import pandas as pd

import numpy as np

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

from sklearn.ensemble import RandomForestRegressor

from sklearn.metrics import mean\_squared\_error, r2\_score

import matplotlib.pyplot as plt

# Suppress warnings

import warnings

warnings.filterwarnings('ignore')

# Load your dataset

# Replace 'your\_dataset.csv' with your actual dataset path

data = pd.read\_csv('your\_dataset.csv')

# Display the first few rows of the dataset

print(data.head())

# Check for missing values

print(data.isnull().sum())

# If there are missing values, you might want to fill or drop them

# data = data.fillna(method='ffill')

# Define features (X) and target (y)

X = data[['charge\_capacity', 'discharge\_capacity', 'temperature', 'voltage']]

y = data['cycles']

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

# Initialize the RandomForestRegressor

model = RandomForestRegressor(n\_estimators=100, random\_state=42)

# Train the model

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

# Initialize the RandomForestRegressor

model = RandomForestRegressor(n\_estimators=100, random\_state=42)

# Train the model

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

# Predict on the test data

y\_pred = model.predict(X\_test)

# Calculate Mean Squared Error and R-squared

mse = mean\_squared\_error(y\_test, y\_pred)

r2 = r2\_score(y\_test, y\_pred)

print(f'Mean Squared Error: {mse}')

print(f'R-squared: {r2}')

# Plotting the results

plt.figure(figsize=(10, 6))

plt.scatter(y\_test, y\_pred, color='blue')

plt.plot([y\_test.min(), y\_test.max()], [y\_test.min(), y\_test.max()], 'k--', lw=4)

plt.xlabel('Actual')

plt.ylabel('Predicted')

plt.title('Actual vs Predicted Battery Life Cycles')

plt.show()

import joblib

# Save the trained model to a file

joblib.dump(model, 'battery\_life\_predictor.pkl')

# Load the model later using:

# model = joblib.load('battery\_life\_predictor.pkl')

**Explanation**

* **Dataset:** The dataset is assumed to have features related to battery performance. You need to replace 'your\_dataset.csv' with the path to your actual dataset.
* **Model:** A RandomForestRegressor is used here, but you can experiment with other models like Gradient Boosting, XGBoost, etc.
* **Evaluation:** The model's performance is evaluated using Mean Squared Error (MSE) and R-squared (R²) metrics. The scatter plot gives a visual indication of how well the model predicts battery life.

This code is a basic template and can be enhanced with additional features, hyperparameter tuning, and cross-validation for improved accuracy. Make sure to pre-process and clean your dataset before training the model for optimal results.