Draft version September 10, 2024
Typeset using LaTeX **twocolumn** style in AASTeX631

Streamlining and standardizing software citations with *The Software Citation Station*

Tom Wagg* [1] and Floor S. Broekgaarden* [2, 3, 4, 5]

[1] *Department of Astronomy, University of Washington, Seattle, WA, 98195*
[2] *Department of Astronomy & Astrophysics, University of California, San Diego, 9500 Gilman Drive, La Jolla, CA 92093, USA*
[3] *Simons Society of Fellows, Simons Foundation, New York, NY 10010, USA*
[4] *Department of Astronomy and Columbia Astrophysics Laboratory, Columbia University, 550 W 120th St, New York, NY 10027, USA*
[5] *William H. Miller III Department of Physics and Astronomy, Johns Hopkins University, Baltimore, Maryland 21218, USA*

ABSTRACT

Software is crucial for the advancement of astronomy especially in the context of rapidly growing datasets that increasingly require algorithm and pipeline development to process the data and produce results. However, software has not always been consistently cited, despite its importance to strengthen support for software development. To encourage, streamline, and standardize the process of citing software in academic work such as publications we introduce '*The Software Citation Station*': a publicly available website and tool to quickly find or add software citations.

Keywords: Astronomy software (1855) – Software documentation (1869) – Open source software (1866)

1. INTRODUCTION

Software development has become an essential part of every sub-field of astronomy, but is often not adequately supported in astronomy (National Academies of Sciences, Engineering, and Medicine 2021). Citations are an important tool in science and astronomy to build upon and properly attribute and credit earlier work, value authorship, motivate funding, enable peer-review, validate and reproduce findings, support collaboration and community, and encourage reuse and building on work of others (Katz et al. 2020). While there exists a well-developed ecosystem of tools and services to assist with citations to traditional publications, such as NASA/ADS and ArXiv, this infrastructure does not work nearly so well for software citation. As a result software is often not cited properly. A major issue is that software lacks a standardized citation practice, leaving it to the software author to point readers to the citation mechanism (Howison & Bullard 2016; Niemeyer et al. 2016; Li et al. 2017; Bouquin et al. 2020; Alsudais 2021; Bouquin et al. 2023).

Some advancements have been made in recent years to promote and standardize software citations in academic journals. This includes the creation of specific recommendations for software citation such as the reports by the Joint Declaration of Data Citation Principles, and the recommendations of the FORCE11 Software Citation Working Group (Martone et al. 2014; Smith et al. 2016; Katz et al. 2020). Several astronomy journals have developed software policies and include software sections or software acknowledgements in their publication templates (e.g., Vishniac & Lintott 2016; Timmes & Muench 2019). Examples include those by AAS publishing (AAS Journals 2024a), and the examples listed on CHORUS (CHORUS 2023). This has helped to increase software citations, as evidenced by IRAF citations increasing by a factor of ∼150% after AAS started actively recommending authors to include their citations in submitted papers (AAS Journals 2019).

Other advancements focus on the review and reproducibility of software accompanying research. An important example is the creation of The Journal of Open Source Software (JOSS[1]; Smith 2020), an open access journal for research software packages that enables peer-review of astronomy software without processing charges or subscription fees. Recently, AAS publishing started a public collaboration with JOSS where authors sub-

* tomjwagg@gmail.com, fsb2127@columbia.edu; both authors contributed equally to the work.

[1] https://joss.theoj.org

mitting to one of the AAS journals can also publish a companion software paper in JOSS (Vishniac & Lintott 2018a). Additionally, AAS publishing started to facilitate 'living articles' – articles that are formally reviewed but that can be updated to accommodate changing conditions, upgrades, or new functionality – to make it easier to publish and update papers describing (long-lived) software (Vishniac & Lintott 2018b). Similarly, *Show your Work!* (Luger et al. 2021) provides a tool to accompany papers with an automatized workflow that creates a self-contained recipe for readers to reproduce results based on accompanying code and software. Similarly, *papers with code* and *Figshare* are websites aimed to increase visibility of software accompanying (ArXiv) papers.

Another important focus has been on creating unique identifiers to ensure citations are standardised and accurately counted. The Astrophysics Source Code Library (ASCL) is a website and tool designed with this goal, specifically dedicated to curating and indexing software used in astronomy-based literature by providing a library of astronomy software with unique identifiers. In a similar vein, GitHub has created an integration with Zenodo to make it easier to reference and cite GitHub repositories in academic literature (GitHub 2024). Software records in places such as ASCL, Zenodo, and JOSS, which provide the software with an unique identifier, will typically also be included in ADS (with citations counted based on the unique identifier), and efforts are on the way to streamline this between software versions and platforms, e.g., through the AAS *Asclepias* project (Nielsen et al. 2018; van de Sandt et al. 2019; Henneken et al. 2022).

Finally, several python packages, including `makecite` (Price-Whelan et al. 2019), `citepy` (Barnes 2021), and `duecredit` (Halchenko et al. 2024), have been developed with the goal of streamlining the process of gathering citations used in research based on the packages imported in a user-provided python script. However, these packages have some limitations given that (a) they only support Python based software, (b) they can be difficult to apply to projects using more than one python script, (c) they require installing and running the code (d) and still rely on (incomplete) libraries with software references.

Despite these many advancements, many authors still do not include proper software citations. This is because it can be challenging to find the citation for many tools (or users might not be aware that their software has references), and different software users might end up using different key words or references, making it often difficult to identify and quantify contributions of software and give proper credits to all software contributors (see Katz & Chue Hong 2018; Katz et al. 2019; Katz et al. 2020; van de Sandt et al. 2019; Bouquin et al. 2020, 2023; Druskat et al. 2021, 2024, and references therein). As citing software is becoming increasingly important to support the careers and field in this area, it is important that astronomy increases the practice of software citation in academic work. To this end we have created '*The Software Citation Station*'[2], a publicly available website and tool to make it extremely easy to find and add software citations to academic work.

2. *The Software Citation Station*

The primary goal of *The Software Citation Station* is to streamline and standardise the processing of citing software in academic work. We streamline the process by designing the site's core functionality to be a simple three-step process (see Figure 1). Citations are standardised by adhering to the requirements outlined by each software package.

2.1. *Core functionality*

Users first select the software that they have used in their academic work in panel 1 (see Figure 1). One can search for particular keywords or package names, as well as filter by category or programming language to refine the list. Additionally, one can right-click each item to see further details about the package such as a short description, a link to the documentation and the associated Zenodo record.

As each package is selected, if it is indexed on Zenodo, the *The Software Citation Station* queries the Zenodo API for the latest list of available versions of the software and populates a dropdown with the options. Users can then select the version used in their research for each package.

As this selection occurs, an acknowledgement and associated BibTeX is generated in panel 3. Users can directly copy the acknowledgement into their LaTeX manuscript, either in the acknowledgements section or, for AAS journals, in the `Software` section. The BibTeX can either be downloaded directly as a BibTeX file or copied to the clipboard to insert into another file.

2.2. *Custom citation statements*

There is not a universally accepted software citation standard and as such in many cases software packages have differing requirements in how they are cited. Many code request that users write a specifically worded acknowledgement in their work in addition to certain ci-

[2] https://www.tomwagg.com/software-citation-station/

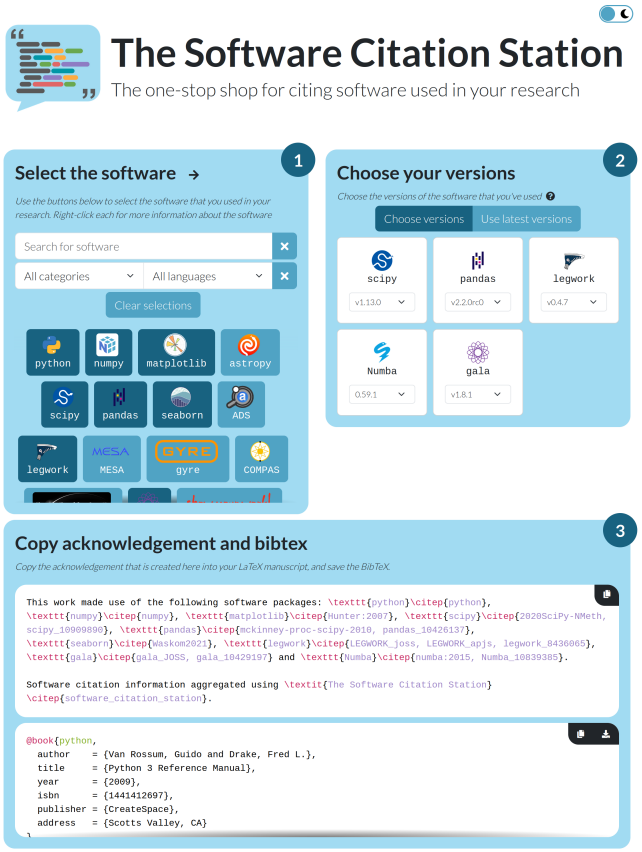

Figure 1. Screenshot demo of using *The Software Citation Station* to cite software. Panel 1 allows users to search and filter software packages in order to select those used in their research. Panels 2 lets the user select the specific version of the selected software packages (for those that are indexed on Zenodo) used in their research. Panel 3 is then populated with an acknowledgement for a LaTeX manuscript and the associated BibTeX respectively.

tations. We allow for flexibility in how a software package is cited and arbitrary customisation is possible when adding a new package to *The Software Citation Station* (see Section 2.4).

2.3. *Package dependencies*

It is very common for open-source projects to depend on other software packages. It is important to consistently cite dependencies of software (Sochat 2022). Therefore, each time a package is selected on *The Software Citation Station*, its dependencies (and dependencies of *those* dependencies) that are not already selected will be added to the list of packages to cite.

2.4. *Adding new software*

We aim for *The Software Citation Station* to be open to all software packages used in academic work. As such, the site includes a form for submitting new software packages, which guides users through the submission process. Figure 2 shows an example of the online form. Each package requires a data entry of the following form:

```
"package name": {
    "tags": [],
    "logo": "",
    "language": "",
    "category": "",
    "keywords": [],
    "description": "",
    "link": "",
    "attribution_link": "",
    "zenodo_doi": "",
    "custom_citation": "",
    "dependencies": []
},
```

where `tags` is a list of the BibTeX tags associated with this package, `logo` is a path to the logo image file, `language` is the programming language of the software, `category` is the category of the package (e.g. "population synthesis" or "visualisation"), `keywords` are a list of relevant keywords for searching, `description` is a short 1-2 sentence description of the package, `link` is a URL to the documentation (or code repository if no documentation is available) and `attribution_link` a link to a website that outlines the preferred citation format of the package. Optionally each package can also have a `zenodo_doi`, which is the DOI associated with all versions of the package on Zenodo,`custom_citation`, which is a custom citation string that overwrites the default generated citation (see Section 2.2 for more details) and `dependencies`, which is a list of package names that should also be cited if this package is cited (see Section 2.3).

The form is designed to streamline the submission of new packages and catch simple errors automatically. `tags` are extracted directly from submitted BibTeX and languages/categories can be selected from a dropdown menu populated based on currently known packages. Additionally, prior to submission the form is validated in several ways, such as requiring URLs to direct to existing websites. In particular, the form checks that the Zenodo DOI corresponds to an existing record with multiple versions and, if a user has submitted the DOI that only corresponds to a specific version, it will correct it.

3. FUTURE WORK

We created *The Software Citation Station* to make it incredibly easy for astronomers to add citations of commonly-used software to their academic publications.

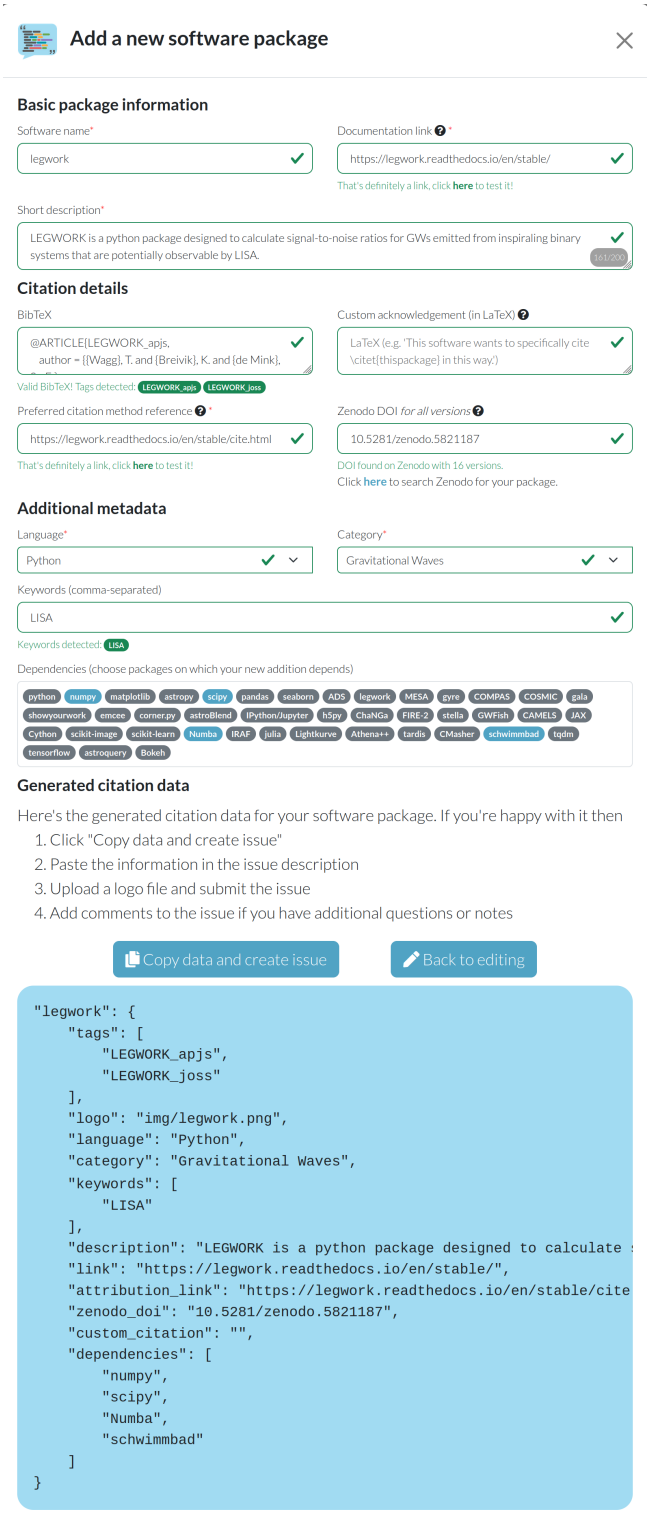

Figure 2. Screenshot of the online form to help users add new software to *The Software Citation Station*. `LEGWORK` (Wagg et al. 2022a,b) is used as an example. The data generated using this form can directly be uploaded as a GitHub issue.

This is in line with recommendations by Bouquin et al. (2023) who mentioned the need to "Create a website that can act as a central source of information on software citation in practice".[3] On the longer term it would be beneficial to integrate *The Software Citation Station* with other existing efforts to improve citations of software. This includes connecting or integrating our website as part of ArXiv, ADS, and/or ASCL, as well as connecting to citation software like `makecite` (Price-Whelan et al. 2019), `citepy` (Barnes 2021), and `duecredit` (Halchenko et al. 2024). Specifically, it would be useful to create a consistent library with citation entries that is community-updated to streamline consistency between different software citation tools and improve completeness of the libraries. Ideally such a library could be generated from data provided by the software developer. An important step to achieve this will be for software-developers in astronomy to use consistent citation practices, which could include adding a citation file format file (e.g. `CITATION.cff`, Druskat et al. 2021) to GitHub or other software repositories as well as a CodeMeta file (CodeMeta.json, Jones et al. 2017) and assigning software references a unique identifier (see Katz & Smith 2014; Cosmo et al. 2020, and references therein). Additionally, such software citations should also support version control (e.g. Chen & Davidson 2020). This work is out of scope of this paper, but would be a useful topic to discuss and pursue in the future (cf. Katz & Chue Hong 2018; Katz et al. 2019; Katz et al. 2020; van de Sandt et al. 2019; Cosmo et al. 2020; Allen 2021; Druskat et al. 2022; Bouquin et al. 2023) and examples of recent efforts in this direction are the HERMES (Druskat et al. 2022), CiteAs (Du et al. 2021), and Asclepias (Henneken et al. 2022) projects.

4. ADDITIONAL RESOURCES

Several resources for best software citation practices have been created. In particular, The FORCE11 Software Citation Implementation Working Group (SCIWG[4]), which has been dedicated to implement the software principles from the FORCE11 Software Citation Working Group provides an excellent collection of resources.

First, several groups have written software citation checklists for authors (Smith et al. 2016; Chue Hong et al. 2019a; Katz et al. 2020), developers (Chue Hong

[3] See also CiteSoftware.org, CiteAs.org, and cite.research-software.org which are online guides and tools on how to cite software, but which do not yet create the BibTeX for a specific software package for you as *The Software Citation Station* does.

[4] https://github.com/force11/force11-sciwg

et al. 2019b; Albert et al. 2019; Best Practices for Software Registries et al. 2020)[5], and journals (Katz et al. 2020; Stall et al. 2023). A summary is also provided by Katz et al. (2019) in section 3.5 (how to cite software in text) and section 3.2 and 3.3 (metadata for software citation). There is also the guide by Puebla et al. (2024) on recognizing data and software contributions in hiring, promotion, and tenure in academia, and the work by Wofford et al. (2020); Erdmann et al. (2021); AAS Journals (2024b) on recommendations for making jupyter notebooks reproducible and citeable. We also direct the reader to the CiteSoftware.org webpage with resources related to software citation, credit, and preservation.

Several organizations are leading software and coding focused academic schools, many of which also include publicly available recordings and lecture notes. Examples include Code/Astro, the LSSTC data science fellowship program, and the La Serena School for data science. In addition, the Astronomical Data Analysis Software and Systems (ADASS) organizes annual software-focused conferences and workshops in astronomy.

Other software-focused developments in astronomy include several organizations initiating the awarding of software development prizes to recognize and highlight software contributions in the field, such as the ADASS Prize for an Outstanding Contribution to Astronomical Software and the NASA software of the year award. Additionally, there are efforts to promote more stable career opportunities for software focused astronomers including the Software Research Faculty Award from the Simons Foundation. These efforts are important to address the issues beyond citing software, such as supporting and funding the careers of software developers and software engineers in astronomy (see the discussions in Anzt et al. 2020; Bouquin et al. 2023; Carlin et al. 2023; Sochat 2024). Moreover the development of tools such as CiteLang (Sochat 2022) move beyond simple software citations to contributions of the dependencies for software. We also point readers to Smith (2022) for an example of a career path as a software engineer in astronomy and a discussion of the challenges.

TW acknowledges support from NASA ATP grant 80NSSC24K0768. FSB is supported by Simons Foundation award 1141468 hosted at Columbia University. This research has made use of NASA's Astrophysics Data System.

[5] See also the guidelines by GitHub, ASCL, astrobetter, and cite-software.

REFERENCES

AAS Journals. 2019, Software citation suggestions, https://joss.theoj.org/

—. 2024a, policy statement on software, https://journals.aas.org/policy-statement-on-software/

—. 2024b, policy statement on notebooks, https://journals.aas.org/policy-statement-on-notebooks/

Albert, K., Bouquin, D., Farber, A., & Hoover, R. 2019, Copyright Guide for Scientific Software, Zenodo, doi: 10.5281/zenodo.3581326

Allen, A. 2021, arXiv e-prints, arXiv:2111.14278, doi: 10.48550/arXiv.2111.14278

Alsudais, A. 2021, Journal of Informetrics, 15, 101139, doi: https://doi.org/10.1016/j.joi.2021.101139

Anzt, H., Bach, F., Druskat, S., et al. 2020, arXiv e-prints, arXiv:2005.01469, doi: 10.48550/arXiv.2005.01469

Barnes, C. 2021, citepy, https://github.com/clbarnes/citepy

Best Practices for Software Registries, T. F. o., :, Monteil, A., et al. 2020, arXiv e-prints, arXiv:2012.13117, doi: 10.48550/arXiv.2012.13117

Bouquin, D., Trisovic, A., Bertuch, O., & Colón-Marrero, E. 2023, arXiv e-prints, arXiv:2302.07500, doi: 10.48550/arXiv.2302.07500

Bouquin, D. R., Chivvis, D. A., Henneken, E., et al. 2020, ApJS, 249, 8, doi: 10.3847/1538-4365/ab7be6

Carlin, D., Rainer, A., & Wilson, D. 2023, PeerJ Computer Science, 9, e1546

Chen, L., & Davidson, S. B. 2020, in 2020 IEEE 36th International Conference on Data Engineering (ICDE), 1754–1757, doi: 10.1109/ICDE48307.2020.00162

CHORUS. 2023, SOFTWARE CITATION POLICIES INDEX, https://www.chorusaccess.org/resources/software-citation-policies-index/

Chue Hong, N. P., Allen, A., Gonzalez-Beltran, A., et al. 2019a, Software Citation Checklist for Authors, 0.9.0, Zenodo, doi: 10.5281/zenodo.3479199

Chue Hong, N. P., Allen, A., Gonzalez-Beltran, et al. 2019b, Software Citation Checklist for Developers, 0.9.0, Zenodo, doi: 10.5281/zenodo.3482769

Cosmo, R. D., Gruenpeter, M., & Zacchiroli, S. 2020, Computing in Science & Engineering, 22, 33, doi: 10.1109/MCSE.2019.2963148

Druskat, S., Bertuch, O., Juckeland, G., Knodel, O., & Schlauch, T. 2022, arXiv e-prints, arXiv:2201.09015, doi: 10.48550/arXiv.2201.09015

Druskat, S., Chue Hong, N. P., Buzzard, S., Konovalov, O., & Kornek, P. 2024, arXiv e-prints, arXiv:2402.14602, doi: 10.48550/arXiv.2402.14602

Druskat, S., Katz, D. S., & Todorov, I. T. 2021, arXiv e-prints, arXiv:2103.06681, doi: 10.48550/arXiv.2103.06681

Druskat, S., Spaaks, J. H., Chue Hong, N., et al. 2021, Citation File Format, 1.2.0, Zenodo, doi: 10.5281/zenodo.5171937

Du, C., Cohoon, J., Priem, J., et al. 2021, in Companion Publication of the 2021 Conference on Computer Supported Cooperative Work and Social Computing, CSCW '21 Companion (New York, NY, USA: Association for Computing Machinery), 218–221, doi: 10.1145/3462204.3482889

Erdmann, C., Stall, S., Gentemann, C., et al. 2021, Guidance for AGU Authors - Jupyter Notebooks, Zenodo, doi: 10.5281/zenodo.5651648

GitHub. 2024, referencing and citing content, https://docs.github.com/en/repositories/archiving-a-github-repository/referencing-and-citing-content

Halchenko, Y. O., Visconti di Oleggio Castello, M., Hanke, M., et al. 2024, duecredit/duecredit: 0.10.1, 0.10.1, Zenodo, doi: 10.5281/zenodo.11209252

Henneken, E. A., Blanco-Cuaresma, S., Accomazzi, A., et al. 2022, Bulletin of the AAS, 54

Howison, J., & Bullard, J. 2016, Journal of the Association for Information Science and Technology, 67, 2137

Jones, M. B., Boettiger, C., Mayes, A. C., et al. 2017, Github

Katz, D. S., & Chue Hong, N. P. 2018, arXiv e-prints, arXiv:1807.08149, doi: 10.48550/arXiv.1807.08149

Katz, D. S., & Smith, A. M. 2014, arXiv e-prints, arXiv:1407.5117, doi: 10.48550/arXiv.1407.5117

Katz, D. S., Bouquin, D., Chue Hong, N. P., et al. 2019, arXiv e-prints, arXiv:1905.08674, doi: 10.48550/arXiv.1905.08674

Katz, D. S., Hong, N. P. C., Clark, T., et al. 2020, F1000Research, 9

Li, K., Yan, E., & Feng, Y. 2017, Journal of Informetrics, 11, 989, doi: https://doi.org/10.1016/j.joi.2017.08.003

Luger, R., Bedell, M., Foreman-Mackey, D., et al. 2021, arXiv e-prints, arXiv:2110.06271. https://arxiv.org/abs/2110.06271

Martone, M., et al. 2014, San Diego CA: Force11, 10

National Academies of Sciences, Engineering, and Medicine. 2021, Pathways to Discovery in Astronomy and Astrophysics for the 2020s (Washington, DC: The National Academies Press), doi: 10.17226/26141

Nielsen, L. H., Muench, A., Accomazzi, A., et al. 2018, Asclepias: Enabling Software Citation and Discovery, Zenodo, doi: 10.5281/zenodo.1283381

Niemeyer, K. E., Smith, A. M., & Katz, D. S. 2016, arXiv e-prints, arXiv:1601.04734, doi: 10.48550/arXiv.1601.04734

Price-Whelan, A., Mechev, A., Sipocz, B., et al. 2019, adrn/makecite v0.5, v0.5, Zenodo, doi: 10.5281/zenodo.3533303

Puebla, I., Ascoli, G., Blume, J., et al. 2024, Ten simple rules for recognizing data and software contributions in hiring, promotion and tenure, Tech. rep., Center for Open Science

Smith, A. M. 2020, Journal of Open Source Software, Zenodo, doi: 10.5281/zenodo.4316639

—. 2022, "I Am Pretty Interested in Coding, Technology, and Infrastructure" (Cham: Springer International Publishing), 91–99, doi: 10.1007/978-3-030-82606-2_10

Smith, A. M., Katz, D. S., & Niemeyer, K. E. 2016, PeerJ Computer Science, 2, e86

Sochat, V. 2022, Journal of Open Source Software, 7, 4458

—. 2024, arXiv preprint arXiv:2405.10473

Stall, S., Bilder, G., Cannon, M., et al. 2023, Scientific Data, 10, 656

Timmes, F. X., & Muench, A. 2019, in Astronomical Society of the Pacific Conference Series, Vol. 523, Astronomical Data Analysis Software and Systems XXVII, ed. P. J. Teuben, M. W. Pound, B. A. Thomas, & E. M. Warner, 693

van de Sandt, S., Holm Nielsen, L., Ioannidis, A., et al. 2019, arXiv e-prints, arXiv:1911.00295, doi: 10.48550/arXiv.1911.00295

Vishniac, E. T., & Lintott, C. 2016, AJ, 151, 21, doi: 10.3847/0004-6256/151/2/21

Vishniac, E. T., & Lintott, C. 2018a, The Astrophysical Journal, 869, 156. https://iopscience.iop.org/article/10.3847/1538-4357/aaf876

—. 2018b, The Astrophysical Journal, 868, 78

Wagg, T., Breivik, K., & de Mink, S. 2022a, The Journal of Open Source Software, 7, 3998, doi: 10.21105/joss.03998

Wagg, T., Breivik, K., & de Mink, S. E. 2022b, ApJS, 260, 52, doi: 10.3847/1538-4365/ac5c52

Wofford, M. F., Boscoe, B. M., Borgman, C. L., Pasquetto, I. V., & Golshan, M. S. 2020, Computing in Science & Engineering, 22, 5, doi: 10.1109/MCSE.2019.2932067