ALPhA Week 13 Presentation

PHY 496

BRADEN KRONHEIM

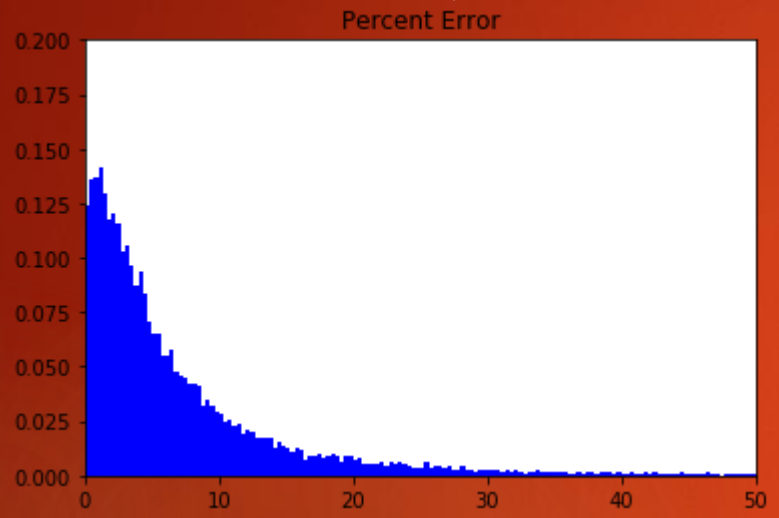
APRIL 19, 2019

Summary

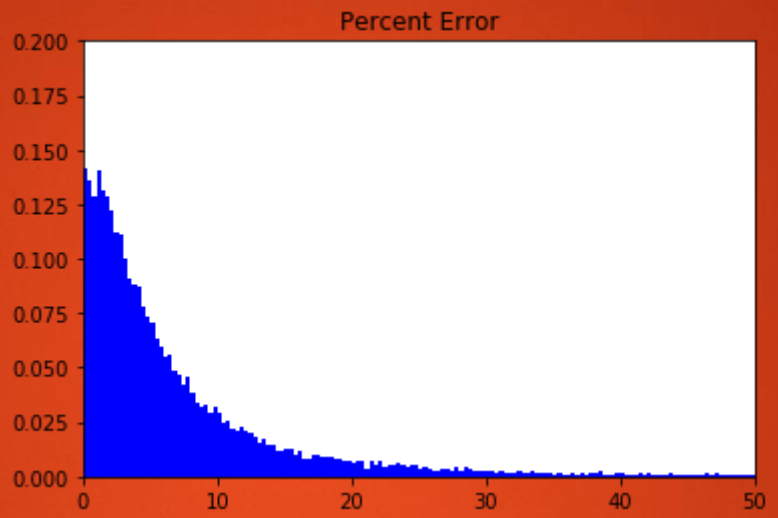
- Finished Poster
- Ran code to generate 50,000 networks, saving every 100
- Ran code to generate 1000 networks starting with the last network from the 50,000, step size of 1000
- Looked at possible extensions to HMC

General % Error

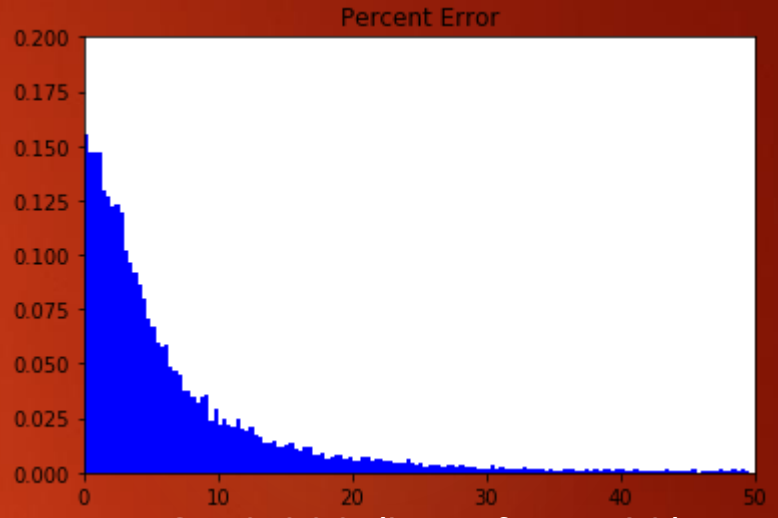
HMC 5,000



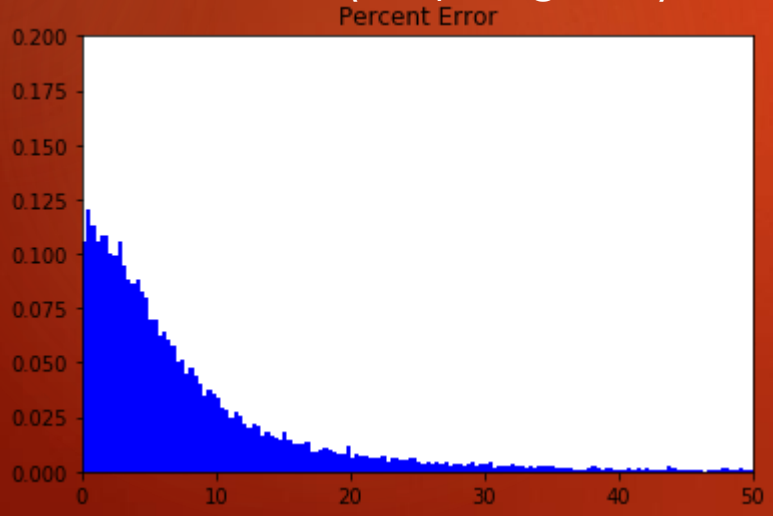
HMC 10,000



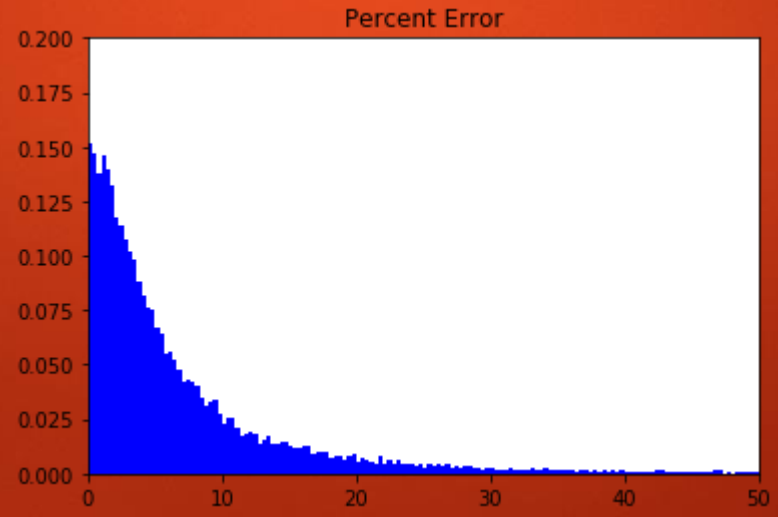
HMC 900 (leapfrog = 1000)



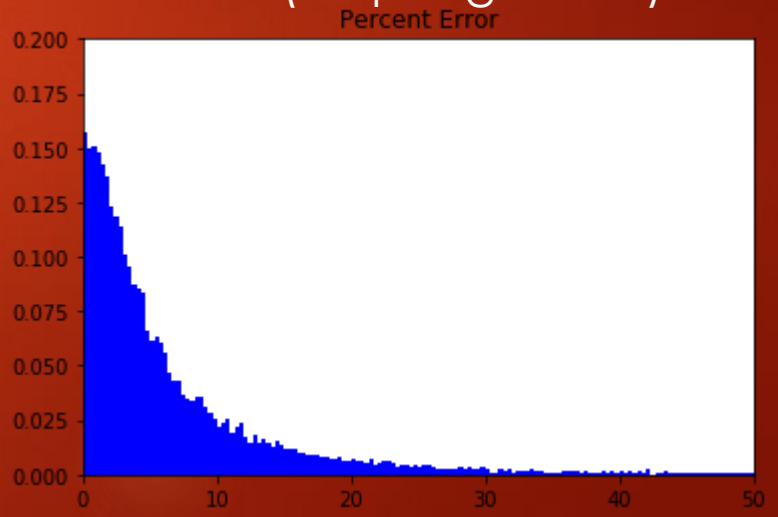
HMC (leapfrog =50)



HMC 50,000



HMC 50,000 (leapfrog=20) and HMC 900 (leapfrog= 1000)



Summary

Network	Inside 1 SD	Inside 2 SDs	Inside 3 SD3	Percent Error
HMC 5,000	55.95	81.65	91.29	9.01
HMC 1,000 (leapfrog = 50)	55.90	81.95	91.09	8.99
HMC 10,000	62.01	85.00	93.19	8.05
HMC 50,000	64.21	86.33	93.78	7.85
HMC 900 (leapfrog = 1000)	67.27	87.95	94.76	7.74
Combined 50,000 and 900	67.83	88.21	94.90	7.68

Possible HMC extensions

- Parallel Tempering:
  - Built into TensorFlow probability
  - Parallel iterations running at different temperatures
    - Higher temperature makes it easier for states to be accepted
  - States are exchanged between iterations
    - Allows low temperature states to jumpy between modes
- Riemann Manifold
  - Assigns masses other than 1 to the particles in the Hamiltonian
- Adaptive sampler
  - Uses Bayesian inference to adjust leapfrog steps and step size
  - Beats No-U-Turn Sampler
  - Can be applied to Riemann Manifold method

Goals for next week

- Try and implement the parallel tempering algorithm
- Further investigate impact of number of leapfrog steps
- Read up more on the other two possible extensions