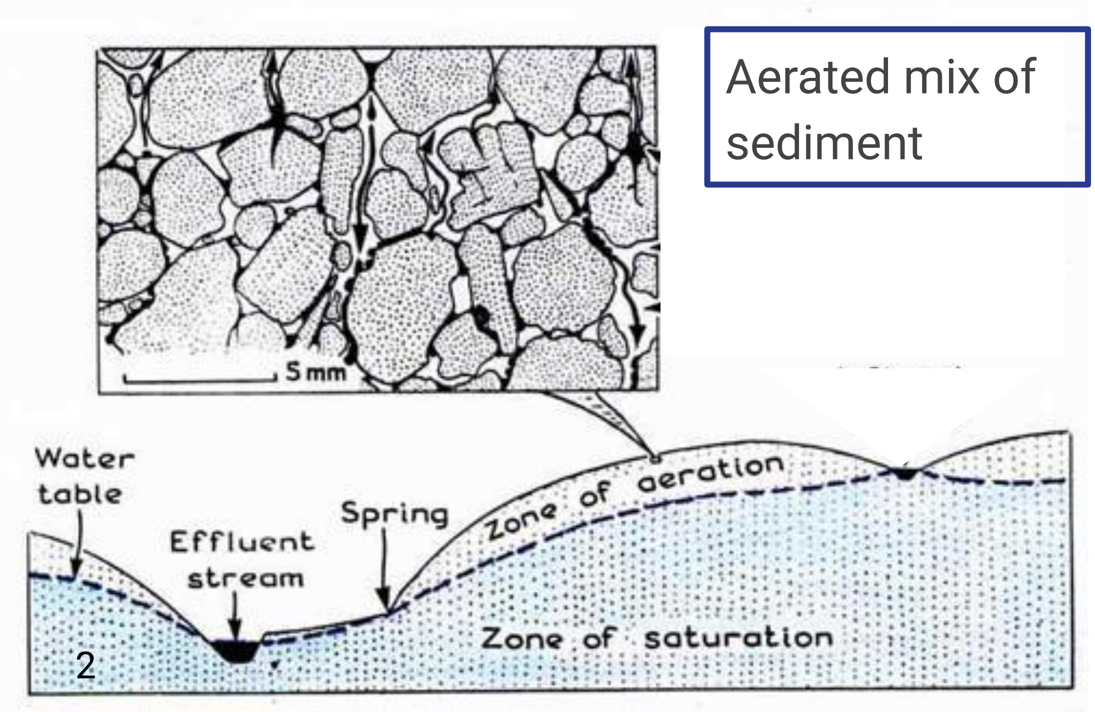
Deep Learning and Deep Saturation

What Neural Networks Can Learn about Percolation

A Final Project by Shannon Gallagher

Pacific University Oregon

Background



Aerated mix of sediment

Applications:

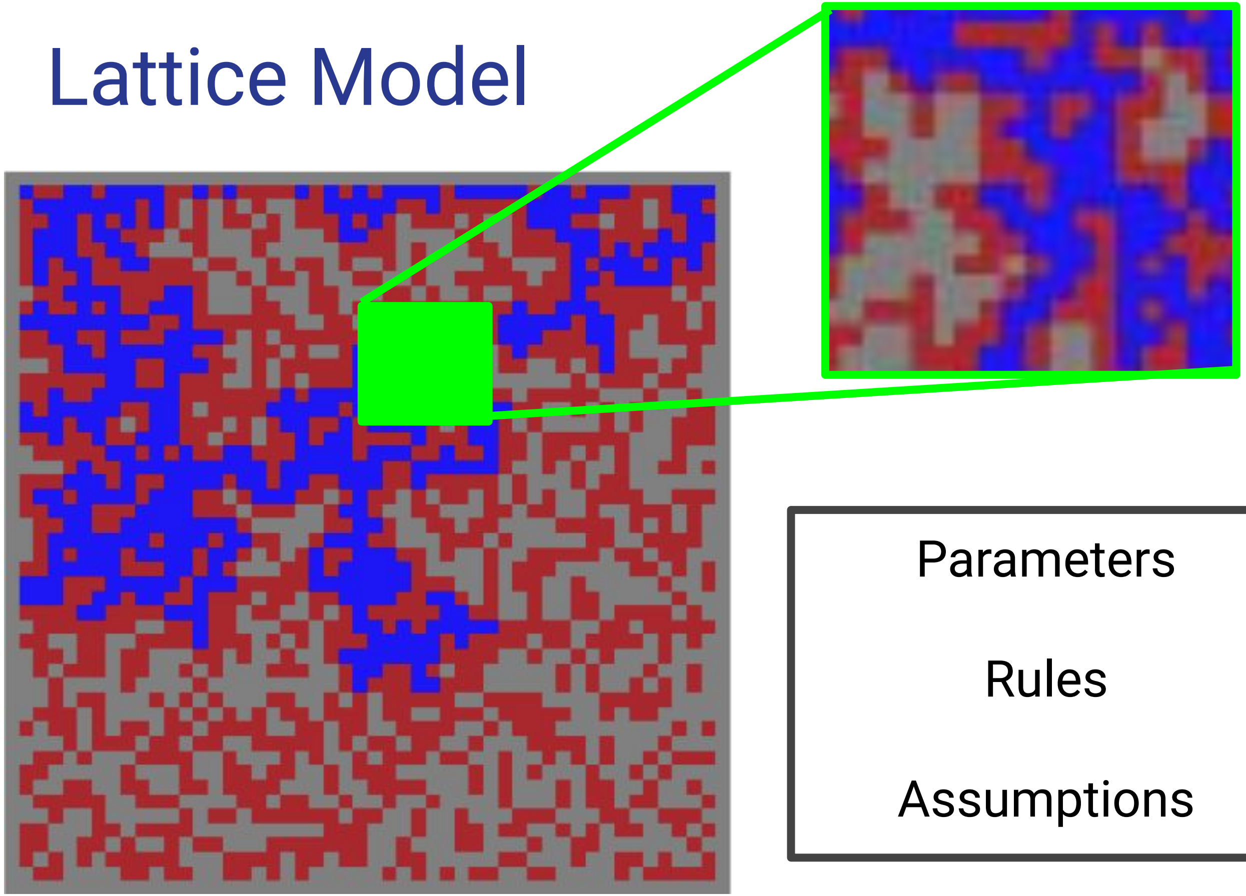
Agriculture

Civil Engineering

Geology

Question:
Can a neural network determine if water percolates through a 2D lattice if trained with extreme data?

Lattice Model



Rock

Water

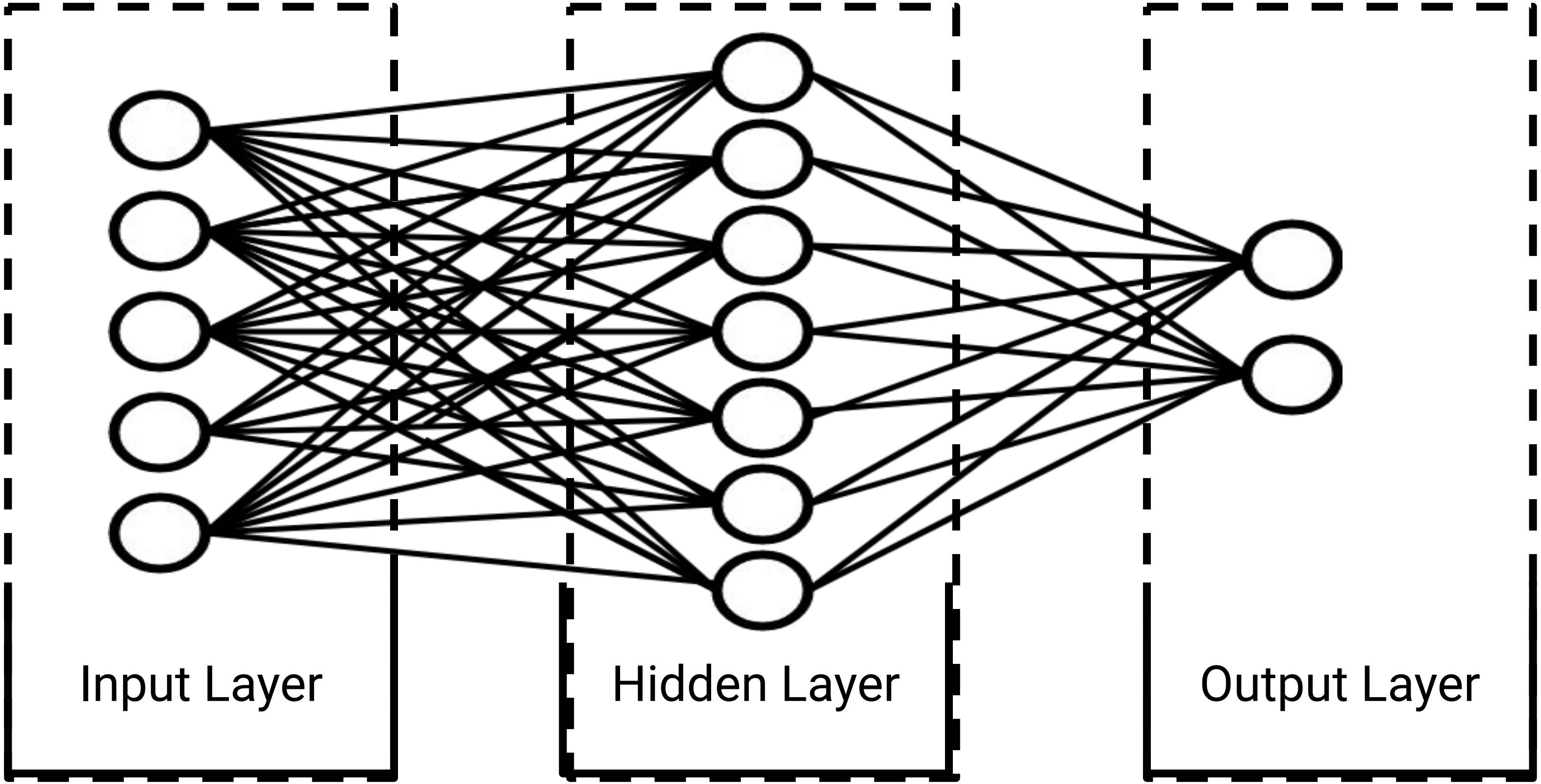
Air

Parameters

Rules

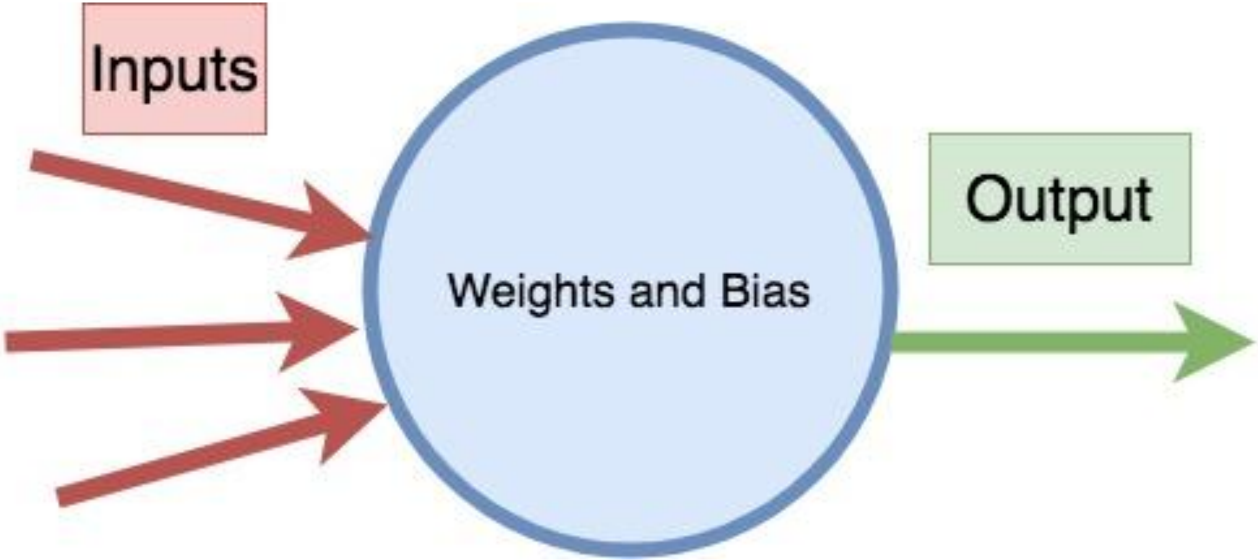
Assumptions

Neural Network Model

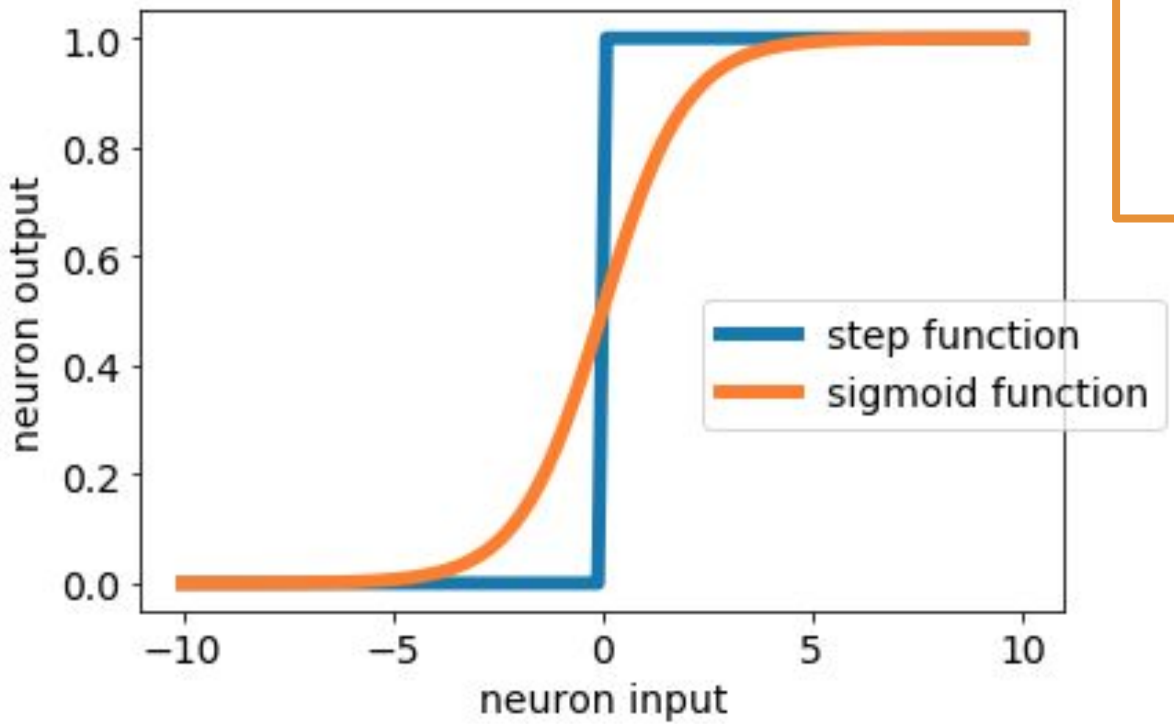


Numerical Methods

$$output = f\left(\left(\sum (weights)(inputs)\right) - bias\right)$$

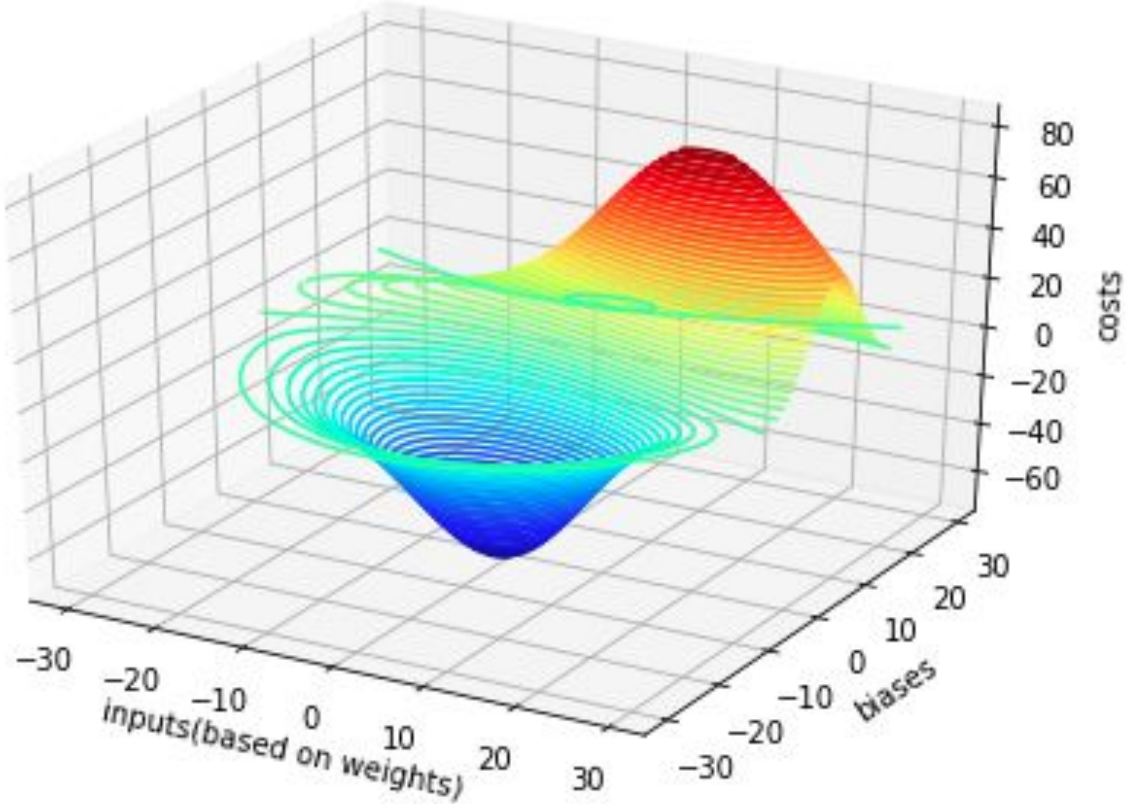


Which function?



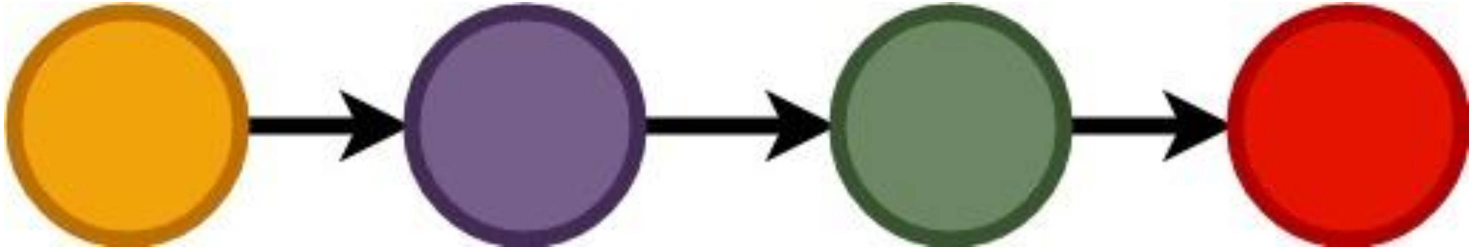
$$s(z) = \frac{1}{(1 + e^{-z})}$$

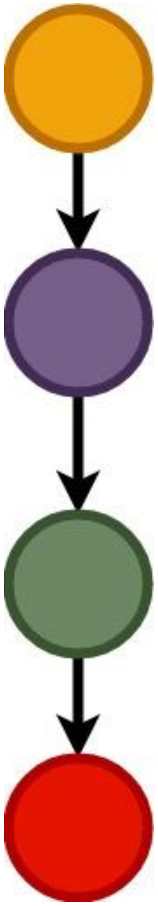
$$Cost(w,b) = \frac{1}{2n}\left(\sum_{k=1}^{j} abs(actual - guess)^2\right)$$

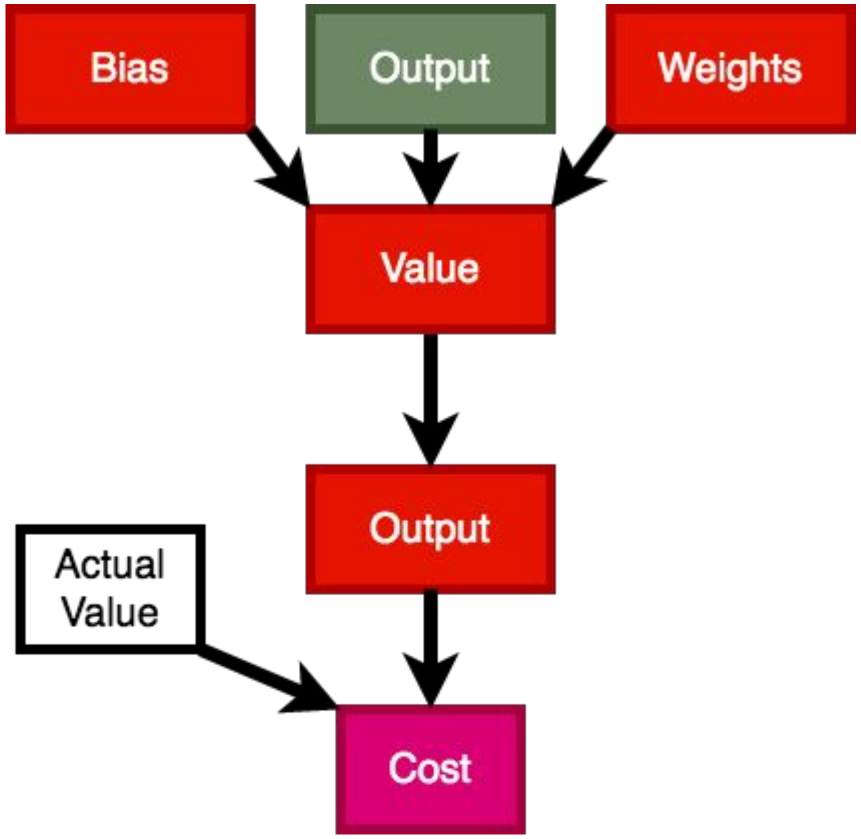


How do we Train the Network?

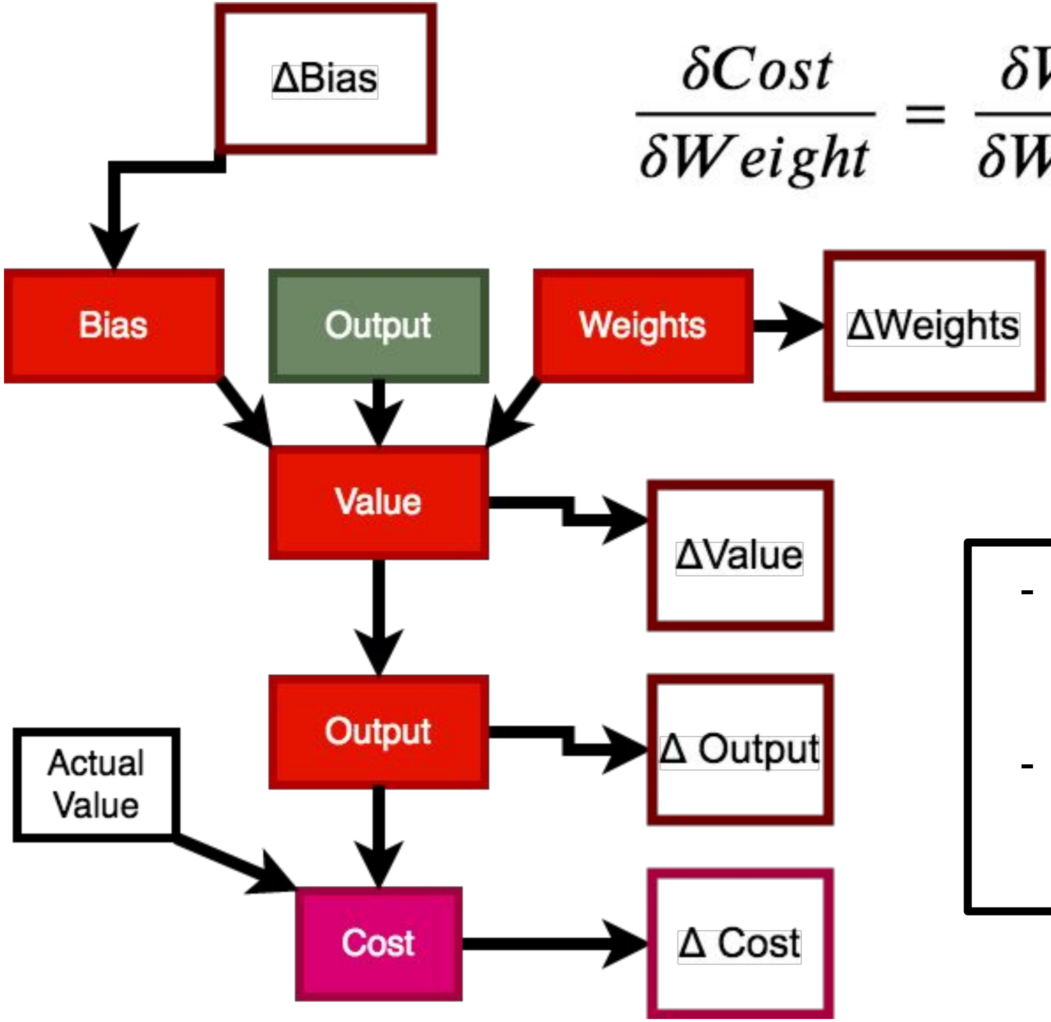
Gradient?






Bias
Output
Weights
Value
Output
Actual Value
Cost

$$\frac{\delta Cost}{\delta Weight} = \frac{\delta Value}{\delta Weight} \frac{\delta Output}{\delta Value} \frac{\delta Cost}{\delta Output}$$



- Backpropagation is an application of the chain rule

- Scaling up is trivial when we use matrices

Training Parameters

Training Data:

10000 samples

Rock Fractions [10% 20% 80% and 90%]

Test Data:

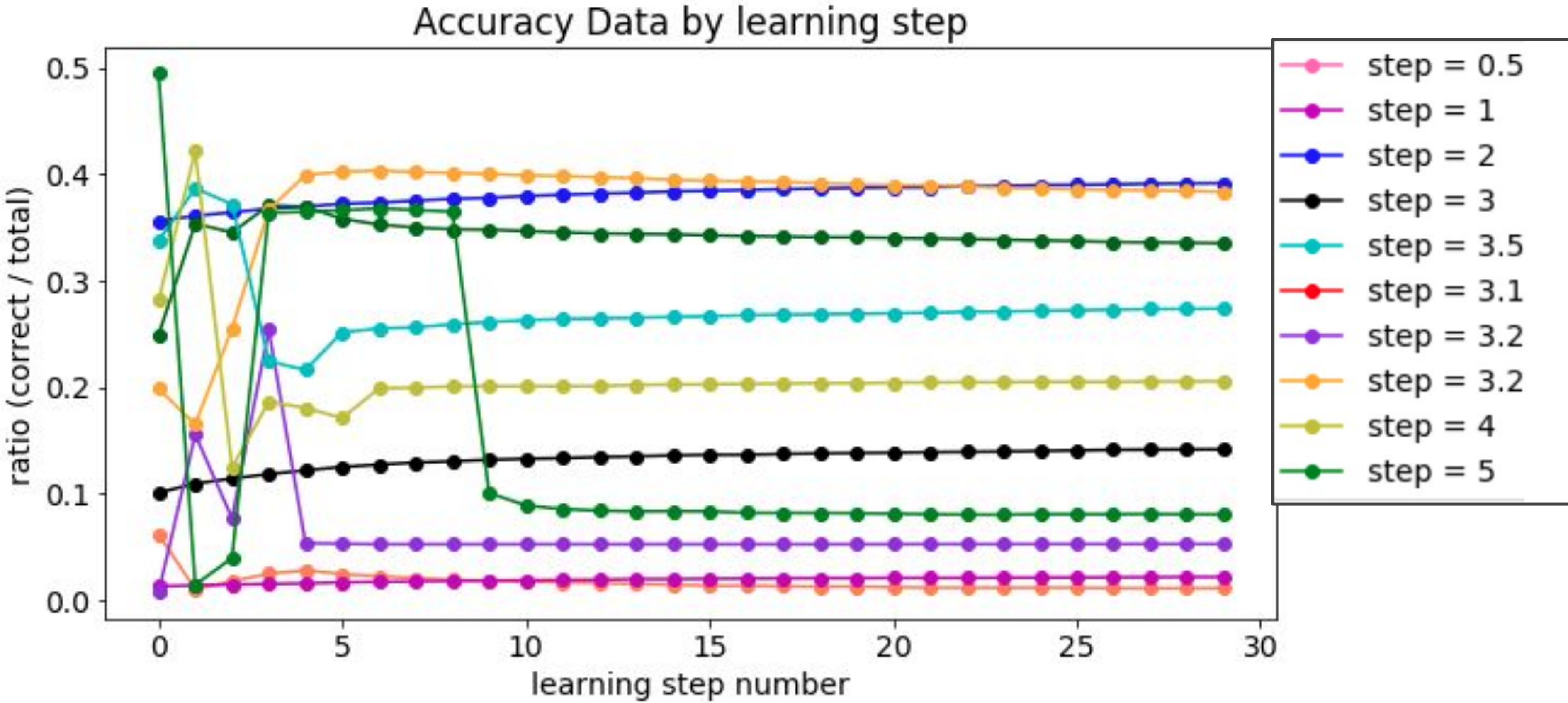
10000 samples

Near Critical point [30%-60%]

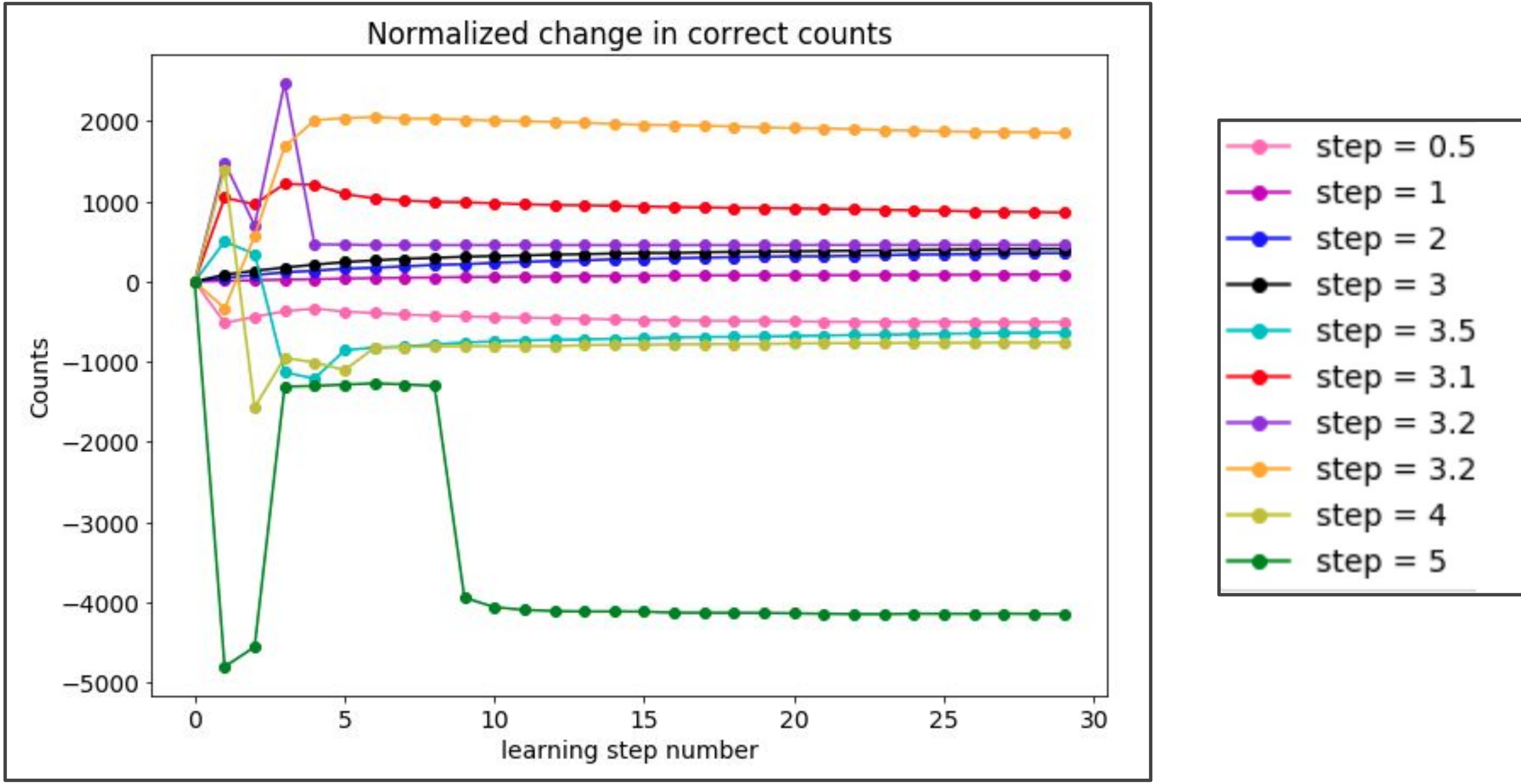
Independent Variable:

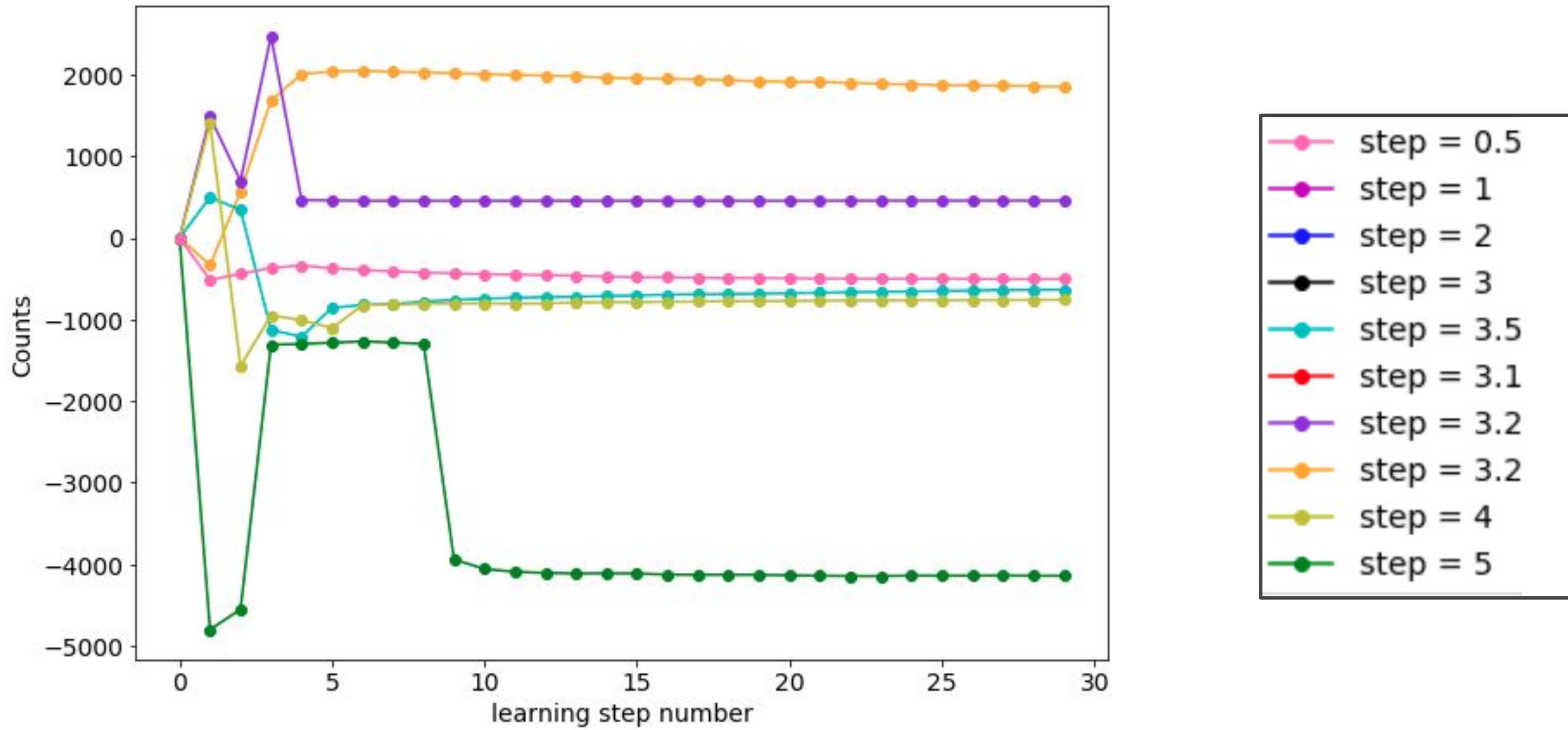
Learning Step Size

Results

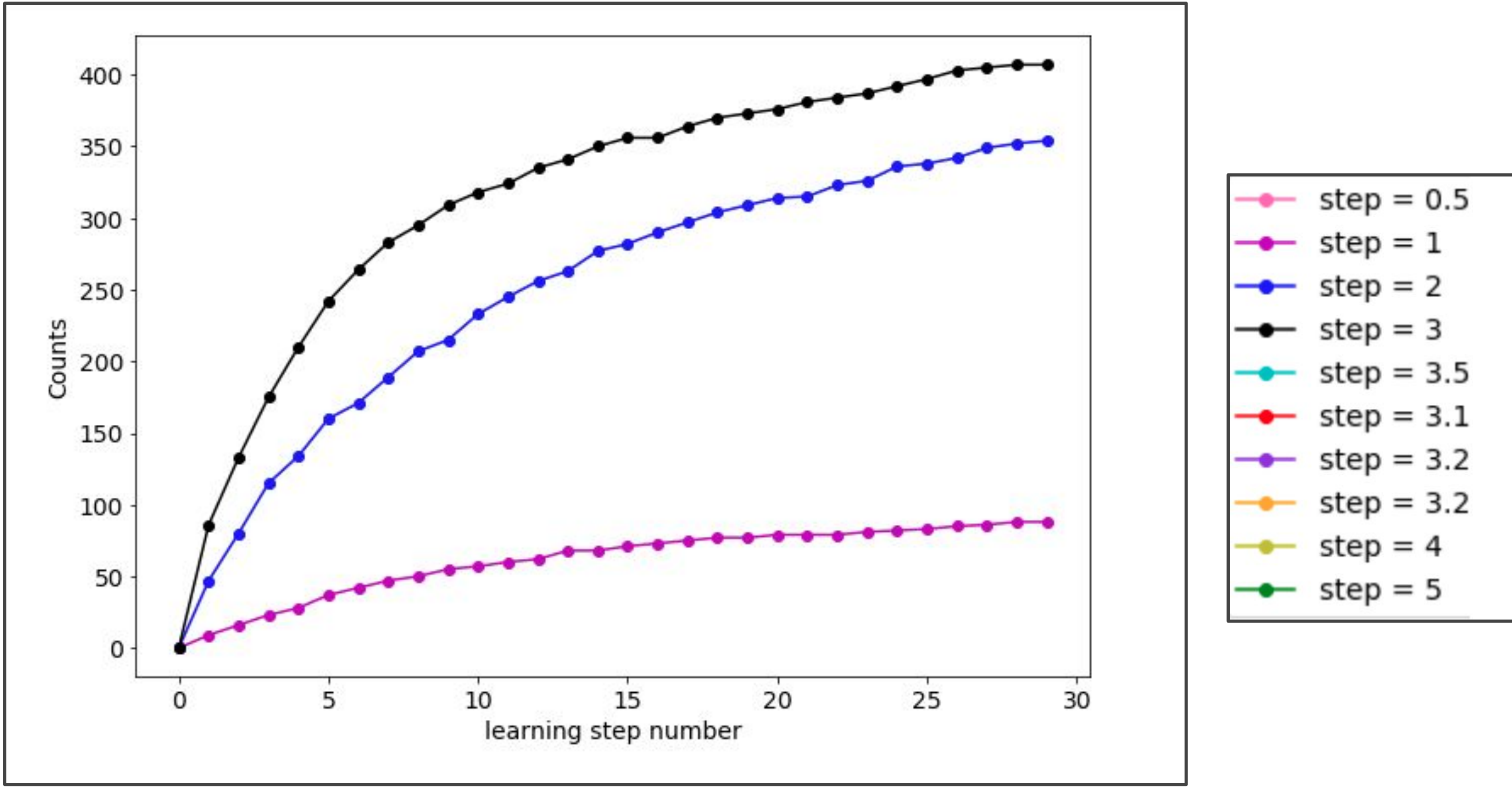


Accuracy Data by learning step
ratio (correct / total)
learning step number
0.0
0.1
0.2
0.3
0.4
0.5
0
5
10
15
20
25
30
step = 0.5
step = 1
step = 2
step = 3
step = 3.5
step = 3.1
step = 3.2
step = 3.2
step = 4
step = 5


Normalized change in correct counts
Counts
learning step number
2000
1000
0
−1000
−2000
−3000
−4000
−5000
0
5
10
15
20
25
30
step = 0.5
step = 1
step = 2
step = 3
step = 3.5
step = 3.1
step = 3.2
step = 3.2
step = 4
step = 5

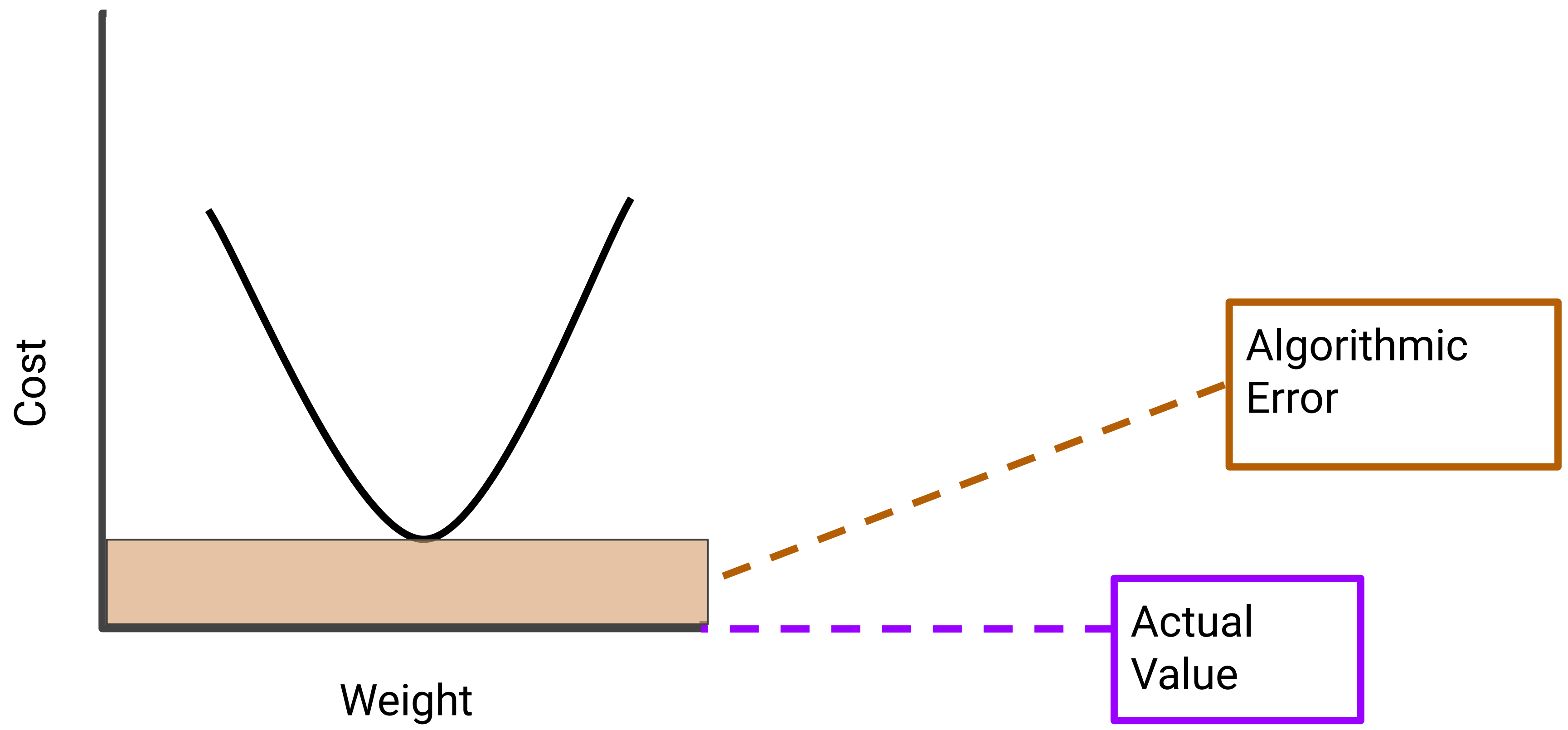


Counts
learning step number
step = 0.5
step = 1
step = 2
step = 3
step = 3.5
step = 3.1
step = 3.2
step = 3.2
step = 4
step = 5



Counts
learning step number
step = 0.5
step = 1
step = 2
step = 3
step = 3.5
step = 3.1
step = 3.2
step = 3.2
step = 4
step = 5

Analysis



Summary

Project Goals:

- 2D Lattice successfully models percolation

Can a neural network determine if water percolates through a 2D lattice if trained with extreme data?

- No, not with the simplifications of this model (Accuracy Plateau 48.3%)

Personal Goals:

- I built my first neural network
- I interpreted the behavior of the network to improve it
- I learned how to tune the network's parameters to minimize error

References

[1]Berkowitz, Brian & Balberg, Isaac. (1993)[Online]. Percolation theory and its application to groundwater hydrology. Water Resources Research

[2] Anderson W. Geology and Water [Online]. *Geology and Water* University of Northern Iowa. https://uni.edu/~andersow/geologyandwater.html [11 May 2020].

[3]Forest Fire Model [Online]. *The Forest-fire model*. https://scipython.com/blog/the-forest-fire-model/ [10 May 2020].

[4]Voyle G, Hudson H. What to do about compacted soil [Online]. *MSU Extension*: 2018. https://www.canr.msu.edu/news/what_to_do_about_compacted_soil [11 May 2020].

[5]Nielsen, A. M. Neural Networks and Deep Learning [Online]. *Neural networks and deep learning* Determination Press: 2019. http://neuralnetworksanddeeplearning.com/index.html [11 May 2020].

Uncertainty

Values accepted within 10% to avoid rounding error

$$\begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0.934 \\ 0.066 \end{bmatrix} \qquad \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} .75 \\ 0.25 \end{bmatrix}$$

Uncertainty Continued

-Each run is based off of a randomized initial network so each run was selected because it was characteristic of those hyper-parameters.

-Could be improved with more computational power