The Journal of Open Source Software

kilianr: an R package for structural vector autoregressive analysis

Rich Ryan[1]

1 California State University, Bakersfield, USA

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Open Journals

Reviewers:

- @openjournals

Submitted: 01 January 1970
Published: unpublished

Summary

Given time series on, say, gloabl oil production, global economic activity, and the price of oil, a vector autoregressive model identifies the correlations between the data series. How are the correlations to be interpreted causally? How does a disruption in oil supply affect the price of oil? That is, how does a disruption in oil supply affect the price of oil? A prominent way to assign causality uses economic theory. Economic theory provides short-run restrictions that can be used. Economic theory posits restrictions among the correlations. There is good reason and evidence that oil production within the month does not respond to global economic activity or the price of oil (Anderson et al., 2018; Kilian, 2009). Restrictions like these can be translated into features of the statistical model, Restrictions like these can be translated into the statistical model, Timing restrictions like these can be used in the statistical model to generate a structural vector autoregression that allows correlations to be interpreted causally. Within this class of models, inference about responses of variables to unexpected shocks is straightforward The kilianr package implements these procedures (short-run, recursive identifying restrictions). The package builds directly upon Kilian (2009) and Kilian & Lütkepohl (2017).

—in a code base.

How are those correlations causally? One way uses economic theory to specify restrictions that allow the correlations to be identified causally. One of the most prominent ways uses short-run identifying restrictions that recursively identify the model. In precise terms, TKTK.

The forces on stars, galaxies, and dark matter under external gravitational fields lead to the dynamical evolution of structures in the universe. The orbits of these bodies are therefore key to understanding the formation, history, and future state of galaxies. The field of "galactic dynamics," which aims to model the gravitating components of galaxies to study their structure and evolution, is now well-established, commonly taught, and frequently used in astronomy. Aside from toy problems and demonstrations, the majority of problems require efficient numerical tools, many of which require the same base code (e.g., for performing numerical orbit integration).

Pfaff (2008)

Lange et al. (2021)

NOTES: What to include

- software has obvious research application
- JOSS publishes articles about research software. This definition includes software that: solves complex modeling problems in a scientific context (physics, mathematics, biology, medicine, social science, neuroscience, engineering); supports the functioning of research instruments or the execution of research experiments; extracts knowledge from large data

sets; offers a mathematical library, or similar. While useful for many areas of research, pre-trained machine learning models and notebooks are not in-scope for JOSS.

- As a rule of thumb, JOSS' minimum allowable contribution should represent not less than three months of work for an individual.
- **Co-publication**: Sometimes authors prepare a JOSS publication alongside a contribution describing a science application, details of algorithm development, and/or methods assessment. In this circumstance, JOSS considers submissions for which the implementation of the software itself reflects a substantial scientific effort. This may be represented by the design of the software, the implementation of the algorithms, creation of tutorials, or any other aspect of the software. We ask that authors indicate whether related publications (published, in review, or nearing submission) exist as part of submitting to JOSS.
- Submission process
  - Make your software available in an open repository (GitHub, Bitbucket, etc.) and include an OSI approved open source license.
  - Make sure that the software complies with the JOSS review criteria. In particular, your software should be full-featured, well-documented, and contain procedures (such as automated tests) for checking correctness.
  - Write a short paper in Markdown format using paper.md as file name, including a title, summary, author names, affiliations, and key references. See our example paper to follow the correct format.
  - (Optional) create a metadata file describing your software and include it in your repository. We provide a script that automates the generation of this metadata.
- Contents of paper: JOSS welcomes submissions from broadly diverse research areas. For this reason, we require that authors include in the paper some sentences that explain the software functionality and domain of use to a non-specialist reader. We also require that authors explain the research applications of the software. The paper should be between 250-1000 words. Authors submitting papers significantly longer than 1000 words may be asked to reduce the length of their paper.
- What to include
  - A list of the authors of the software and their affiliations, using the correct format (see the example below).
  - A summary describing the high-level functionality and purpose of the software for a diverse, non-specialist audience.
  - A Statement of need section that clearly illustrates the research purpose of the software and places it in the context of related work.
  - A list of key references, including to other software addressing related needs. Note that the references should include full names of venues, e.g., journals and conferences, not abbreviations only understood in the context of a specific discipline.
  - Mention (if applicable) a representative set of past or ongoing research projects using the software and recent scholarly publications enabled by it.
  - Acknowledgement of any financial support.

Statement of need

`Gala` is an Astropy-affiliated Python package for galactic dynamics. Python enables wrapping low-level languages (e.g., C) for speed without losing flexibility or ease-of-use in the user-interface. The API for `Gala` was designed to provide a class-based and user-friendly interface to fast (C or Cython-optimized) implementations of common operations such as gravitational potential and force evaluation, orbit integration, dynamical transformations, and chaos indicators for nonlinear dynamics. `Gala` also relies heavily on and interfaces well with the implementations of physical units and astronomical coordinate systems in the `Astropy` package (**?**) (`astropy.units` and `astropy.coordinates`).

`Gala` was designed to be used by both astronomical researchers and by students in courses on gravitational dynamics or astronomy. It has already been used in a number of scientific publications (?) and has also been used in graduate courses on Galactic dynamics to, e.g., provide interactive visualizations of textbook material (?). The combination of speed, design, and support for Astropy functionality in `Gala` will enable exciting scientific explorations of forthcoming data releases from the *Gaia* mission (?) by students and experts alike.

Mathematics

Single dollars ($) are required for inline mathematics e.g. $f(x) = e^{\pi/x}$

Double dollars make self-standing equations:

$$\Theta(x) = \left\{\begin{array}{l} 0\textrm{ if } x < 0\cr 1\textrm{ else} \end{array}\right.$$

You can also use plain LaTeXfor equations

$$\hat f(\omega) = \int_{-\infty}^{\infty} f(x) e^{i\omega x} dx \tag{1}$$

and refer to Equation 1 from text.

Citations

Citations to entries in paper.bib should be in rMarkdown format.

If you want to cite a software repository URL (e.g. something on GitHub without a preferred citation) then you can do it with the example BibTeX entry below for (?).

For a quick reference, the following citation commands can be used: - `@author:2001` -> "Author et al. (2001)" - `[@author:2001]` -> "(Author et al., 2001)" - `[@author1:2001; @author2:2001]` -> "(Author1 et al., 2001; Author2 et al., 2002)"

Figures

Figures can be included like this: Caption for example figure. and referenced from text using section .

Figure sizes can be customized by adding an optional second parameter: Caption for example figure.

Acknowledgements

We acknowledge contributions from Brigitta Sipocz, Syrtis Major, and Semyeong Oh, and support from Kathryn Johnston during the genesis of this project.

References

Anderson, S. T., Kellogg, R., & Salant, S. W. (2018). Hotelling under pressure. *Journal of Political Economy*, *126*(3), 984–1026. https://doi.org/10.1086/697203

Kilian, L. (2009). Not all oil price shocks are alike: Disentangling demand and supply shocks in the crude oil market. *American Economic Review*, *99*(3), 1053–1069. https://doi.org/10.1257/aer.99.3.1053

Kilian, L., & Lütkepohl, H. (2017). *Structural vector autoregressive analysis*. Cambridge University Press. https://doi.org/10.1017/9781108164818

Lange, A., Dalheimer, B., Herwartz, H., & Maxand, S. (2021). Svars: An r package for data-driven identification in multivariate time series analysis. *Journal of Statistical Software*, *97*(5), 1–34. https://doi.org/10.18637/jss.v097.i05

Pfaff, B. (2008). VAR, SVAR and SVEC models: Implementation within r package vars. *Journal of Statistical Software*, *27*(4), 1–32. https://doi.org/10.18637/jss.v027.i04