# Summary of NASA and Oxford Battery Dataset Preprocessing

Detailed Summary of NASA and Oxford Battery Dataset Preprocessing (As of February 27, 2025)  
Overview  
We’ve been working on preprocessing the NASA Li-ion Battery Aging Dataset (B0005–B0056) and the Oxford Battery Degradation Dataset (Cell1–Cell8) to extract key features (Cycle, Voltage (V), Current (A), Temperature (°C), Capacity (Ah), SoH (%), RUL (Cycles)), align them, and combine for modeling (e.g., BiLSTM, survival analysis) and dashboard development. Below is a comprehensive breakdown of our progress, challenges, and solutions.  
1. Oxford Dataset Preprocessing  
Started With: The Oxford dataset (Oxford\_Battery\_Degradation\_Dataset\_1.mat) contains 8 cells (Cell1–Cell8), each with characterization cycles (e.g., cyc0000, cyc0100, etc.) under 40°C, 1-C charge/discharge, with data fields like C1ch, C1dc, OCVch, OCVdc, and discharge features (t, v, q, T).  
  
Progress:  
Successfully processed all 8 cells, extracting Cycle, Voltage (V), Current (A) (negative for discharge), Temperature (°C), Capacity (Ah), SoH (%), and RUL (Cycles) for each.  
  
Used a threshold of 0.592 Ah (80% SoH for initial 0.740 Ah) to determine failure and calculate RUL.  
  
Saved preprocessed data as CSVs (e.g., Cell8\_processed.csv) in data/oxford/.  
  
Fixed issues like current sign (negative for discharge), missing cycles, and data gaps, ensuring alignment with NASA.  
  
Challenges:  
Initial TypeError with capacity scalars, resolved by extracting .item() from NumPy arrays.  
  
Missing cycles in CSV (e.g., 83 rows but output showing 76), fixed by debugging cycle\_keys.  
  
Current units were positive, corrected to negative for discharge.  
  
Current State: Oxford preprocessing is complete and ready for combining, with all cells processed correctly and saved.  
  
2. NASA Dataset Preprocessing (B0005–B0056)  
Started With: The NASA dataset includes batteries B0005–B0056 (excluding gaps: B0008–B0017, B0019–B0024, B0035, B0037) with operational profiles (charge, discharge, impedance) at various temperatures (24°C, 43°C, 4°C) and currents (2A, 4A, 1A). Data structure: top-level cycle with type, ambient\_temperature, time, and data (fields like Voltage\_measured, Current\_measured, Temperature\_measured, Capacity for discharge).  
  
Progress:  
Initially processed B0005–B0050 assuming battery or data top-level keys, but confirmed all use cycle top-level.  
  
Struggled with B0005–B0056, facing issues like:  
"No cycle data found" warnings, fixed by trying cycle, battery, data, battery\_id keys.  
  
Type: unknown for cycles, no valid capacities, empty CSVs, and ValueError/IndexError errors.  
  
Key fixes included:  
Simplified cycle['type'][0] access to detect 'discharge'.  
  
Used cycle['data']['Capacity'] with capacity\_scalar = raw\_capacity[0][0][0][0] for nested arrays.  
  
Removed / 1000 scaling for Current\_measured (amps, not mA), ensuring ~ -2 A for B0005.  
  
Set RUL thresholds: 1.38 Ah (70% SoH) for most, 1.6 Ah (20% fade) for B0033–B0036/B0038–B0040, 1.4 Ah (30% fade) for B0041–B0044.  
  
Saved to ../data/NASA/preprocessed/B0005\_processed.csv, handling empty DataFrames with NaN.  
  
Current state: B0005 finds 1 discharge cycle, but capacities are NaN, and CSVs are empty. We’re close but missing proper type and Capacity extraction.  
  
Challenges:  
Mismatched data structures (assumed battery/data keys, corrected to cycle).  
  
Type: unknown due to complex cycle\_type detection—simplified to cycle['type'][0].  
  
Capacity extraction failed due to incorrect nesting (cycle['data'][0, 0] vs. cycle['data']['Capacity']).  
  
Tiny currents (e.g., 0.0049 A) fixed by removing / 1000, but we need to verify raw Current\_measured.  
  
Empty CSVs due to NaN data, fixed by ensuring valid cycle and feature extraction.  
  
3. Key Issues and Fixes Identified  
Cycle Type Detection:  
Issue: cycle\_type returned 'unknown' for B0005 because cycle['type'] is a NumPy array or string requiring cycle['type'][0].  
  
Fix: Use cycle['type'][0] == 'discharge' directly, as in the working B0025 code.  
  
Capacity Extraction:  
Issue: raw\_capacity = data['Capacity'] with data = cycle['data'][0, 0] didn’t find valid Capacity, returning NaN. The working code used raw\_capacity = cycle['data']['Capacity'] with capacity\_scalar = raw\_capacity[0][0][0][0].  
  
Fix: Update to raw\_capacity = cycle['data']['Capacity'] and unwrap with raw\_capacity[0][0][0][0].  
  
Current Scaling:  
Issue: Tiny currents (e.g., 0.0049 A) due to / 1000 scaling, assuming mA. README confirms amps, so scaling was wrong.  
  
Fix: Use currents.append(-current\_scalar) (no scaling), verifying Current\_measured is ~ -2 A for B0005.  
  
CSV Empty:  
Issue: CSVs were empty or all NaN due to no valid discharges or capacities.  
  
Fix: Ensure all 168 discharges are found, extract valid Capacity, and populate DataFrame.  
  
RUL Calculation:  
Issue: RUL skipped due to no valid capacities, with incorrect thresholds (e.g., 1.380 Ah for B0005 should be 70% of ~1.97 Ah, or 1.38 Ah).  
  
Fix: Use 1.38 Ah for B0005–B0032, B0045–B0056, and specific fades (1.6 Ah, 1.4 Ah) for other batches.  
  
4. Current Status and Next Steps  
Oxford: Fully preprocessed, ready to combine.  
  
NASA: B0005–B0056 are partially processed, but B0005 finds only 1 discharge, no valid capacities, and empty CSVs. We’re close—fixes are:  
Use cycle['type'][0] == 'discharge' for cycle detection.  
  
Use cycle['data']['Capacity'] with capacity\_scalar = raw\_capacity[0][0][0][0] for capacities.  
  
Remove / 1000 from currents, verify ~ -2 A for B0005.  
  
Ensure 168 sequential cycles, valid Capacity, and RUL at 129 for B0005.  
  
Next Steps:  
Run the updated script, verify B0005 output and CSV.  
  
Combine NASA and Oxford, check alignment (columns, units, thresholds).  
  
Move to modeling (BiLSTM, survival analysis) and dashboard.