Neural Networks Case Study: Face Potential

Prisca Calkins[1] (pcalki1@students.towson.edu), Maria Cameron[2] (mariakc@umd.edu) , Margot Yuan[2] (jyuan98@umd.edu)

[1]Department of Mathematics, Towson University [2]Department of Mathematics, University of Maryland

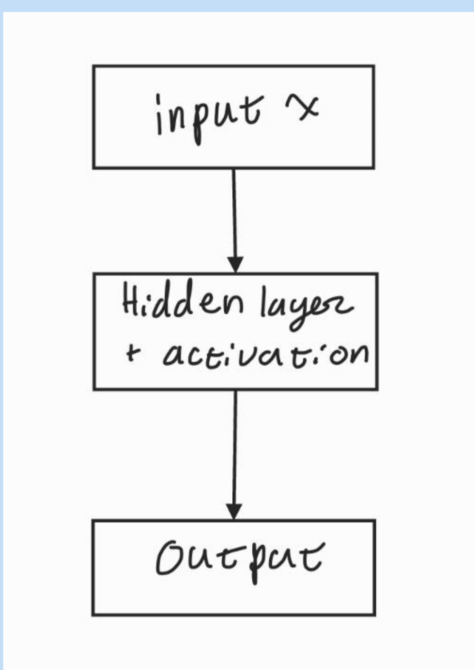
Background

Committor:
Probability that, given a set of data with two states A and B and starting at some point x, you will reach B before reaching A:
$q(x) = \text{Prob}\{\tau_B(x) < \tau_A(x)\}$

Equation:
$dx = -\nabla V dt + \sqrt{2\beta^{-1}} dw$

Gradient Value Problem:
$-\nabla V \cdot \nabla q + \beta^{-1}\Delta q = 0$, $q(\partial A) = 0$, $q(\partial B) = 1$
with boundary conditions q at boundary A = 0 and q at boundary B = 1.

Neural Networks:



Fully Connected Neural Network with one hidden layer Architecture:

$f(x;\theta) = \sigma_1(W_1\sigma_0(W_0x + b_0) + b_1)$, where σ_1 is the sigmoid and σ_0 is the hyperbolic tangent:
$\sigma_1 = \frac{1}{1+\exp(-x)}$, $\sigma_0 = \tanh(x)$.

W_1 is full matrix with unknown entries that are discovered through optimization

Goal

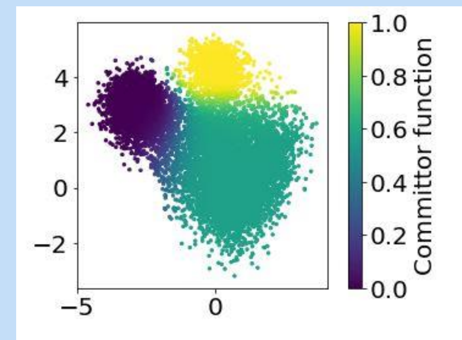
Explore neural network(NN)-based committor solvers and increase accuracy in a predictable manner.

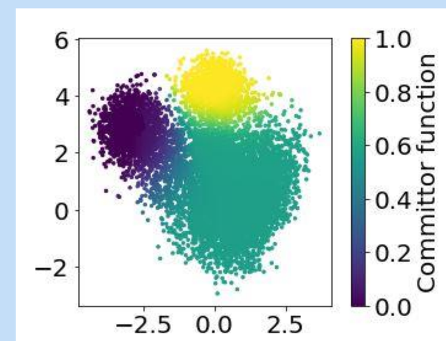
Methods

Replicating Li, Lin, Ren (2019)[1] Neural Network
Committor Model:
$q(x,\theta) = (1 - X_A(x))[(1 - X_B(x))q_{NN}(x) + X_B(x)]$
where
$q_{NN}(x,\theta) = \sigma_1(W_1(\sigma_0(W_0x + b_0)) + b_1)$ and X_A and X_B are smooth functions such that $X_A(x)|\partial A = 1$, $X_A(x)|\partial B = 0$, and $X_B(x)|\partial B = 1$.

Experimentation:
- Neurons
- Hidden layers
- Scheduler
- Training set

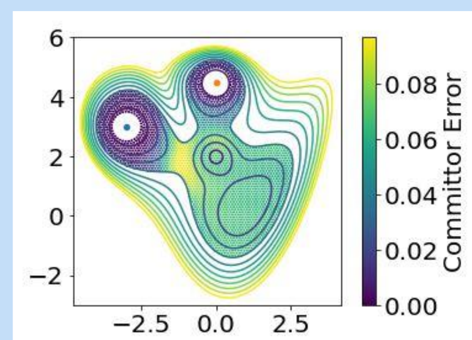
Artificial Temperature





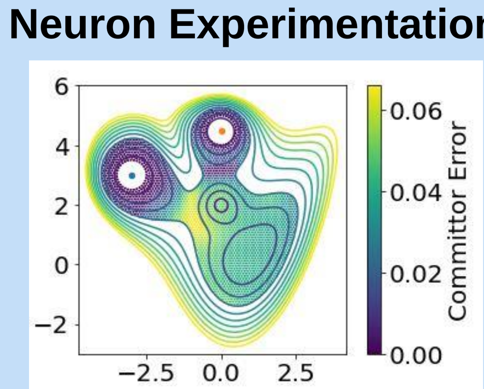
- Artificial temperature training set with 10000 training points.
- Artificial temperature involves artificially raising the temperature of the function
- Loss function: $f(\theta) = \frac{1}{N}\sum_{j=1}^{N} \|\nabla_x q(x_j,\theta)\|^2 e^{-(\beta-\beta')V(x_j)}$ where the data points were sampled at β'.
- $\beta' = 0.5$, and $\beta = 1$.
- Solution model is based on the one outlined in Li, Lin, Ren (2019) [1].
- Loss function is in Dirichlet form.
- Original committor and best committor shown here

Original Neural Network



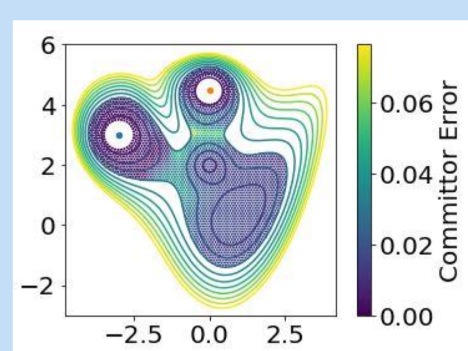
MAE	Neurons	Layers	Learning Rate	Epochs
0.046	10	One hidden layer	5e-3	1000

Neuron Experimentation



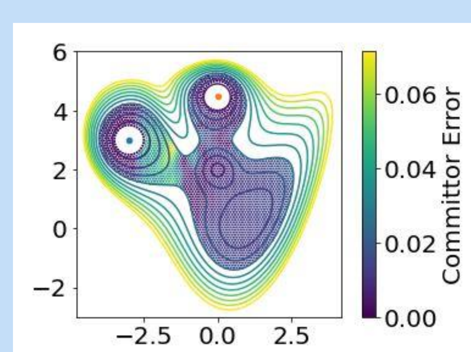
MAE	Neurons	Layers	Learning Rate	Epochs
0.029	20	One hidden layer	5e-3	1000

Hidden Layer Experimentation



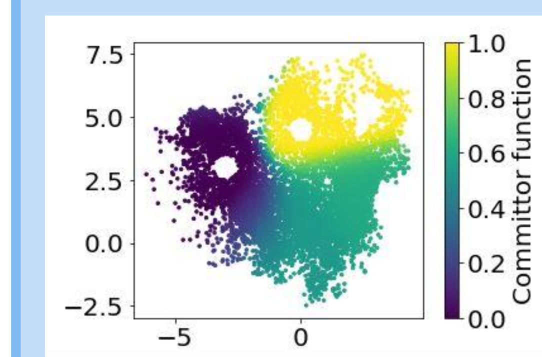
MAE	Neurons	Layers	Learning Rate	Epochs
0.015	10 on first layer, 20 on second layer	Two hidden layers	5e-3	1000

Scheduler Experimentation



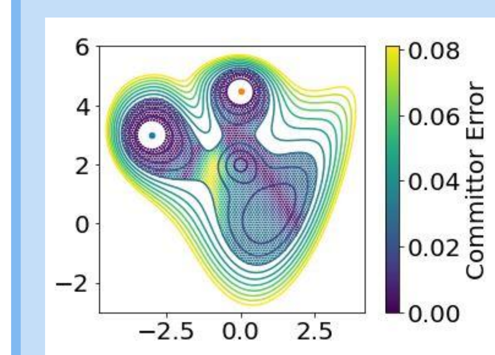
MAE	Neurons	Layers	Learning Rate	Epochs
0.014 (Best Error)	10 on first layer, 20 on second layer	Two hidden layer	Started at 5e-3, reduced by 0.2 every 500 epochs	2000

Metadynamics



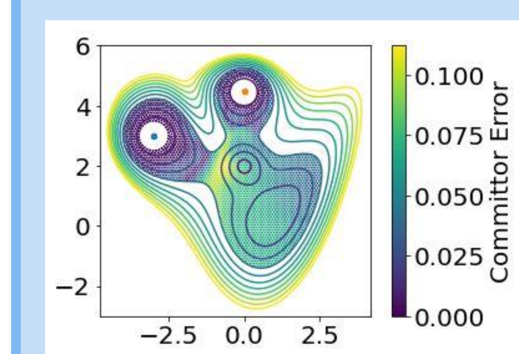
- Metadynamics training set with 7000 training points.
- Metadynamics sampling uses Gaussian functions to fill the areas around local minima of potential energy.
- Loss function: $f(\theta) = \frac{1}{N}\sum_{j=1}^{N} \|\nabla_x q(x_j,\theta)\|^2 e^{(\beta V_{bias}(x_j))}$ where the data points were sampled under the biased potential $V+V_{bias}$.
- Beta, solution model, and loss function same as artificial temperature
- Original committor shown here

Original:



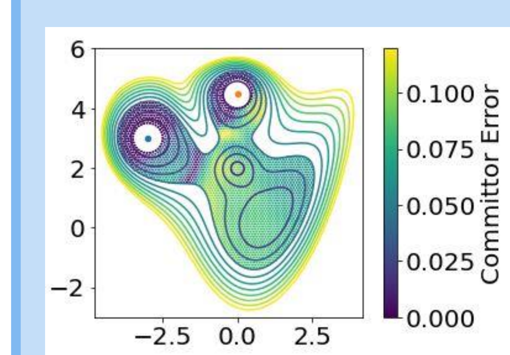
MAE	Neurons	Layers	Learning Rate	Epochs
0.019	10	One hidden layer	5e-3	1000

Hidden Layers



MAE	Neurons	Layers	Learning Rate	Epochs
0.042	10 on first layer, 20 on second layer	Two hidden layers	5e-3	1000

Scheduler



MAE	Neurons	Layers	Learning Rate	Epochs
0.066	10 on first layer, 20 on second layer	Two hidden layers	Started at 5e-3, reduced by 0.2 every 500 epochs	2000

Sources of Error

The unexpected errors produced by the metadynamics training set could be a result of inaccurate code for the training set. It also could mean that metadynamics is less reliable than temperature acceleration, although this contradicts the findings by Li, Lin, Ren (2016) [1]. Further research should be conducted.

Further Work
- Further experimentation with the metadynamics training set
- Experiment with different kinds of neural networks, such as residual neural networks
- Further explore sources of error and ways to reduce them

Literature cited
[1] Li, Lin, Ren. (2019). Computing committor functions for the study of rare events using deep learning. J. Chem. Phys. 151, 054112 (2019); https://doi.org/10.1063/1.5110439

Acknowledgements: Special thanks to the NSF for funding the 2022 UMD Math REU