NASA Battery Analysis

This project analyzes battery aging data collected from charge and discharge cycles. The goal is to visualize how battery parameters like impedance, electrolyte resistance (Re), and charge transfer resistance (Rct) evolve as the battery ages.

In [1]:
```
import pandas as pd
import plotly.express as px
import plotly.io as pio

# Use matplotlib backend as a workaround for static rendering
pio.renderers.default = "plotly_mimetype+notebook"

df = pd.read_csv('Battery_data.csv')

df.head(10)
```

Out[1]:

	type	start_time	ambient_temperature	battery_id	test_id	uid	filename	Capacity	Re	
0	discharge	[2010. 7. 21. 15. 0. ...	4	B0047	0	1	00001.csv	1.6743047446975208	NaN	
1	impedance	[2010. 7. 21. 16. 53. ...	24	B0047	1	2	00002.csv	NaN	0.05605783343888099	0.20097
2	charge	[2010. 7. 21. 17. 25. ...	4	B0047	2	3	00003.csv	NaN	NaN	
3	impedance	[2010 7 21 20 31 5]	24	B0047	3	4	00004.csv	NaN	0.05319185850921101	0.16473
4	discharge	[2.0100e+03 7.0000e+00 2.1000e+01 2.1000e+01 2...	4	B0047	4	5	00005.csv	1.5243662105099023	NaN	
5	charge	[2010. 7. 21. 22. 38. ...	4	B0047	5	6	00006.csv	NaN	NaN	
6	discharge	[2.010e+03 7.000e+00 2.200e+01 1.000e+00 4.000...	4	B0047	6	7	00007.csv	1.5080762969973425	NaN	
7	charge	[2010. 7. 22. 3. 14. ...	4	B0047	7	8	00008.csv	NaN	NaN	
8	discharge	[2010. 7. 22. 6. 16. ...	4	B0047	8	9	00009.csv	1.4835577960067696	NaN	
9	charge	[2010. 7. 22. 7. 50. ...	4	B0047	9	10	00010.csv	NaN	NaN	

In [2]:
```
print("Initial Data Overview:")
print(df.info())
```

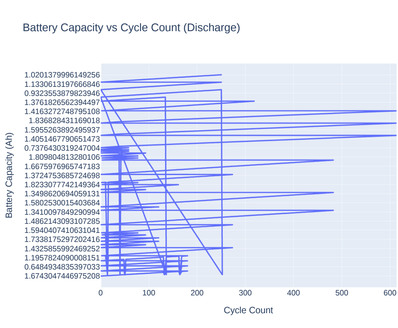
```
Initial Data Overview:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7565 entries, 0 to 7564
Data columns (total 10 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   type                7565 non-null   object
 1   start_time          7565 non-null   object
 2   ambient_temperature  7565 non-null   int64
 3   battery_id          7565 non-null   object
 4   test_id             7565 non-null   int64
 5   uid                 7565 non-null   int64
 6   filename            7565 non-null   object
 7   Capacity            2794 non-null   object
 8   Re                  1956 non-null   object
 9   Rct                 1956 non-null   object
dtypes: int64(3), object(7)
memory usage: 591.1+ KB
None
```

In [3]:
```
import plotly.io as pio
print(pio.renderers)
```

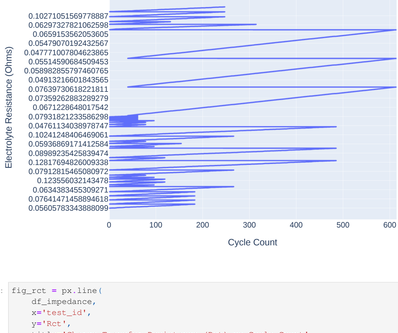
```
Renderers configuration
-----------------------
    Default renderer: 'plotly_mimetype+notebook'
    Available renderers:
        ['plotly_mimetype', 'jupyterlab', 'nteract', 'vscode',
         'notebook', 'notebook_connected', 'kaggle', 'azure', 'colab',
         'cocalc', 'databricks', 'json', 'png', 'jpeg', 'jpg', 'svg',
         'pdf', 'browser', 'firefox', 'chrome', 'chromium', 'iframe',
         'iframe_connected', 'sphinx_gallery', 'sphinx_gallery_png']
```

In [4]:
```
df_discharge = df[df['type'] == 'discharge'].dropna(subset=['Capacity'])
df_impedance = df[df['type'] == 'impedance'].dropna(subset=['Re', 'Rct'])
```

In [5]:
```
fig_capacity = px.line(
    df_discharge,
    x='test_id',
    y='Capacity',
    title='Battery Capacity vs Cycle Count (Discharge)',
    labels={'test_id': 'Cycle Count', 'Capacity': 'Battery Capacity (Ah)'},
)
fig_capacity.show()
```



In [6]:
```
fig_re = px.line(
    df_impedance,
    x='test_id',
    y='Re',
    title='Electrolyte Resistance (Re) vs Cycle Count',
    labels={'test_id': 'Cycle Count', 'Re': 'Electrolyte Resistance (Ohms)'},
)
fig_re.show()
```



In [7]:
```
fig_rct = px.line(
    df_impedance,
    x='test_id',
    y='Rct',
    title='Charge Transfer Resistance (Rct) vs Cycle Count',
    labels={'test_id': 'Cycle Count', 'Rct': 'Charge Transfer Resistance (Ohms)'},
)
fig_rct.show()
```

In [ ]: