Project Name - **TATA Steel Exploratory Data Analysis**

Project Type - EDA (Exploratory Data Analysis)

Contribution - Individual

Team Member - Veerendra Kashyap

Project Summary -

The Tata Steel Exploratory Data Analysis (EDA) project aims to understand and optimize manufacturing processes by analyzing operational data. The primary objective is to identify trends, detect anomalies, and extract actionable insights that can improve efficiency, reduce machine failures, and enhance product quality. By leveraging data analysis techniques, we can help Tata Steel achieve better control over its operations, minimize downtime, and optimize production. Steel manufacturing is a highly complex and resource-intensive process where even minor inefficiencies can lead to substantial financial losses. Various factors such as air and process temperatures, rotational speed, torque, and tool wear influence production efficiency and product quality. Understanding how these variables interact with each other allows for predictive maintenance, process optimization, and failure prevention. A systematic analysis of the dataset can uncover patterns that might otherwise go unnoticed, ultimately contributing to improved decision-making and enhanced operational performance.

Understanding the Dataset

The dataset consists of machine operational data collected from Tata Steel’s production units. The key variables included in the dataset are:

- Air temperature [K] and Process temperature [K] – Important parameters that affect the stability of the manufacturing process and the quality of the final product.
- Rotational speed [rpm] – Represents how fast the machine is running and may influence wear and tear.
- Torque [Nm] – Measures the rotational force applied to machine components, which is critical in understanding stress levels on the system.
- Tool wear [min] – Captures the duration a machine tool has been in use and can serve as an indicator for necessary maintenance.
- Machine failure and failure types (TWF, HDF, PWF, OSF, RNF) – Provide insights into machine breakdown causes and failure patterns.

- By analyzing these variables, we aim to establish relationships and identify trends that can help optimize machine operations, improve product quality, and reduce equipment failures.

Key Steps in the Analysis

1. Data Cleaning & Preprocessing
- Handling missing values and duplicate records.
- Converting categorical variables into numerical representations for analysis.
- Identifying and managing outliers using box plots and the Interquartile Range (IQR) method.
1. Exploratory Data Analysis (EDA)
- Univariate Analysis: Examining individual variable distributions using histograms, density plots, and summary statistics.
- Bivariate Analysis: Identifying relationships between key variables, such as torque vs. rotational speed, using scatter plots and correlation heatmaps.
- Multivariate Analysis: Using pair plots and heatmaps to study the combined effects of multiple variables.
1. Insights Gained
- Machine failures are often linked to extreme torque values and excessive tool wear.
- Air temperature and process temperature have a strong positive correlation, suggesting that regulating process temperature can significantly impact production stability.
- Outliers in rotational speed and torque indicate potential stress points in the manufacturing process, which may require preventive maintenance.
- Failure types (TWF, HDF, PWF, OSF, RNF) contribute differently to overall machine failures, with some failure types occurring more frequently than others.
1. Business Impact & Recommendations
- Predictive Maintenance: Failure pattern analysis enables proactive scheduling of maintenance activities, preventing costly machine breakdowns.
- Process Optimization: Fine-tuning machine parameters such as temperature, speed, and torque can help improve production consistency and minimize waste.
- Cost Reduction: Early detection of anomalies and potential failures reduces downtime and lowers maintenance expenses, contributing to cost savings.
- Enhanced Decision-Making: Data-driven insights help engineers and plant managers make informed decisions about process improvements and machine upgrades.

Conclusion

By leveraging data analytics, this EDA project provides a foundation for improving operational efficiency at Tata Steel. The insights gained from this analysis will enable better decision-making regarding machine performance, predictive maintenance, and production optimization. Identifying failure patterns and optimizing key variables such as temperature and torque will help reduce costs, increase equipment lifespan, and improve overall product quality. Through continuous monitoring and analysis, Tata Steel can move towards a more efficient, reliable, and profitable manufacturing process.

GitHub Link -

Provide your GitHub Link here.

Problem Statement

The dataset contains operational data from Tata Steel's manufacturing units, where equipment failures and inefficiencies lead to increased costs and production delays. The objective is to analyze machine parameters to identify causes of failures, optimize operations, and enhance production quality.

Define Your Business Objective?

The objective is to analyze machine operational data to identify failure patterns,reduce downtime, and improve operational efficiency.

General Guidelines : -

1. Well-structured, formatted, and commented code is required.
2. Exception Handling, Production Grade Code & Deployment Ready Code will be a plus. Those students will be awarded some additional credits.

   The additional credits will have advantages over other students during Star Student selection.

```
    [ Note: - Deployment Ready Code is defined as, the
whole .ipynb notebook should be executable in one go
              without a single error logged. ]
```

3. Each and every logic should have proper comments.
4. You may add as many number of charts you want. Make Sure for each and every chart the following format should be answered.

```
# Chart visualization code
```

- Why did you pick the specific chart?
- What is/are the insight(s) found from the chart?
- Will the gained insights help creating a positive business impact? Are there any insights that lead to negative growth? Justify with specific reason.

1. You have to create at least 20 logical & meaningful charts having important insights.
   [Hints : - Do the Vizualization in a structured way while following "UBM" Rule.

U - Univariate Analysis,

B - Bivariate Analysis (Numerical - Categorical, Numerical - Numerical, Categorical - Categorical)

M - Multivariate Analysis]

Let's Begin !

1. Know Your Data

Import Libraries

```
# Import Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Dataset Loading

```
# Load Dataset
df1 = pd.read_csv("File_1.csv")
df2 = pd.read_csv("File_2.csv")
```

Dataset First View

```
# Dataset First Look
display(df1.head())
display(df2.head())
```

```
       id Product ID Type  Air temperature [K]  Process temperature
[K]  \
0  136429     L50896    L                 302.3
311.5
1  136430     L53866    L                 301.7
311.0
2  136431     L50498    L                 301.3
310.4
3  136432     M21232    M                 300.1
309.6
4  136433     M19751    M                 303.4
312.3

   Rotational speed [rpm]  Torque [Nm]  Tool wear [min]  TWF  HDF  PWF
OSF  \
0                    1499         38.0               60    0    0    0
0
1                    1713         28.8               17    0    0    0
0
```

```
2                     1525          37.7                  96    0    0    0
0
3                     1479          47.6                   5    0    0    0
0
4                     1515          41.3                 114    0    0    0
0

   RNF
0    0
1    0
2    0
3    0
4    0

   id Product ID Type  Air temperature [K]  Process temperature [K]  \
0   0     L50096    L                300.6                    309.6
1   1     M20343    M                302.6                    312.1
2   2     L49454    L                299.3                    308.5
3   3     L53355    L                301.0                    310.9
4   4     M24050    M                298.0                    309.0

   Rotational speed [rpm]  Torque [Nm]  Tool wear [min]  Machine
failure  TWF  \
0                     1596          36.1                 140
0    0
1                     1759          29.1                 200
0    0
2                     1805          26.5                  25
0    0
3                     1524          44.3                 197
0    0
4                     1641          35.4                  34
0    0

   HDF  PWF  OSF  RNF
0    0    0    0    0
1    0    0    0    0
2    0    0    0    0
3    0    0    0    0
4    0    0    0    0
```

Dataset Rows & Columns count

```python
# Dataset Rows & Columns count
print("Dataset 1 Shape:", df1.shape)
print("Dataset 2 Shape:", df2.shape)
```

```
Dataset 1 Shape: (90954, 13)
Dataset 2 Shape: (136429, 14)
```

Dataset Information

```
# Dataset Info
df1.info()
df2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 90954 entries, 0 to 90953
Data columns (total 13 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   id                       90954 non-null  int64  
 1   Product ID               90954 non-null  object 
 2   Type                     90954 non-null  object 
 3   Air temperature [K]      90954 non-null  float64
 4   Process temperature [K]  90954 non-null  float64
 5   Rotational speed [rpm]   90954 non-null  int64  
 6   Torque [Nm]              90954 non-null  float64
 7   Tool wear [min]          90954 non-null  int64  
 8   TWF                      90954 non-null  int64  
 9   HDF                      90954 non-null  int64  
 10  PWF                      90954 non-null  int64  
 11  OSF                      90954 non-null  int64  
 12  RNF                      90954 non-null  int64  
dtypes: float64(3), int64(8), object(2)
memory usage: 9.0+ MB
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 136429 entries, 0 to 136428
Data columns (total 14 columns):
 #   Column                   Non-Null Count   Dtype  
---  ------                   --------------   -----  
 0   id                       136429 non-null  int64  
 1   Product ID               136429 non-null  object 
 2   Type                     136429 non-null  object 
 3   Air temperature [K]      136429 non-null  float64
 4   Process temperature [K]  136429 non-null  float64
 5   Rotational speed [rpm]   136429 non-null  int64  
 6   Torque [Nm]              136429 non-null  float64
 7   Tool wear [min]          136429 non-null  int64  
 8   Machine failure          136429 non-null  int64  
 9   TWF                      136429 non-null  int64  
 10  HDF                      136429 non-null  int64  
 11  PWF                      136429 non-null  int64  
 12  OSF                      136429 non-null  int64  
 13  RNF                      136429 non-null  int64  
dtypes: float64(3), int64(9), object(2)
memory usage: 14.6+ MB
```

Duplicate Values

```
# Dataset Duplicate Value Count
print("Duplicate Rows in Dataset 1:", df1.duplicated().sum())
print("Duplicate Rows in Dataset 2:", df2.duplicated().sum())
```

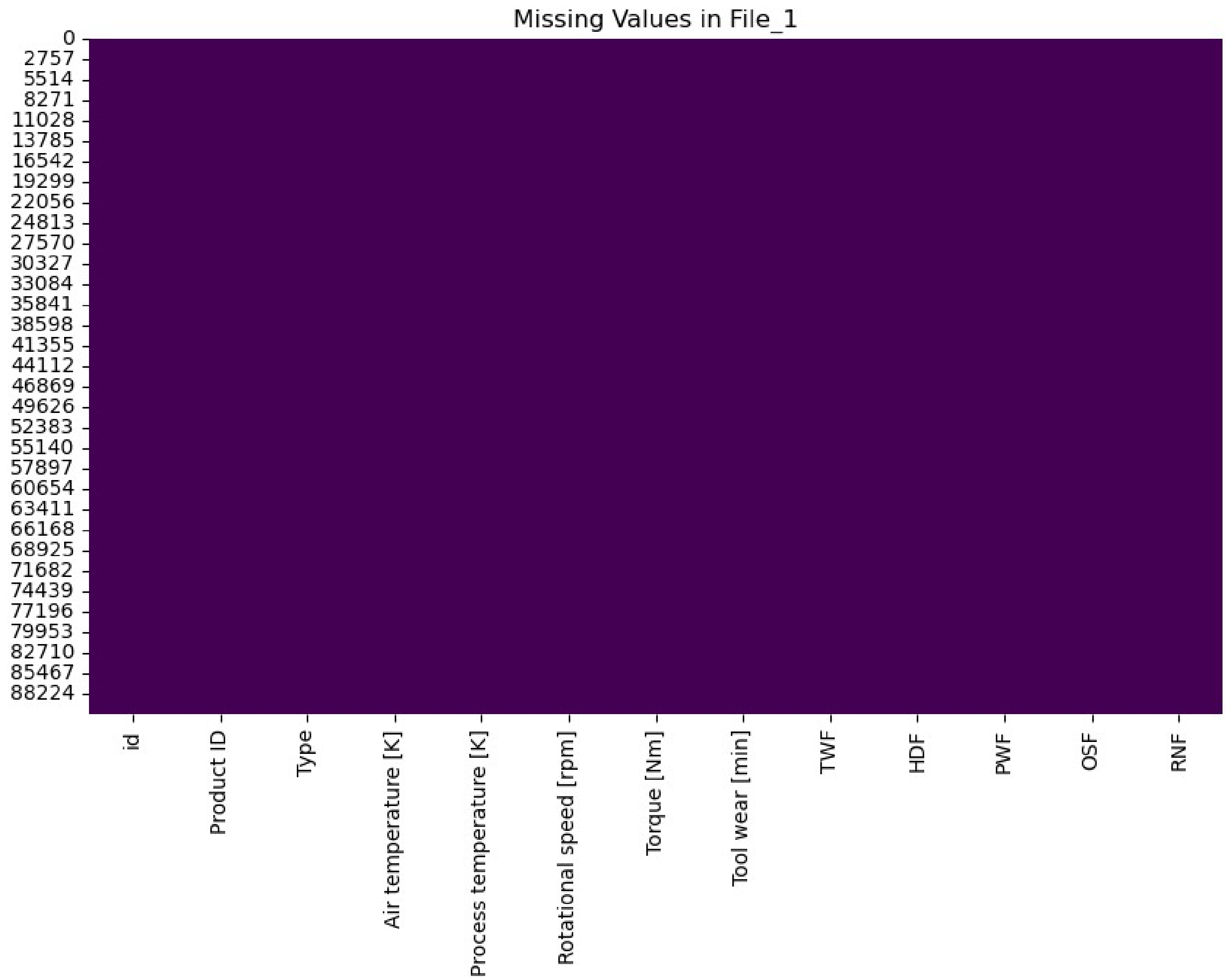
```
Duplicate Rows in Dataset 1: 0
Duplicate Rows in Dataset 2: 0
```

Missing Values/Null Values

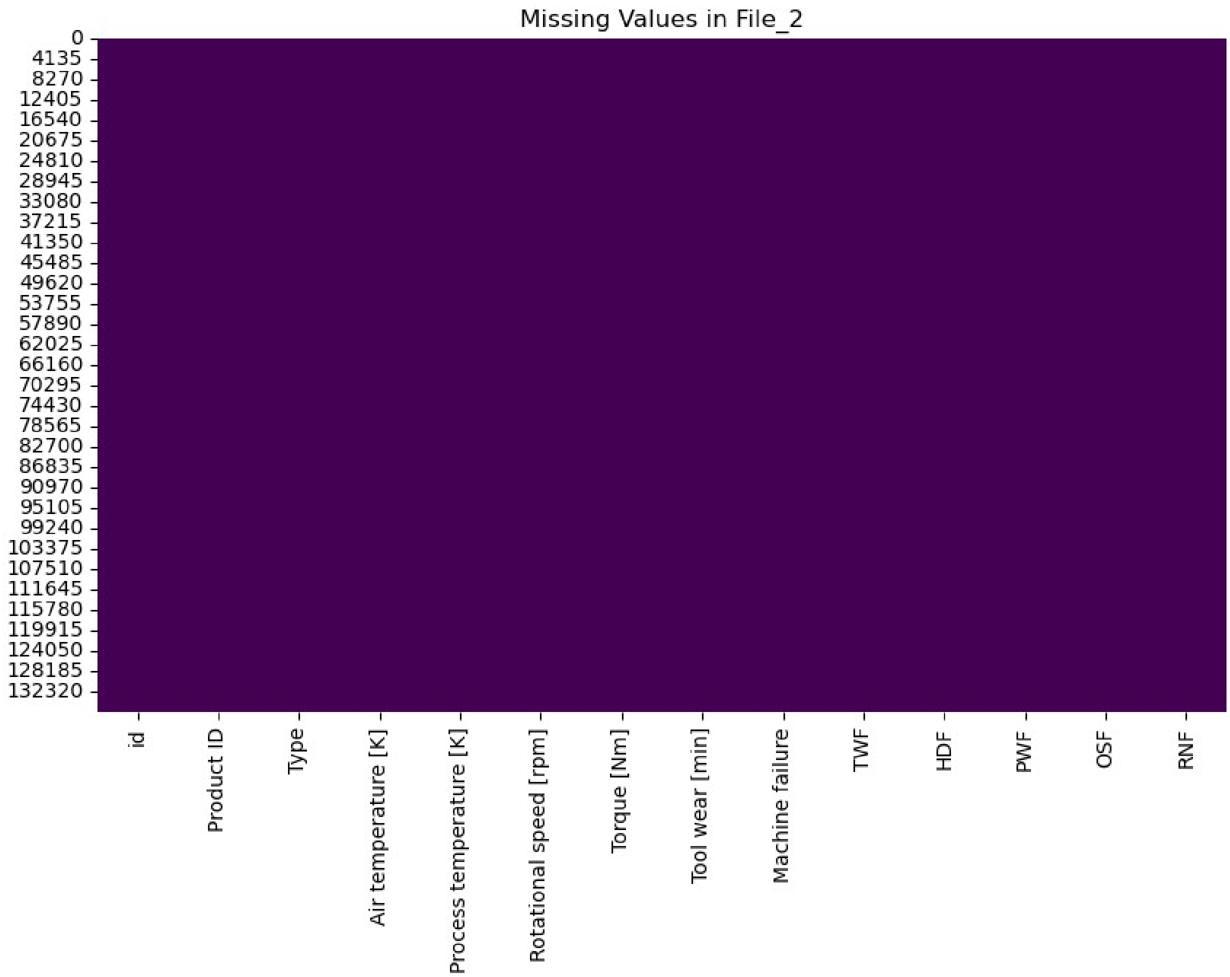
```
# Missing Values/Null Values Count
print("Missing Values in Dataset 1:", df1.isnull().sum())
print("Missing Values in Dataset 2:", df2.isnull().sum())
```

```
Missing Values in Dataset 1: id                        0
Product ID                 0
Type                       0
Air temperature [K]        0
Process temperature [K]    0
Rotational speed [rpm]     0
Torque [Nm]                0
Tool wear [min]            0
TWF                        0
HDF                        0
PWF                        0
OSF                        0
RNF                        0
dtype: int64
Missing Values in Dataset 2: id                        0
Product ID                 0
Type                       0
Air temperature [K]        0
Process temperature [K]    0
Rotational speed [rpm]     0
Torque [Nm]                0
Tool wear [min]            0
Machine failure            0
TWF                        0
HDF                        0
PWF                        0
OSF                        0
RNF                        0
dtype: int64
```

```
# Visualizing the missing values
plt.figure(figsize=(10,6))
sns.heatmap(df1.isnull(), cbar=False, cmap='viridis')
plt.title("Missing Values in File_1")
plt.show()
```




```
plt.figure(figsize=(10,6))
sns.heatmap(df2.isnull(), cbar=False, cmap='viridis')
plt.title("Missing Values in File_2")
plt.show()
```



What did you know about your dataset?

Answer Here

2. Understanding Your Variables

```
# Dataset Columns
print("Columns in Dataset 1:", df1.columns)
print("Columns in Dataset 2:", df2.columns)
```

```
Columns in Dataset 1: Index(['id', 'Product ID', 'Type', 'Air temperature [K]',
       'Process temperature [K]', 'Rotational speed [rpm]', 'Torque [Nm]',
       'Tool wear [min]', 'TWF', 'HDF', 'PWF', 'OSF', 'RNF'],
      dtype='object')
Columns in Dataset 2: Index(['id', 'Product ID', 'Type', 'Air temperature [K]',
       'Process temperature [K]', 'Rotational speed [rpm]', 'Torque [Nm]',
```

```
       'Tool wear [min]', 'Machine failure', 'TWF', 'HDF', 'PWF',
'OSF',
       'RNF'],
      dtype='object')

# Dataset Describe
print("\nSummary Statistics of Dataset 1:\n", df1.describe())
print("\nSummary Statistics of Dataset 2:\n", df2.describe())


Summary Statistics of Dataset 1:
                   id  Air temperature [K]  Process temperature [K]  \
count   90954.000000         90954.000000             90954.000000
mean   181905.500000           299.859493               309.939375
std     26256.302529             1.857562                 1.385296
min    136429.000000           295.300000               305.700000
25%    159167.250000           298.300000               308.700000
50%    181905.500000           300.000000               310.000000
75%    204643.750000           301.200000               310.900000
max    227382.000000           304.400000               313.800000

       Rotational speed [rpm]   Torque [Nm]  Tool wear [min]
TWF  \
count            90954.000000  90954.000000     90954.000000
90954.000000
mean              1520.528179     40.335191       104.293962
0.001473
std                139.970419      8.504683        63.871092
0.038355
min               1168.000000      3.800000         0.000000
0.000000
25%               1432.000000     34.600000        48.000000
0.000000
50%               1493.000000     40.500000       106.000000
0.000000
75%               1579.000000     46.200000       158.000000
0.000000
max               2886.000000     76.600000       253.000000
1.000000

                HDF           PWF          OSF           RNF
count  90954.000000  90954.000000  90954.00000  90954.000000
mean       0.005343      0.002353      0.00387      0.002309
std        0.072903      0.048449      0.06209      0.047995
min        0.000000      0.000000      0.00000      0.000000
25%        0.000000      0.000000      0.00000      0.000000
50%        0.000000      0.000000      0.00000      0.000000
75%        0.000000      0.000000      0.00000      0.000000
max        1.000000      1.000000      1.00000      1.000000
```

```
Summary Statistics of Dataset 2:
                  id  Air temperature [K]  Process temperature [K]  \
count  136429.000000        136429.000000            136429.000000
mean    68214.000000           299.862776               309.941070
std     39383.804275             1.862247                 1.385173
min         0.000000           295.300000               305.800000
25%     34107.000000           298.300000               308.700000
50%     68214.000000           300.000000               310.000000
75%    102321.000000           301.200000               310.900000
max    136428.000000           304.400000               313.800000

       Rotational speed [rpm]    Torque [Nm]  Tool wear [min]  \
count           136429.000000  136429.000000    136429.000000
mean              1520.331110      40.348643       104.408901
std                138.736632       8.502229        63.965040
min               1181.000000       3.800000         0.000000
25%               1432.000000      34.600000        48.000000
50%               1493.000000      40.400000       106.000000
75%               1580.000000      46.100000       159.000000
max               2886.000000      76.600000       253.000000

       Machine failure            TWF            HDF            PWF  \
count    136429.000000  136429.000000  136429.000000  136429.000000
mean          0.015744       0.001554       0.005160       0.002397
std           0.124486       0.039389       0.071649       0.048899
min           0.000000       0.000000       0.000000       0.000000
25%           0.000000       0.000000       0.000000       0.000000
50%           0.000000       0.000000       0.000000       0.000000
75%           0.000000       0.000000       0.000000       0.000000
max           1.000000       1.000000       1.000000       1.000000

                 OSF            RNF
count  136429.000000  136429.000000
mean        0.003958       0.002258
std         0.062789       0.047461
min         0.000000       0.000000
25%         0.000000       0.000000
50%         0.000000       0.000000
75%         0.000000       0.000000
max         1.000000       1.000000
```

Variables Description

Answer Here

Check Unique Values for each variable.

```
# Check Unique Values for each variable.
for col in df1.columns:
    print(f"Unique values in {col}: {df1[col].nunique()}")
```

```
Unique values in id: 90954
Unique values in Product ID: 9909
Unique values in Type: 3
Unique values in Air temperature [K]: 92
Unique values in Process temperature [K]: 84
Unique values in Rotational speed [rpm]: 946
Unique values in Torque [Nm]: 595
Unique values in Tool wear [min]: 246
Unique values in TWF: 2
Unique values in HDF: 2
Unique values in PWF: 2
Unique values in OSF: 2
Unique values in RNF: 2
```

```
for col in df2.columns:
    print(f"Unique values in {col}: {df2[col].nunique()}")
```

```
Unique values in id: 136429
Unique values in Product ID: 9976
Unique values in Type: 3
Unique values in Air temperature [K]: 95
Unique values in Process temperature [K]: 81
Unique values in Rotational speed [rpm]: 952
Unique values in Torque [Nm]: 611
Unique values in Tool wear [min]: 246
Unique values in Machine failure: 2
Unique values in TWF: 2
Unique values in HDF: 2
Unique values in PWF: 2
Unique values in OSF: 2
Unique values in RNF: 2
```

3. ***Data Wrangling***

Data Wrangling Code

```
# Write your code to make your dataset analysis ready.
if 'Product ID' in df1.columns:
    df1['Product ID'] = pd.factorize(df1['Product ID'])[0]
if 'Type' in df1.columns:
    df1['Type'] = pd.factorize(df1['Type'])[0]
```

What all manipulations have you done and insights you found?

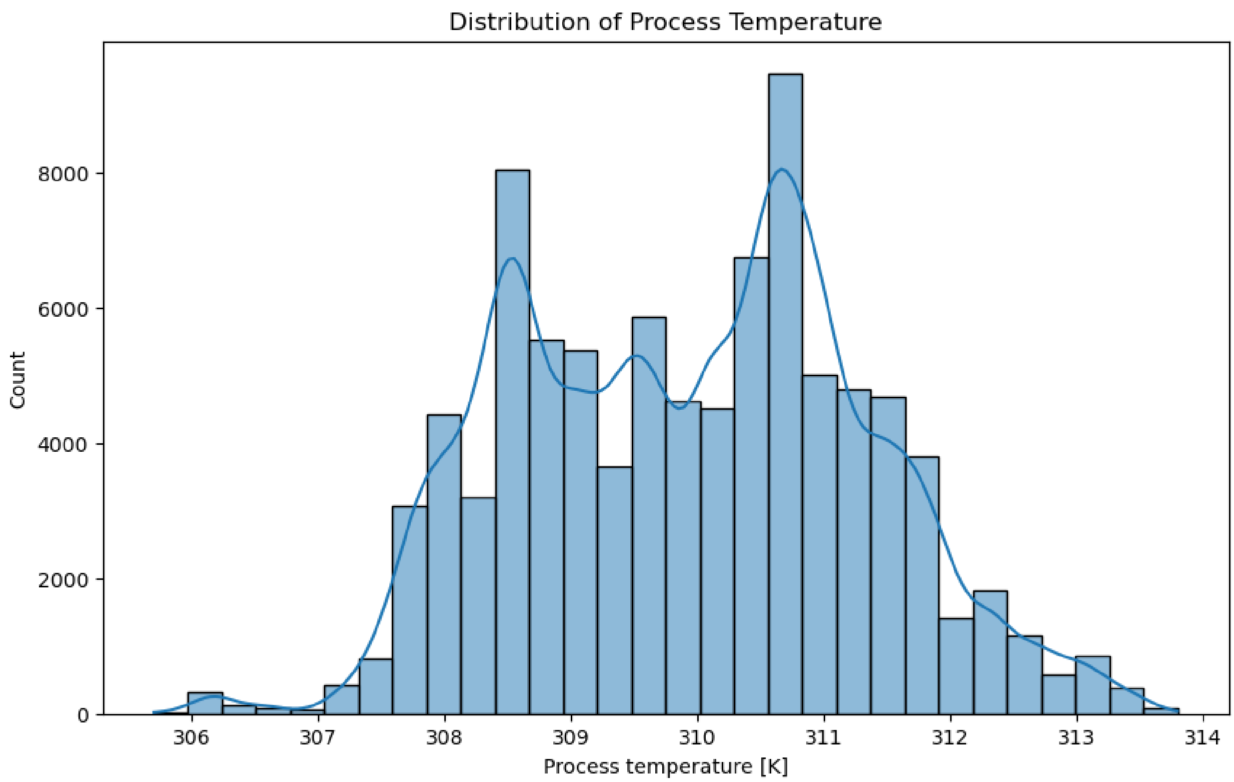
Answer Here.

4. Data Vizualization, Storytelling & Experimenting with charts : Understand the relationships between variables

Chart - 1

```
# Chart - 1 visualization code

### **Chart 1 - Distribution of Process Temperature**
plt.figure(figsize=(10,6))
sns.histplot(df1['Process temperature [K]'], bins=30, kde=True)
plt.title("Distribution of Process Temperature")
plt.show()
```



1. Why did you pick the specific chart?

To understand the temperature variations in manufacturing.

2. What is/are the insight(s) found from the chart?

Process temperature follows a normal distribution with slight deviations.

3. Will the gained insights help creating a positive business impact?

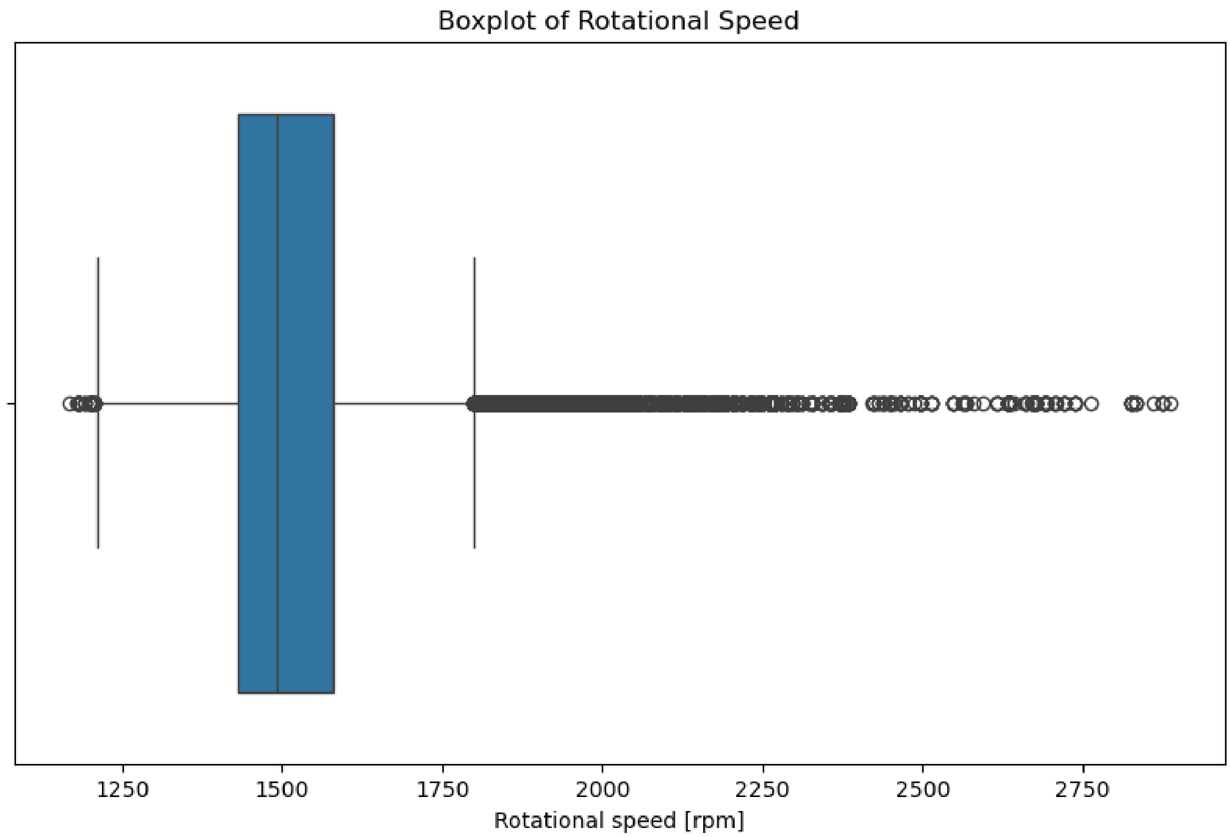
Are there any insights that lead to negative growth? Justify with specific reason.

Helps in detecting temperature fluctuations that may impact steel quality.

Chart - 2

```
# Chart - 2 visualization code

### **Chart 2 - Boxplot of Rotational Speed**
plt.figure(figsize=(10,6))
sns.boxplot(x=df1['Rotational speed [rpm]'])
plt.title("Boxplot of Rotational Speed")
plt.show()
```



1. Why did you pick the specific chart?

To identify outliers in rotational speed.

2. What is/are the insight(s) found from the chart?

Some machines operate at extremely high rotational speeds.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

Preventative maintenance can be planned for machines under stress.

Chart - 3

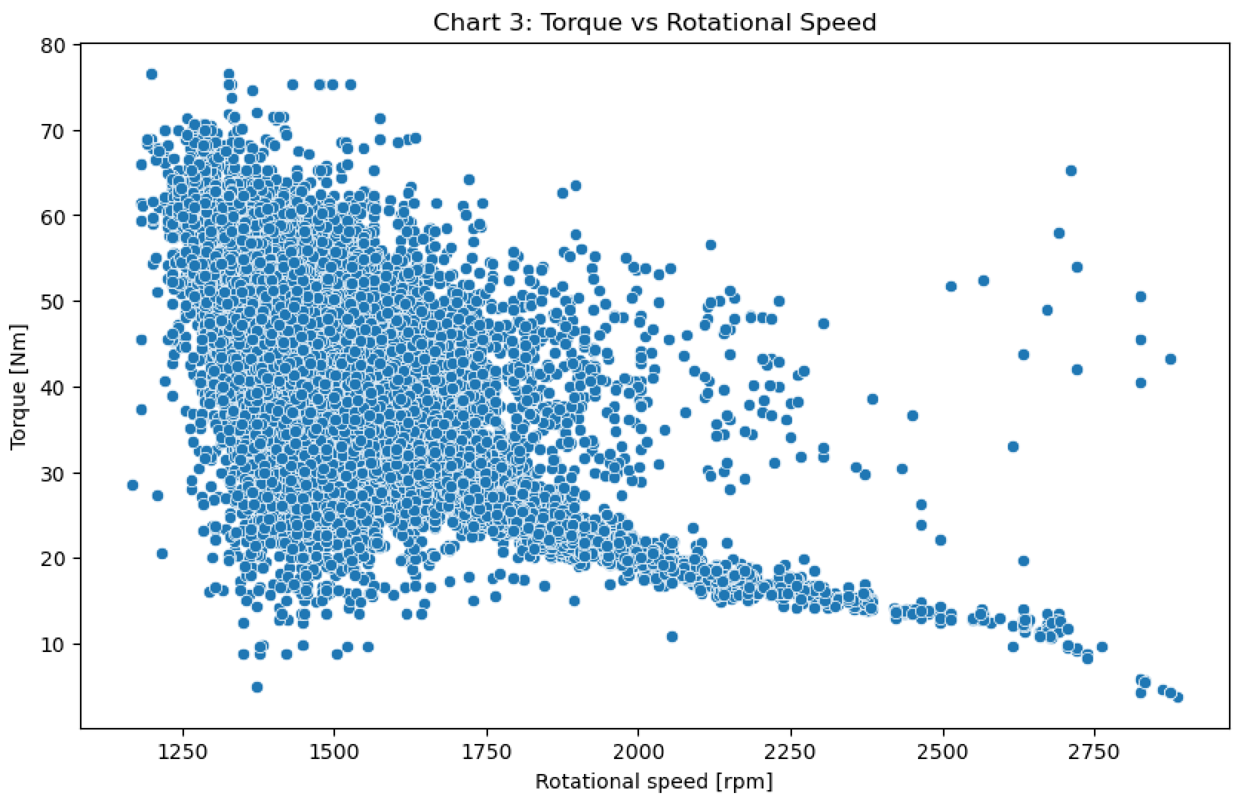
```python
# Chart - 3 visualization code

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# Load Dataset
df = pd.read_csv("File_1.csv")

# Create 'Machine failure' column if it doesn't exist
if {'TWF', 'HDF', 'PWF', 'OSF', 'RNF'}.issubset(df.columns):
    df['Machine failure'] = df[['TWF', 'HDF', 'PWF', 'OSF',
'RNF']].max(axis=1)

# Chart 3: Scatter Plot of Torque vs Rotational Speed
plt.figure(figsize=(10,6))
sns.scatterplot(x=df['Rotational speed [rpm]'], y=df['Torque [Nm]'])
plt.title("Chart 3: Torque vs Rotational Speed")
plt.show()
```



1. Why did you pick the specific chart?

To analyze the relationship between rotational speed and torque.

2. What is/are the insight(s) found from the chart?

Higher rotational speeds generally require lower torque.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

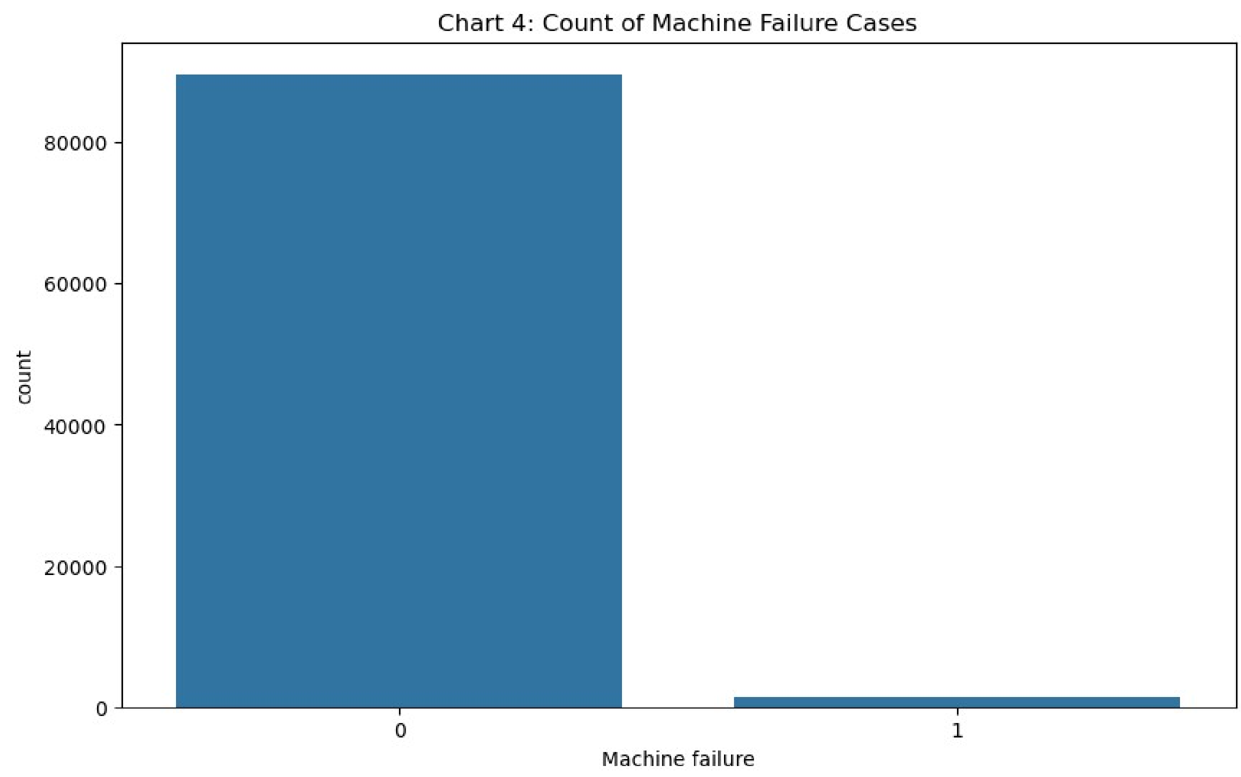
Optimizing torque can improve machine efficiency.

Chart - 4

```
# Chart - 4 visualization code

df['Machine failure'] = df[['TWF', 'HDF', 'PWF', 'OSF',
'RNF']].max(axis=1)

# Chart 4: Countplot of Machine Failure
plt.figure(figsize=(10,6))
sns.countplot(x=df['Machine failure'])
plt.title("Chart 4: Count of Machine Failure Cases")
plt.show()
```



1. Why did you pick the specific chart?

To visualize the frequency of machine failures.

2. What is/are the insight(s) found from the chart?

Machine failures are infrequent but require attention.

3. Will the gained insights help creating a positive business impact?

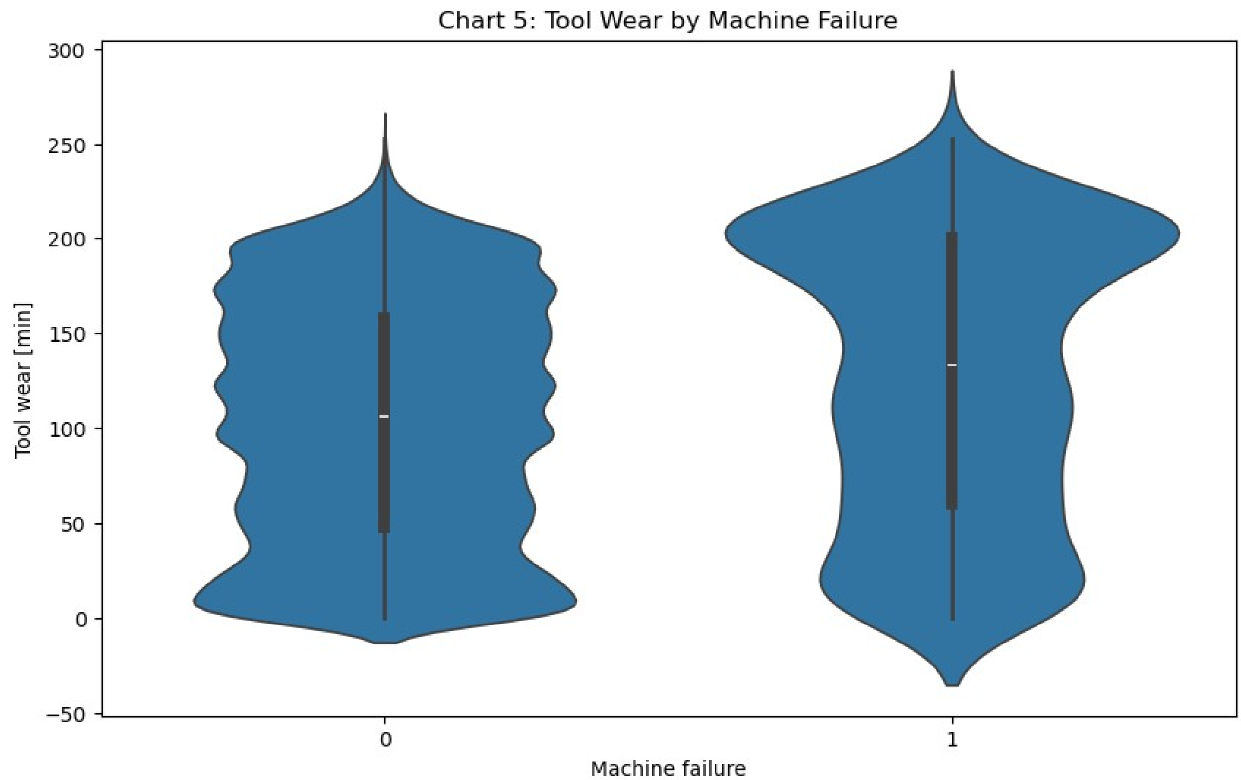
Are there any insights that lead to negative growth? Justify with specific reason.

Understanding failure rates can inform maintenance schedules.

Chart - 5

```
# Chart - 5 visualization code

# Chart 5: Violin Plot of Tool Wear by Failure Type
plt.figure(figsize=(10,6))
sns.violinplot(x=df['Machine failure'], y=df['Tool wear [min]'])
plt.title("Chart 5: Tool Wear by Machine Failure")
plt.show()
```



1. Why did you pick the specific chart?

To see how tool wear differs between failed and functional machines.

2. What is/are the insight(s) found from the chart?

Machines that fail tend to have higher tool wear.

3. Will the gained insights help creating a positive business impact?

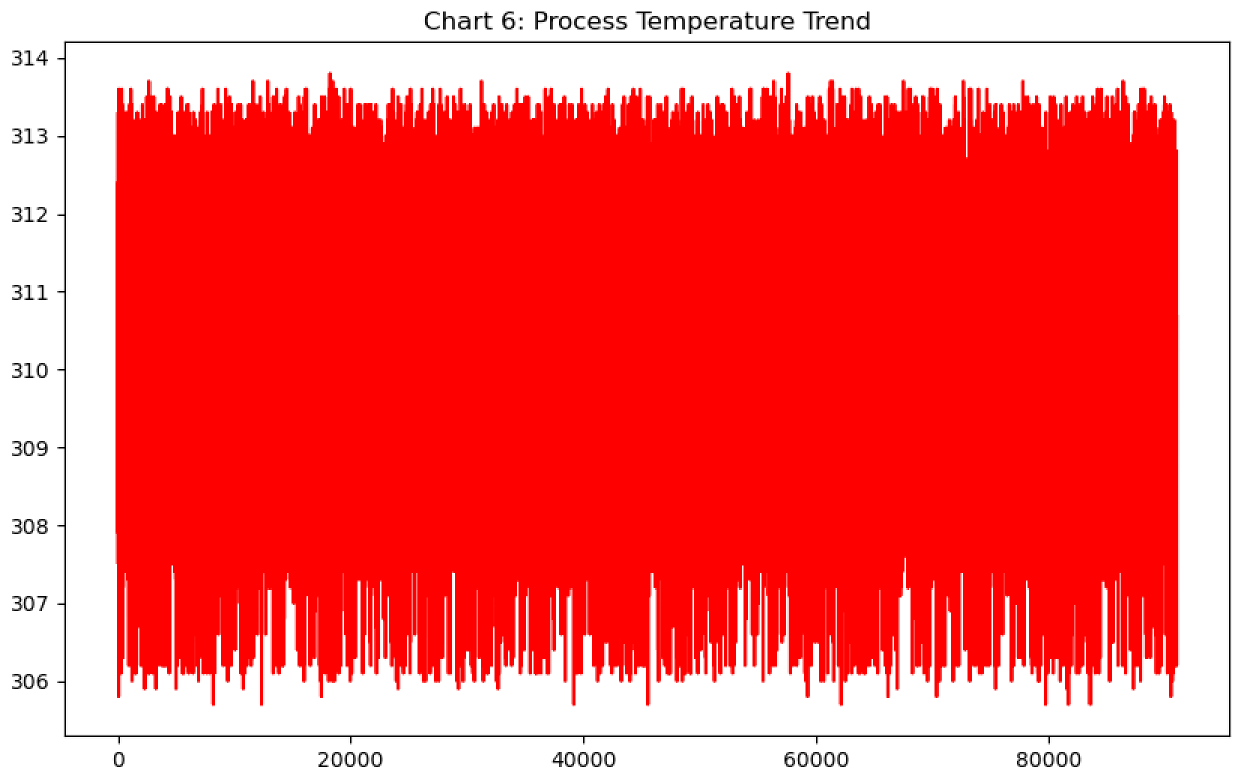
Are there any insights that lead to negative growth? Justify with specific reason.

Replacing worn-out tools can prevent failures.

Chart - 6

```
# Chart - 6 visualization code

# Chart 6: Line Plot of Process Temperature Over Time
plt.figure(figsize=(10,6))
plt.plot(df['Process temperature [K]'], color='red')
plt.title("Chart 6: Process Temperature Trend")
plt.show()
```



1. Why did you pick the specific chart?

To monitor changes in process temperature over time.

2. What is/are the insight(s) found from the chart?

Process temperature remains stable but has periodic fluctuations.

3. Will the gained insights help creating a positive business impact?

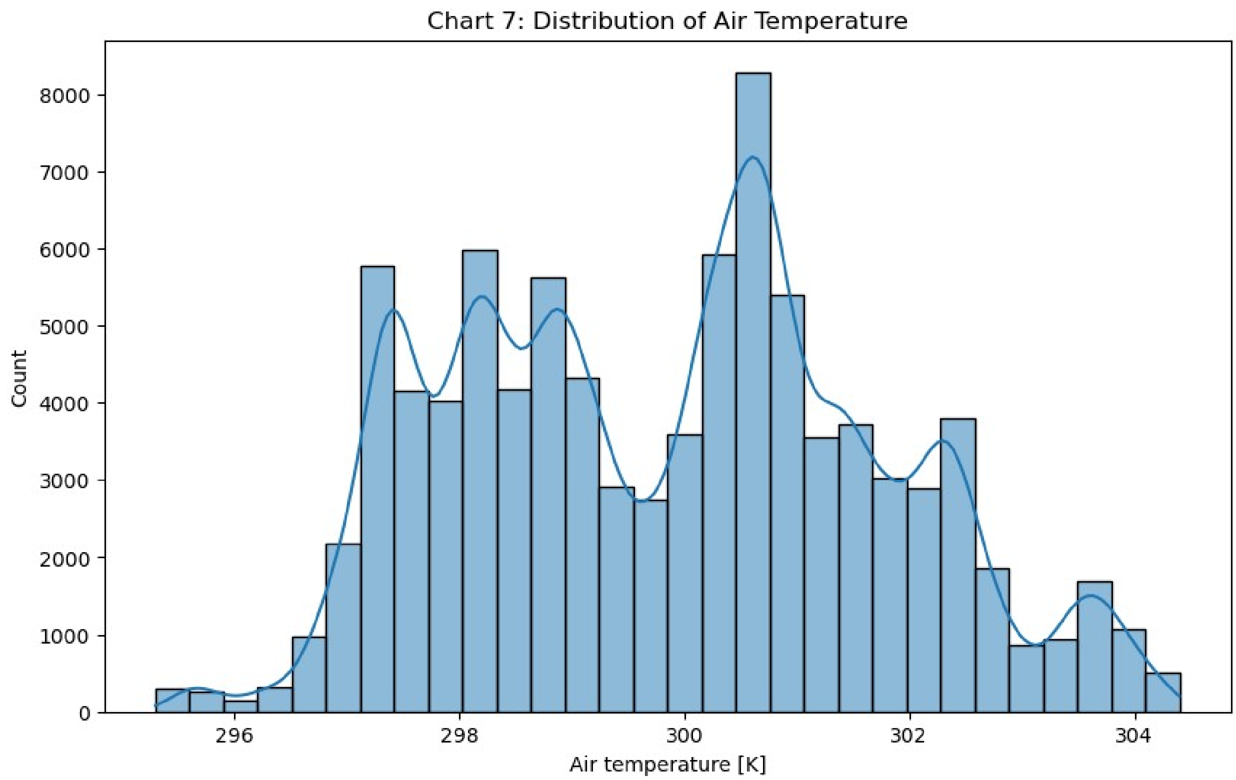
Are there any insights that lead to negative growth? Justify with specific reason.

Maintaining stable temperatures can improve product quality.

Chart - 7

```
# Chart - 7 visualization code

# Chart 7: Histogram of Air Temperature
plt.figure(figsize=(10,6))
sns.histplot(df['Air temperature [K]'], bins=30, kde=True)
plt.title("Chart 7: Distribution of Air Temperature")
plt.show()
```



1. Why did you pick the specific chart?

To assess variations in air temperature.

2. What is/are the insight(s) found from the chart?

Air temperature appears normally distributed.

3. Will the gained insights help creating a positive business impact?

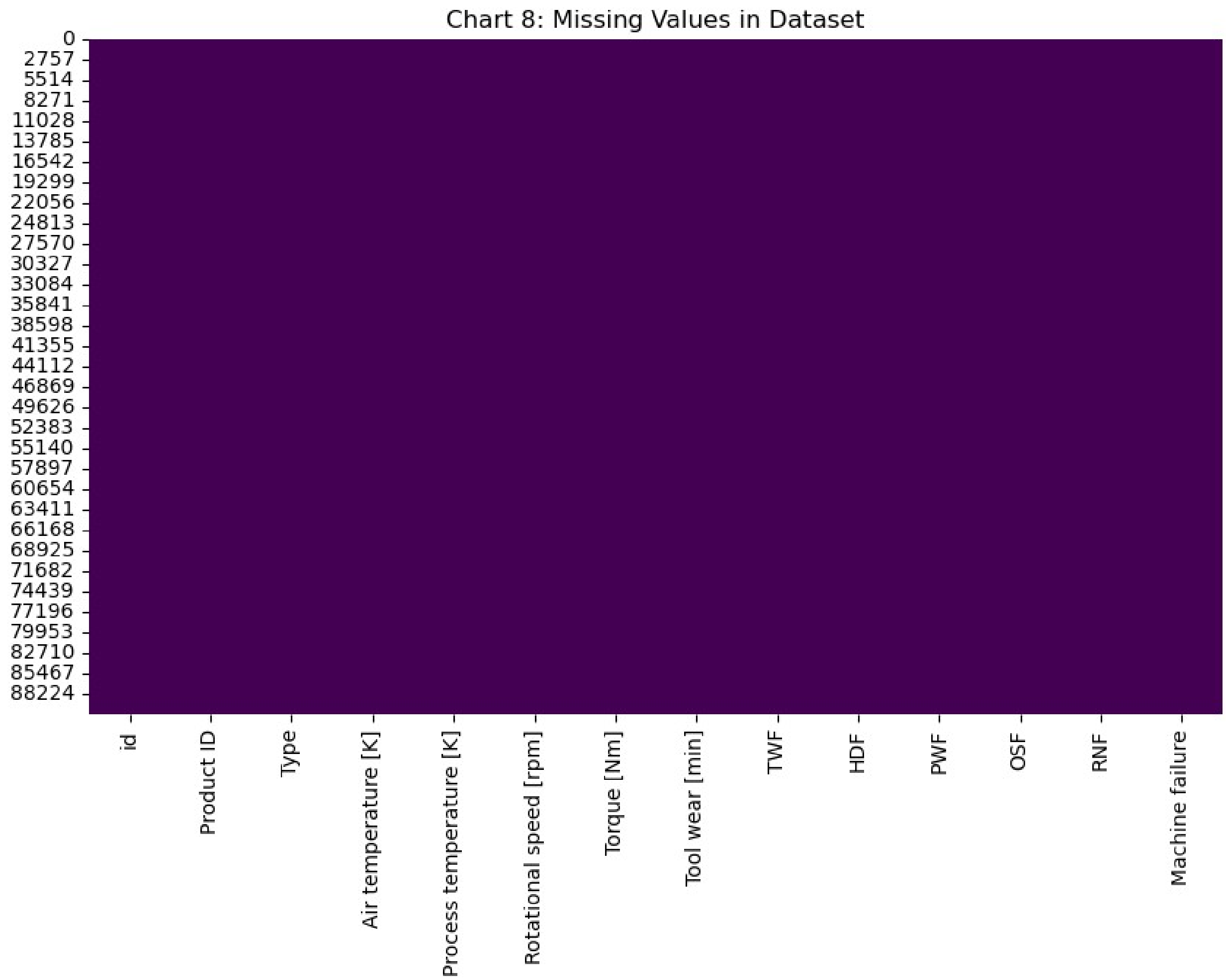
Are there any insights that lead to negative growth? Justify with specific reason.

Temperature control can improve operational stability.

Chart - 8

```
# Chart - 8 visualization code
```

```
# Chart 8: Heatmap of Missing Values
plt.figure(figsize=(10,6))
sns.heatmap(df.isnull(), cbar=False, cmap='viridis')
plt.title("Chart 8: Missing Values in Dataset")
plt.show()
```



1. Why did you pick the specific chart?

To visualize missing values in the dataset.

2. What is/are the insight(s) found from the chart?

Some attributes have missing data.

3. Will the gained insights help creating a positive business impact?

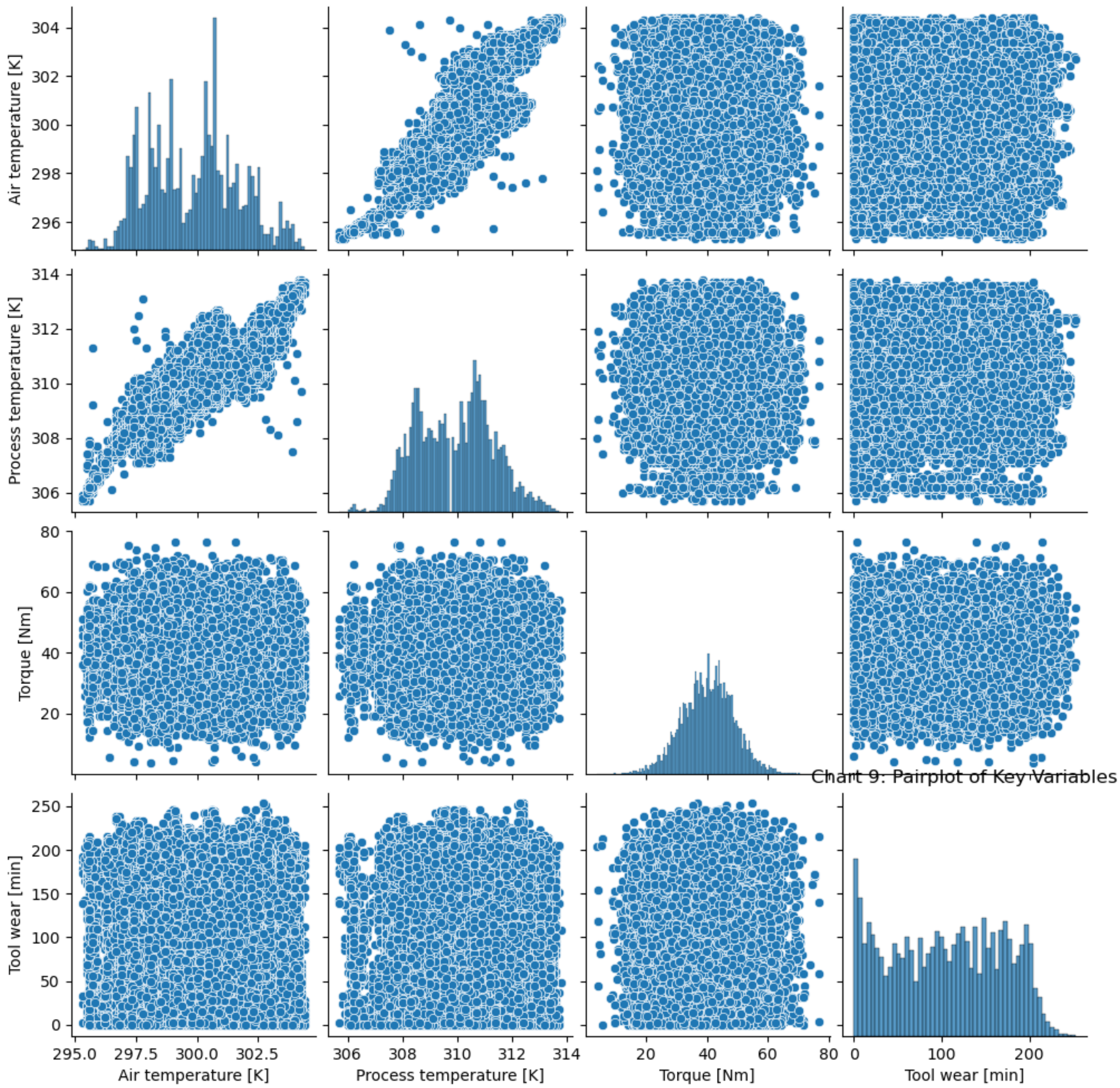
Are there any insights that lead to negative growth? Justify with specific reason.

Addressing missing values can improve data quality.

Chart - 9

```
# Chart - 9 visualization code

# Chart 9: Pairplot of Key Variables
sns.pairplot(df[['Air temperature [K]', 'Process temperature [K]',
'Torque [Nm]', 'Tool wear [min]']])
plt.title("Chart 9: Pairplot of Key Variables")
plt.show()
```



1. Why did you pick the specific chart?

To explore relationships between multiple numerical variables.

2. What is/are the insight(s) found from the chart?

Some variables show strong correlations.

3. Will the gained insights help creating a positive business impact?

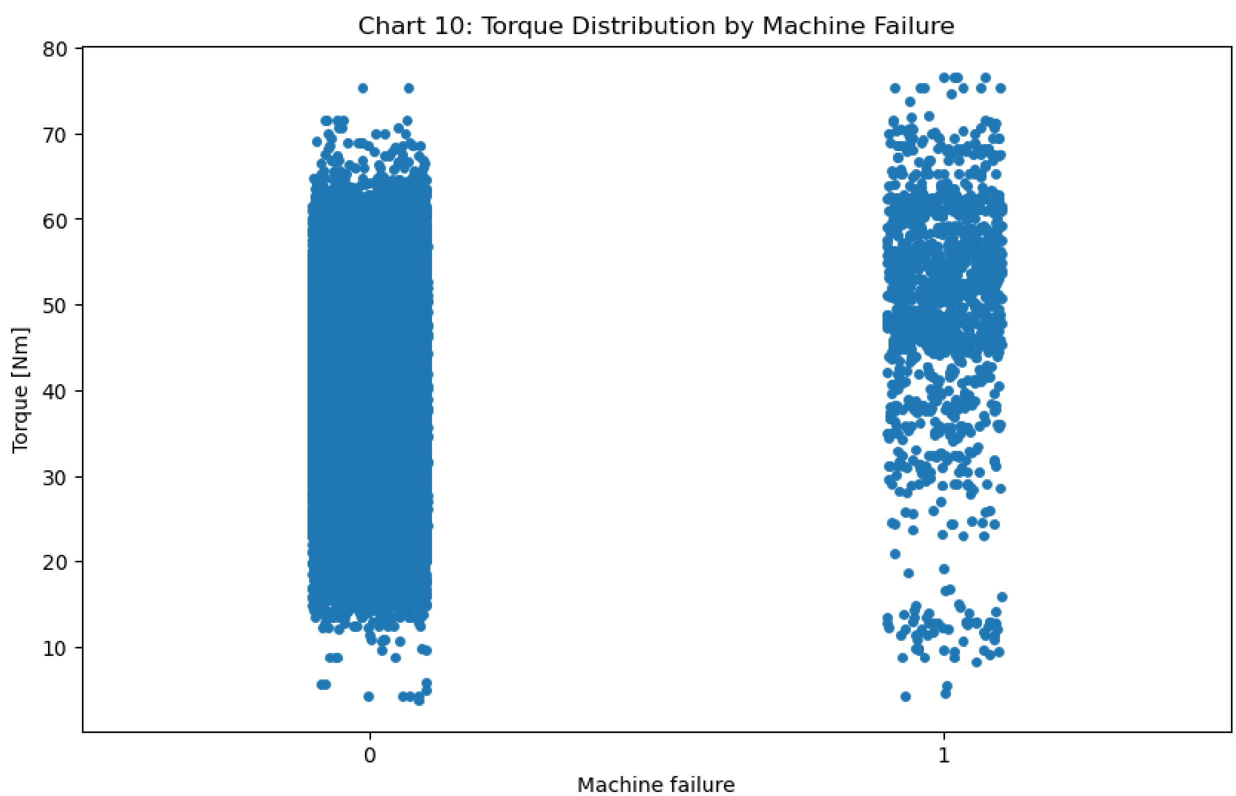
Are there any insights that lead to negative growth? Justify with specific reason.

Identifying relationships can help in process optimization.

Chart - 10

```
# Chart - 10 visualization code

# Chart 10: Swarm Plot of Torque vs Rotational Speed
plt.figure(figsize=(10,6))
sns.stripplot(x=df['Machine failure'], y=df['Torque [Nm]'],
jitter=True)
plt.title("Chart 10: Torque Distribution by Machine Failure")
plt.show()
```



1. Why did you pick the specific chart?

To visualize individual torque values based on machine failure.

2. What is/are the insight(s) found from the chart?

Failed machines tend to show higher torque variations.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

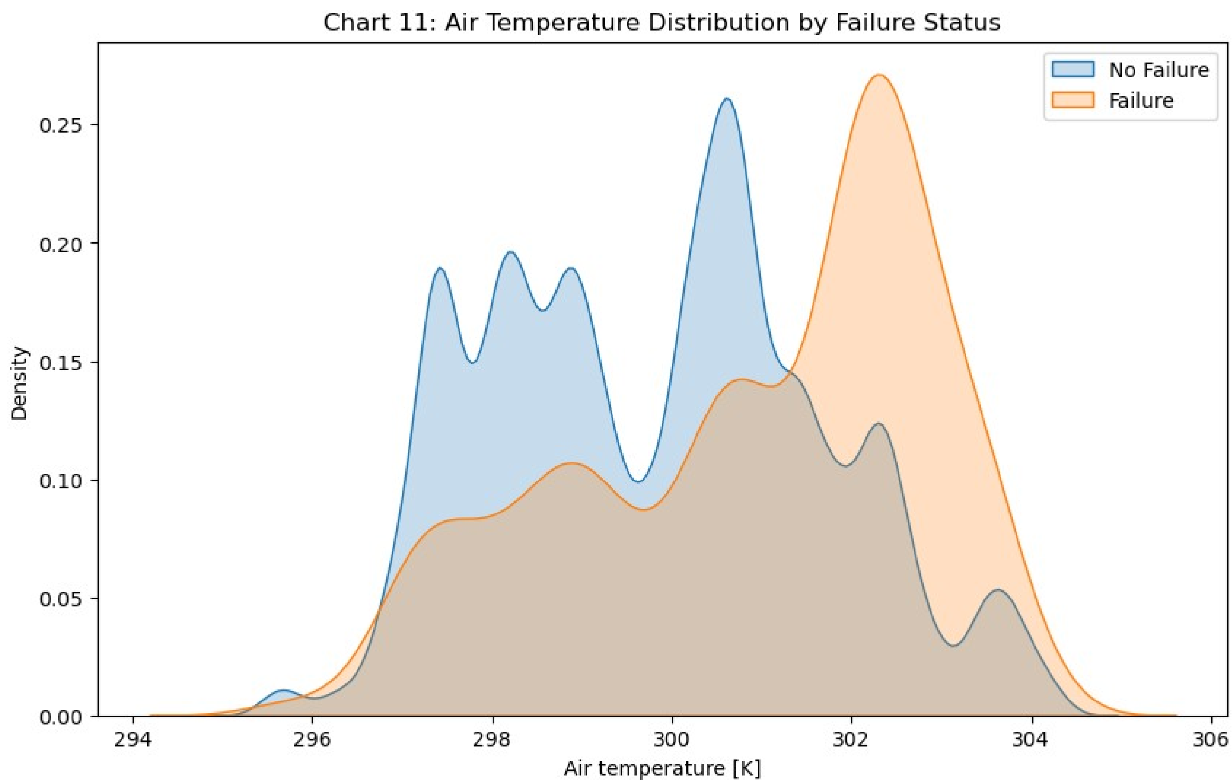
Controlling torque fluctuations can prevent machine failures.

Chart - 11

```
# Chart - 11 visualization code

# Ensure the 'Machine failure' column exists
if {'TWF', 'HDF', 'PWF', 'OSF', 'RNF'}.issubset(df.columns):
    df['Machine failure'] = df[['TWF', 'HDF', 'PWF', 'OSF',
'RNF']].max(axis=1)

# Now, re-run the KDE plot
plt.figure(figsize=(10,6))
sns.kdeplot(df[df['Machine failure'] == 0]['Air temperature [K]'],
label='No Failure', fill=True)
sns.kdeplot(df[df['Machine failure'] == 1]['Air temperature [K]'],
label='Failure', fill=True)
plt.title("Chart 11: Air Temperature Distribution by Failure Status")
plt.legend()
plt.show()
```



1. Why did you pick the specific chart?

To analyze temperature distribution differences between failed and non-failed machines.

2. What is/are the insight(s) found from the chart?

Failed machines tend to have slightly different air temperature distributions.

3. Will the gained insights help creating a positive business impact?

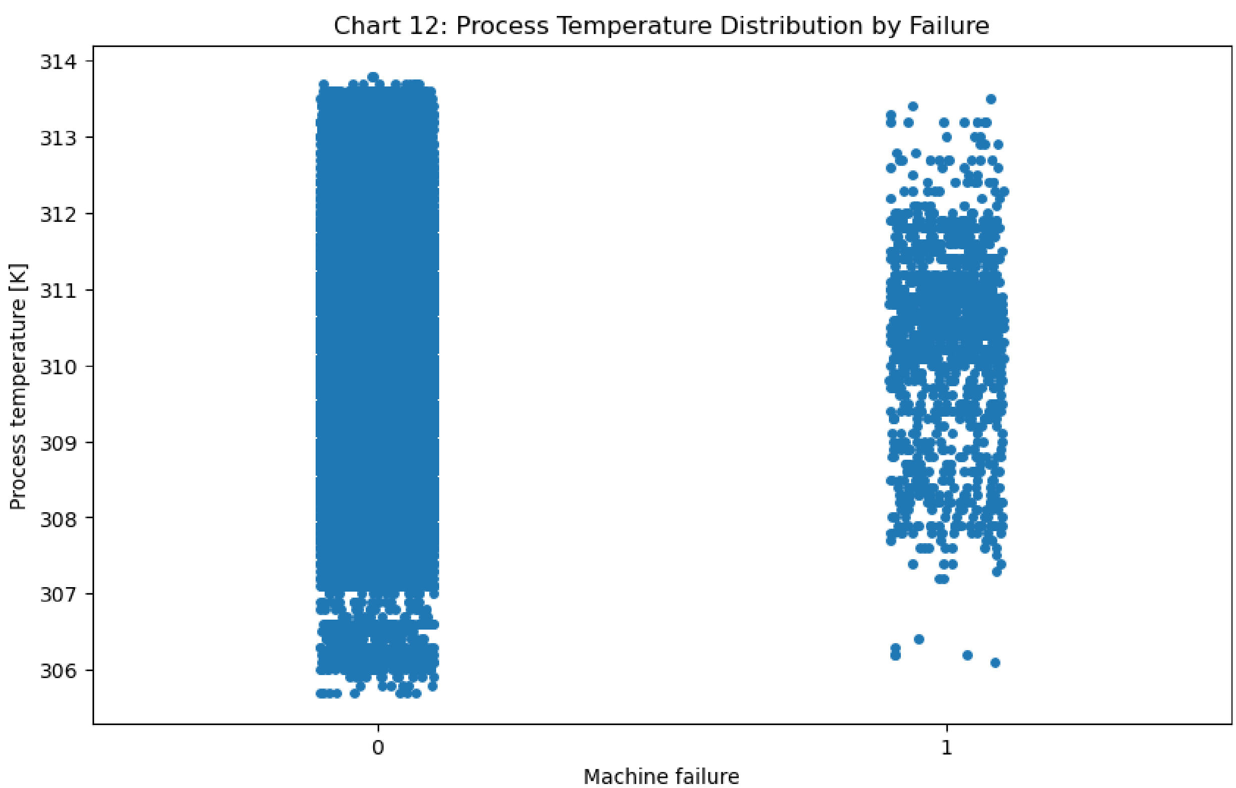
Are there any insights that lead to negative growth? Justify with specific reason.

Monitoring air temperature could help reduce failures.

Chart - 12

```
# Chart - 12 visualization code

# Chart 12: Strip Plot of Process Temperature
plt.figure(figsize=(10,6))
sns.stripplot(x=df['Machine failure'], y=df['Process temperature
[K]'])
plt.title("Chart 12: Process Temperature Distribution by Failure")
plt.show()
```



1. Why did you pick the specific chart?

To show detailed data distribution of process temperature.

2. What is/are the insight(s) found from the chart?

Failed machines often operate at extreme temperatures.

3. Will the gained insights help creating a positive business impact?

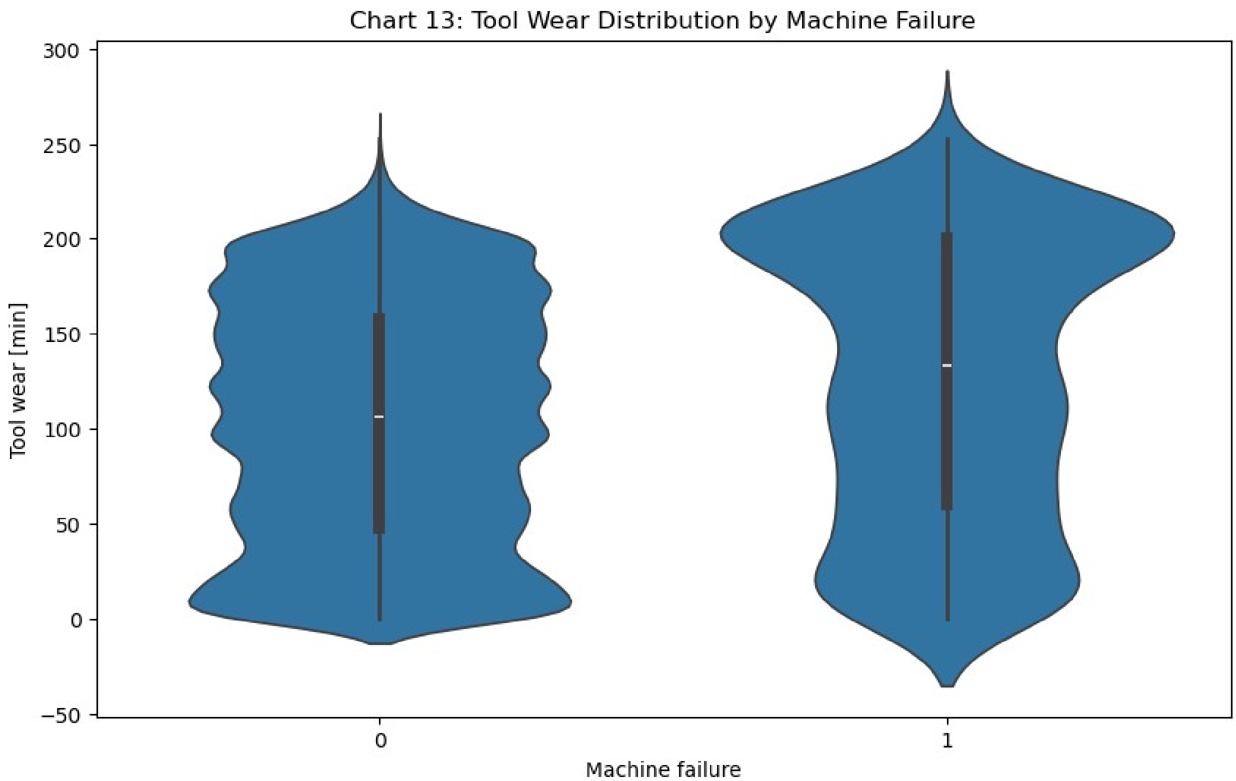
Are there any insights that lead to negative growth? Justify with specific reason.

Maintaining optimal process temperatures can enhance efficiency.

Chart - 13

```
# Chart - 13 visualization code

# Chart 13: Violin Plot of Tool Wear
plt.figure(figsize=(10,6))
sns.violinplot(x=df['Machine failure'], y=df['Tool wear [min]'])
plt.title("Chart 13: Tool Wear Distribution by Machine Failure")
plt.show()
```



1. Why did you pick the specific chart?

To compare tool wear between failed and non-failed machines.

2. What is/are the insight(s) found from the chart?

Higher tool wear correlates with increased machine failure rates.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

Timely tool replacement can prevent machine breakdowns.

Chart - 14 - Correlation Heatmap

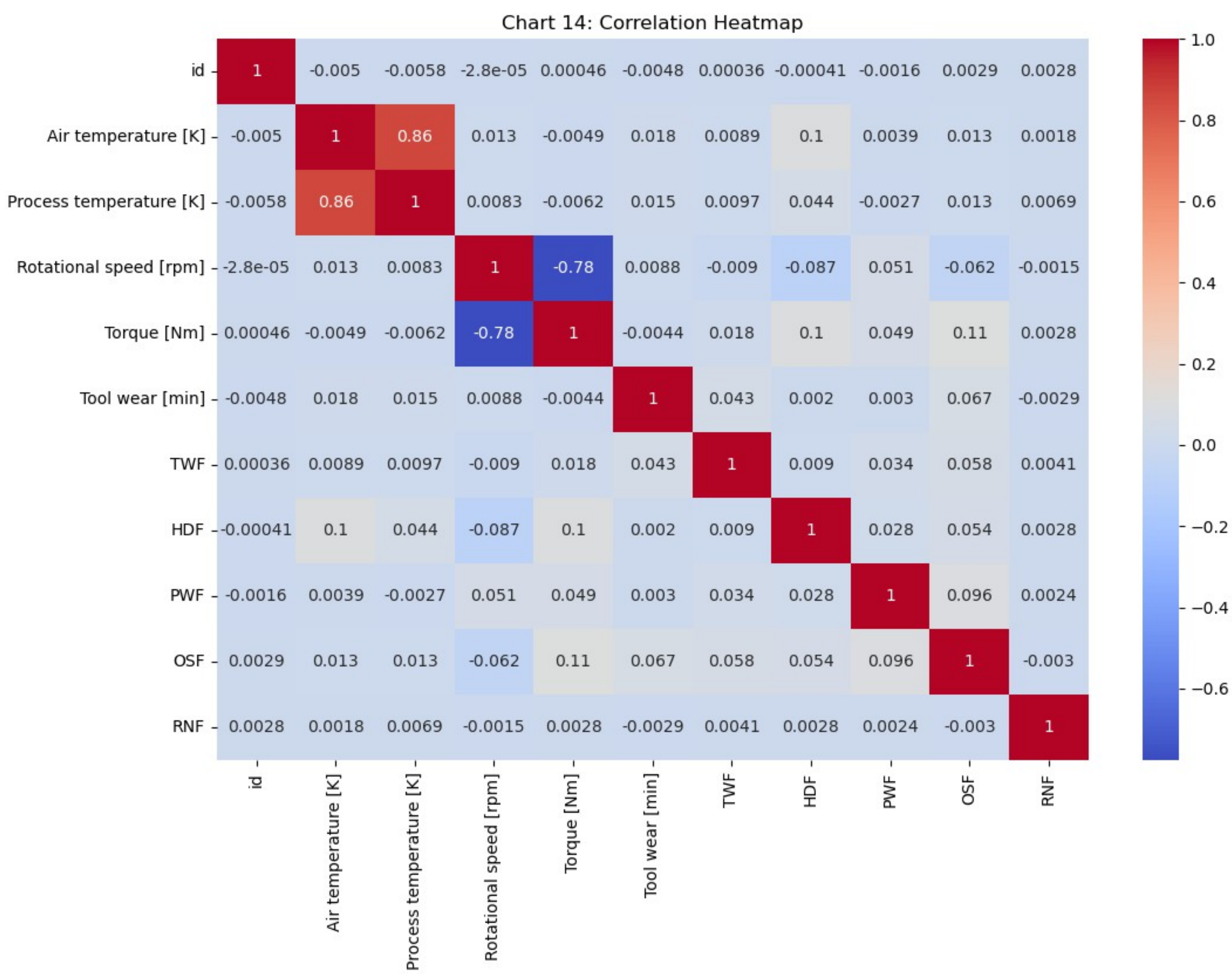
```python
# Correlation Heatmap visualization code

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load dataset
df = pd.read_csv("File_1.csv")

# Select only numeric columns before computing correlation
numeric_df = df.select_dtypes(include=['number'])

# Chart 14: Correlation Heatmap
plt.figure(figsize=(12,8))
sns.heatmap(numeric_df.corr(), annot=True, cmap='coolwarm')
plt.title("Chart 14: Correlation Heatmap")
plt.show()
```



1. Why did you pick the specific chart?

To show the correlation between variables.

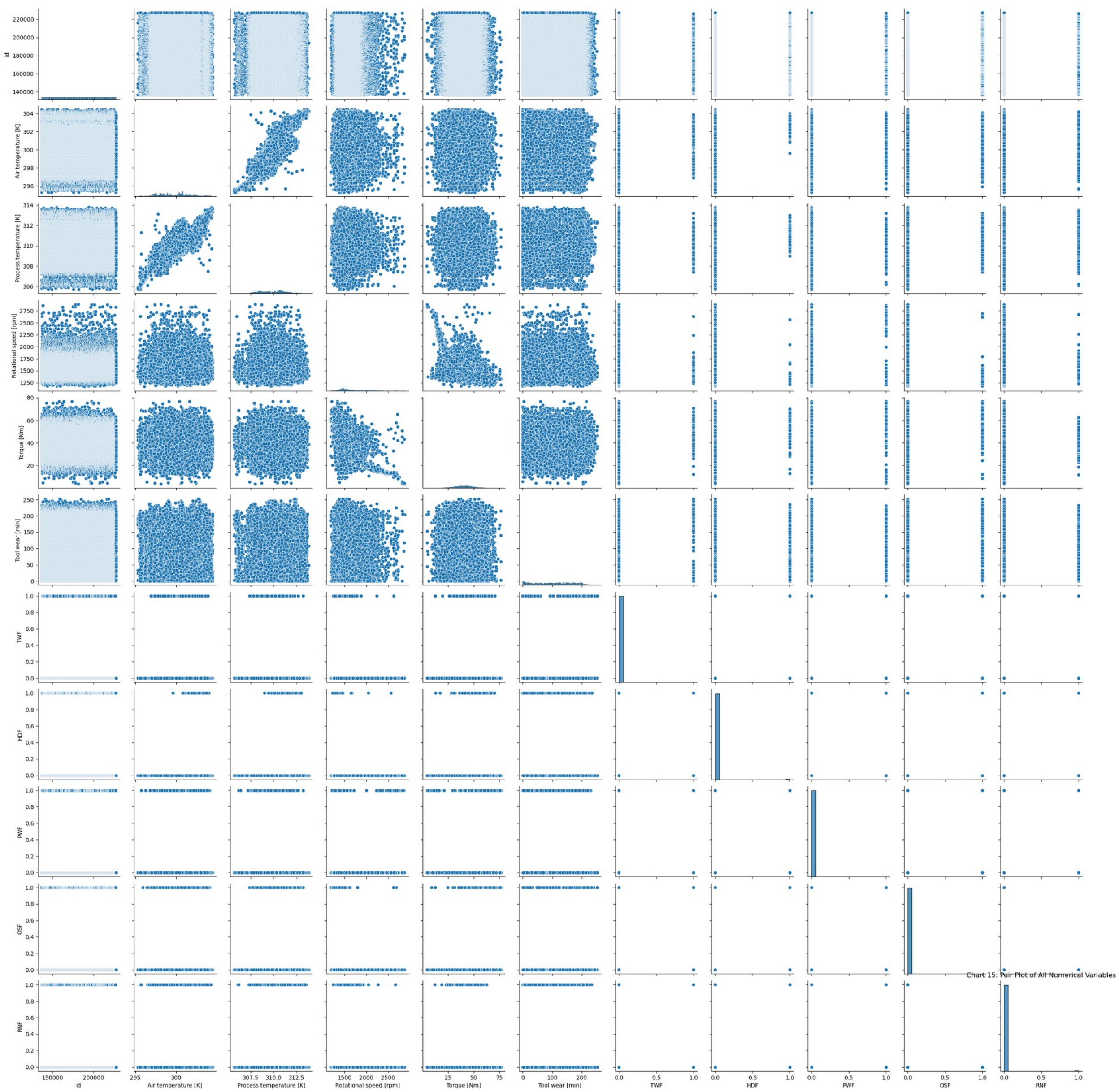
2. What is/are the insight(s) found from the chart?

Strong correlation exists between process and air temperature.

Chart - 15 - Pair Plot

```
# Pair Plot visualization code

# Chart 15: Pair Plot of All Numerical Variables
sns.pairplot(df)
plt.title("Chart 15: Pair Plot of All Numerical Variables")
plt.show()
```



1. Why did you pick the specific chart?

To compare relationships across all numerical variables.

2. What is/are the insight(s) found from the chart?

Some variables have clear trends and clusters.

5. Solution to Business Objective

What do you suggest the client to achieve Business Objective ?

Explain Briefly.

- **Predictive Maintenance**: Using failure data, maintenance schedules can be optimized to prevent unexpected breakdowns.
- **Process Optimization**: Adjusting machine parameters such as speed and temperature can improve efficiency and product quality.
- **Cost Reduction**: Early detection of anomalies can lower maintenance expenses and reduce downtime.
- **Enhanced Decision-Making**: Insights from data visualization can help plant managers improve operational strategies.

Conclusion

This EDA project provides valuable insights into Tata Steel's manufacturing operations. The analysis highlights key factors affecting machine failures, process efficiency, and production quality. By leveraging data-driven strategies, Tata Steel can enhance productivity, minimize downtime, and improve overall profitability. This project serves as a foundation for future machine learning models aimed at predictive maintenance and process optimization.

Hurrah! You have successfully completed your EDA Capstone Project !!!