emcee v3: A Python ensemble sampling toolkit for affine-invariant MCMC


1 Center for Computational Astrophysics, Flatiron Institute 2 Department of Physics and Astronomy, Stony Brook University, United States 3 Centre for Astrophysics & Supercomputing, Swinburne University of Technology, Australia 4 ARC Centre of Excellence for All Sky Astrophysics in 3 Dimensions (ASTRO 3D) 5 University of Newcastle 6 Center for Cosmology and Particle Physics, Department of Physics, New York University 7 Max Planck Institute for Extraterrestrial Physics 8 Institute for Astronomy, University of Edinburgh, Edinburgh, EH9 3HJ, UK 9 Center for Astrophysics | Harvard & Smithsonian 10 American Nuclear Science and Technology Organisation, NSW, Australia 12 Planetary Science Institute, 1700 East Fort Lowell Rd., Suite 106, Tucson, AZ 85719, USA 13 Climate and Environmental Physics and Oeschger Center for Climate Change Research, University of Bern, Bern, Switzerland 15 Universitat Pompeu Fabra, Barcelona

Summary

emcee is a Python library implementing a class of affine-invariant ensemble samplers for Markov chain Monte Carlo (MCMC). This package has been widely applied to probabilistic modeling problems in astrophysics where it was originally published (Foreman-Mackey et al., 2013), with some applications in other fields. When it was first released in 2012, the interface implemented in emcee was fundamentally different from the MCMC libraries that were popular at the time, such as PyMC, because it was specifically designed to work with “black box” models instead of structured graphical models. This has been a popular interface for applications in astrophysics because it is often non-trivial to implement realistic physics within the modeling frameworks required by other libraries. Since emcee’s release, other libraries have been developed with similar interfaces, such as dynesty (Speagle, 2019). The version 3.0 release of emcee is the first major release of the library in about 6 years and it includes a full re-write of the computational backend, several commonly requested features, and a set of new “move” implementations.

This new release includes both small quality of life improvements—like a progress bar using tqdm—and larger features. For example, the new backends interface implements real time serialization of sampling results. By default emcee saves its results in memory (as in the original implementation), but it now also includes a HDFBackend class that serializes the chain to disk using h5py.

The most important new feature included in the version 3.0 release of emcee is the new moves interface. Originally, emcee implemented the affine-invariant “stretch move” proposed by Goodman & Weare (2010), but there are other ensemble proposals that can get better performance for certain applications. emcee now includes implementations of several other ensemble moves and an interface for defining custom proposals. The implemented moves include:

- The “stretch move” proposed by Goodman & Weare (2010),
emcee has been widely used and the original paper has been highly cited, but there have been many contributions from members of the community. This paper is meant to highlight these contributions and provide citation credit to the academic contributors. A full up-to-date list of contributors can always be found on GitHub.

References


