This form documents the artifacts associated with the article (i.e., the data and code supporting the computational findings) and describes how to reproduce the findings.

Part 1: Data

☐ This paper does not involve analysis of external data (i.e., no data are used or the only data are generated by the authors via simulation in their code).

☒ I certify that the author(s) of the manuscript have legitimate access to and permission to use the data used in this manuscript.

Abstract

Availability

☐ Data **are** publicly available.
☒ Data **cannot be made** publicly available.

If the data are publicly available, see the *Publicly available data* section. Otherwise, see the *Non-publicly available data* section, below.

Publicly available data

☐ Data are available online at:

☐ Data are available as part of the paper's supplementary material.

☐ Data are publicly available by request, following the process described here:

☐ Data are or will be made available through some other mechanism, described here:

Non-publicly available data

The data that support the findings of this study are available from a ride-hailing platform. Restrictions apply to the availability of these data, which were used under license for this study.

Description

File format(s)

☐ CSV or other plain text.
☐ Software-specific binary format (.Rda, Python pickle, etc.): pkcle
☐ Standardized binary format (e.g., netCDF, HDF5, etc.):
☐ Other (please specify):

Data dictionary

☐ Provided by authors in the following file(s):
☐ Data file(s) is(are) self-describing (e.g., netCDF files)
☐ Available at the following URL:

Additional Information (optional)

Part 2: Code

Abstract

The proposed estimators are implemented in `opeuc.py` including direct estimator, importance sampling estimator and confounded off-policy estimator. The other code are implemented for estimating nuisance parameters, sampling, and numerical experiments.

Description

Code format(s)

- ☒ Script files
  - ☐ R
  - ☒ Python
  - ☐ Matlab
  - ☐ Other:
- ☒ Package
  - ☐ R
  - ☒ Python
  - ☐ MATLAB toolbox
  - ☐ Other:
- ☐ Reproducible report
  - ☐ R Markdown
  - ☐ Jupyter notebook
  - ☐ Other: a readme
- ☐ Shell script
- ☐ Other (please specify):

Supporting software requirements

Version of primary software used Python version 3.7.8

Libraries and dependencies used by the code

- `numpy`==1.20.3
- `scikit-learn`==0.24.2
- `scipy`==1.6.3
- `tensorflow-cpu`==2.6.0
- `pandas`==1.2.4

Supporting system/hardware requirements (optional)

Windows 11, Intel(R) Core(TM) i9-9940X CPU @ 3.30GHz 3.31 GHz, 48.0 GB

Parallelization used

- ☒ No parallel code used
- ☐ Multi-core parallelization on a single machine/node
  - Number of cores used:
- ☐ Multi-machine/multi-node parallelization
  - Number of nodes and cores used:

License

- ☐ MIT License (default)
- ☐ BSD
- ☒ GPL v3.0
- ☐ Creative Commons
- ☐ Other: (please specify)

Additional information (optional)

Part 3: Reproducibility workflow

Scope

The provided workflow reproduces:

- ☐ Any numbers provided in text in the paper
- ☒ The computational method(s) presented in the paper (i.e., code is provided that implements the method(s))
- ☐ All tables and figures in the paper
- ☒ Selected tables and figures in the paper, as explained and justified below:

- Figure 2
- Figure 3
- Figure S1
- Figure S2
- Figure S3
- Table S1
- Table S2

Workflow

Location

The workflow is available:

- ☐ As part of the paper's supplementary material.
- ☒ In this Git repository: the `main` branch
- ☐ Other (please specify):

Format(s)

- ☐ Single master code file
- ☐ Wrapper (shell) script(s)
- ☐ Self-contained R Markdown file, Jupyter notebook, or other literate programming approach
- ☒ Text file (e.g., a readme-style file) that documents workflow
- ☐ Makefile
- ☐ Other (more detail in *Instructions* below)

Instructions

- `sim_robust.py` –> Figure 2
- `sim_trajectory_compare_multdim.py` & `sim_time_compare_multdim.py` –> Figure 3
- `sim_trajectory_compare.py` & `sim_time_compare.py` –> Figure S3
- `sim_ratiolearner_compare.py` –> Table S1, Table S2, Figure S1
- `sim_ratio_features_number_compare.py` –> Figure S2

Expected run-time

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

- ☐ < 1 minute
- ☐ 1-10 minutes
- ☐ 10-60 minutes
- ☒ 1-8 hours
- ☐ > 8 hours
- ☐ Not feasible to run on a desktop machine, as described here:

Additional information (optional)

Notes (optional)