Procedural Guide for Replicating Daily U.S. Protest Attendance Graphs Using the Crowd Counting Consortium Dataset (2025)

Paul Garrett Hugel

ORCID iD: 0000-0001-8082-7208

September 2025

1. Citations

Author citation: Hugel, Paul G. 2025. *Strategic Nonviolent Campaign 2025–2029 to Resist Autocratic Consolidation.* OSF. July 16. doi:10.17605/OSF.IO/X9QA6.

Data citation: Ulfelder, Jay. 2025. *Crowd Counting Consortium U.S. Protest Event Data, 2025–.* Harvard Dataverse. https://doi.org/10.7910/DVN/RI9JFU.

—

2. Objective

To create an interactive and replicable visualization of daily U.S. protest attendance between January 1, 2025 and August 31, 2025, including significant protest events and methodological transparency for replication and validation.

—

3. Data Acquisition

- Source: Harvard Dataverse (see data citation above).
- File: `ccc-phase3-public.csv`
- Variables of interest: `date`, `size_mean`.

—

4. Preprocessing

1. Load dataset with `pandas`, using encoding `latin1`.
2. Convert `date` to ISO datetime format.
3. Filter by date range: 2025–01–01 to 2025–08–31.
4. Drop rows with missing `size_mean`.
5. Aggregate by day: compute daily totals as the sum of `size_mean`.

—

5. Analytical Method

- Metric: daily protest attendance estimated by aggregated `size_mean`.
- Highlight days with >10,000 participants.
- Annotate known events:
  - April 5, 2025 — Hands Off Nationwide Protest.
  - June 14, 2025 — No Kings Nationwide Protest.

—

6. Visualization Procedure

1. Toolchain: Python 3.11, `pandas`, `plotly`.

2. Graph construction:

   - Blue line with dots for daily totals.
   - Red "X" for days exceeding 10,000, with vertical stems and numeric labels.
   - Vertical reference lines and annotations for significant events.

3. Output:

   - Self-contained interactive HTML file (`include_plotlyjs=inline`).
   - Static PNG image for publication.

—

7. Replication Instructions

1. Install dependencies:

```
pip install pandas plotly matplotlib
```

2. Download CCC dataset from Harvard Dataverse.

3. Execute preprocessing and visualization steps described above.

4. Validate by comparing computed daily totals against raw event-level sums.

—

8. Validation

- Cross-check totals against raw `size_mean`.
- Confirm that annotated peaks align with external protest records.
- Archive code, dataset DOI, and output for reproducibility.

—

9. AI Transparency Notice

This document and associated graphs were prepared with the assistance of OpenAI's ChatGPT (GPT-5). AI was used to:

1. Ingest the CCC dataset and perform preprocessing.
2. Generate reproducible Python code for aggregation and visualization.
3. Produce explanatory text and this LaTeX-formatted guide.

Final outputs were verified and curated by the author.

—