Take-Home Midterm: Visualizing War, Regime Type, Distance, and Trade (R + GitHub / Canvas)

Overview. This take-home midterm asks you to construct a small analytical dataset using the Correlates of War framework and visualize relationships between war, regime type, distance, and trade. Your goal is not to estimate causal effects, but to use visualization to explore patterns, diagnose structure, and communicate tradeoffs clearly.

Allowed resources. You may use course materials, textbooks, documentation, and online resources.

AI restriction (required).

- You **may not** use AI tools (e.g., ChatGPT, Copilot, or similar systems) to write your answers. Your written interpretations and explanations must be your own.
- The exam is open book, note, and web. **However, you are responsible for understanding and explaining your code and results.**
- You may not work with anyone else on the exam. This has to be your own work.

Provided dataset (required). You will begin from a provided dyad-year dataset (not from `peacesciencer`). The dataset is a dyad-year panel with one row per country-pair per year.

Mini codebook (variables you will use). The dataset includes the following columns:

- `year`: calendar year.
- `ccode1`, `ccode2`: COW country codes for state 1 and state 2.
- `iso3c1`, `iso3c2`: ISO-3 abbreviations for state 1 and state 2.
- `capdist`: capital-to-capital distance (km). (May contain missing values.)
- `trade`: dyadic trade volume (may be zero; may contain missing values).
- `cowinteronset`: indicator (0/1) for an inter-state war onset in that dyad-year (may contain missing values).
- `cowinterongoing`: indicator (0/1) for an inter-state war ongoing in that dyad-year (may contain missing values).
- `polity21`, `polity22`: regime category for each state in the dyad-year. Values include `Democracy`, `Autocracy`, and `Anocracy`.
- `initiator1`, `initiator2`: Country that started the conflict.

Important note on aggregation. You will need to construct at least one **system-year** (or year-level) dataset from this dyad-year data. For example, you may aggregate dyad-year war indicators into counts per year (total wars per year; wars between democracies per year), and you may aggregate dyad-year relationships into year-level summaries as needed for visualization.

Submission options (choose ONE): Canvas *or* GitHub.

- **Option A (Canvas submission).**
  - Upload **one ZIP file** to Canvas.
  - The ZIP must contain (at minimum) these folders and files:

    - `scripts/midterm.R` *(preferred)* or `midterm.qmd` / `midterm.Rmd`
    - `outputs/writeup.md`
    - `figures/` (your saved figures)
  - **Do not** upload many loose files individually. Submit a single ZIP that preserves the folder structure.
- **Option B (GitHub submission).**
  - Push your work to your GitHub repo with the same required structure:
    - `scripts/midterm.R` *(preferred)* or `midterm.qmd` / `midterm.Rmd`
    - `outputs/writeup.md`
    - `figures/` (your saved figures)

Reproducibility requirements.

- Use an **R Project**.
- Use a **sequential, hard-coded workflow** (no user-defined functions).
- Save figures using `ggsave()` (no screenshots).
- Organize your work with `scripts/`, `outputs/`, and `figures/`.

Visualization requirements.

- You must produce at least **four** figures, each answering a **different** analytical question.
- At least one figure must be shown in **two versions**:
  - raw scale, and
  - transformed scale (e.g., log, rate, proportion, binning).
- You must use **color** intentionally:
  - color mappings must be intuitive for the task,
  - color choices must be accessibility-conscious (colorblind-friendly),
  - and you must justify your color choices in writing.

Tasks

1. **Data construction transparency (required).**
   Start from the provided dyad-year dataset and construct the analysis datasets you need for the tasks below. At minimum, you must construct:
   - a **year-level (system-year)** dataset that supports "wars per year" summaries, and
   - a **dyad-year** dataset (possibly filtered) that supports distance–trade–war visualizations.

   Requirement: In your write-up, briefly describe how you aggregated from dyads to years (what you counted/summed, and how you handled missing values).

   In your write-up, include a section titled **Data construction decisions** listing:
   - how you created your year-level dataset from dyad-year data (aggregation rule),
   - the unit(s) of analysis you created (system-year vs dyad-year),
   - and what observations you filtered or dropped (and why).
2. **System-level war trends (required).**
   Using your system-year dataset, create a visualization showing:
   - the total number of wars per year, and
   - the number of wars between democracies per year.

   Requirement: You must justify your design choice (counts vs rates, smoothing vs no smoothing, scale choices) in 3–5 sentences.
3. **Regime type comparisons (required).**
   Create at least one visualization comparing war patterns across dyad regime types:

- democracy–democracy,
- mixed dyads,
- autocracy–autocracy.

Requirement: Regime type must be encoded clearly (color or faceting). If you use color, justify palette choice and accessibility.

4. **Distance and war (required).**
   Using dyad-year data, visualize the relationship between:
   - distance between states, and
   - war occurrence or war frequency.

   Requirement: You must explain how your visualization approach addresses the fact that war is a rare event.
5. **Trade, distance, and conflict (required).**
   Create a visualization that jointly considers:
   - trade volume,
   - distance,
   - and war occurrence.

   You may choose the form (scatter, faceting, binned summaries, transformed axes), but you must justify why your approach is informative given the distribution of the data.
6. **Raw vs transformed scale comparison (required).**
   Choose one of your figures and produce two versions:
   - a raw-scale version, and
   - a transformed-scale version.

   In your write-up, explain:
   - what becomes easier to see after transformation,
   - what becomes harder to interpret,
   - and which version you would show to (i) a policymaker and (ii) a methods audience (with justification).
7. **One "bad" visualization (required).**
   Produce one intentionally misleading or poorly designed visualization related to this midterm.

   In your write-up, include 3–5 bullet points explaining precisely why it is misleading (e.g., inappropriate scale, poor color encoding, missing context, distortion, or ambiguous labeling).
8. **Interpretation and narrative (required).**
   In `outputs/writeup.md`, write a coherent narrative (approximately 800–1200 words) that:
   - introduces what you attempted to learn from the data,
   - walks through your figures in a logical order,
   - interprets what the figures show,
   - and highlights limitations and design tradeoffs.

   Color requirement (repeat): For each figure that uses color, include 1–2 sentences justifying:
   - what color encodes,
   - why color is appropriate,
   - and why your palette is accessibility-conscious.
9. **Submission proof (required). Choose the proof that matches your submission method.**
   - **If you submit via GitHub:** In your write-up, paste:
     - the output of `git status` after committing (clean working tree), and
     - the output of `git log -3` showing at least three commits.

- **If you submit via Canvas:** In your write-up, include:
  - a short **file tree** of what is inside your ZIP (you can type it manually), showing `scripts/`, `outputs/`, and `figures/`, and
  - the **exact filename** of the ZIP you uploaded to Canvas (e.g., `Lastname_Firstname_Midterm.zip`).

Checklist (before submitting)

- Your code runs top-to-bottom in a fresh R session
- `scripts/midterm.R` (or `midterm.qmd`/`midterm.Rmd`) exists
- `outputs/writeup.md` exists and contains all required sections
- At least four distinct figures saved in `figures/`
- One figure appears in raw and transformed versions
- One intentionally "bad" visualization included with critique bullets
- You submitted via **either** Canvas **or** GitHub (not both)
- You included the correct submission proof for your chosen method (GitHub proof *or* Canvas ZIP proof)