import numpy as np

import pandas as pd

from sklearn.ensemble import GradientBoostingRegressor

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import mean\_squared\_error

# Load the dataset

data\_filtered = pd.read\_csv(r" ")

# Prepare the data

X = data\_filtered[['Temperature (°C)', 'Concentration', 'Response Time']] # Features

y = data\_filtered['Response'] # Target variable

# Split the data into training and testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Train a Gradient Boosting Regressor model

model = GradientBoostingRegressor(random\_state=100)

model.fit(X\_train, y\_train)

# Define the input conditions for prediction

concentration = 50 # Fixed concentration

temperatures = [25, 40, 55, 70, 85, 100, 115, 130, 145, 160] # Temperatures to predict at

response\_times = [100] \* len(temperatures) # Assume a fixed response time for prediction

# Create a DataFrame for the prediction inputs

predict\_data = pd.DataFrame({

'Temperature (°C)': temperatures,

'Concentration': [concentration] \* len(temperatures),

'Response Time': response\_times

})

# Predict the responses

predicted\_responses = model.predict(predict\_data)

# Display the prediction results

prediction\_results = predict\_data.copy()

prediction\_results['Predicted Response'] = predicted\_responses

print(prediction\_results)