

Dynamicall: Tools for Modeling Resolved Galaxies Using Jax and Numpyro

Juan Guerra¹ and Marla Geha¹

¹ Yale University, New Haven, CT, USA

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review](#)
- [Repository](#)
- [Archive](#)

Editor: [Josh Borrow](#)

Reviewers:

- [@sophialilleengen](#)
- [@michael-petersen](#)

Submitted: 19 October 2024

Published: unpublished

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

Summary

Studying the underlying dark matter distribution of galaxies relies on single snapshots of kinematic tracers. In the regime where individual stars can be resolved, various methods have been developed to infer the underlying mass distribution of these galaxies (Binney & Tremaine, 2008; ?; ?; ?; ?).

We present a new tool, Dynamicall, a pure python package for modeling the dynamics of galaxies using Jax and Numpyro. Dynamicall provides a suite of functions for modeling the dynamics of galaxies including functions for all basic dynamical quantities like density, mass, potential, distribution functions, actions, J-Factors and D-Factors. A variety of common models used in astronomy are implemented for ease of use such as Plummer, NFW, and Hernquist-Zhao profiles. We provide a flexible and extensible framework for users to implement their own models with a minimal amount of code. Leveraging Jax and Numpyro allows for the just-in-time (JIT) compilation, GPU acceleration and automatic differentiation, greatly speeding up the process of fitting models to data.

We showcase the use of Dynamicall by implementing spherical Jeans modeling and applying it to a mock data set of a dwarf galaxy. We additionally show 1) how to use the Fisher information matrix to forecast the uncertainties on model parameters, 2) how to use built in functions to generate mock data sets to test the code, and 3) how to build additional models using the base classes provided.

Statement of need

The use of GPU accelerated codes, automatic differentiation and Hamiltonian Monte Carlo has become increasingly widely adopted in the field of astronomy. As datasets become larger and more complex, the need for efficient and scalable tools to model the dynamics of galaxies has become more pressing. Although there are many tools available to model the dynamics of galaxies, many of them are not built with the latest advances in computational tools in mind and require significant rewrites to take advantage of these tools.

Additionally the ability to calculate the gradients of likelihood functions leads to the ability to forecast uncertainties on model parameters via Fisher information matrix calculations (?; ?; ?). This has traditionally been done with finite difference methods which can require a lot of computational time, but can also be very sensitive to the choice of step size. The use of automatic differentiation allows for the calculation of the gradients of the likelihood function with respect to the model parameters to be done with machine precision and in a fraction of the time, as it would not require various call to the likelihood function with different step sizes.

We provide spherical mass modeling as a built-in example of the capabilities of Dynamicall but the framework is also flexible enough to be extended to other models.

software citations

Dynamicall uses the following software packages:

- Jax (Bradbury et al., 2018)
- Numpyro (Bingham et al., 2019; Phan et al., 2019)
- matplotlib (Hunter, 2007)
- emcee (D. Foreman-Mackey et al., 2013)
- corner (Daniel Foreman-Mackey, 2016)
- arviz (Kumar et al., 2019)
- astropy (Astropy Collaboration et al., 2013, 2018, 2022)
- pytest (Krekel et al., 2004)
- agama (Vasiliev, 2019)

Acknowledgements

References

- Astropy Collaboration, Price-Whelan, A. M., Lim, P. L., Earl, N., Starkman, N., Bradley, L., Shupe, D. L., Patil, A. A., Corrales, L., Brasseur, C. E., N'otche, M., Donath, A., Tollerud, E., Morris, B. M., Ginsburg, A., Vaher, E., Weaver, B. A., Tocknell, J., Jamieson, W., ... Astropy Project Contributors. (2022). The Astropy Project: Sustaining and Growing a Community-oriented Open-source Project and the Latest Major Release (v5.0) of the Core Package. *935*(2), 167. <https://doi.org/10.3847/1538-4357/ac7c74>
- Astropy Collaboration, Price-Whelan, A. M., Sipőcz, B. M., Günther, H. M., Lim, P. L., Crawford, S. M., Conseil, S., Shupe, D. L., Craig, M. W., Dencheva, N., Ginsburg, A., VanderPlas, J. T., Bradley, L. D., Pérez-Suárez, D., de Val-Borro, M., Aldcroft, T. L., Cruz, K. L., Robitaille, T. P., Tollerud, E. J., ... Astropy Contributors. (2018). The Astropy Project: Building an Open-science Project and Status of the v2.0 Core Package. *156*(3), 123. <https://doi.org/10.3847/1538-3881/aabc4f>
- Astropy Collaboration, Robitaille, T. P., Tollerud, E. J., Greenfield, P., Droettboom, M., Bray, E., Aldcroft, T., Davis, M., Ginsburg, A., Price-Whelan, A. M., Kerzendorf, W. E., Conley, A., Crighton, N., Barbary, K., Muna, D., Ferguson, H., Grollier, F., Parikh, M. M., Nair, P. H., ... Streicher, O. (2013). Astropy: A community Python package for astronomy. *558*, A33. <https://doi.org/10.1051/0004-6361/201322068>
- Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P. A., Horsfall, P., & Goodman, N. D. (2019). Pyro: Deep universal probabilistic programming. *J. Mach. Learn. Res.*, *20*, 28:1–28:6. <http://jmlr.org/papers/v20/18-403.html>
- Binney, J., & Tremaine, S. (2008). *Galactic Dynamics: Second Edition*. Princeton University Press.
- Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., & Zhang, Q. (2018). *JAX: Composable transformations of Python+NumPy programs* (Version 0.3.13). <http://github.com/google/jax>
- Foreman-Mackey, Daniel. (2016). Corner.py: Scatterplot matrices in python. *The Journal of Open Source Software*, *1*(2), 24. <https://doi.org/10.21105/joss.00024>
- Foreman-Mackey, D., Hogg, D. W., Lang, D., & Goodman, J. (2013). Emcee: The MCMC hammer. *PASP*, *125*, 306–312. <https://doi.org/10.1086/670067>
- Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science &*

- 85 *Engineering*, 9(3), 90–95. <https://doi.org/10.1109/MCSE.2007.55>
- 86 Krekel, H., Oliveira, B., Pfannschmidt, R., Bruynooghe, F., Laughner, B., & Bruhin, F. (2004).
87 *Pytest* x.y. <https://github.com/pytest-dev/pytest>
- 88 Kumar, R., Carroll, C., Hartikainen, A., & Martin, O. (2019). ArviZ a unified library for
89 exploratory analysis of bayesian models in python. *Journal of Open Source Software*, 4(33),
90 1143. <https://doi.org/10.21105/joss.01143>
- 91 Phan, D., Pradhan, N., & Jankowiak, M. (2019). Composable effects for flexible and accelerated
92 probabilistic programming in NumPyro. *arXiv Preprint arXiv:1912.11554*.
- 93 Vasiliev, E. (2019). AGAMA: action-based galaxy modelling architecture. 482(2), 1525–1544.
94 <https://doi.org/10.1093/mnras/sty2672>

DRAFT