Panel Data Quality Checklist

Collin Zoeller

Structural Integrity Checks

- ☐ **Check Panel Structure:** The dataset has unique keys for individuals (`id`) and time periods (`t`). Remove Duplicates.
- ☐ **Balanced vs. Unbalanced Panel:** Dataset is balanced or unbalanced if appropriate.
- ☐ **Expected Number of Observations:** The total number of observations matches expectations ($N \times T$ for balanced panels).
- ☐ **Missing `id` or `t` Values:** Identify missing `id` or `t` values that could distort the panel structure.
- ☐ **Repeated Observations:** Remove duplicate observations across all variables, not just (`id`, `t`).

Consistency Checks

- ☐ **Consistent Identifiers:** `id` values are consistently formatted and do not change over time.
- ☐ **Time Gaps:** Missing time periods within individual panels could indicate data collection issues.
- ☐ **Logical Time Order:** `t` values are in the correct sequence and there are no backward jumps.
- ☐ **Time-Invariant Variables:** Time-invariant variables (e.g., gender, country of birth) unchanged within individuals.
- ☐ **Stationary vs. Non-Stationary Trends:** Assess whether key variables exhibit trends over time that should be accounted for.
- ☐ **Seasonal Patterns:** Check for seasonal effects in time-series variables.
- ☐ **Lagged Values Consistency:** Lagged variables (e.g., `x[t-1]`) properly align in the dataset.

Data Quality and Completeness

- ☐ **Missing Values:** Identify the extent of missing values in key variables.
- ☐ **Imputation Strategy:** Decide whether missing data should be dropped, interpolated, or imputed.
- ☐ **Outliers:** Detect and investigate extreme values that may indicate errors or unusual cases.
- ☐ **Data Entry Errors:** Check for implausible values (e.g., negative income, impossible dates).
- ☐ **Extreme Values Across Time:** Outliers remain consistent over time and aren't due to data recording errors.
- ☐ **Heaping in Numeric Variables:** Detect rounding or heaping issues (e.g., income values clustered at multiples of 10 or 100).

Distributional and Grouping Checks

- ☐ **Expected Group Sizes:** Stratified groups (e.g., age brackets, income deciles) contain reasonable sample sizes.
- ☐ **Distribution Stability:** Compare distributions of key variables across time to detect data shifts.
- ☐ **Heterogeneity:** Test for group-specific differences that may indicate model misspecification.
- ☐ **Percentile Breakpoints:** Distributional splits (e.g., quartiles) match expectations and do not create artificial biases.
- ☐ **Sample Representativeness:** Demographic, regional, industry-level distributions stable over time.
- ☐ **Normalization of Continuous Variables:** Transformations (e.g., logs, standardization) correctly applied.

Cross-Sectional and Longitudinal Relationships

- ☐ **Autocorrelation Patterns:** Examine correlations across time within units for key variables.
- ☐ **Key Ratio Consistency:** Ratios (e.g., revenue per employee, debt-to-income) remain stable unless justified by economic events.
- ☐ **Lag and Lead Effects:** Lagged and lead variables align correctly with time indexes.
- ☐ **Attrition and Truncation Bias:** Assess whether dropout patterns over time create biases or if there is a period cutoff.
- ☐ **Structural Breaks:** Identify sudden changes in key relationships that might suggest data recording issues.

Merging and Integration Checks

- ☐ **Key Identifiers in Merged Data:** `id` and `t` remain unique after merge.
- ☐ **Data Merge Completeness:** Expected number of observations remains correct post-merge.
- ☐ **Unit Consistency Across Datasets:** Variables like firm size remain comparable when merging external data sources.
- ☐ **Check for Spurious Merges:** Merge has not unintentionally increased the number of observations.
- ☐ **Duplicate Merges:** Merge does not create multiple matches for a single `id`, `t` pair.

Code and Documentation Checks

- ☐ **Replication:** Ensure that code produces identical datasets when rerun.
- ☐ **Variable Documentation:** Maintain clear documentation for all transformations and derived variables.
- ☐ **Version Control:** Keep a log of dataset versions and changes to track modifications.
- ☐ **Code Optimization:** Identify and eliminate inefficient operations that slow processing time.

Final Review

- ☐ **Exploratory Summary Statistics:** Run summary statistics (`sum`, `tab`, `xtsum`, `summarize`) to check basic properties.
- ☐ **Visualization Checks:** Use time series plots, histograms, and boxplots to inspect trends and distributions.
- ☐ **Sanity Checks:** Ensure results align with prior knowledge and theoretical expectations.
- ☐ **Cross-validation with External Sources:** Compare key statistics with external benchmarks or prior research.
- ☐ **Longitudinal Balance in Summary Stats:** Compare pre/post-split statistics when filtering data (e.g., before/after imputation).