University College of Engineering,Dindigul.

Department of Computer Science and Engineering


AI
BASED
PREDICTIVE
MAINTENANE

AIM

- This project aims to implement a predictive maintenance system and utilizing advanced predictive analytics techniques.
- The project's overarching goal is to enhance asset reliability, reduce unplanned downtime, and optimize maintenance costs.

PROCEDURE

DATA COLLECTION

DATA CLEANING

MODEL TRAINING

EVALUTION

EVALUTION

MODEL TRAINING

DATA CLEANING

DATA COLLECTION

- Gather Historical maintenance records contain information about past maintenance activities, repairs, replacements, failures, information about equipment usage, operating conditions, load levels, and environmental factors, external data sources, such as weather data, supply chain data, and equipment specifications, can provide supplementary information relevant to equipment performance and maintenance

EVALUTION

MODEL TRAINING

DATA CLEANING

- Identify and rectify errors, missing values, outliers, and inconsistencies in the data to ensure its quality and reliability
- Find the missing values in dataset if missing values found drop that column.
- Ensuring data consistency across different columns.

EVALUTION

MODEL TRAINING

- Data is split into training and testing sets using train_test_split of sklearn. The training set is used to train the model, and the testing set is used to evaluate its performance.
- Random forest model is trained using X_train and y_train data using fit() method
- Gridsearch Hyperparameter technique is used for improve the model's performance

EVALUATION

- **MSE (Mean Squared Error) :** This metric squares the absolute differences before taking the average. Squaring the errors gives more weight to larger errors in the penalty
- **R^2 (R-squared) :** R^2 measures the proportion of variance in the dependent variable(what you're trying to predict) that's explained by the independent variables (features used for prediction). A higher R^2 indicates a better fit, but it doesn't directly tell you how far off the predictions are from the actual values.
- **Accuracy_score :** Accuracy score is a metric used in machine learning classification tasks to assess how well a model predicts the correct category for a given data point.

A higher accuracy score (closer to 1) indicates better performance, meaning the model is accurately classifying most of the data points.

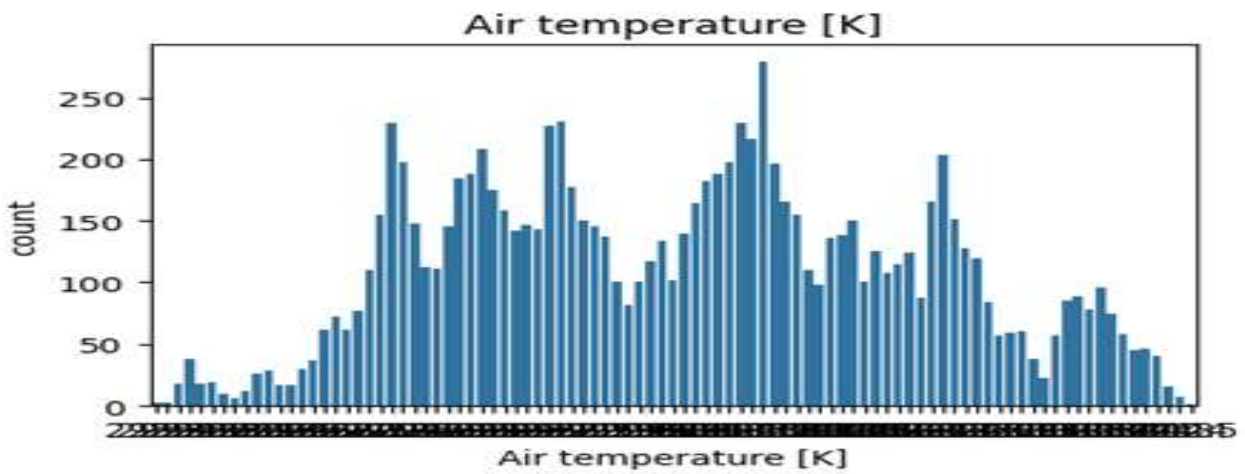
A lower accuracy score (closer to 0) suggests the model is struggling to make accurate classifications.

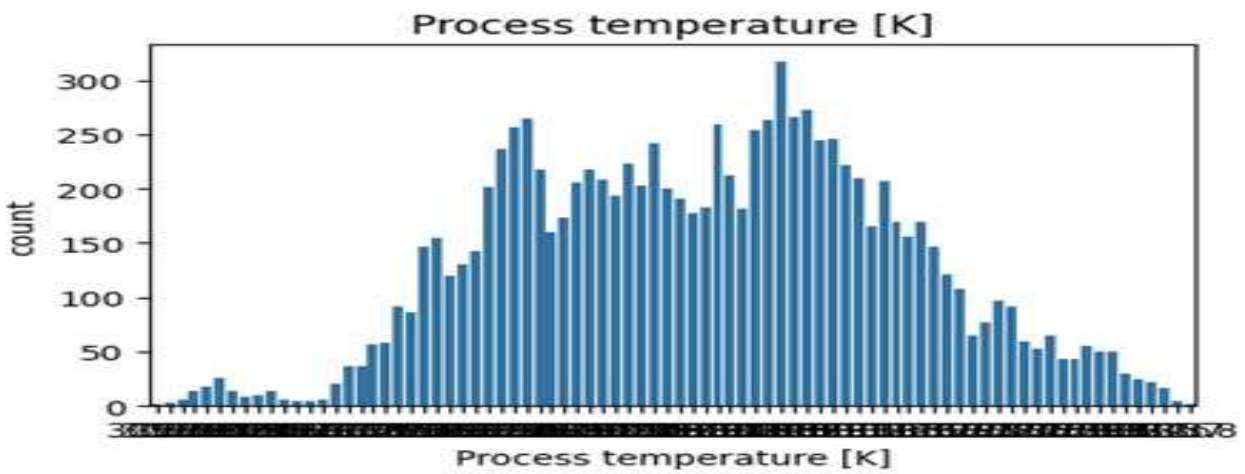
PROGRAM

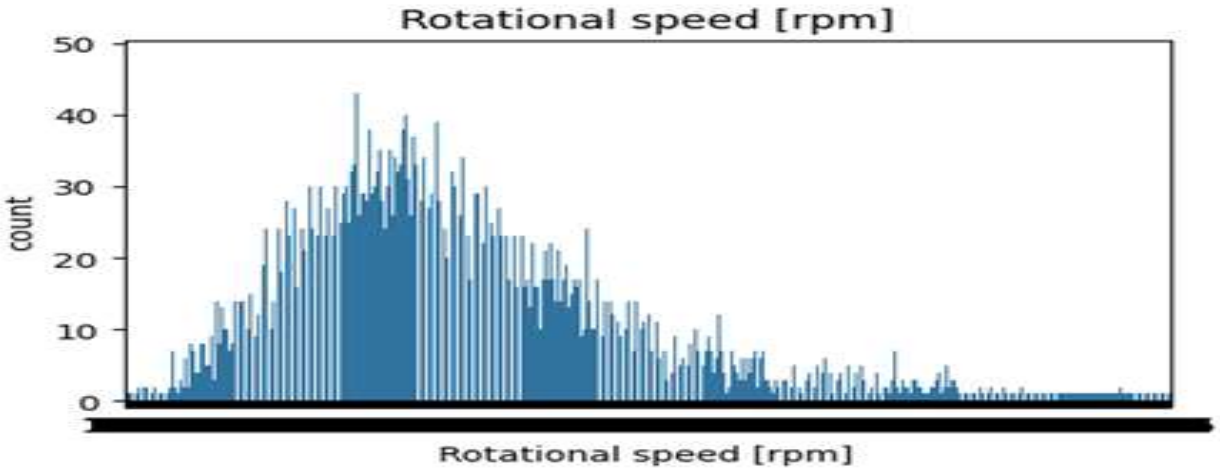
```
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error,
r2_score,accuracy_score
import pandas as pd
import matplotlib.pyplot as plt
data = pd.read_csv(“\predictive maintenance.csv”)
data=dataset.dropna()
data=data.drop_duplicates()
columns = ['Air temperature [K]', 'Process temperature [K]',
'Rotational speed [rpm]', 'Torque [Nm]', 'Tool wear [min]’]
 titles = ["Air temperature [K]", "Process temperature [K]",
"Rotational speed [rpm]", "Torque [Nm]", "Tool wear [min]"]
plt.figure(figsize=(10, 10))
```

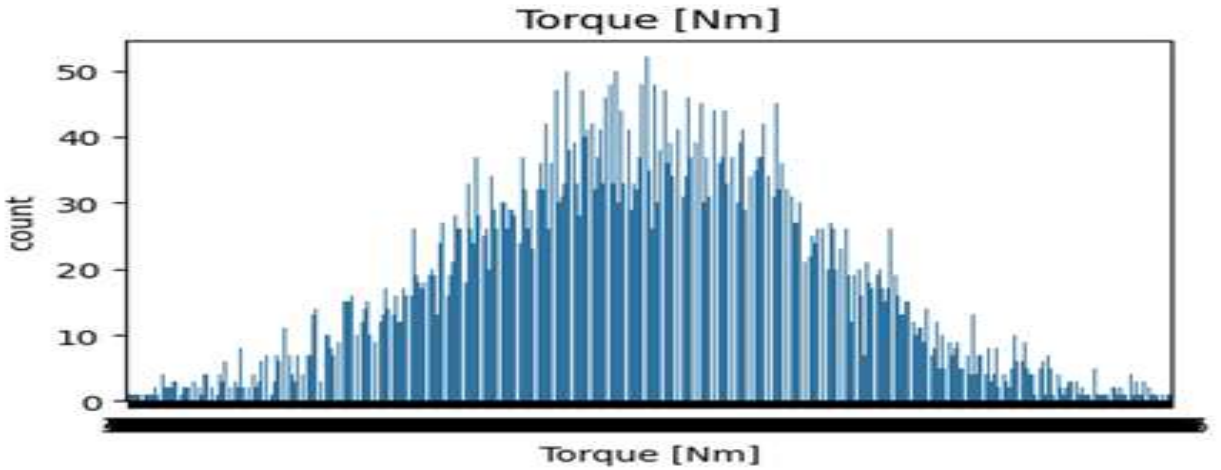
PROGRAM

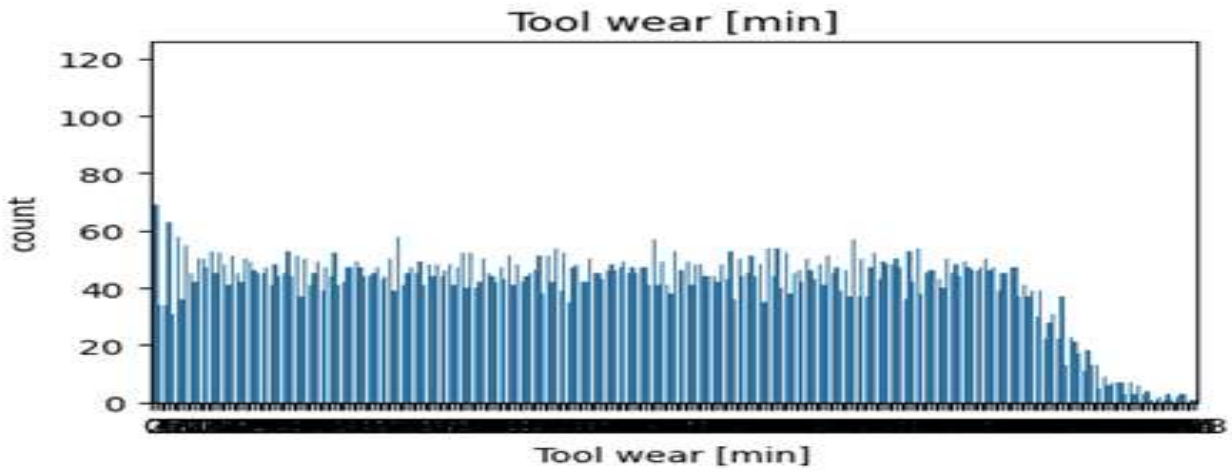
```
for i, col in enumerate(columns, 1):
  plt.subplot(3, 2, i)
  sns.countplot(x=col, data=data)
  plt.title(titles[i-1])
  plt.tight_layout()
Plt.show()
X = data.drop(columns=["Failure Type","Product ID", "Type"])
y = data["Target"]
X_train, X_test, y_train, y_test =
train_test_split(X,y,test_size=0.2,random_state=42)
model=RandomForestRegressor(n_estimators=100,random_state=42)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
print(f'MSE: {mean_squared_error(y_test, y_pred)}')
print(f'R^2: {r2_score(y_test, y_pred)}')
print(f'accuracy: {accuracy_score(y_test,y_pred)}')
```


Air temperature [K]
count
Air temperature [K]


Process temperature [K]
count
Process temperature [K]


Rotational speed [rpm]
count
Rotational speed [rpm]


Torque [Nm]
count
Torque [Nm]


Tool wear [min]
count
Tool wear [min]

OUTPUT

```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error, r2_score,accuracy_score
import pandas as pd
df = pd.read_csv('/content/predictive_maintenance[1].csv')
X = df.drop(columns=["Failure Type","Product ID", "Type"])
y = df["Target"]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
model=RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
print(f'MSE: {mean_squared_error(y_test, y_pred)}')
print(f'R^2: {r2_score(y_test, y_pred)}')
print(f'accuracy: {accuracy_score(y_test,y_pred)}')
```

OUTPUT

```
MSE: 0.0
R^2: 1.0
accuracy: 1.0
```

CONCLUSION

- Ai Predictive maintenance system is implemented successful
- The system predict equipment failure
- We get better performance than expection
- Documentation of AI has been done successfully

PRESENTED BY

- SATHYAN . P
- BHARATH . V
- VIJAYAN . S
- ARTHI . K
- KAVI SRI MAHA . P