Replication Package for “Binary Choice under Asymmetric Loss in a Data-Rich Environment: Theory and an Application to Algorithmic Fairness”

Andrii Babii, Xi Chen, Eric Ghysels, and Rohit Kumar

November 3, 2025

Overview

The code in this replication package constructs a cleaned dataset from ProPublica (2016) raw files and performs the empirical analysis and simulations. The code is stored in two `Python` and one `R` Jupyter Notebooks. These three notebooks contain all the code to generate results for four tables and six figures in the paper, as well as five tables and one figure in the Online Appendix.

Data Availability and Provenance Statements

The data used in this study come from ProPublica (2016) and is publicly available at their Data Store Archive. ProPublica’s terms of use do not allow republishing the raw data. Our `data_cleaning.ipynb` notebook, provided as part of the replication package, contains all the code necessary to download the raw ProPublica (2016) data from their repository, clean it, and save the cleaned dataset to `data/data_clean.csv`.

Summary of Availability

☒ All data **are** publicly available.
☐ Some data **cannot be made** publicly available.
☐ **No data can be made** publicly available.

Details on each Data Source

The raw dataset can be downloaded from the ProPublica Data Store Archive. We use the raw dataset consisting of a single file, named `compas.db`:

This file can be downloaded directly from https://github.com/propublica/compas-analysis/blob/master/compas.db.

Data Name	Data Files	Location	Provided	Citation
COMPAS Data	`compas.db`	ProPublica Repository	No	ProPublica (2016)

Dataset list

The cleaned dataset, provided as part of the replication package is:

Data Name	Data Files	Location	Provided	Source
Clean COMPAS Data	`data_clean.csv`	`/data` folder	Yes	ProPublica (2016)

Computational requirements

Software Requirements

- `Python` 3.12.2
  - `numpy`: 1.26.4
  - `pandas`: 2.2.2
  - `sklearn`: 1.7.2
  - `tensorflow`: 2.19.0
  - `keras`: 3.9.2
  - `matplotlib`: 3.9.2
  - `xgboost`: 3.0.1
  - `sqlite3`: 3.46.0
  - `urllib`: 3.12
- `R` 4.4.1
  - `dplyr`: 1.1.4
  - `magrittr`: 2.0.3
  - `purrr`: 1.0.2
  - `pROC`: 1.18.5
  - `xgboost`: 1.7.8.1
  - `Matrix`: 1.7.0
  - `caret`: 6.0.94
  - `glmnet`: 4.1.8
  - `reshape2`: 1.4.4

- stargazer: 5.2.3
- hdi: 0.1.10
- ggplot2: 3.5.1

Controlled Randomness

☒ Random seed is set at line 39 in the first cell of notebook `empirical_application.ipynb`
☒ Random seeds are set at line 45-52 in the first cell and lines 3-5 in cells 2-6 and lines 61-63 in cell 1 of notebook `simulations.ipynb`

Memory, Runtime, Storage Requirements

Approximate time needed to reproduce the analyses on a standard (2025) desktop machine:

☐ <10 minutes
☐ 10-60 minutes
☐ 1-2 hours
☐ 2-8 hours
☐ 8-24 hours
☒ 1-3 days
☐ 3-14 days
☐ > 14 days

Approximate storage space needed:

☐ < 25 MBytes

☒ 25 MB - 250 MB

☐ 250 MB - 2 GB

☐ 2 GB - 25 GB

☐ 25 GB - 250 GB

☐ > 250 GB

☐ Not feasible to run on a desktop machine, as described below.

All the computations were conducted using parallel computing on a **Apple M2 Macbook Air laptop with MacOS Sequoia, version 15.6.1**.

Description of programs/code

- The Python Jupyter notebook `data_cleaning.ipynb` will extract and reformat the raw data from the ProPublica repository into a cleaned dataset `data_clean.csv`. These computations can be run in less than 5 minutes.

- The R Jupyter notebook `empirical_application.ipynb` produces all empirical results in the paper and Online Appendix from the cleaned data `data_clean.csv`. These computations can be run under three hours.
- The Python Jupyter notebook in `simulations.ipynb` produces all simulation results in the paper and Online Appendix. These computations can be run in under 30 hours.

Instructions to Replicators

- The `code/data_cleaning.ipynb` notebook should be run first. It will save the cleaned data to `data_clean.csv`. It will also print the `LaTeX` code with summary statistics for Table OA.3 and save it to `tab/tab_oa3.tex`.
- The `code/empirical_application.ipynb` notebook should be run next to reproduce all empirical results in the paper and Online Appendix. It will print the `LaTeX` output for tables, save it to `/tab` folder, and save all figures to `/fig` folder.
- The `code/simulations.ipynb` notebook can be run independently of the other two notebooks to reproduce all simulation results in the paper and Online Appendix. It will print the `LaTeX` output for tables and save all figures to the `/fig` folder, and save all tables to the `/tab` folder.

List of tables and programs

The provided code reproduces:

☒ All numbers provided in text in the paper
☒ All tables and figures in the paper

Figure/Table	Notebook	Cells	Output
Table 1	`simulations.ipynb`	#2	`tab/tab_1_n1000.tex`
Table 1	`simulations.ipynb`	#7	`tab/tab_1_n10000.tex`
Table 2	`simulations.ipynb`	#3	`tab/tab_2.tex`
Table 3	`simulations.ipynb`	#4	`tab/tab_3.tex`
Table 4	`empirical_application.ipynb`	#4	`tab/tab_4.tex`
Table OA.1	`simulations.ipynb`	#3	`tab/tab_oa1.tex`
Table OA.2	`simulations.ipynb`	#4	`tab/tab_oa2.tex`
Table OA.3	`data_cleaning.ipynb`	#126	`tab/tab_oa3.tex`
Table OA.4	`empirical_application.ipynb`	#3	`tab/tab_oa4.tex`
Table OA.5	`empirical_application.ipynb`	#13	`tab/tab_oa5.tex`
Figure 1	`simulations.ipynb`	#6	`fig/fig_1.pdf`
Figure 2	Generated directly in manuscript using `tikz` graphics		
Figure 3 (a)	`simulations.ipynb`	#5	`fig/fig_3a.pdf`
Figure 3 (b)	`simulations.ipynb`	#5	`fig/fig_3b.pdf`
Figure 3 (c)	`simulations.ipynb`	#5	`fig/fig_3c.pdf`
Figure 3 (d)	`simulations.ipynb`	#5	`fig/fig_3d.pdf`

Figure 4	`empirical_application.ipynb`	#2	`fig/fig_4.pdf`
Figure 5 (a)	`empirical_application.ipynb`	#6	`fig/fig_5a.pdf`
Figure 5 (b)	`empirical_application.ipynb`	#10	`fig/fig_5b.pdf`
Figure 6 (a)	`empirical_application.ipynb`	#8	`fig/fig_6a.pdf`
Figure 6 (b)	`empirical_application.ipynb`	#8	`fig/fig_6b.pdf`
Figure 6 (c)	`empirical_application.ipynb`	#12	`fig/fig_6c.pdf`
Figure 6 (d)	`empirical_application.ipynb`	#12	`fig/fig_6d.pdf`
Figure 7 (a)	`empirical_application.ipynb`	#8	`fig/fig_7a.pdf`
Figure 7 (b)	`empirical_application.ipynb`	#8	`fig/fig_7b.pdf`
Figure 7 (c)	`empirical_application.ipynb`	#12	`fig/fig_7c.pdf`
Figure 7 (d)	`empirical_application.ipynb`	#12	`fig/fig_7d.pdf`
Figure OA.1	Generated directly in manuscript using `tikz` graphics		
Figure OA.2 (a)	`empirical_application.ipynb`	#6	`fig/fig_oa2a.pdf`
Figure OA.2 (b)	`empirical_application.ipynb`	#6	`fig/fig_oa2b.pdf`
Figure OA.2 (c)	`empirical_application.ipynb`	#10	`fig/fig_oa2c.pdf`
Figure OA.2 (d)	`empirical_application.ipynb`	#10	`fig/fig_oa2d.pdf`

References

ProPublica. 2016. COMPAS Recidivism Risk Score Data and Analysis (Broward County, FL, 2013—2014). ProPublica Data Archive/GitHub. https://github.com/propublica/compas-analysis