**Project 5 DAILY WEATHER DATA ANALYSIS USING DECISION TREE CLASSIFICATION**

import pandas as pd  
from sklearn.metrics import accuracy\_score  
from sklearn.model\_selection import train\_test\_split  
from sklearn.tree import DecisionTreeClassifier  
data = pd.read\_csv('C:/Users/Lenovo PC 16/Downloads/data/data/data\_weather.csv')  
print("Columns are: ",data.columns)  
print("Data: \n",data)  
print("Null Data: \n",data[data.isnull().any(axis=1)])  
*#Data Cleaning Steps*del data['number']  
before\_rows = data.shape[0]  
print(before\_rows)  
data = data.dropna()  
after\_rows = data.shape[0]  
print(after\_rows)  
print("Total rows dropped: ",before\_rows - after\_rows)  
*#Convert to a Classification Task*clean\_data = data.copy()  
clean\_data['high\_humidity\_label'] = (clean\_data['relative\_humidity\_3pm'] > 24.99)\*1  
print(clean\_data['high\_humidity\_label'])  
y=clean\_data[['high\_humidity\_label']].copy()  
clean\_data['relative\_humidity\_3pm'].head()  
print("Y Data: \n",y.head())  
*#Use 9am Sensor Signals as Features to Predict Humidity at 3pm*morning\_features = ['air\_pressure\_9am','air\_temp\_9am','avg\_wind\_direction\_9am','avg\_wind\_speed\_9am', 'max\_wind\_direction\_9am','max\_wind\_speed\_9am','rain\_accumulation\_9am', 'rain\_duration\_9am']  
X = clean\_data[morning\_features].copy()  
print("Columns in X: ",X.columns)  
print("Columns in Y: ",y.columns)  
*#Perform Test and Train split*X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.33, random\_state=324)  
print ("X\_train is as under:")  
print(X\_train.head())  
print ("X\_test is as under:")  
print(X\_test.head())  
print ("y\_train is as under:")  
print(y\_train.head())  
print ("y\_test is as under:")  
print(y\_test.head())  
print ("Let us describe y\_train \n",y\_train.describe())  
humidity\_classifier = DecisionTreeClassifier(max\_leaf\_nodes=10, random\_state=0)  
humidity\_classifier.fit(X\_train, y\_train)  
print(type(humidity\_classifier))  
*#Predict on Test Set*predictions = humidity\_classifier.predict(X\_test)  
print("Sample Predictions: \n",predictions[:10])  
print("Sample Y Test(Actual Data): \n",y\_test['high\_humidity\_label'][:10])  
*#Measure Accuracy of the Classifier*print("Accuracy: \n",accuracy\_score(y\_true = y\_test, y\_pred = predictions))