Assignment 2 — Team Data Analytics & Visualization Project

Teams: 3 students ·

Duration: ~4 weeks ·

Weight: 25%

Deliverables: proposal, repo, pipeline notebook/script(s), dashboard/visuals, report, presentation, peer review

---

1) Overview

Choose a **business or technology area** where data visualization can materially improve decisions, communication, or operations.

Build an **end-to-end data analytics pipeline** (from problem framing → data acquisition → cleaning → analysis/modelling → validation) and **communicate the outcome** with clear, audience-appropriate visualizations (dashboard and/or report graphics).

You may select any topic with accessible data and a realistic stakeholder.

---

2) Learning goals

By the end you should be able to:

- Frame a decision-centric question and define success metrics.
- Source, evaluate, and responsibly use real-world datasets.
- Implement a reproducible analytics pipeline (scripts/notebooks; configs; seeds).
- Design exploratory and explanatory visualizations following best practices.
- Build a small interactive artifact (dashboard/story) or a well-structured static narrative.
- Communicate insights to non-technical stakeholders.

---

3) Team roles (recommended)

Every member contributes to all parts, but pick a primary role:

- **Data Engineer:** ingestion, cleaning, joins, data dictionary, reproducibility.
- **Analyst/Modeler:** EDA, features, modeling/validation, metrics.
- **Visualization Lead:** visual grammar, accessibility, narrative, dashboard/report polish.

> Submit a short **team contract** (1/2 to one page) with roles, meeting cadence, deadlines, and conflict-resolution steps.

---

4) Scope & constraints

- **Data:** at least **10,000 rows** (or equivalent complexity); may combine sources.
    - Tiny datasets are fine only if the problem is analysis-rich (e.g., time series at high frequency) or if you build a sophisticated visualization.
- **Sources:** public/open, or your own collected data with consent. Respect licenses.
- **Complexity:** show non-trivial cleaning/EDA and at least one of: feature engineering, simple model, or statistically sound comparison (e.g., A/B test analysis).
- **No black box:** explain all methods and assumptions.

---

5) Deliverables (summary)

1. **1-page Proposal (PDF)** — problem, stakeholder, datasets, success metrics, risks, and mockups (hand-sketch ok). Due end of Week 1. It is an informal assessment, I will provide feedback on proposals.
2. **Git Repository** — all code, data instructions, and documentation.
    - `README.md` with quickstart, project structure, and reproduction steps.
    - Environment: `environment.yml` / `requirements.txt` or `renv.lock`.
    - `data/` with README on how to obtain raw data (don't commit large proprietary files).
    - `src/` or notebooks with clear naming and execution order; configs/seeds.
3. **End-to-End Pipeline** — Jupyter/R Markdown/py scripts showing ingestion → cleaning → EDA → modelling/analysis → outputs.
4. **Visualization Artifact** — interactive dashboard (e.g., Plotly/Dash) **or** a well-designed static narrative (multi-figure report). Must include:
    - 1+ **exploratory** view(s) used during analysis; and
    - 1+ **explanatory** view(s) tailored to the stakeholder.
5. **Written Report (1,500–2,000 words)** — audience: informed non-technical stakeholder. Due the same day of data visualisation exam.
    - Executive summary (≤200 words)
    - Problem & success metrics
    - Data & methods (incl. limitations)
    - Results & visuals (captioned; references to figures)
    - Ethics, bias, and accessibility considerations
    - Recommendations & next steps
    - Short appendix: reproduction notes
6. **Presentation (8–10 min + 2 min Q&A)** — slide deck with key visuals. Due week 12.

---

6) Milestones & Timeline (suggested)

- **Week 1:** Proposal + team contract; dataset verified; initial sketches.

- **Week 2:** Data ingestion & cleaning complete; EDA and draft charts; layout wireframe.
- **Week 3:** Analysis/modeling validated; dashboard/story prototype; accessibility pass.
- **Week 4:** Final polish; report & repo freeze; presentation delivery.

> TA/Instructor check-ins at the end of Weeks 1 & 3 (5–10 minutes/team).

---

7) Technical requirements

Languages/Tools (choose one stack):

- **Python:** pandas, numpy, scipy/scikit-learn (if data modelling), Plotly/Matplotlib/Altair/Bokeh, Dash/Streamlit (optional);

Reproducibility:

- Deterministic seeds; config file for paths/params; such as `Makefile` or simple run script is a plus.
- Data provenance and licensing noted; any manual steps documented.
- Data of different stages (e.g., raw and cleaned) must be stored and documented.

Visualization quality:

- Label axes/units; readable titles; define colours/encodings; legends clear.
- Use **colour-blind-safe** palettes; avoid misleading scales; consider uncertainty (Confidence Interval/CI, error bars) when appropriate.
- Provide **alt text**/descriptions; maintain sufficient contrast.

Performance:

- Interactive artifacts should load within ~3 seconds on provided sample data.

---

8) Ethics, privacy, and bias

- Check and discuss for sampling bias, missingness patterns, leakage, and fairness impacts.
- **Document data licenses**. **Web scraping** is allowed as a supportive data source only when compliant with the website's Terms of Service** and robots.txt; prefer official APIs/open datasets. Do not bypass logins, paywalls, or CAPTCHAs; respect rate limits; collect only non-PII unless you have documented consent.
- Include a very short **Ethics & Limitations** subsection in the report.

---

9) Potential data sources (examples)

World Bank/IMF, Eurostat, OECD, WHO, Our World in Data, data.gov portals, Open Data portals for cities (NYC TLC, London, Dublin, etc.), UCI ML Repository, Kaggle open datasets, GDELT/news, app store reviews, GitHub archives, OpenWeather (terms permitting), Yelp Open, OpenStreetMap/GTFS.

---

10) Topic ideas (pick one or propose your own)

Business: churn analysis & retention dashboard; dynamic pricing diagnostics; demand forecasting for retail; marketing attribution sanity-check; supply-chain delays & inventory heatmap; A/B test readouts; sales funnel + anomalies.

Technology/Engineering: model drift/monitoring dashboard; IoT sensor anomaly detection timeline; incident/root-cause exploration for service outages; open-source project health; security events overview (synthetic/redacted data).

Public sector / society: road safety hotspots; air quality vs. health outcomes; housing affordability; emergency response times; education outcomes and equity; energy consumption & renewables integration.

---

11) Evaluation rubric (100 pts + up to 5 extra)

Area	Excellent (A)	Adequate (B/C)	Needs work (D/F)	Pts
Problem framing	Clear stakeholder, measurable success metric, realistic scope	Somewhat clear; metrics vague	Unclear; misaligned	**10**
Data sourcing & management	Suitable, well-licensed data; provenance documented; dictionary present	Usable but gaps in documentation	Poor fit; licensing unclear	**10**
Pipeline correctness	Clean, modular, reproducible; light tests; no brittle steps	Mostly reproducible; minor manual steps	Not reproducible; errors	**20**
Analysis/modeling	Sound methods; validation/uncertainty addressed	Basic methods; limited validation	Flawed methods; claims unsupported	**15**
Visualization quality	Clear, accurate, accessible; thoughtful encodings; tells a story	Readable but some issues (scales/labels)	Misleading or hard to read	**25**
Communication	Concise report; compelling slides; executive summary & recommendations	Understandable but unfocused	Disorganized; jargon-heavy	**10**
Reproducibility & documentation	Excellent README; environment; run instructions	Some gaps	Major gaps	**5**
Teamwork & professionalism	Balanced contributions; on-time; peer reviews positive	Uneven but functional	Dysfunctional; missed deadlines	**5**

Area	Excellent (A)	Adequate (B/C)	Needs work (D/F)	Pts
Extra credit	Deployed demo; novel interaction; reusable package	—	—	**+5**

12) Submission format

- Submit **repo link** and a zipped release (excluding large raw data). Include a small sample dataset if original is huge.
- Upload **PDF report** and **PDF/PPTX slides** to the LMS.
- Provide a **live link** or **recorded demo** if using a web dashboard.

Filename convention: `A2_TeamName_Title_report.pdf`, `A2_TeamName_slides.pdf`.

13) Teamwork & peer assessment

- List contributions (1–2 sentences per member) in the report appendix.
- Complete the individual **peer evaluation** (confidential, simple rubric: effort, reliability, quality, communication; 1–5 scale with comments). Large disparities may adjust individual grades.

14) Academic integrity & AI tools policy

- You may use coding assistants (e.g., Copilot/ChatGPT) for **boilerplate or refactoring**. You must add a short **"AI assistance log"** in the appendix describing what was assisted.
- **Do not** fabricate data, results, or citations. All sources must be cited. Generated text/figures must be checked for correctness and edited into your own voice.
- If a tool generates code/visuals, **you are responsible** for understanding them.

15) What "Good" looks like

- A realistic question tied to a stakeholder and metric.
- Clean, modular code with one-command reproduction.
- Visuals that make the insight obvious without verbal explanation.
- Honest discussion of limitations and risks.
- Actionable recommendations tied to the data.

16) Templates

Proposal (max 1 page):

- Title & team name

- Stakeholder & decision context
- Primary question & success metric(s)
- Datasets (source, license, size, fields, refresh)
- Risks/assumptions
- Sketch of key chart(s) or dashboard layout

Final report (structure):

1. Executive summary
2. Problem statement & success metrics
3. Data (sources, schema, quality, ethics)
4. Methods (pipeline, EDA, modeling/validation)
5. Results & visualizations (with captions)
6. Recommendations & impact
7. Limitations & future work
8. Reproduction notes (how to run)
9. Appendix (AI assistance log, team contributions)

README (minimum sections): project overview · environment/setup · data access instructions · how to run · file map · license

---

17) Presentation guidance (8–10 minutes)

- **Slide 1:** Problem & stakeholder
- **Slide 2:** Data sources & quality
- **Slide 3–4:** Key visuals & findings
- **Slide 5:** Recommendations (tied to metrics)
- **Slide 6:** Limitations & next steps
- **Backup:** Method details

Tips: big fonts, few numbers per slide, annotate charts, practice time.

---

18) FAQ

- **What qualifies as a full pipeline?** You show the path from raw data to decision-ready visuals, with code/artifacts that someone else can run.
- **Do we need a predictive model?** Not required, but you should include rigorous analysis (e.g., hypothesis tests, baselines, or simple models) where appropriate.
- **Can we scrape websites?** Yes—**as a supportive data source** when APIs/open datasets don't suffice, **only if** the site's terms permit it. Respect robots.txt and rate limits; don't bypass authentication, paywalls, or CAPTCHAs; avoid collecting PII; include a link to the site's terms in your appendix and provide your scraping code or a clear description.

- **Can we use proprietary data?** No, 4 weeks are not sufficient to have all document signed, with all other works done.
- **How big should the dashboard be?** 2–5 linked views is typical; prioritize clarity over feature count.