***PYTHON -EDA***

**PYTHON CODE:**

import pandas as pd

import numpy as np

from matplotlib import pyplot as plt

import seaborn as sns

from mysql import connector

import scipy.stats as stats

#establishing conection

db\_connection={“host”:”localhost”,“user”:”subhash”,”password”:”guru”,”database”:”DBA”}

my\_connection=connector.connect(\*\*db\_connection)

my\_cursor=my\_connection.cursor()

select\_query=”select \*from new\_machine\_downtime;”

machine\_downtime=pd.read\_sql(select\_query, my\_connection)

my\_cursor.close()

my\_connection.close()

machine\_downtime1.head()

machine\_downtime1.tail()

machine\_downtime1.shape()

machine\_downtime1.columns()

machine\_downtime1.to\_csv(‘downtime.csv’, index=false)

machine\_downtime=pd.read\_csv(“downtime.csv”)

machine\_downtime.head()

machine\_downtime.tail()

machine\_downtime.shape()

machine\_downtime.info()

machine\_downtime.dtypes()

machine\_downtime.columns()

machine\_downtime.describe()

print(machine\_downtime.isnull().sum())

machine\_downtime=machine\_downtime.replace(‘ ‘, np.nan)

machine\_downtime.isnull().sum()

machine\_downtime[machine\_downtime['Hydraulic Pressure\_bar']==’ ‘].shape

machine\_downtime=machine\_downtime.replace(‘’, np.nan)

machine\_downtime.isnull().sum()

cols=['Hydraulic\_Pressure’, 'Coolant Pressure', 'Air\_System\_Pressure’, 'Coolant\_Temperature', ' Hydraulic\_Oil\_Temperature', ‘Spindle\_Bearing\_Temperature', 'Spindle\_Vibration', ‘Tool\_Vibration’, ‘Torque',’Cutting ']

machine\_downtime.isnull().sum()

#changing column data types

for col in cols:

machine\_downtime[col]=machine\_downtime[col].astype(float) machine\_downtime['Spindle\_Speed']=machine\_downtime['Spindle\_Speed ].astype(float)

machine\_downtime["Voltage']=machine\_downtime[‘Voltage'].astype(float)

machine\_downtime.info()

for col in machine\_downtime.columns:

if machine\_downtime[col].dtype=’object':

             machine\_downtime[col]=machine\_downtime[col].fillna(machine\_downtime[col].mode()[0])

else:

        machine\_downtime[col]=machine\_downtime[col].fillna(machine\_downtime[col].mean())

machine\_downtime.isnull().sum().sum()

print("Num of duplicates in each row: {}".format(machine\_downtime [machine\_downtime.duplicated()].shape[0]))

machine\_downtime.describe().

num\_cols=machine\_downtime.select\_dtypes(include=float).columns

fig, axes=plt.subplots(nrows=4,ncols=3, figsize=(18,9))

axes=axs.flatten()

import warnings

warnings.filterwarnings(‘ignore’)

machine\_downtime.head()

for 1, col in enumerate(num\_cols):       sns.boxplot(x=col,data=machine\_downtime,ax=axes[1])

fig.tight\_layout()

plt.show()

for 1,col in enumerate(num\_cols):

sns.boxplot(x=col, data=f\_machine\_downtime,ax=axes[i])

fig.tight\_layout()

plt.show()

z\_score=np, abs((machine\_downtime-machine\_downtime.mean())/machine\_downtime.std())

threshold=3

f\_machine\_downtime=machine\_downtime[(z\_score<threshold).all(axis=1)]

fig, axs=plt.subplots(nrows=4,ncols=3, figsize=(18,9))

axes=axs.flatten()

plt.style.use(‘classic’)

fig,axes = plt.subplots(nrows=3, ncols=4, figsize=(20,10))

axes=axs.flatten()

for i, col in enumerate(num\_cols):

sns.histplot(x=col,data=f\_machine\_downtime,ax=axes[i])

fig.tight\_layout()

plt.show()

from sklearn.preprocessing import MinMaxScaler

scaler=MinMaxScaler()

final\_machine\_downtime[‘downtime’]=pd.DataFrame(data=scaler.fit\_transform(f\_machine\_downtime.drop((‘downtime’),axis=1)), columns=f\_machine\_downtime.drop((downtime’),axis=1))

final\_machine\_downtime[‘downtime’]=f\_machine\_downtime[‘downtime’]

final\_machine\_downtime.head()

sns.countplot(f\_machine\_downtime[‘Downtime’])

plt.title(‘Downtime distribuion’)

plt.show()

fromscipy.stats import chi2\_contingency

cr=pd.crosstab(df[‘Downtime’],df[‘Assembly’])

mcr = sns.moaicplot(data=machine\_downtime, title=’downtime vs analysis’, gap=0.02, statistic=False)

plt.xlabel(‘assembly’)

plt.ylabel(‘Downtime’)

plt.show()

f\_machine\_downtime.skew()

f\_machine\_downtime[num\_cols].skew().plot(kind=’bar’)

plt.show()

from sklearn.preprocessing import PowerTransformer

skewed\_features = [‘Hydraulic\_Pressure’, ‘coolant\_temperature’, ‘spindle\_speed’,’cutting\_kn’]

pt=PowerTransformer(method=’yeo-johnson’, standardize=false)

f\_machine\_downtime[skewed\_features] = pt.fit\_transform(f\_machine\_downtime[skewed\_features])

f\_machine\_downtime.skew()

f\_machine\_downtime.drop(‘Downtime’, axis=1).columns

#for reducing skewness to zzero:

machine\_downtime[‘Downtime’]=np.log(machine\_downtime[‘Downtime’])

for col in cols:

machine\_downtime[‘col’]=np.log(machine\_downtime[‘col’])

#for increasing skewness to zero

machine\_downtime[‘Downtime’]=np.exp(machine\_downtime[‘Downtime’])

for col in cols

machine\_downtime[‘col’]=np.exp(machine\_downtime[‘col’])

#checking all the skewness

machine\_downtime.skew()

f\_machine\_downtime.skew()

f\_machine\_downtime[num\_cols].skew().plot(kind=’bar’)

plt.show()

machine\_downtime.corr()

plt.style.use(‘ggplot’)

plt.figure(figsize=(20,10))

sns.heatmap(machine\_downtime.corr(),annot=True,cmap="rainbow")

plt.show()

***BUSINESS INSIGHTS:***

* I observe that some of the column has different data types. Wvwn if they are object type it is not in categorical data type. So we need to change the data type first.
* Then after verify the duplicates in all columns.
* Drop he null values & replace with suitable mean, meadian & mode values to it. Reverify the data.
* Data preprocessing is done & then we need to perform EDA(exploratory data analysis).
* Univariate, using predefined functions visualize the box plot on all columns. And checking the box plot variations in those.
* It gives the each identation of columns.
* Now bi-variate, apply scatter plot on the data & visualize the data where is high & find the outliers(away from remaining scatters/top of the graph.
* Now bi-variate with categorical vs categorical values, in this we find the relationship between the two categorical columns. It visualize the graph contains which among what is there
* By using histograms & histplots we visualize the data structure or data information in the form of tail.
* Now, find the skewness of data using correlation matrix.
* If shows the skewness of the data, if it’s zero then it will has no oultiers. If skewness of data is positive then we need to reduce the skewness of skewed column. If skewness of the data is negative then we need to increase the skewness of the column.
* Using exp function to increase the skewness of the skewed column.
* Using log function to reduce the skewness of the skewed column.
* Most frequently machine failures are increased day by day.
* Due to this machine failures it impacts on the financial, including lost production, maintenance costs and potential revenue loses. This insights can help prioritize preventive measures.
* So, our main objective is to be reduce the machine failures or machine downtime for overcome the financial loses.
* Unplanned machine downtime leads to financial loses. So, we maintain the effectiveness of activities by comparing downtime. This pattern is indicating the reducing the unplanned machine downtime.
* The performance for identify the areas for improvement the data with data preprocessing.
* Firstly we observe the statistical analysis & observe business insights.
* After all that we need to visualize the full data using the heatmap function on that we find the outliers, regression values, etc…
* I also used z-score model to remove outliers
* Normally, distributed features are highly skewed.
* Checking the distribution of the dataset using normalized data with bar plot.