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

Two public datasets are used in this paper. Instructions on how to download the white wine quality data can be found here. Instructions on how to download the bike sharing system data can be found here.

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: White Wine Quality and Bike Sharing

☐ 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

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: White Wine Quality and Bike Sharing

Additional Information (optional)

Part 2: Code

Abstract

The code files can reproduce simulation results, figures and real data analysis results from the paper "A unified framework for residual diagnostics in generalized linear models and beyond"

Description

Code format(s)

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

Supporting software requirements

Version of primary software used

- R version 4.2.1

Libraries and dependencies used by the code

- R Packages:
  - brglm2 0.9
  - ExtDist 0.7-2
  - forecast 8.21.1
  - ggplot2 3.4.4
  - ggpointdensity 0.1.0
  - gridExtra 2.3
  - lubridate 1.8.0
  - MASS 7.3-58.2
  - mgcv 1.8-40
  - np 0.60-17
  - parallel 4.2.1
  - pbmcapply 1.5.1
  - PAsso 0.1.10
  - plyr 1.8.9
  - pscl 1.5.5
  - Rmisc 1.5.1
  - tidyverse 1.3.2

- tscount 1.4.3
- tseries 0.10-55
- tsibble 1.1.3
- vcd 1.4-12
- VGAM 1.1-9

Supporting system/hardware requirements (optional)

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:

Workflow

Location

The workflow is available:

☒ As part of the paper's supplementary material.
☐ In this Git repository:
☐ 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

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: