UIDAI Aadhaar Enrolment and Update Intelligence (2025)

Team: Vikas Kalsariya, Rishabh Bhattacharya

Website: https://best-uidai-project.streamlit.app/

GitHub: https://github.com/Quainsami/uidai-project

Data coverage: 2025-03-01 to 2025-12-31. Daily patterns use dense months with continuous data (September to December 2025).

Problem Statement and Approach

We analyze anonymized Aadhaar enrolment and update transactions to identify temporal patterns, update-heavy geographies, operational anomalies, and socio-infrastructure signals that can guide capacity planning and system improvements. The approach combines multi-level aggregation (state, district, pincode), anomaly scoring, and spatial visualization with external normalization data for per-capita comparisons.

Datasets Used

- UIDAI Aadhaar Enrolment dataset: date, state, district, pincode, age_0_5, age_5_17, age_18_greater
- UIDAI Aadhaar Demographic Update dataset: date, state, district, pincode, demo_age_5_17, demo_age_17_
- UIDAI Aadhaar Biometric Update dataset: date, state, district, pincode, bio_age_5_17, bio_age_17_
- Census 2011 district indicators (population, households, electricity, internet) for per-capita normalization
- Pincode lookup table for state/district cleanup and pincode boundary geojsons for city maps

Data Coverage and Quality Notes

The dataset spans March 1, 2025 to December 31, 2025. Some months are reported as single snapshot days (March to July 2025), and August 2025 is absent. Daily seasonality analysis is therefore restricted to September-December 2025.

Month	Days with updates	Total updates
2025-03	1	19,395,529
2025-04	1	10,126,439
2025-05	1	9,420,790
2025-06	1	9,541,141
2025-07	1	11,952,114
2025-09	20	13,924,974
2025-10	18	9,540,026
2025-11	24	16,604,303
2025-12	28	18,061,175

Missing months in UIDAI daily data: 2025-08

Methodology

- Standardize date, state, district, and pincode fields; resolve inconsistent state labels using a pincode lookup.
- Aggregate transactions at state, district, and pincode levels; compute enrolment totals and update totals.
- Normalize district activity by population to derive enrolments per 1k and updates per 1k.
- Compute update-to-enrolment ratios and z-score anomalies at pincode level.
- Classify districts into four quadrants using median enrolment and update intensity.
- Overlay pincode totals on city boundary geojsons for spatial hotspots.

Key Findings

- Updates dominate activity: 21.9x more updates than enrolments; biometric updates are 58.6% of total updates.
- Top 5 states account for 50.5% of all updates, indicating concentrated operational load.
- Sunday update volume is 49% of Friday in dense months, highlighting a strong weekly cycle.
- Median district update/enrol ratio is 23.1; 90th percentile is 47.2. About 14.7% of districts exceed a ratio of 40.
- Update-to-enrol ratio correlates with electricity access (r = 0.37), suggesting infrastructure readiness supports more frequent updates.

Top Update Concentration

State	Updates	Share
Uttar Pradesh	18,120,063	15.3%
Maharashtra	14,280,742	12.0%
Bihar	9,711,939	8.2%
Andhra Pradesh	8,886,150	7.5%
Madhya Pradesh	8,836,709	7.5%

Most Update-Heavy Districts (ratio)

State	District	Update/Enrol Ratio
Manipur	Thoubal	108.0
Manipur	Imphal East	101.8
Manipur	Imphal West	97.6
Chhattisgarh	Balod	84.0
Maharashtra	Gadchiroli	81.5

Pincode Anomaly Hotspots

Pincode	State	District	z-score
431001	Maharashtra	Aurangabad	13.97
743329	West Bengal	South 24 Parganas	13.25
110094	Delhi	North East Delhi	12.52
700135	West Bengal	South 24 Parganas	11.59
500005	Andhra Pradesh	K.v. Rangareddy	10.80
852210	Bihar	Madhepura	10.70
713146	West Bengal	Barddhaman	10.69
711315	West Bengal	Howrah	10.64

Visualizations

Figure 1. Monthly enrolment and update volume by type.

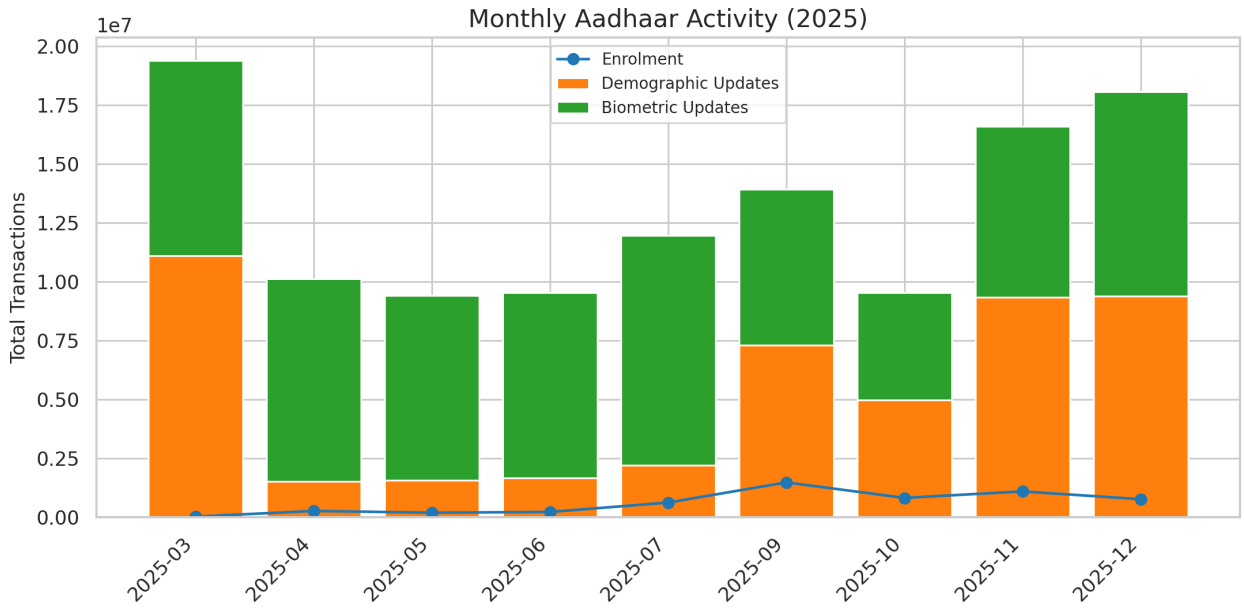


Figure 2. Daily trends (7-day rolling average) for dense months.

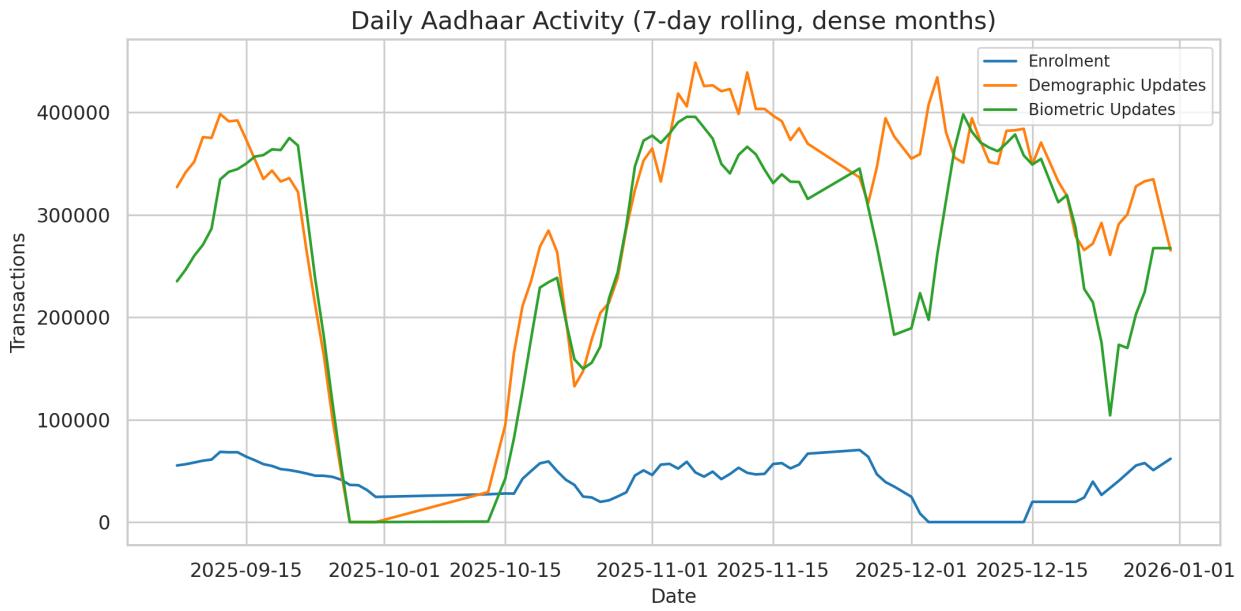


Figure 3. Average daily volume by day of week (dense months).

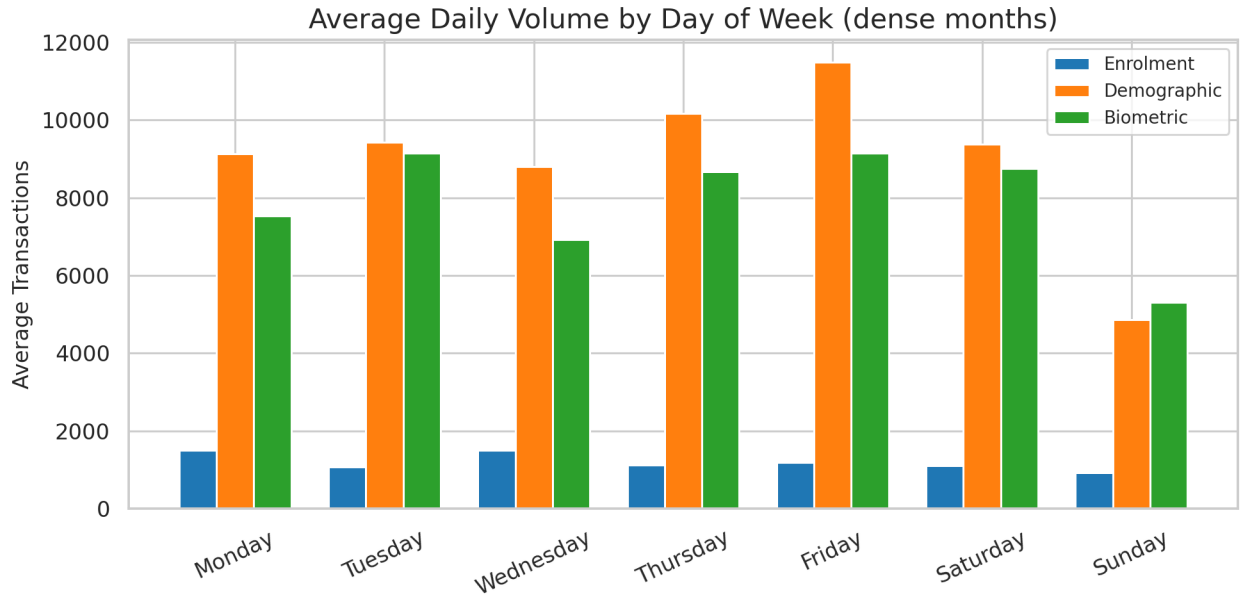


Figure 4. Enrolment age mix across top 12 states.

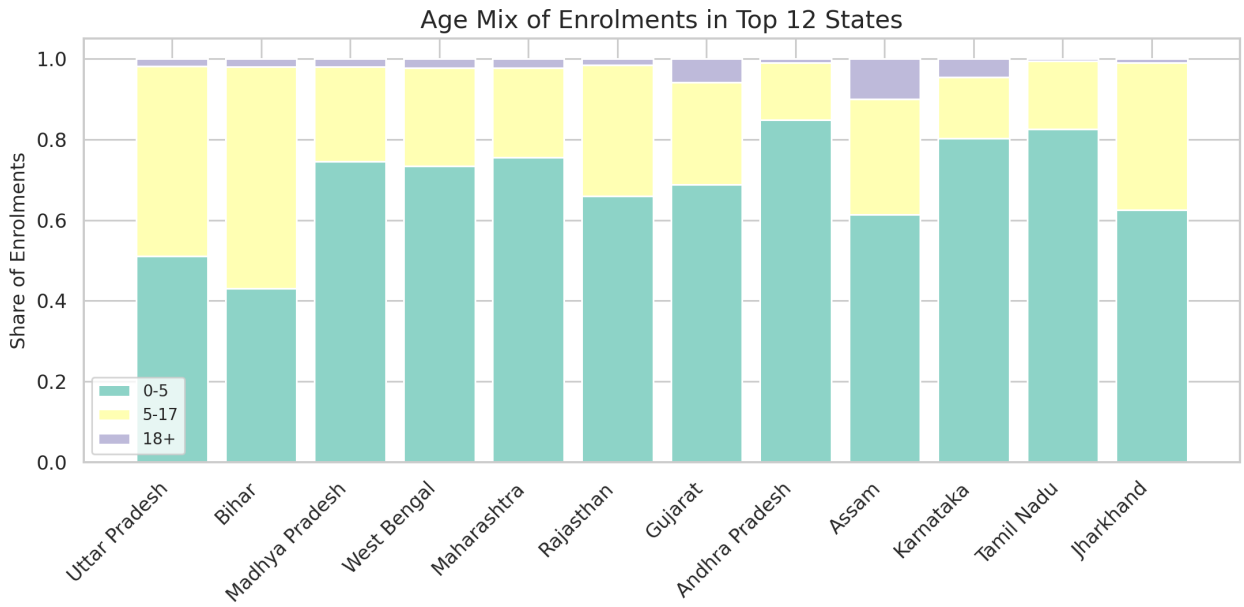


Figure 5. State update mix: demographic vs biometric.

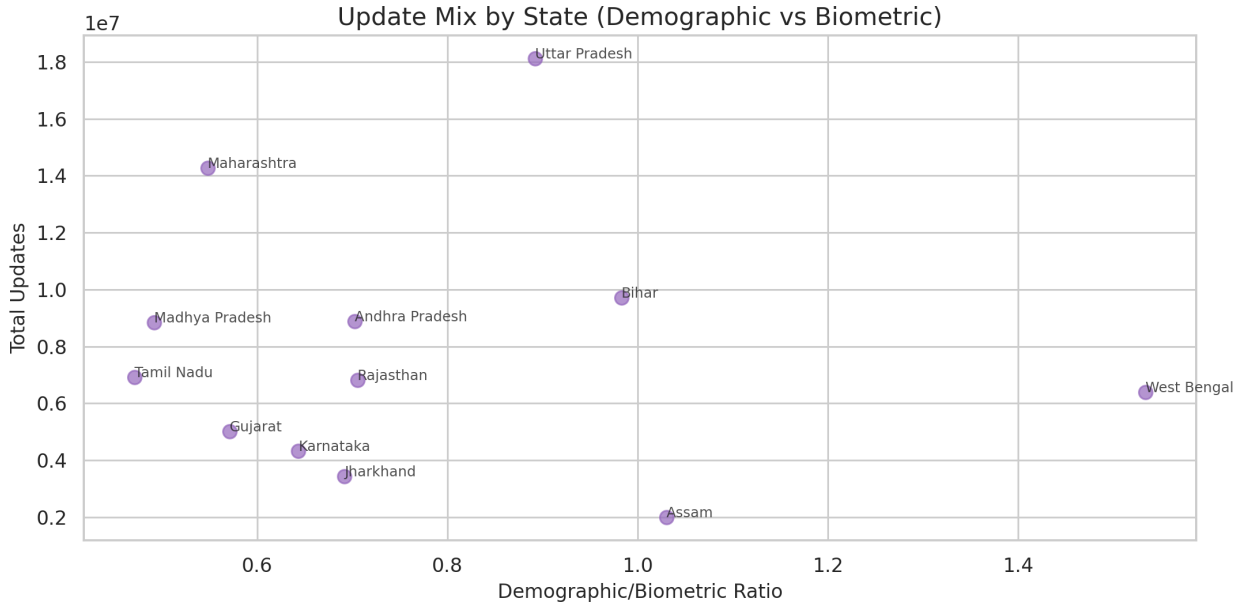


Figure 6. District enrolment vs update intensity per 1k population.

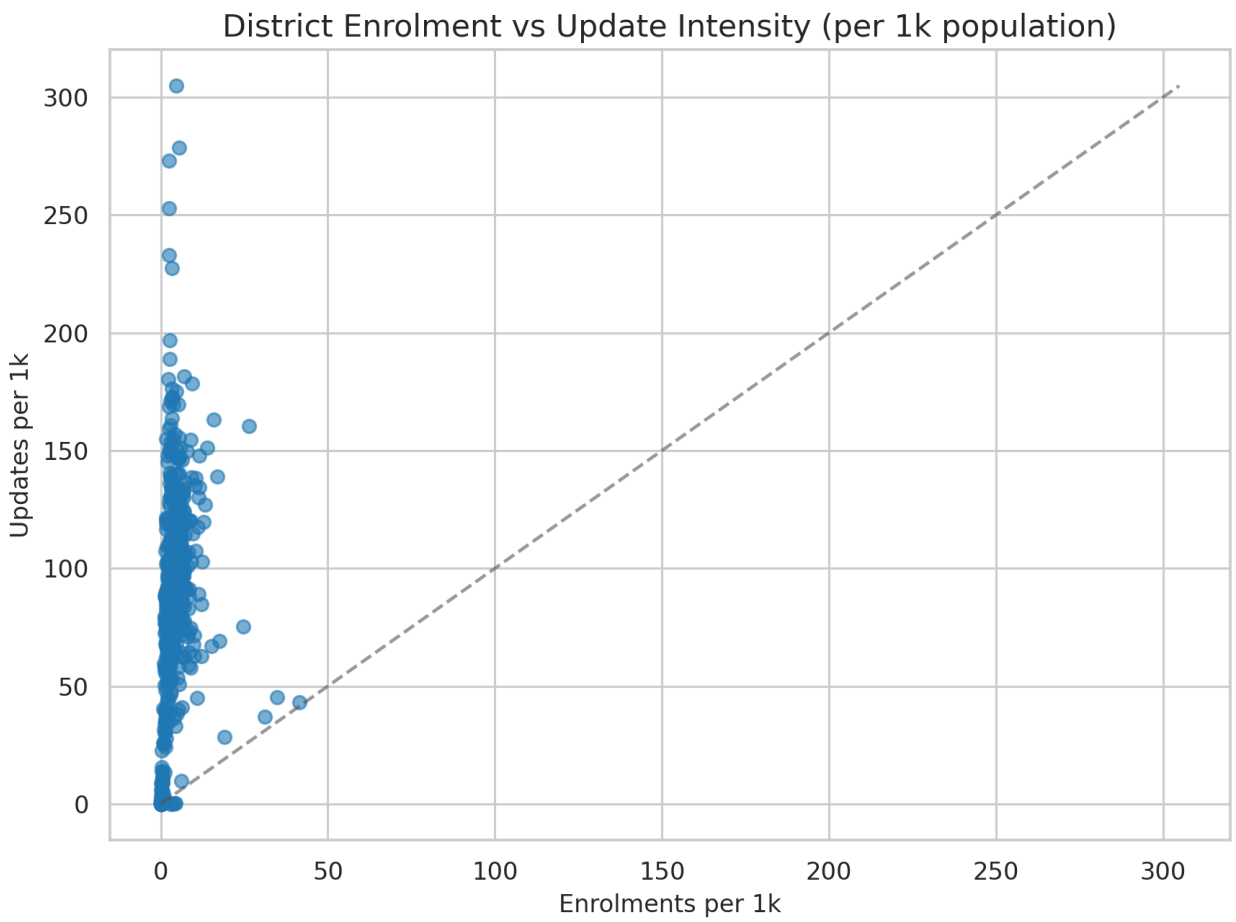

District Enrolment vs Update Intensity (per 1k population)
Updates per 1k
Enrolments per 1k
0
50
100
150
200
250
300

Figure 7. Electricity access vs update-to-enrol ratio.

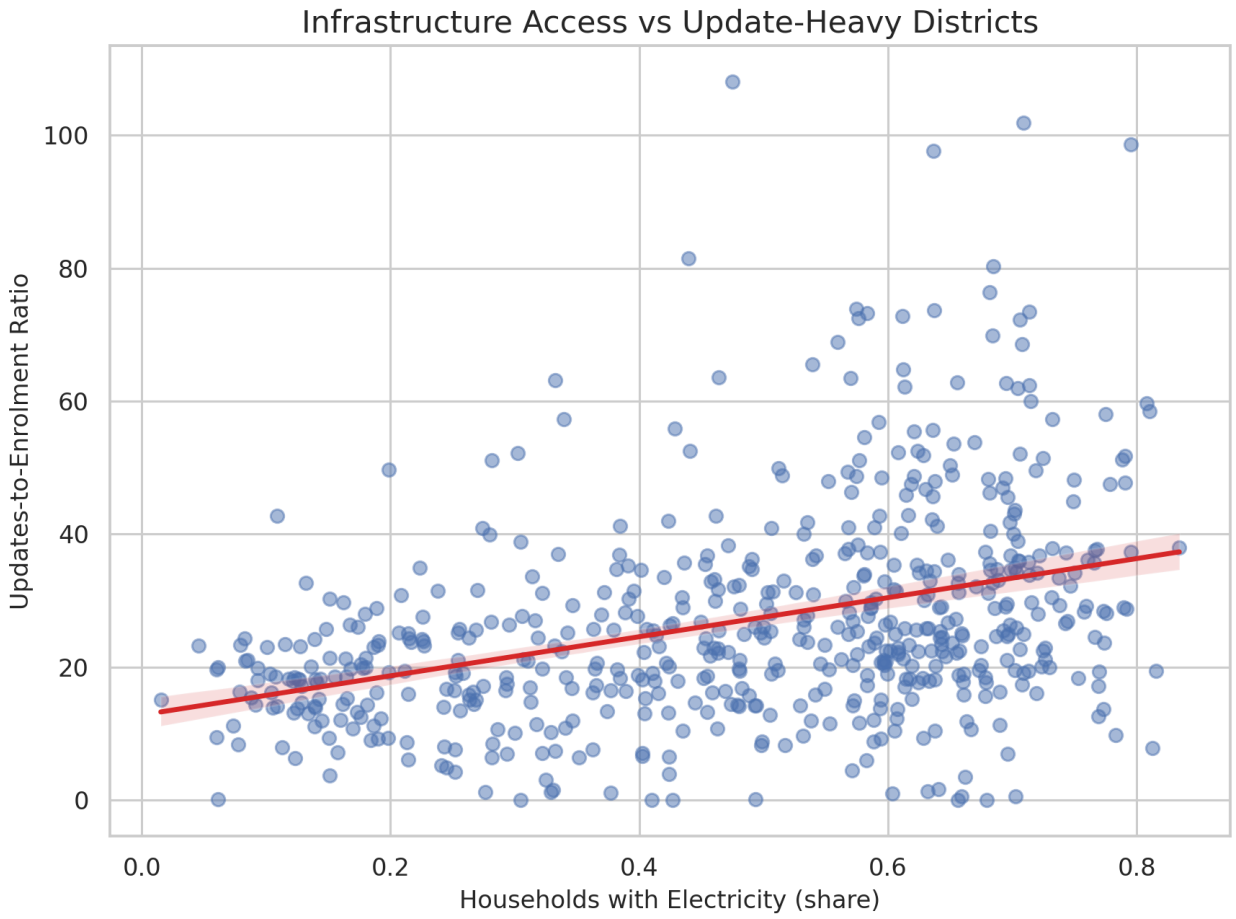


Figure 8. Top pincode anomalies by update-to-enrol ratio z-score.

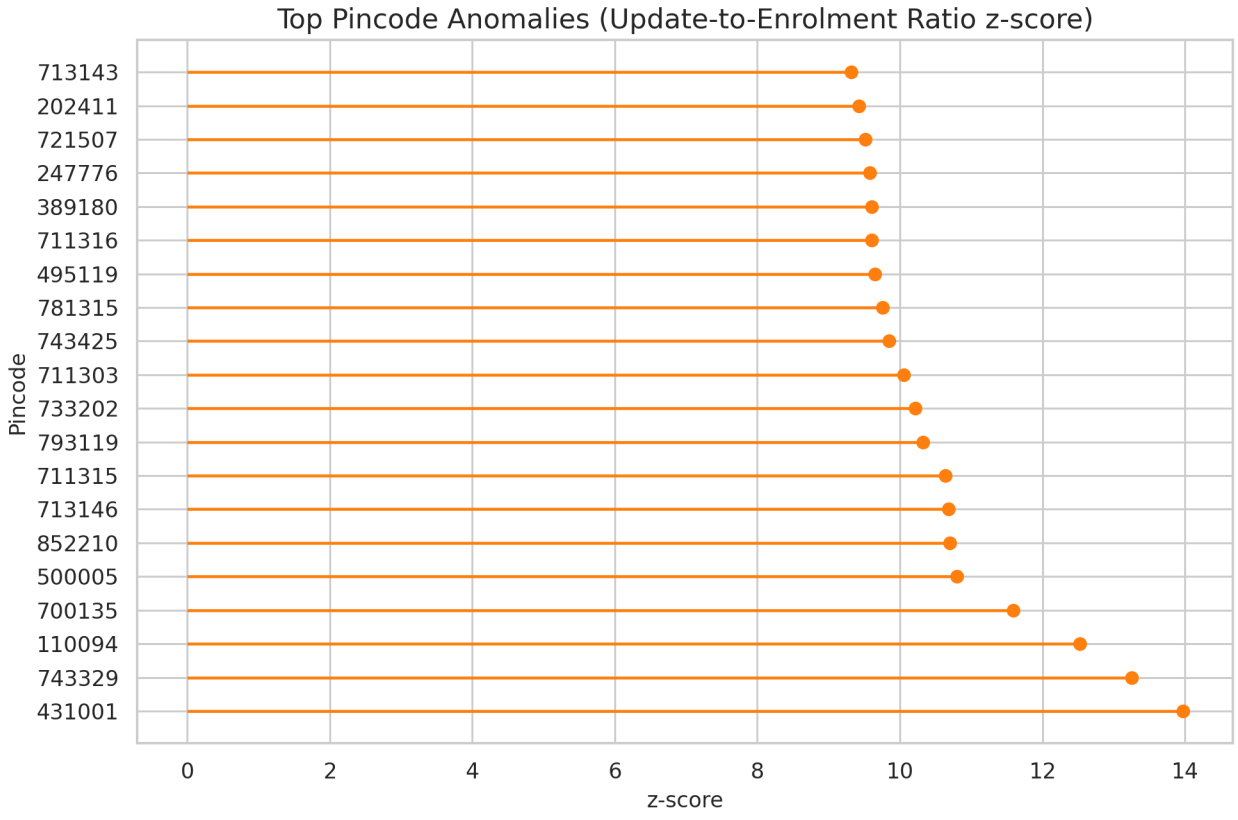


Figure 9. Bangalore pincode update intensity heatmap.

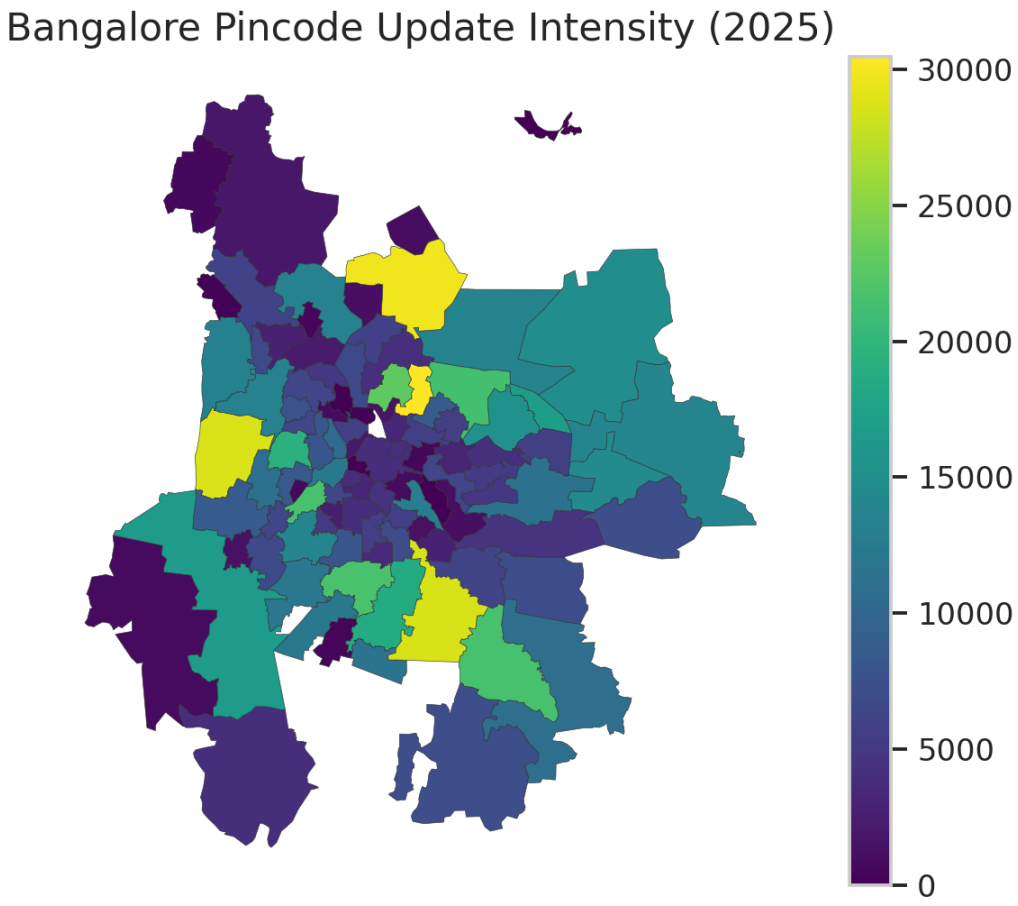


Figure 10. Mumbai pincode update intensity heatmap.

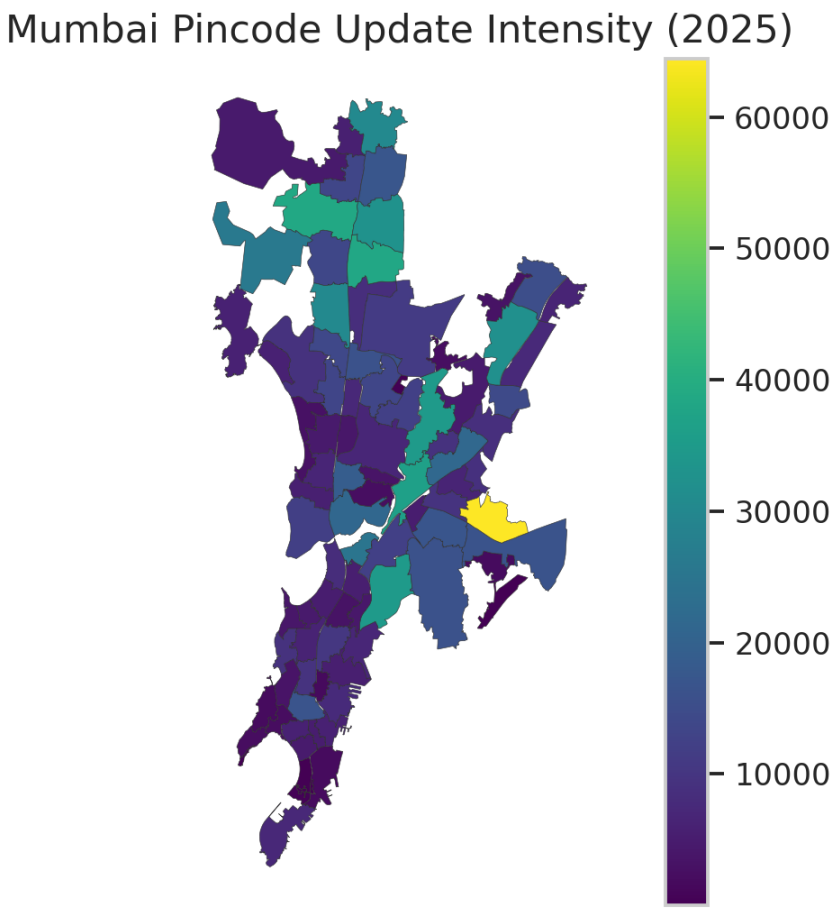


Figure 11. Monthly update load by state (top 12).

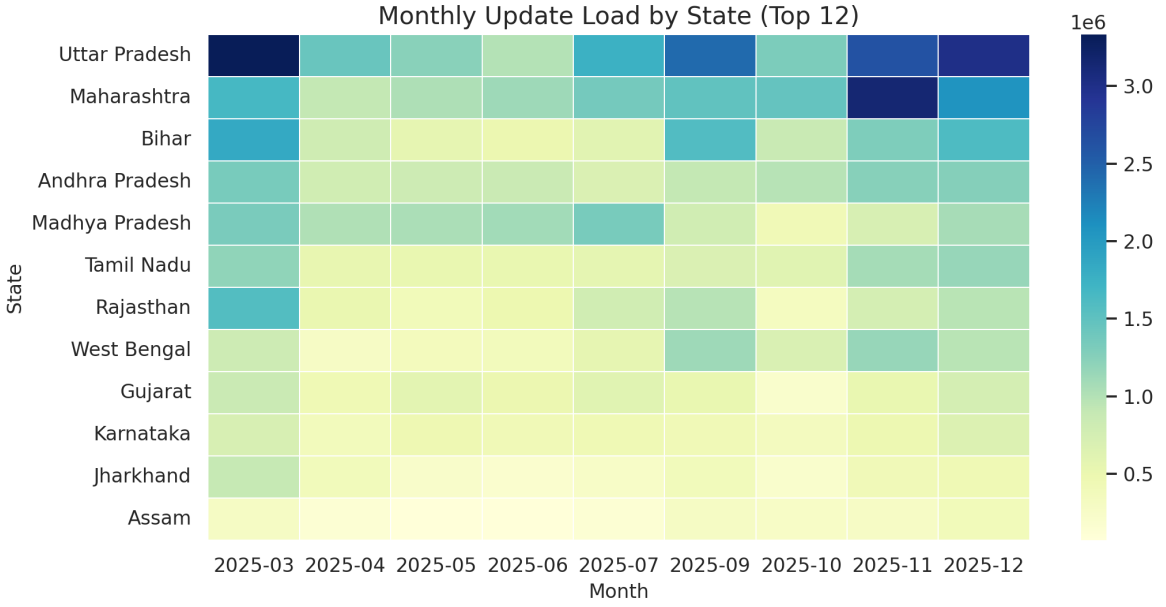


Figure 12. District quadrant segmentation.

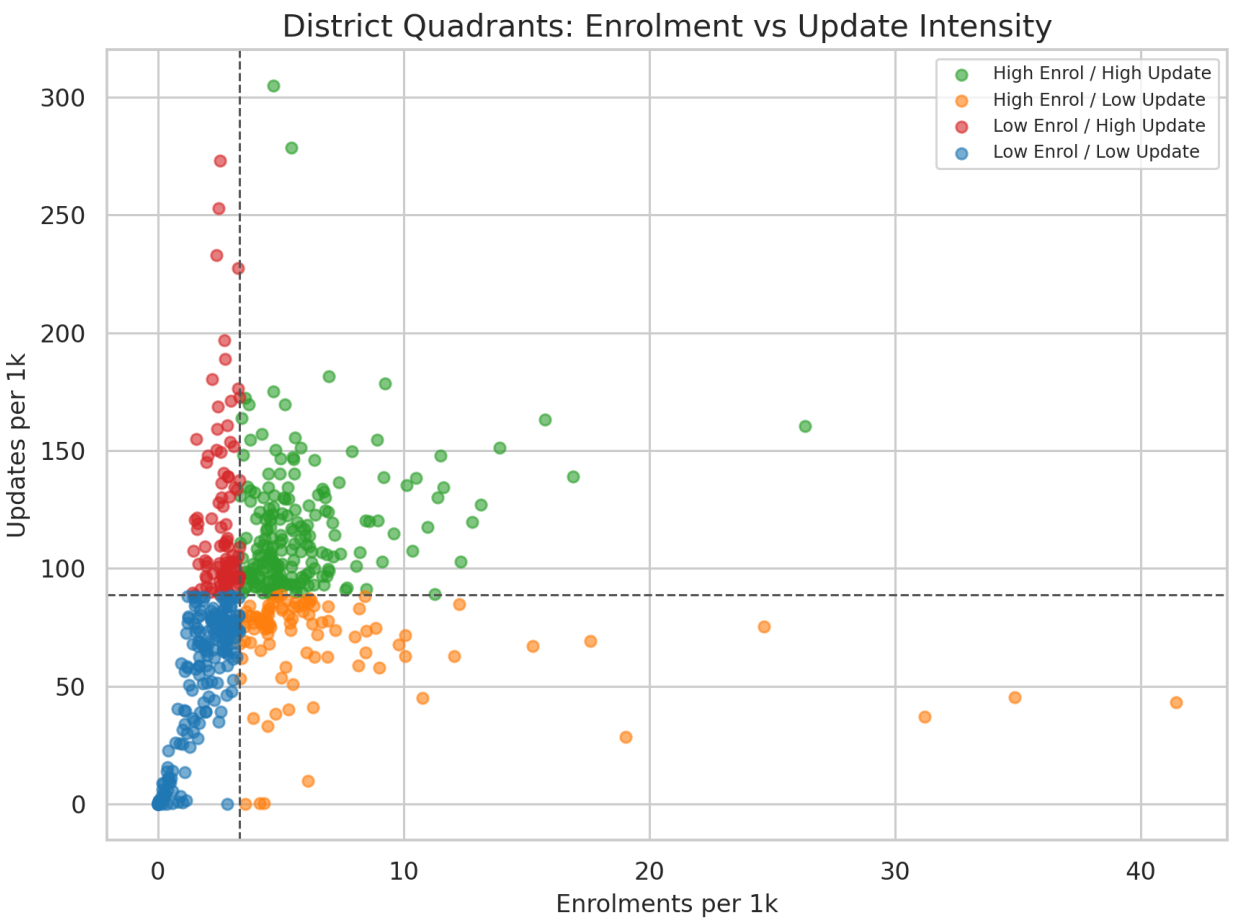

District Quadrants: Enrolment vs Update Intensity
High Enrol / High Update
High Enrol / Low Update
Low Enrol / High Update
Low Enrol / Low Update
Updates per 1k
Enrolments per 1k
0
50
100
150
200
250
300
0
10
20
30
40

Figure 13. Distribution of update-to-enrol ratio.

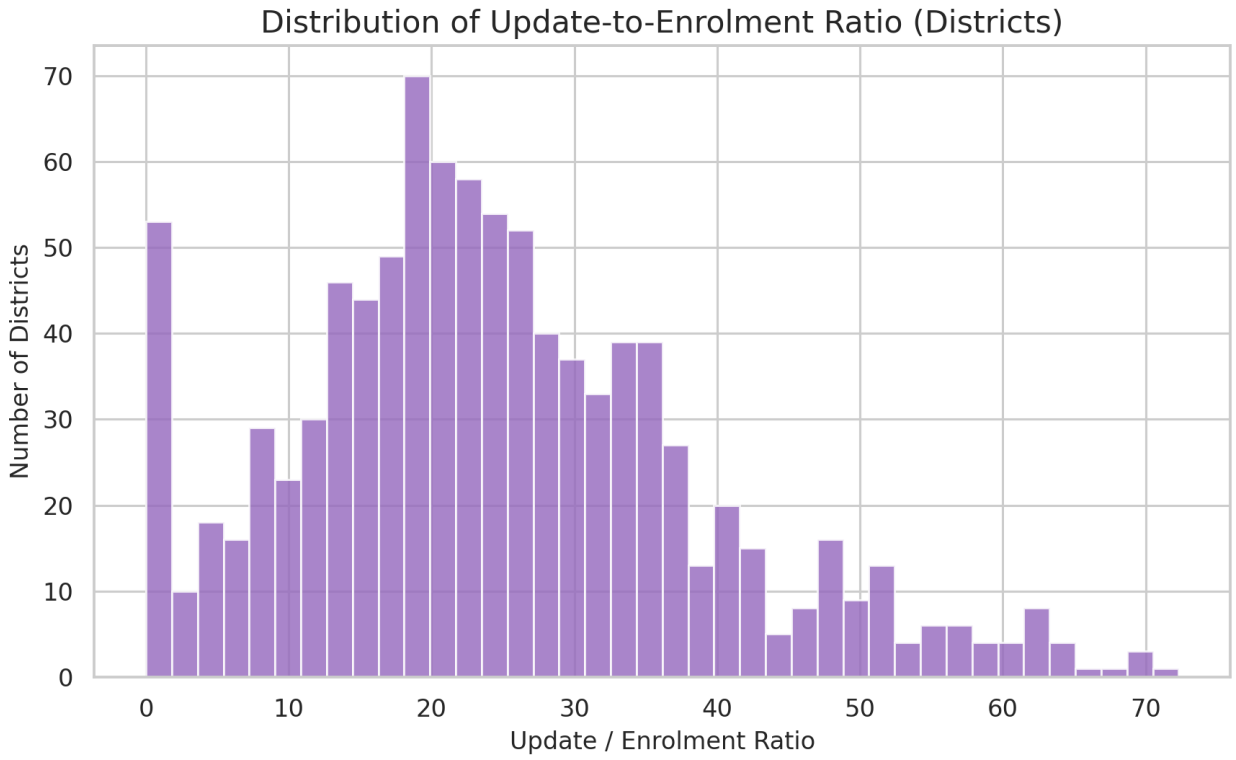


Figure 14. State-level updates per 1k population.

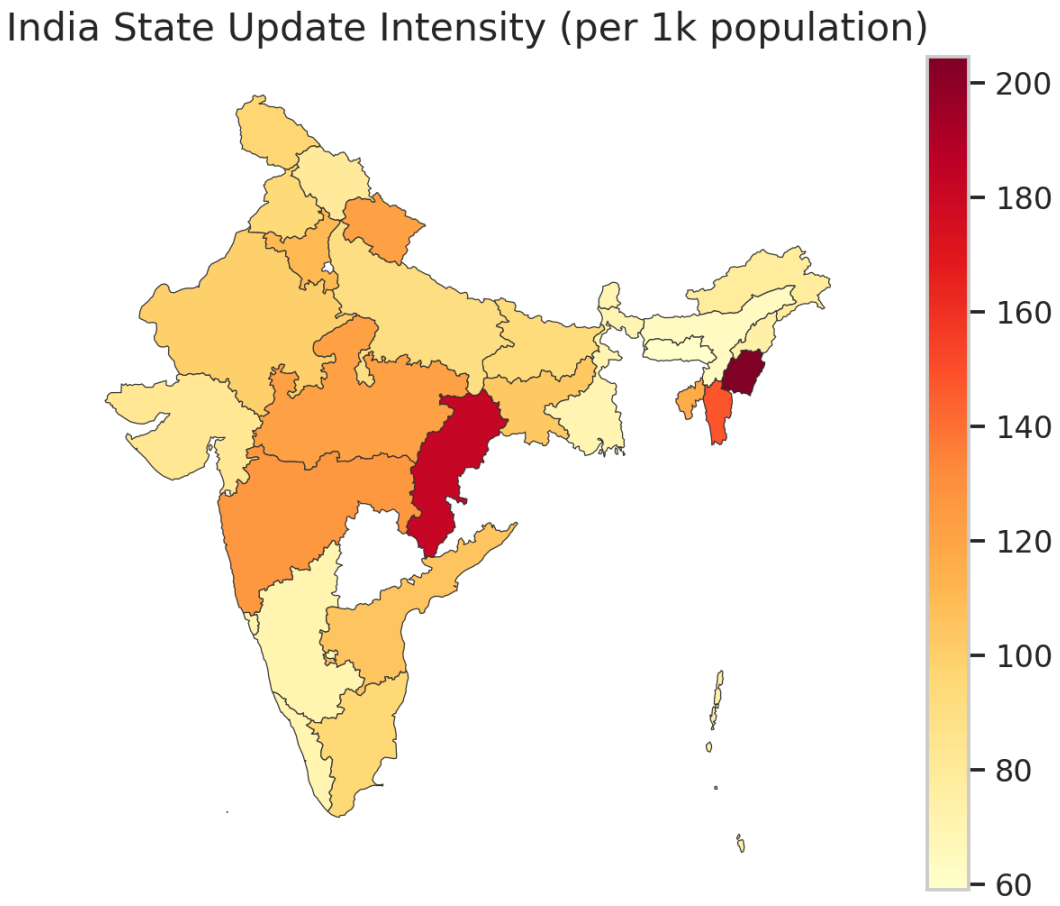
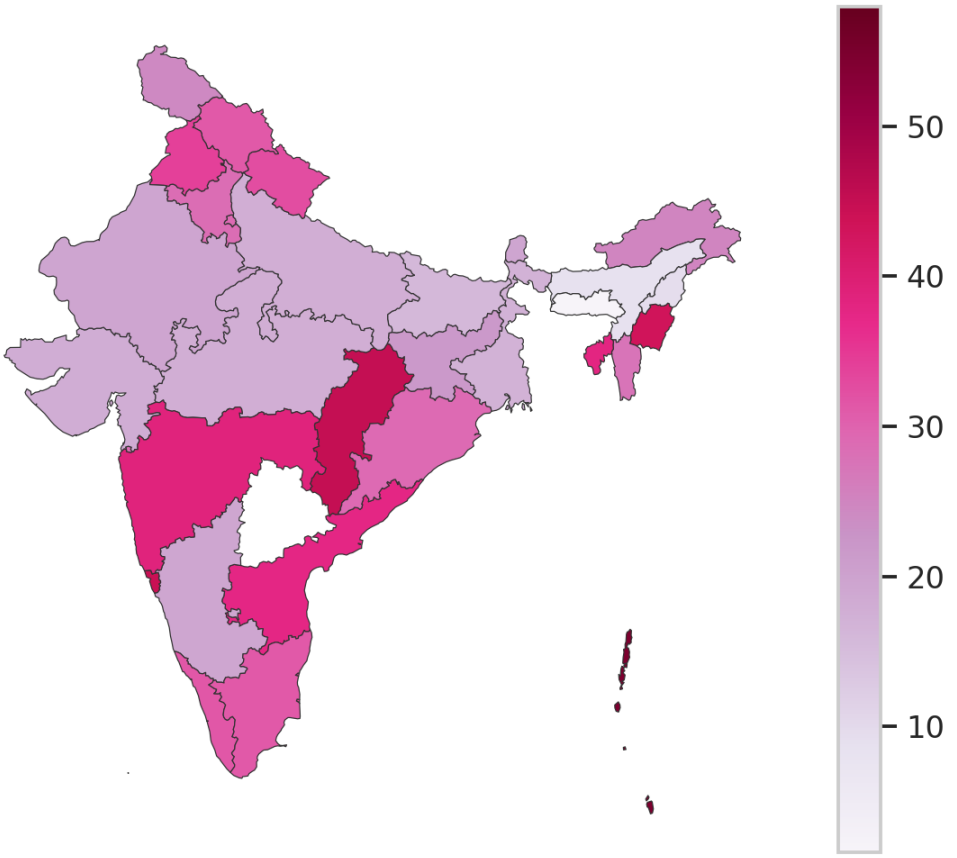


Figure 15. State-level update-to-enrolment ratio.


India State Update-to-Enrolment Ratio
50
40
30
20
10

Impact and Applicability

- Staffing and device allocation can prioritize Thursday-Friday peaks and reduce idle capacity on Sundays.
- States with high update-to-enrol ratios should be targeted for workflow automation and self-service kiosks.
- Districts in the Low Enrol / High Update quadrant indicate maintenance-heavy regions; support via update camps.
- Anomaly pincodes should trigger operational audits or data-quality reviews.
- Infrastructure metrics (electricity access) can guide rollout of high-throughput centres.

Streamlit Website

The interactive Streamlit site provides dashboards for trends, state deep dives, district segmentation, pincode anomaly exploration, city-level choropleths, and a Gemini-powered analyst chat.

- Local run command: streamlit run app.py
- Gemini integration uses GEMINI_API_KEY stored in .streamlit/secrets.toml
- Deployment link: https://best-uidai-project.streamlit.app/
- GitHub repository: https://github.com/Quainsami/uidai-project

Reproducibility

- Run analysis pipeline: python analysis/run_analysis.py
- Generate report: python reports/build_report.py
- App reads aggregated parquet outputs from analysis/outputs