# SUBMISSION REQUIREMENTS

* The software should be open source as per the OSI definition.
* The software should have an obvious research application.
* You should be a major contributor to the software you are submitting.
* The software should be a significant contribution to the available open source software that either enables some new research challenges to be addressed or makes addressing research challenges significantly better (e.g., faster, easier, simpler).
* The software should be feature complete (no half-baked solutions) and designed for maintainable extension (not one-off modifications). Minor ‘utility’ packages, including ‘thin’ API clients, are not acceptable.
* In addition, the software associated with your submission must:
  + Be stored in a repository that can be cloned without registration.
  + Be stored in a repository that is browsable online without registration.
  + Have an issue tracker that is readable without registration.
  + Permit individuals to create issues/file tickets against your repository.

# TO DO ITEMS

* Make 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.
* Author a short Markdown paper paper.md with a title, summary, author names, affiliations, and key references.
  + Paper should be 250-1000 words
  + Note the paper should not include software documentation such as API (Application Programming Interface) functionality, as this should be outlined in the software documentation.
* (Optional) create a metadata file and include it in your repository describing your software.

# PAPER OUTLINE

TITLE

AUTHORS/AFFILIATIONS

1. Summary
   1. A summary describing the high-level functionality and purpose of the software for a diverse, *non-specialist audience*.
   2. A clear statement of need that illustrates the purpose of the software.
   3. Mentions (if applicable) of any ongoing research projects using the software or recent scholarly publications enabled by it.
2. Acknowledgements
   1. Acknowledgement of any financial support.
3. References
   1. A list of key references including a link to the software archive.

# CODE DOCUMENTATION

There should be sufficient documentation for the reviewer to understand the core functionality of the software under review. A high-level overview of this documentation should be included in a README file (or equivalent). There should be:

1. A statement of need: The authors should clearly state what problems the software is designed to solve and who the target audience is.
2. Installation instructions: There should be a clearly-stated list of dependencies. Ideally these should be handled with an automated package management solution.
   * Good: A package management file such as a Gemfile or package.json or equivalent
   * OK: A list of dependencies to install
   * Bad (not acceptable): Reliance on other software not listed by the authors
3. Example usage: The authors should include examples of how to use the software (ideally to solve real-world analysis problems).
4. API documentation: Reviewers should check that the software API is documented to a suitable level.
   * Good: All functions/methods are documented including example inputs and outputs
   * OK: Core API functionality is documented
   * Bad (not acceptable): API is undocumented
5. Community guidelines: There should be clear guidelines for third-parties wishing to
   * Contribute to the software
   * Report issues or problems with the software
   * Seek support

# OTHER

* Tests: Authors are strongly encouraged to include an automated test suite covering the core functionality of their software.
  + Good: An automated test suite hooked up to an external service such as Travis-CI or similar
  + OK: Documented manual steps that can be followed to objectively check the expected functionality of the software (e.g. a sample input file to assert behaviour)
  + Bad (not acceptable): No way for you the reviewer to objectively assess whether the software works

# EXAMPLE paper.md

---

title: 'Gala: A Python package for galactic dynamics'

tags:

- Python

- astronomy

- dynamics

- galactic dynamics

- milky way

authors:

- name: Adrian M. Price-Whelan

orcid: 0000-0003-0872-7098

affiliation: "1, 2" # (Multiple affiliations must be quoted)

- name: Author 2

orcid: 0000-0000-0000-0000

affiliation: 2

affiliations:

- name: Lyman Spitzer, Jr. Fellow, Princeton University

index: 1

- name: Institution 2

index: 2

date: 13 August 2017

bibliography: paper.bib

---

# Summary

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).

``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] (``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 [@Pearson:2017] and has also been

used in graduate courses on Galactic dynamics to, e.g., provide interactive

visualizations of textbook material [@Binney:2008]. 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

[@gaia] 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.$$

# Citations

Citations to entries in paper.bib should be in

[rMarkdown](http://rmarkdown.rstudio.com/authoring\_bibliographies\_and\_citations.html)

format.

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: ![Example figure.](figure.png)

# Acknowledgements

We acknowledge contributions from Brigitta Sipocz, Syrtis Major, and Semyeong

Oh, and support from Kathryn Johnston during the genesis of this project.

# References

# EXAMPLE paper.bib

@article{Pearson:2017,

Adsnote = {Provided by the SAO/NASA Astrophysics Data System},

Adsurl = {http://adsabs.harvard.edu/abs/2017arXiv170304627P},

Archiveprefix = {arXiv},

Author = {{Pearson}, S. and {Price-Whelan}, A.~M. and {Johnston}, K.~V.},

Eprint = {1703.04627},

Journal = {ArXiv e-prints},

Keywords = {Astrophysics - Astrophysics of Galaxies},

Month = mar,

Title = {{Gaps in Globular Cluster Streams: Pal 5 and the Galactic Bar}},

Year = 2017

}

@book{Binney:2008,

Adsnote = {Provided by the SAO/NASA Astrophysics Data System},

Adsurl = {http://adsabs.harvard.edu/abs/2008gady.book.....B},

Author = {{Binney}, J. and {Tremaine}, S.},

Booktitle = {Galactic Dynamics: Second Edition, by James Binney and Scott Tremaine.~ISBN 978-0-691-13026-2 (HB).~Published by Princeton University Press, Princeton, NJ USA, 2008.},

Publisher = {Princeton University Press},

Title = {{Galactic Dynamics: Second Edition}},

Year = 2008

}

@article{gaia,

author = {{Gaia Collaboration}},

title = "{The Gaia mission}",

journal = {\aap},

archivePrefix = "arXiv",

eprint = {1609.04153},

primaryClass = "astro-ph.IM",

keywords = {space vehicles: instruments, Galaxy: structure, astrometry, parallaxes, proper motions, telescopes},

year = 2016,

month = nov,

volume = 595,

doi = {10.1051/0004-6361/201629272},

adsurl = {http://adsabs.harvard.edu/abs/2016A%26A...595A...1G},

}

@article{astropy,

author = {{Astropy Collaboration}},

title = "{Astropy: A community Python package for astronomy}",

journal = {\aap},

archivePrefix = "arXiv",

eprint = {1307.6212},

primaryClass = "astro-ph.IM",

keywords = {methods: data analysis, methods: miscellaneous, virtual observatory tools},

year = 2013,

month = oct,

volume = 558,

doi = {10.1051/0004-6361/201322068},

adsurl = {http://adsabs.harvard.edu/abs/2013A%26A...558A..33A}

}