predictive maintenance model represents a proactive approach to equipment maintenance, providing actionable insights to optimize maintenance schedules and prevent costly disruptions. By leveraging data analytics and machine learning techniques, we empower industries to achieve greater efficiency, reliability, and safety in their operations.