Battery Impedance Analysis Assignment

Dataset Used: NASA Battery Dataset (from Kaggle) **Link to Dataset:** NASA Battery Dataset

---

Objective

The goal of this assignment is to analyze how key battery parameters change as the battery ages through charge and discharge cycles. Specifically, the following parameters were analyzed:

- **Re**: Estimated Electrolyte Resistance (Ohms)
- **Rct**: Estimated Charge Transfer Resistance (Ohms)

Interactive visualizations were created using **Plotly** to understand trends over time or cycles.

Steps Followed

1. Data Inspection

The first step was to inspect the structure of the dataset, identify column names, check for missing values, and verify the types of data present.

Columns in the Dataset:

- type: Type of data recorded
- start_time: Timestamp of the test run
- ambient_temperature: Temperature during the operation
- battery_id: Battery identification number
- test_id: Test identification number
- uid: Unique ID for each row
- filename: File name corresponding to the data
- Capacity: Remaining battery capacity (Ah)
- Re: Estimated electrolyte resistance (Ohms)
- Rct: Estimated charge transfer resistance (Ohms)

Code for Data Inspection:

```
import pandas as pd


# Load the dataset

file_path = 'battery_dataset.csv'  # Replace with your file path

battery_data = pd.read_csv(file_path)
```

```
# Display first few rows
print(battery_data.head())

# Check column names and missing values
print(battery_data.dtypes)
print(battery_data.isnull().sum())

# Basic statistical summary
print(battery_data.describe())
```

Findings:

- All required columns (Capacity, Re, Rct) were present.
- The uid and filename columns were irrelevant for analysis.
- Missing values were handled by either dropping rows or imputing values.

---

2. Data Cleaning

To ensure clean and structured data:

- Irrelevant columns uid and filename were dropped.
- Missing values in Re and Rct were removed for simplicity.
- The data was sorted based on start_time to analyze aging trends.

Code for Data Cleaning:

```
# Drop irrelevant columns
battery_data = battery_data.drop(columns=['uid', 'filename'])

# Drop rows with missing values
battery_data = battery_data.dropna()

# Convert 'start_time' to datetime format
battery_data['start_time'] = pd.to_datetime(battery_data['start_time'])

# Sort by start_time
battery_data = battery_data.sort_values(by='start_time').reset_index(drop=True)
```

---

3. Adding a Synthetic Cycle Column

Since the dataset did not include a Cycle column, a new column was created to represent the sequential charge/discharge cycles.

Code to Add Cycle Column:

```
# Add a Cycle column representing sequential rows

battery_data['Cycle'] = range(1, len(battery_data) + 1)
```

---

4. Data Visualization Using Plotly

To analyze how the **Electrolyte Resistance (Re)** and **Charge Transfer Resistance (Rct)** change with respect to **Battery Capacity** and time:

4.1 Re vs Capacity

Objective: Show how the electrolyte resistance (Re) changes as the battery capacity decreases.

Code:

```
import plotly.express as px


fig_re = px.line(
    battery_data,
    x='Capacity',
    y='Re',
    title='Electrolyte Resistance (Re) vs Battery Capacity',
    labels={'Capacity': 'Battery Capacity (Ah)', 'Re': 'Electrolyte Resistance (Ohms)'},
    color='battery_id'  # Differentiate by battery ID
)
fig_re.show()
```

4.2 Rct vs Capacity

Objective: Show how the charge transfer resistance (Rct) evolves as battery capacity decreases.

Code:

```
fig_rct = px.line(
    battery_data,
    x='Capacity',
```

```
    y='Rct',
    title='Charge Transfer Resistance (Rct) vs Battery Capacity',
    labels={'Capacity': 'Battery Capacity (Ah)', 'Rct': 'Charge Transfer Resistance (Ohms)'},
    color='battery_id'  # Differentiate by battery ID
)
fig_rct.show()
```

4.3 Re and Rct Over Time

Objective: Show how both resistances (Re and Rct) change over time.

Code:

```
fig_time = px.line(
    battery_data,
    x='start_time',
    y=['Re', 'Rct'],
    title='Electrolyte and Charge Transfer Resistance Over Time',
    labels={'start_time': 'Start Time', 'value': 'Resistance (Ohms)'},
    color_discrete_map={'Re': 'blue', 'Rct': 'red'}
)
fig_time.show()
```

Key Observations

1. **Re vs Capacity:** As the battery capacity decreases, the electrolyte resistance (Re) tends to increase, indicating degradation.
2. **Rct vs Capacity:** The charge transfer resistance (Rct) also shows an increasing trend with capacity loss.
3. **Re and Rct Over Time:** Both resistances increase progressively with time, highlighting the aging process of the battery.

Conclusion

The analysis clearly shows the impact of battery aging through charge/discharge cycles:

- Both **Electrolyte Resistance (Re)** and **Charge Transfer Resistance (Rct)** increase as the battery capacity reduces.

- These trends are consistent over time, demonstrating how repeated cycles lead to internal resistance buildup.

The interactive plots created using **Plotly** effectively visualize these trends, providing insight into the battery degradation process.

Code and Outputs

The cleaned dataset and all the plots have been saved for further reporting and presentation purposes.

References

- NASA Battery Dataset on Kaggle: Link
- Plotly Documentation: Plotly Express