A Comparative Study of Neural Networks and Logistic Regression for High Energy Physics

Nigel Pugh, Joel Gonzalez-Santiago, and Thomas Johnson

Mentor: Jerome E. Mitchell

Abstract

-- Collisions particle colliders are a source of particle discoveries. However, determining these particles requires solving difficult signal-versus-background classification problems

-- Other approaches have relied on machine learning models, which are limited in learning complex functions

-- Recent advances (computation and data size) neural networks provide an opportunity to learn complex functions and better discriminate between signal and background classes

-- We study the design considerations for a neural network model and provide a comparative analysis with logistic regression

Machine Learning Problems

	Supervised Learning	Unsupervised Learning
Discrete	classification or categorization	clustering
Continuous	regression	dimensionality reduction

The Machine Learning Framework

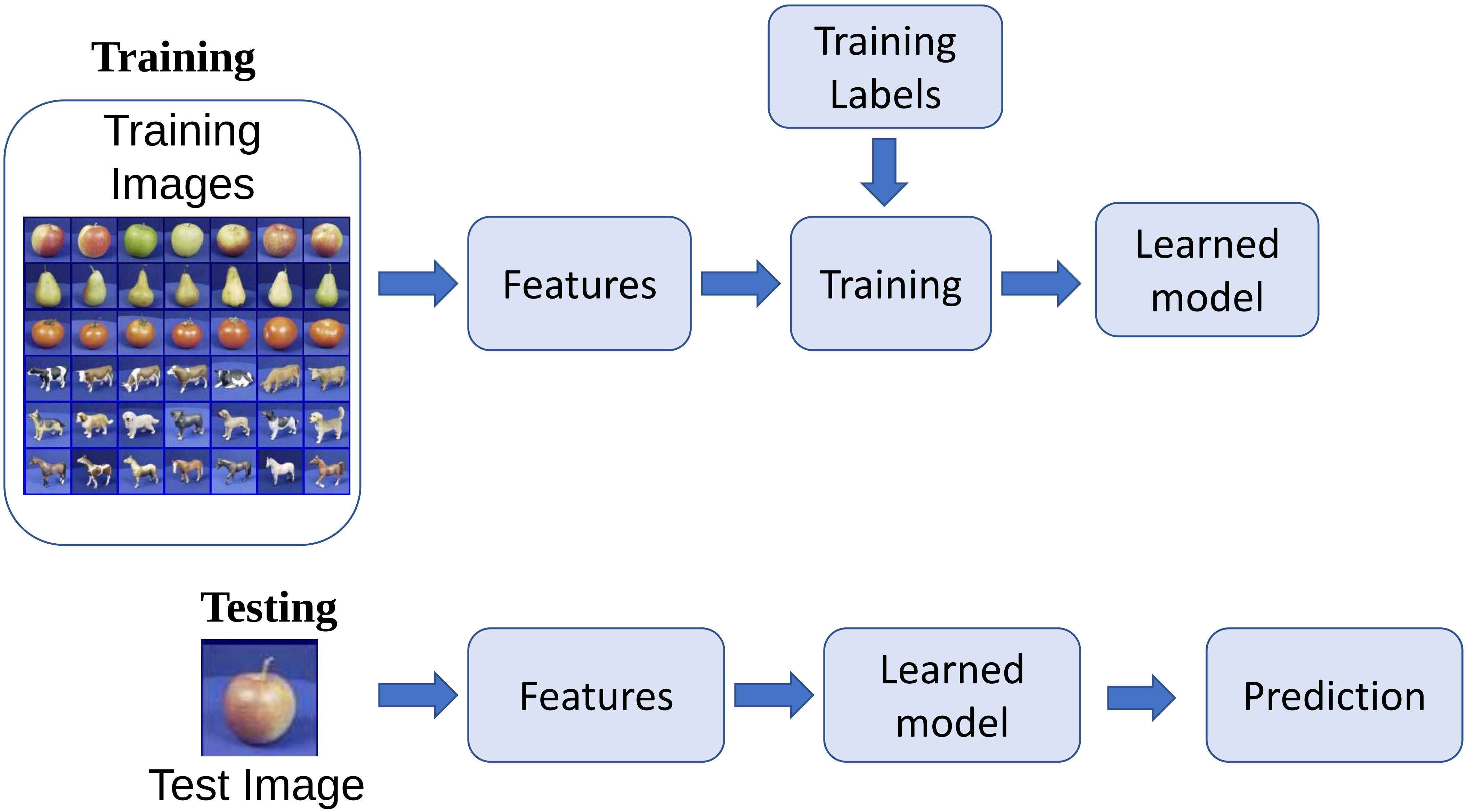
$$y = f(\mathbf{x})$$

output — prediction function — feature

-- **Training:** given a *training set* of labeled examples $\{(\mathbf{x}_1,y_1), \ldots, (\mathbf{x}_N,y_N)\}$, estimate the prediction function f by minimizing the prediction error on the training set

-- **Testing:** apply f to a never before seen *test example* **x** and output the predicted value $y = f(\mathbf{x})$

Steps


Training
Training Images
Training Labels
Features
Training
Learned model
Testing
Test Image
Features
Learned model
Prediction

Generalization

Training set (labels known)

Test set (labels unknown)

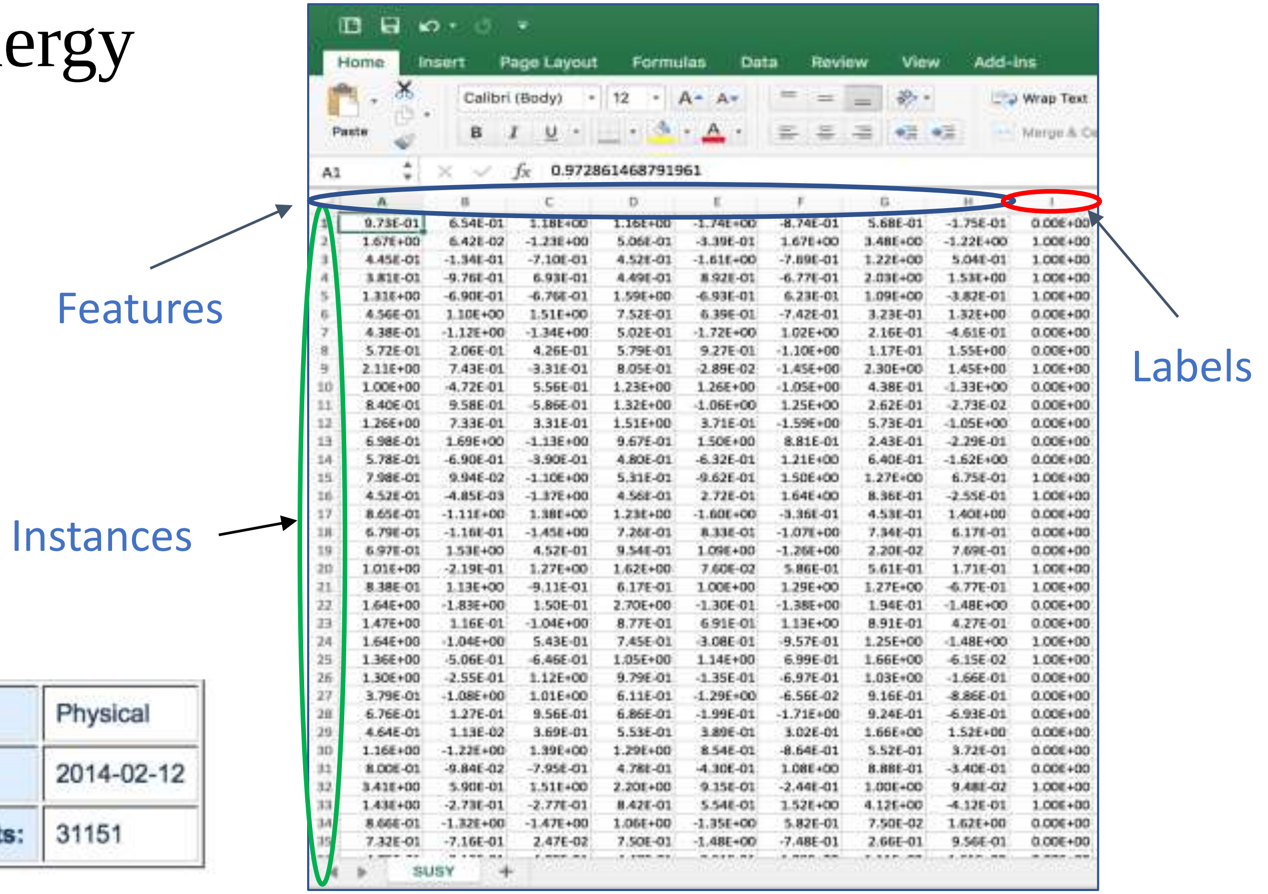
How well does a learned model generalize from the data it was trained on to a new test set?

Our Data…

Classification problem to distinguish between a signal process to background for high energy physics

UCI Machine Learning Repository

Data Set Characteristics:	N/A	Number of Instances:	5000000	Area:	Physical
Attribute Characteristics:	Real	Number of Attributes:	18	Date Donated	2014-02-12
Associated Tasks:	Classification	Missing Values?	N/A	Number of Web Hits:	31151

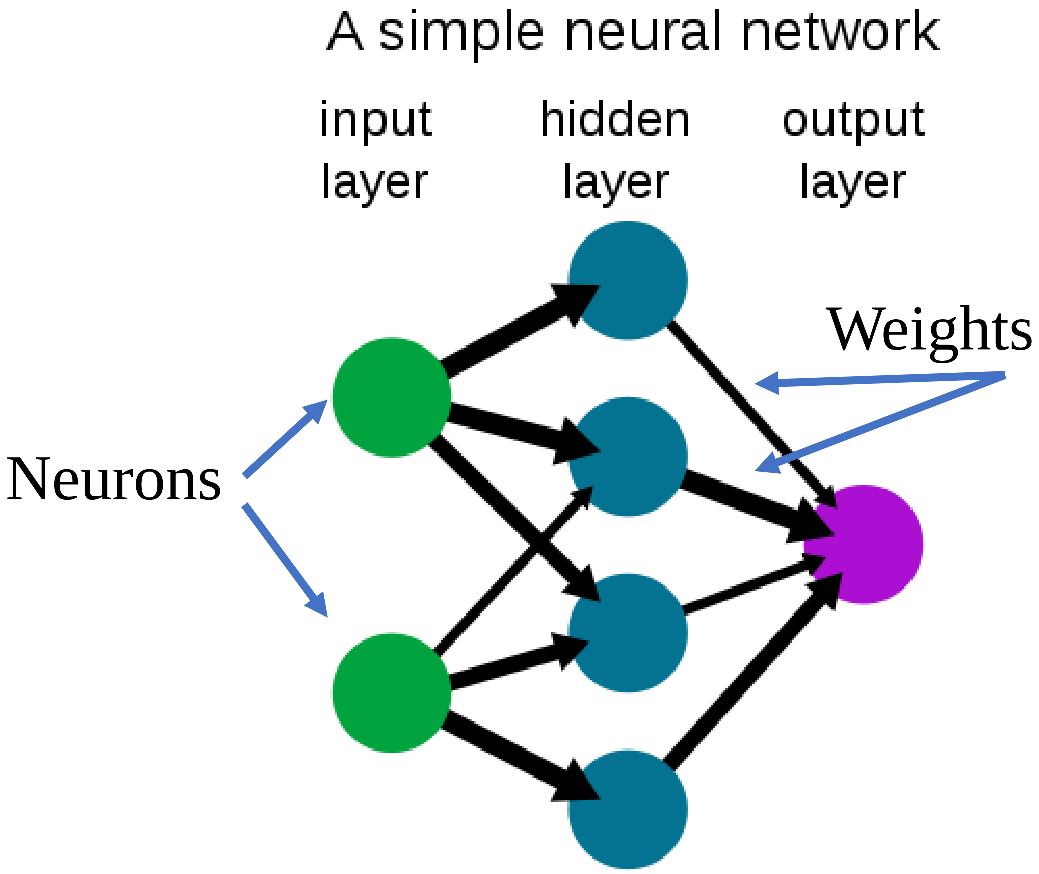


Neural Networks

-- Neural networks are loosely based on the human brain

-- Utilized for classification purposes

-- Popularity of neural networks have increased



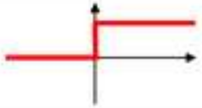
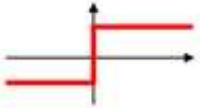
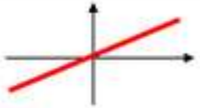
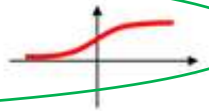
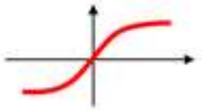
Backpropagation

-- Common Method used to train Neural Network

-- Supervised Learning Technique

-- Reduces error respective to weights

-- Activation function is utilized

Activation function	Equation	Example	1D Graph
Unit step (Heaviside)	$\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Sign (Signum)	$\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Linear	$\phi(z) = z$	Adaline, linear regression	
Piece-wise linear	$\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$	Support vector machine	
Logistic (sigmoid)	$\phi(z) = \frac{1}{1 + e^{-z}}$	Logistic regression, Multi-layer NN	
Hyperbolic tangent	$\phi(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$	Multi-layer NN	

Neural Networks

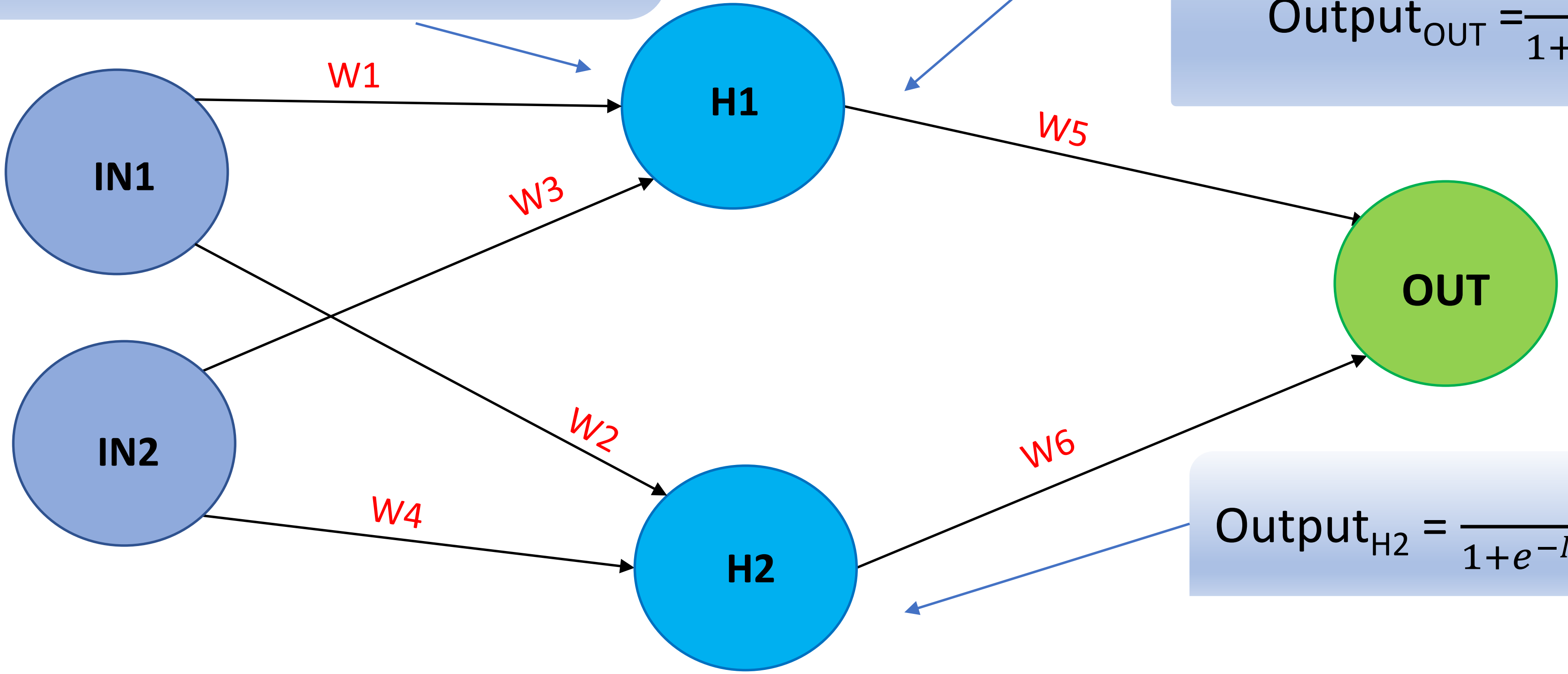
$$Netinput_{H1} = (IN1 * W1) + (IN2 * W3)$$

$$Netinput_{H2} = (IN1 * W2) + (IN2 * W4)$$

$$Output_{H1} = \frac{1}{1+e^{-Netinput_{H1}}}$$

$$Netinput_{OUT} = Output_{H1} * w5 + Output_{H2} * w6$$

$$Output_{OUT} = \frac{1}{1+e^{-Netinput_{OUT}}}$$



$$Output_{H2} = \frac{1}{1+e^{-Netinput_{H2}}}$$

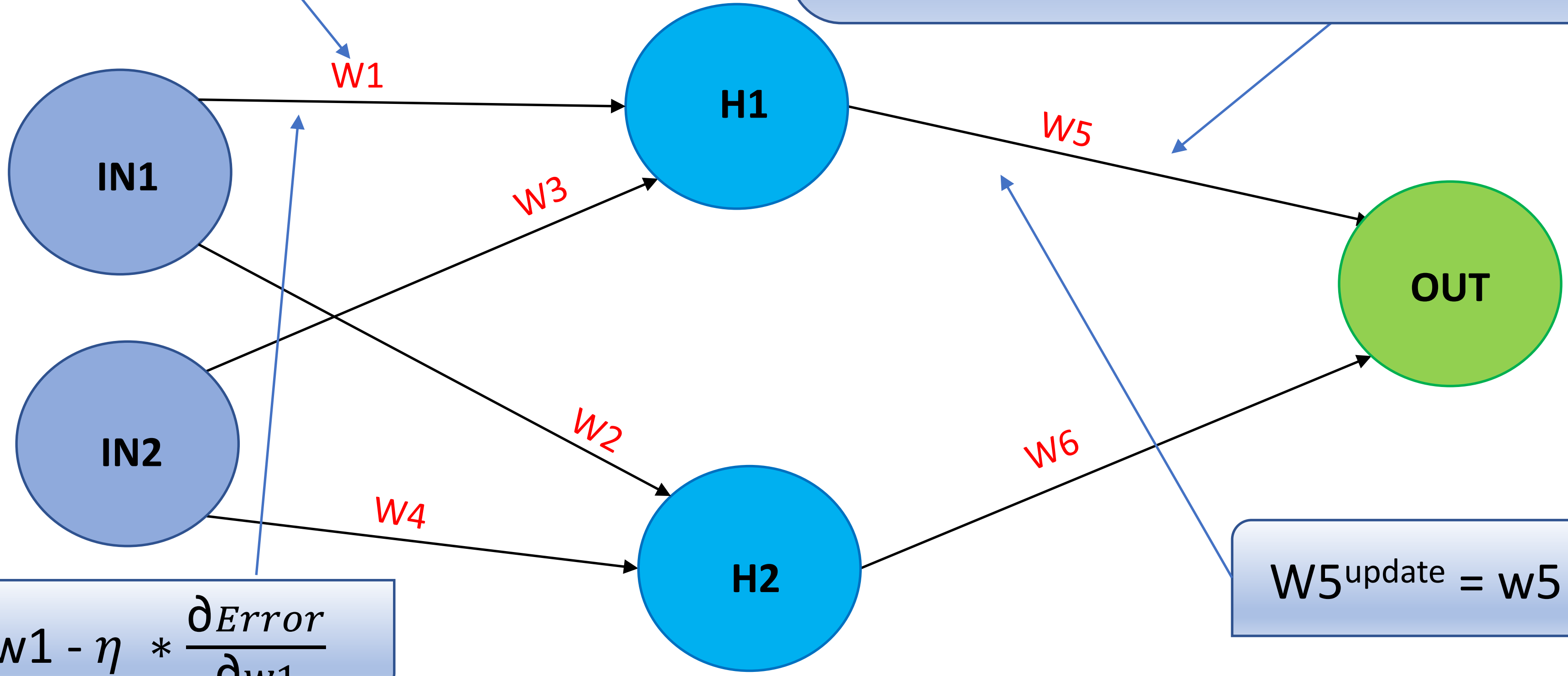
Backpropagation

$$\text{Error} = \tfrac{1}{2}\,(\text{Target} - \text{Output}_{OUT})^2$$

$$\frac{\partial Error}{\partial w_1} = \frac{\partial Error}{\partial Output_{H1}} * \frac{\partial Output_{H1}}{\partial netinput_{H1}} * \frac{\partial netinput_{H1}}{\partial w1}$$

$$\frac{\partial Error}{\partial w_5} = \frac{\partial Error}{\partial Output_{OUT}} * \frac{\partial Output_{OUT}}{\partial netinput_{OUT}} * \frac{\partial netinput_{OUT}}{\partial w5}$$

$$= -(\text{target} - \text{Out}) * \text{Out}(1\text{-Out}) * \text{Output}_{h1}$$



$$W1^{update} = w1 - \eta * \frac{\partial Error}{\partial w1}$$

$$W5^{update} = w5 - \eta * \frac{\partial Error}{\partial w_5}$$

Neural Network Design Considerations

-- What transfer function should be used?

-- How many inputs does the network need?

-- How many hidden neurons per hidden layer?

-- How many outputs should the network have?

There is no standard methodology to determinate these values. Even there is some heuristic points, final values are determinate by a trial and error procedure

Logistic Regression

Assumes the following form for P(Y|X):

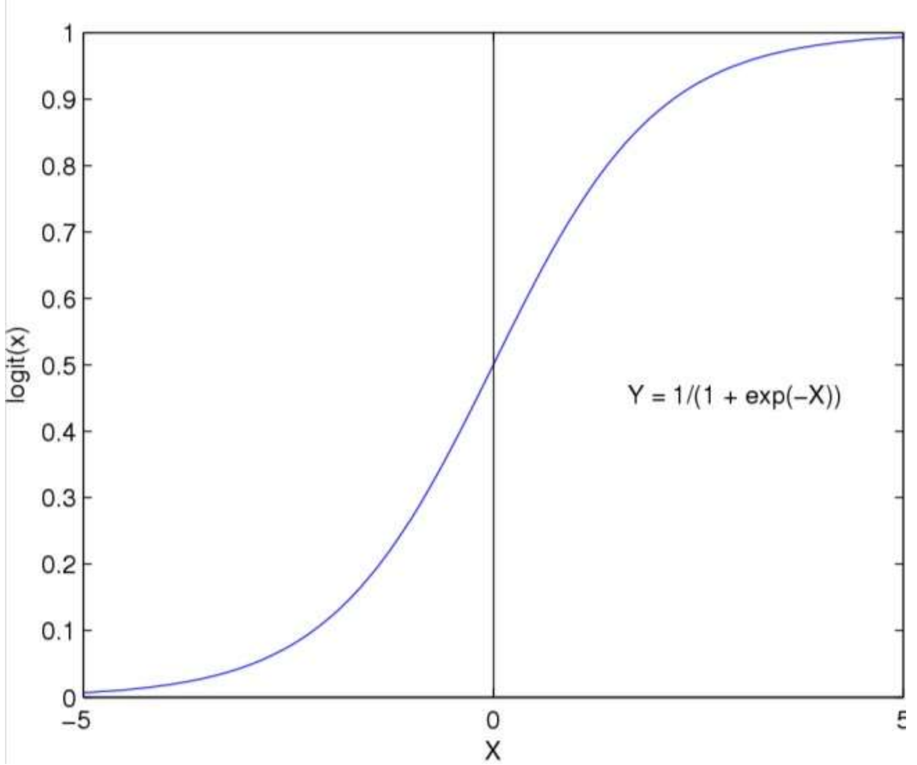
$$P(Y=1|X) = \frac{1}{1+\exp(-(w_0+\sum_i w_i X_i))}$$

Logistic function applied to a linear function of the data

Logistic function (or Sigmoid):

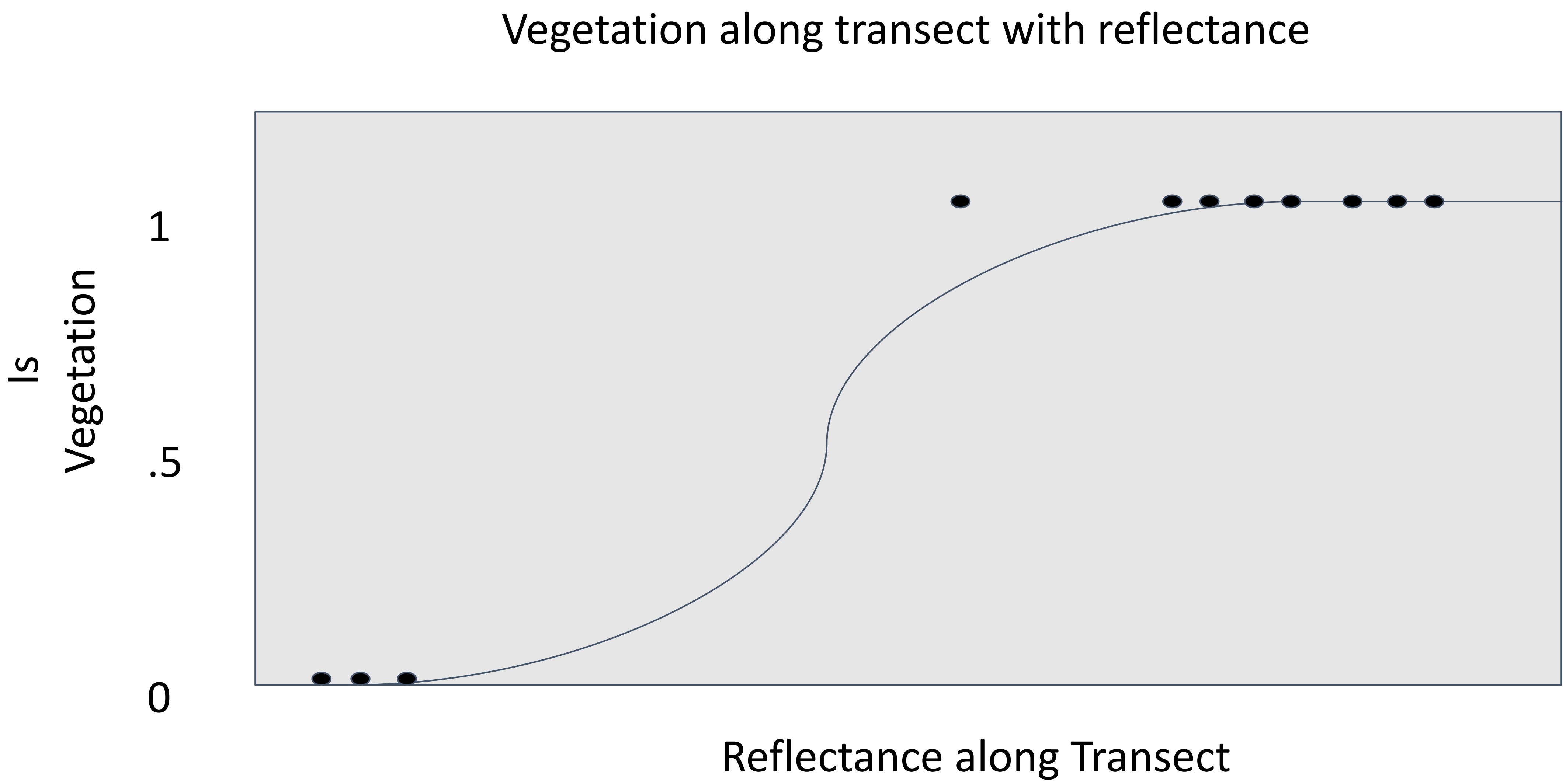
$$\frac{1}{1+exp(-z)}$$

logit (z)



Features can be discrete or continuous!

Logistic Regression Application



Experiment : Our Neural Network Information

--1000 training examples

-- 100 test examples

-- 8 features

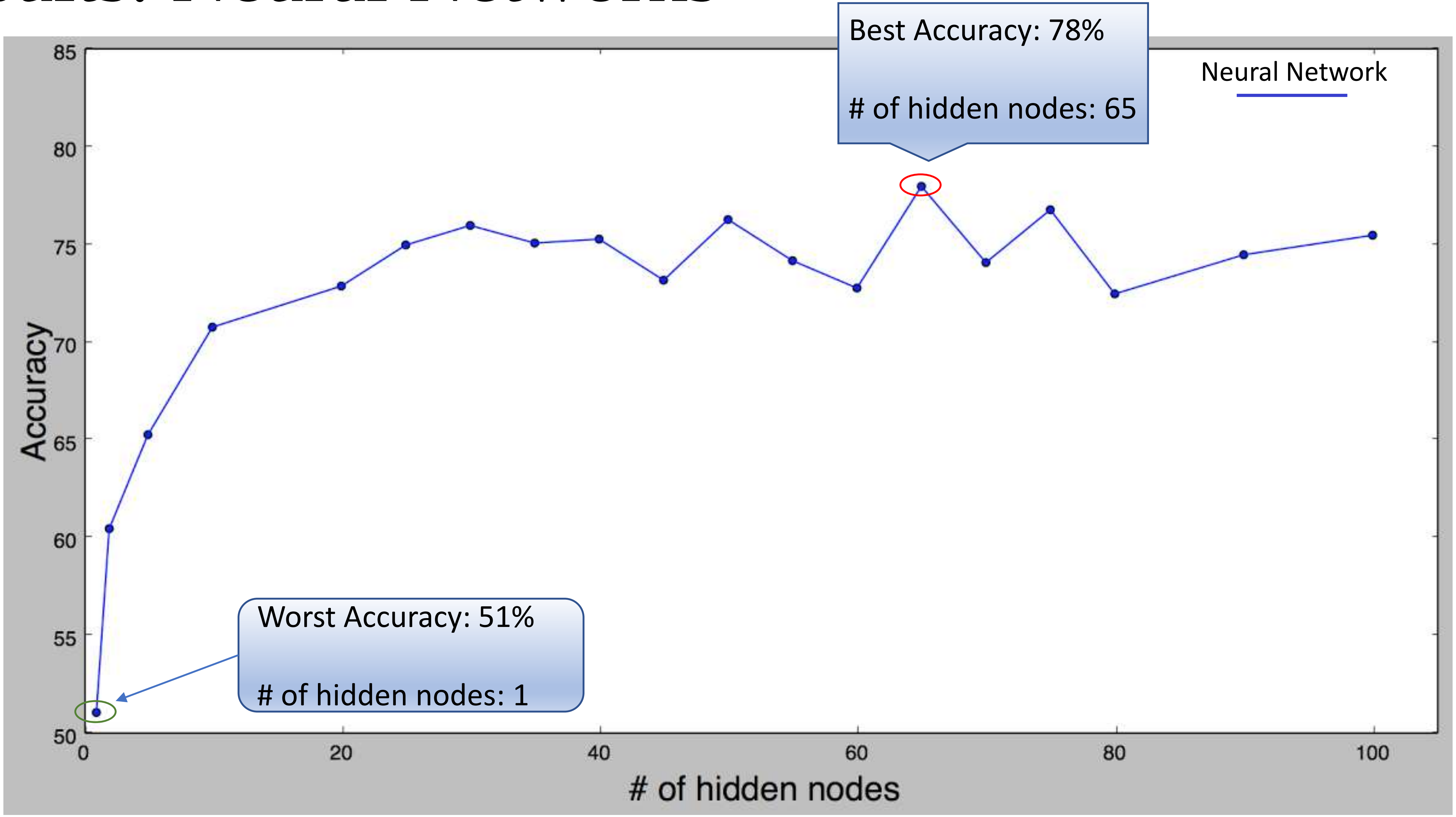
-- 3 Layer Neural Network

-- 10,000 training iterations

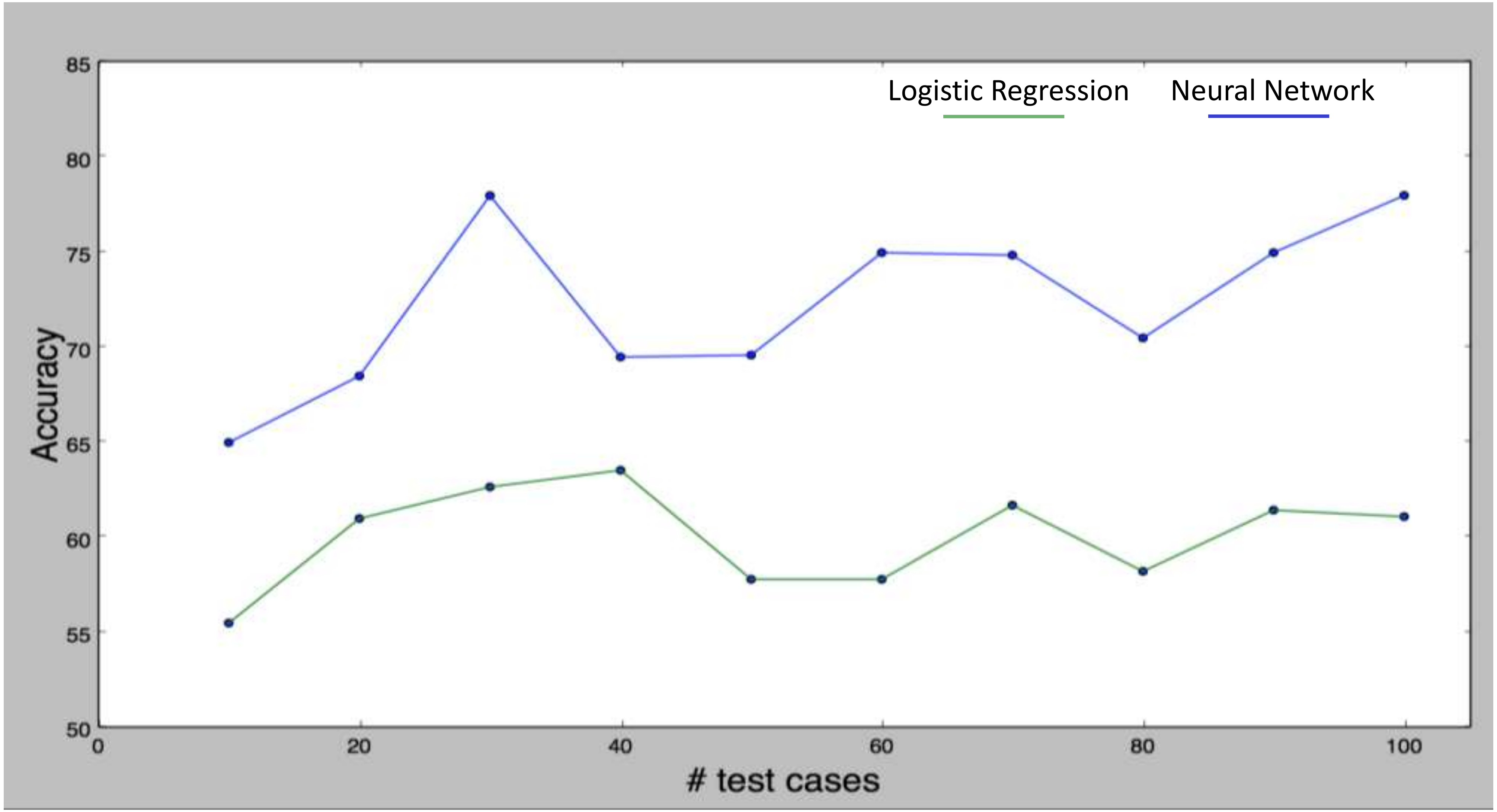
-- Hidden Nodes: Range from 1 – 100

-- Step size : .01

Results: Neural Networks



Results: Best hidden node NN vs Logistic Regression



Conclusion

-- Neural Network with 65 hidden nodes provided better performance than our Logistic Regression model on all test cases

-- We conclude it is better to use a the Backpropagation algorithm to classify the High Energy Physics Data

Future Work

-- Increase the number of hidden layers, to determine if accuracy percentage will increase

-- Compare against different machine learning algorithms against this same data, and determine which are more effective for classification.

References

Baldi, P., P. Sadowski, and D. Whiteson. "Searching For Exotic Particles In High-Energy Physics With Deep Learning". *Nature Communications* 5.4308 (2014): 9. Web. 13 Mar. 2017.

Sadowski, Peter, Julian Collando, and Pierre Baldi. "Deep Learning, Dark Knowledge, And Dark Matter". JMLR Workshop And Conference Proceedings 42. Irvine, California: Journal of Machine Learning Research, 2017. 81-97. Web. 12 Mar. 2017.

O. Ohrimenko, F. Schuster, C. Fournet, A. Mehta, S. Nowozin, K. Vaswani and M. Costa, "Oblivious Multi-Party Machine Learning on Trusted Processors", in *25th USENIX Security Symposium*, Austin, Texas, 2016, pp. 618-636.

D. Marrón, J. Read, A. Bifet and N. Navarro, "Data stream classification using random feature functions and novel method combinations", *Journal of Systems and Software*, vol. 127, pp. 195-204, 2017.

D. Whiteson and M. Lichman, "UCI Machine Learning Repository: SUSY Data Set", *Archive.ics.uci.edu*, 2017. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/SUSY. [Accessed: 03- Apr- 2017].

Acknowledgements

The team would like to acknowledge:

-- Jerome Mitchell, for his guidance, and contributions in completing this research project

-- Dr. Linda Hayden, who provided funding and opportunity of this project through the CERSER program

Questions?