

Computational Reproducibility

C.1 Purpose

All principal empirical stages are implemented programmatically in Python.

The computational design follows three principles:

- transparency;
- modularity;
- reproducibility.

Each stage creates intermediate files that can be inspected before the next stage is executed.

The analysis is therefore not implemented as one opaque program.

C.2 Software Environment

The research environment uses Python and open-source packages including

- `pandas`;
- `numpy`;
- `requests`;
- `scikit-learn`;
- `statsmodels`;
- `matplotlib`.

A Python virtual environment can be created using

```
python3 -m venv .venv
source .venv/bin/activate
```

The required packages can then be installed within the virtual environment.

C.3 Computational Sequence

The final empirical implementation is organized around seven principal scripts.

C.3.1 Script 01: SEC Data Acquisition

```
01_download_sec_data.py
```

The first script

- defines the predetermined sample;
- resolves tickers to CIKs;
- retrieves SEC Company Facts data;
- downloads the required support and analysis years;
- stores source data for subsequent processing.

The SEC request header contains an identifying User-Agent and contact address.

C.3.2 Script 02: Panel Construction

```
02_construct_panel.py
```

The second stage

- reads the downloaded accounting facts;
- resolves preferred and fallback XBRL concepts;
- distinguishes stocks and flows;
- aligns observations to fiscal years;

- selects annual observations;
- constructs one firm-year accounting panel.

C.3.3 Script 03: Proxy Construction

The third stage constructs the Accounting Bias proxy variables and firm-level controls.

Depending on the final local filename, this is the Script 03 responsible for

- ratio construction;
- average-asset calculations;
- lagged variables;
- proxy validation;
- construction of the regression-ready Accounting Bias panel.

The principal resulting variables are

$$RDINT, \quad GWINT, \quad PPEINT, \\ ACCDEPINT, \quad ACCRUALINT, \quad DTINT.$$

C.3.4 Script 04: Structural Analysis

`04_analyze_abi_structure.py`

The structural-analysis stage generates

- descriptive statistics;
- proxy missingness diagnostics;
- correlation matrices;
- pairwise observation counts;
- firm-level proxy means;
- year-level proxy means;
- within-firm and between-firm diagnostics;
- dimension-specific PCA;
- cross-dimensional PCA.

This stage is exploratory.

Its purpose is to characterize the observed variables before formal hypothesis testing.

C.3.5 Script 05: Hypothesis-Test Preparation

Script 05 prepares the final statistical inputs required for the formal hypothesis-testing stage.

Its role is kept separate from the estimation program so that sample construction and model estimation remain distinguishable.

C.3.6 Script 06: Hypothesis Testing

`06_run_hypothesis_tests.py`

The main statistical script estimates

- industry heterogeneity tests;
- persistence models;
- within-dimension regressions;
- firm-fixed-effect robustness models;
- single-proxy ROA regressions;
- the restricted multivariate ROA model.

Standard errors in the main regression specifications are clustered at the firm level.

C.3.7 Script 07: Publication Output

`07_export_book_results.py`

The final stage transforms validated statistical results into publication-ready

- CSV tables;
- LaTeX tables;
- PDF figures;
- PNG figures;

- output manifests.

The manuscript imports these files directly.

Thus,

data $\rightarrow$ estimation $\rightarrow$ automatic output $\rightarrow$ manuscript.

C.4 Running the Pipeline

A complete local execution follows the sequence

```
cd ~/Desktop/book_code
source .venv/bin/activate

python 01_download_sec_data.py
python 02_construct_panel.py
python 03_construct_abi_proxies.py
python 04_analyze_abi_structure.py
python 05_prepare_hypothesis_tests.py
python 06_run_hypothesis_tests.py
python 07_export_book_results.py
```

The exact names of Script 03 and Script 05 should correspond to the final files included with the research project.

C.5 Generated Publication Files

The final export stage generates, among others,

```
book_results/
tables/
table_proxy_coverage.tex
table_descriptive_statistics.tex
table_hypothesis_summary.tex
table_persistence.tex
table_dimension_associations.tex
table_roa_single_proxy.tex
table_roa_multivariate.tex
table_pca_cross_dimension.tex
```

```
figures/  
figure_industry_profiles.pdf  
figure_proxy_persistence.pdf  
figure_within_between_variation.pdf
```

These files are inserted directly into the empirical chapters.
For example,

```
\input{book_results/tables/table_persistence.tex}
```

and

```
\includegraphics[  
width=.9\textwidth  
]{book_results/figures/figure_proxy_persistence.pdf}
```

This reduces manual transcription risk.

C.6 Reproducibility and Version Dependence

The pipeline is reproducible from publicly available SEC data, but exact replication depends on preserving

- the predetermined firm list;
- the XBRL fallback definitions;
- the extraction logic;
- the software versions;
- the downloaded source observations;
- the statistical specifications.

Public filing databases can evolve over time.

For strict archival reproducibility, the raw downloaded source files should therefore be retained together with the analysis code.

C.7 Separation of Construction and Inference

A central design principle is the separation

Data Construction $\neq$ Statistical Inference.

Ticker resolution, concept selection, missing-data treatment, fiscal-year alignment, and proxy construction occur before the main hypothesis-testing stage.

This reduces the risk that data-processing decisions are implicitly optimized to obtain particular regression results.