Documentation for Pseudodata Generation

To facilitate transparency and reproducibility in our analysis while preserving the confidentiality of the proprietary dataset, we generated a synthetic pseudodataset that mirrors the structure, variable types, and roughly approximate distributional properties of the original data. This pseudodataset was created using Python scripts to simulate realistic values for all variables, ensuring that it can be used for code validation, sensitivity testing, and referee review without exposing sensitive information. The generation process adheres to the following principles:

1. **Reproducibility**: All random processes are seeded (e.g., using np.random.seed(123) or equivalents) to ensure identical outputs across runs. This allows exact replication of the pseudodataset.
2. **Structural Fidelity**: The pseudodataset replicates the exact column names, data types (e.g., integers, floats, categoricals, dates), and logical relationships (e.g., conditional missingness based on loan outcome) from the original dataset. Derived variables (e.g., dummy indicators) are computed similarly.
3. **Distributional Approximation**: Values are drawn from distributions (e.g., uniform, normal, discrete choices) that roughly approximate the ranges, based on aggregate summaries. Exact means or variances from the original data were not used; instead, meaningful estimates were employed.
4. **Independence and Realism**: Unless explicitly constrained (e.g., experience capped by age), variables are generated independently. Constraints ensure logical consistency (e.g., ages clipped to 22–65 years; approval amounts set to zero for rejected loans). Probabilities (e.g., 4% for defaults) are calibrated to match observed rates in the real data.
5. **Sampling with Replacement**: Customer managers and loan officers are sampled from fixed pools to simulate assignment in a banking environment, allowing for repeated assignments as in real operations.
6. **No Data Leakage**: The pseudodataset does not incorporate any actual observations, correlations, or patterns from the confidential dataset beyond high-level summaries (e.g., variable ranges and types).

The pseudodataset consists of 1,436 observations (matching the original sample size), saved as a pipe-separated CSV file (`pseudo_data.csv`) with commas as decimal separators for compatibility with certain locales.

Step-by-Step Generation Process

The pseudodataset is generated in three main components: (1) a pool of customer managers, (2) a pool of loan officers, and (3) the main loan application data, which integrates samples from these pools along with independently generated variables. The process is implemented in three Python scripts: customer_managers.py, loan_officers.py, and gen_pseudo_data.py.

1. Customer Managers Pool (from customer_managers.py)

A pool of 200 unique customer managers is created using a CustomerManagers class. This simulates bank employees responsible for initial client interactions. Key features include:

- Initialization: Seeded with np.random.seed(42); generates 200 rows.
- Demographics: Unique IDs, random gender and education, ages from normal distribution clipped to 22–65, experience capped at age − 18, derived dummies for gender and education.
- Biospheric Values: Four Likert items (−1 to 7) and their mean.
- Environmental Beliefs/Knowledge: Eight Likert items (1 to 5).
- Portfolio Statistics: Numbers of loans managed (brown and green) and performance rates within specified ranges.

2. Loan Officers Pool (from loan_officers.py)

A pool of 64 unique loan officers is created using a LoanOfficers class. This simulates employees handling loan approvals. The structure parallels customer managers but with adjusted scales:

- Initialization: Seeded with np.random.seed(123); generates 64 rows.
- Demographics: Similar to managers.
- Biospheric Values: Same structure as managers.
- Environmental Beliefs/Knowledge: Eight Likert items (1 to 5).
- Portfolio Statistics: Green and brown loan portfolios with approval, late, and default rates in specified ranges.

3. Main Loan Application Data (from gen_pseudo_data.py)

This script integrates the sampled managers and officers with independently generated variables to form 1,436 rows. Seeded with np.random.seed(112). Variables include:

- Training Groups: Green, brown, and control groups assigned with probabilities 0.33, 0.33, and 0.34.
- Company Characteristics: Industry, registration type, establishment year, financial metrics, and actual controller gender.
- Risk Assessments: Likert scales for repayment ability, environmental risk, etc.
- Loan Outcomes: Outcome variable (0/1) with conditional assignment of approved amount, interest rate, collateral, contract dates, and performance flags (default, late repayment).
- Integration of Managers and Officers: Columns from sampled pools merged directly.

Additional Manual Steps After Data Generation

To transform the generated data into a compatible Excel format:
- Open the CSV file in Excel.
- Divide the data into columns using the '|' separator (Data > Text to Columns > Delimited > Other: '|').

- Convert the Contract Start Date and Contract expiration date columns to format yyyy-mm-dd hh:mm:ss (or locale equivalent).
- Save the file as .xlsx.