Distribution fit simulations

Pranav Minasandra
Max Planck Institute of Animal Behavior

1 Goal

Here, I intend to simulate many power-law and exponential distributed behavioural sequences, and test how a fictional classifier would affect the perceived bout duration disributions. This is needed because one common criticism of this project is bound to be "but what if the classifier causes these power-laws?"

Intuitively, classifiers or other noise prone methods of behaviour identification must destroy power-laws. Bout lengths are not subject to the law of large numbers: observing longer behavioural sequences doesn't make your predicted bout durations more accurate. An error splits a large bout into two smaller ones. The characteristic heavy-tails of power-law-type distributions must thus vanish. At the extreme, the most stupid classifiers which make their predictions by random guesswork must produce no processes with memory at all, and I predict this stupidity-induced memorilessness will cause exponential best-fits.

2 Simulations

A 2-behaviour discrete time simulation is described in the module `simulations`. `simulator.py` provides class `Simulator()`, which automates simulations for our sake. `simulations.__init__.py` provides a wrapper function which searches parameter space in parallel, finds the best fits for predicted bout lengths, and makes appropriate plots.