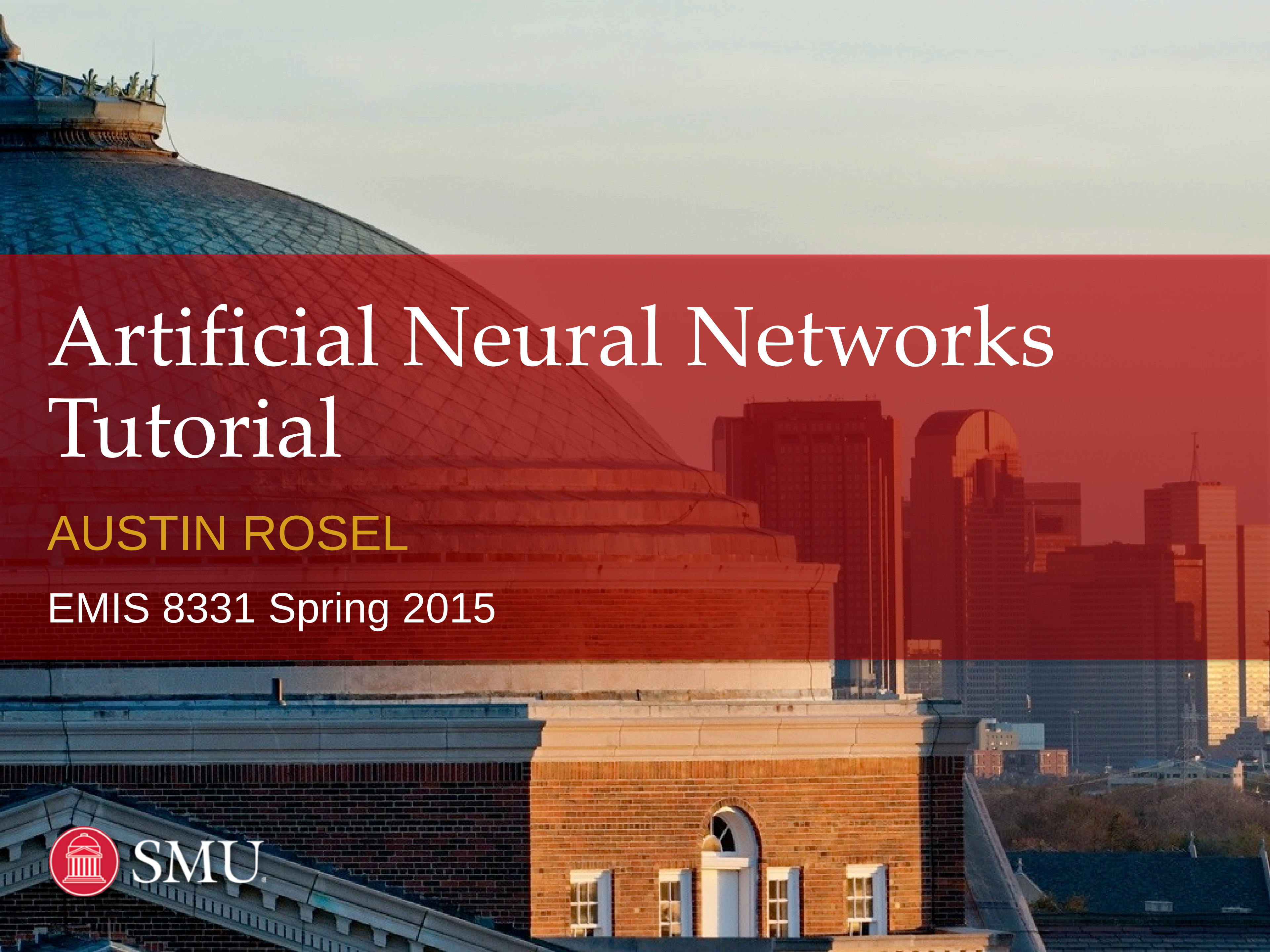


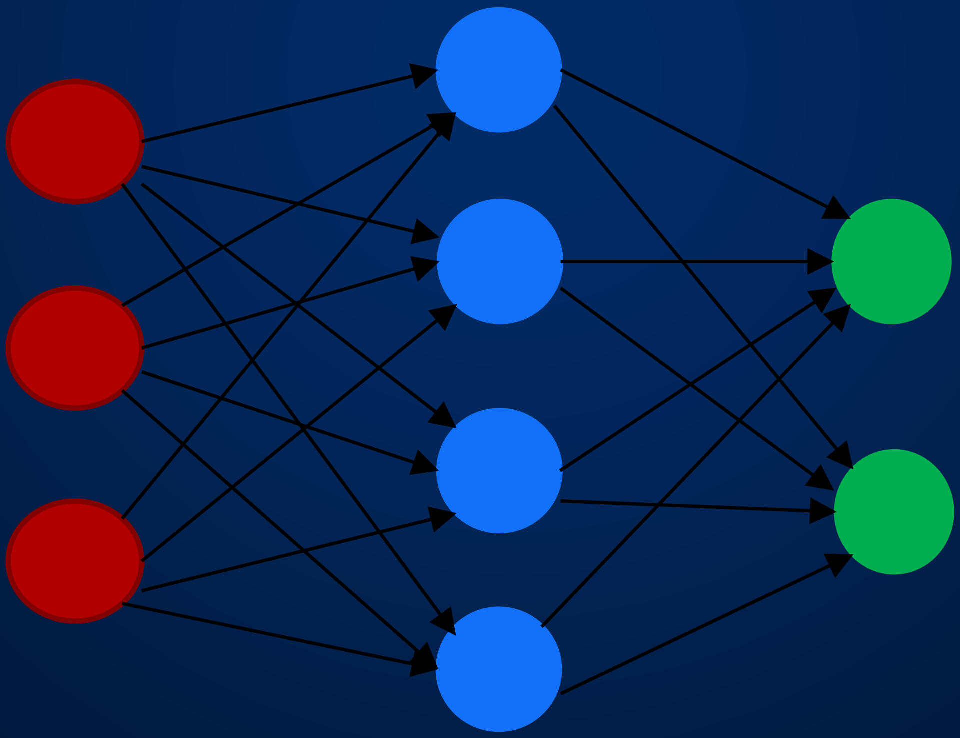
Artificial Neural Networks Tutorial

AUSTIN ROSEL

EMIS 8331 Spring 2015

Artificial Neural Networks

This presentation is intended to be a primer on Artificial Neural Networks. After going through this presentation you should have an understanding on what is going on inside a basic Neural Network, as well as have a good enough understanding of their strengths and weaknesses to help you identify situations for which they are well suited (and not well suited).



SMU

Outline

- Problem Area
- What is an Artificial Neural Network (ANN)
- What are ANNs used for?
- Positive and Negative Features of ANNs
- State-of-the-Art
- References

The terms Artificial Neural Network, ANN, Neural Network and NN are used interchangeably in this presentation.

Problem Area

- Rule Based Programming
  - The rules, or condition-action pairs, need to be known in advanced.
  - An algorithm that picks a rule is executed and an action is taken.
  - Could be thought of as a bunch of “if-then” statements.

When it is not possible, feasible or required to know all the rules within the system an ANN is one machine learning method that can be employed to model the process.

http://inst.eecs.berkeley.edu/~selfpace/studyguide/3S.rdngs/rule-based.prog.pdf

What is a Neural Network?

There is no single formal definition of what a Neural Network is.

- Artificial Neural Networks(ANNs) are a family of statistical learning algorithms inspired by biological neural networks that are used to estimate or approximate functions that can depend on a large number of inputs and are generally unknown. *(wikipedia)*
- A Neural Network(NN) is a two-stage regression or classification model, typically represented by a network diagram. A neural network is a nonlinear statistical model. *(The Elements of Statistical Learning: Hastie, Tibshirani, Friedman)*
- A nonlinear statistical learning algorithm that functions as a universal approximator for an unknown function or functions. *(my definition)*

What is a Neural Network?

It's important not to take the analogy to biological networks too far.

- » The human brain is estimated to contain around 100,000,000,000 neurons, ANNs usually contain less than 1,000 equivalent units.
- » The interconnection of neurons is much bigger in natural systems.
- » The way in which ANNs store and manipulate information is a gross simplification of the way in which networks of neurons work in natural systems.
- » Some methods used in training ANNs are not used in biological networks.

http://www.doc.ic.ac.uk/~sgc/teaching/pre2012/v231/lecture12.html

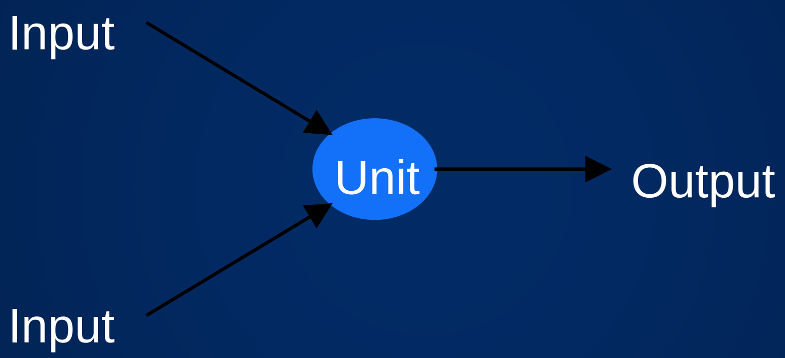
SMU

What is a Neural Network?

- ANN are typically defined by the following:
  - The interconnection pattern between the different layers of neurons
  - The learning process for updating the weights of the interconnections
  - The activation function that converts a neuron's weighted input to its output activation.

Units

Units (or neurons) are mini calculation devices. They take in real-valued input from multiple other nodes and they produce a single valued output (activation function).



Input Units – Make up the input layer, where information is taken into the model.

Hidden Units – Take input from the input layer as well as other hidden units. A weighted sum of the output from the input units forms the input to every hidden unit.

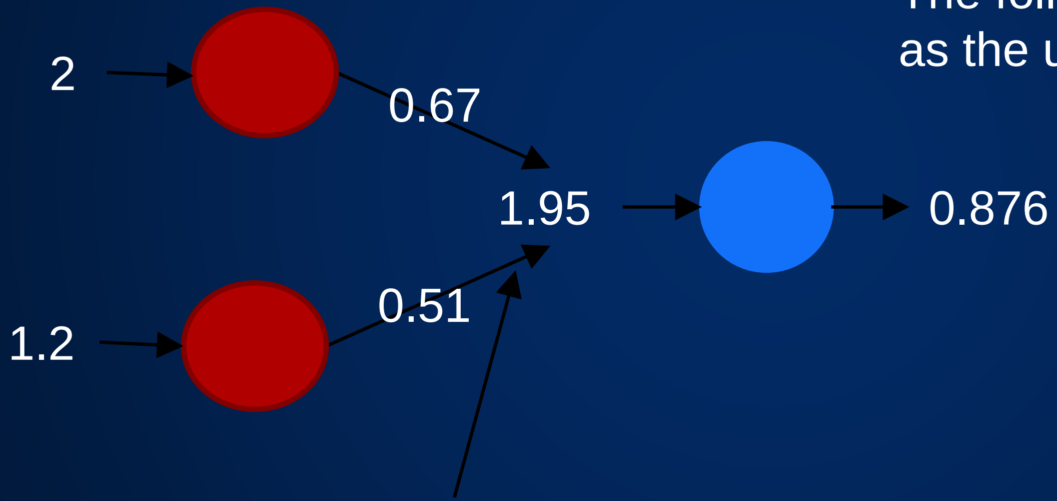
Output Units - A weighted sum of the output from the hidden units forms the input to every output unit. Output can be a real number or a category.

http://www.doc.ic.ac.uk/~sgc/teaching/pre2012/v231/lecture12.html

```
b->h3 i1->h3 i2->h3 0.68 0.67 0.51
```

Unit Calculation Example

The following example takes two weighted inputs and calculates a single output.



The following formula is used as the unit calculation:

$$\frac{1}{1+e^{-v}}$$

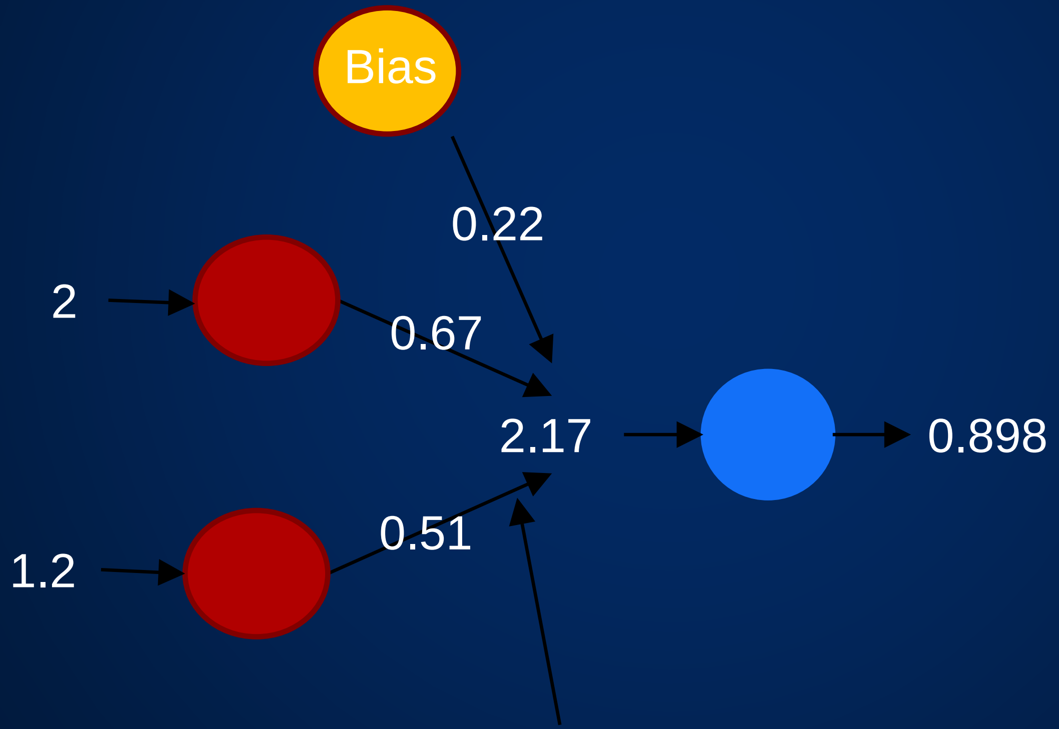
Weighted Input: 0.67 * 2 + 0.51 * 1.2 = 1.95

Unit Calculation: =1/(1+ e $^{-1.95}$) = 0.876

The output value is changed by adjusting the weights, this is important for the learning process.

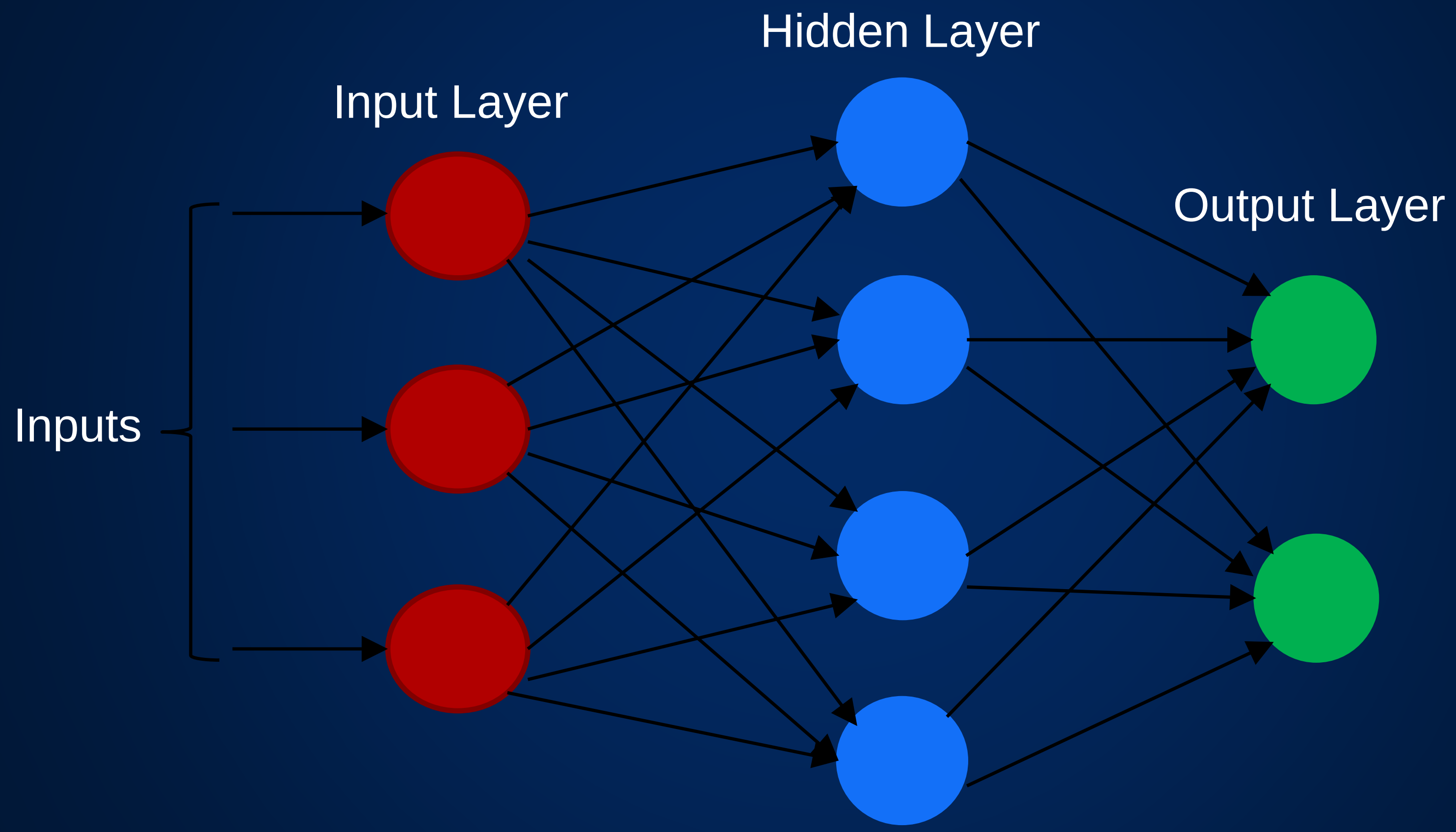
Bias

NN commonly use a bias variable when calculating weighted sums for inputs.



Weighted Input: 0.22 + 0.67 * 2 + 0.51 * 1.2 = 2.17

Neural Network is a Network of Units

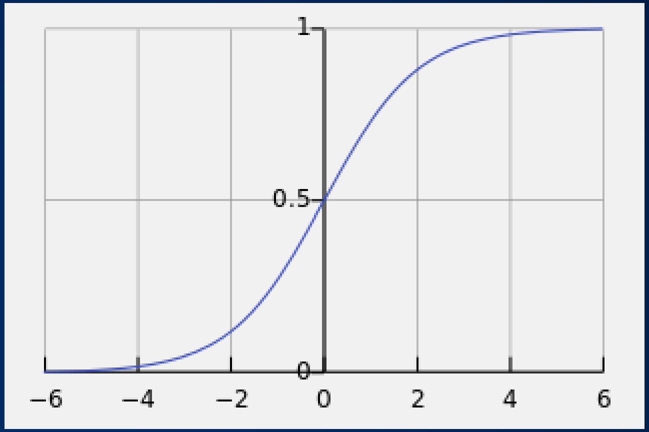


This is a single layer feed-forward Neural Network. Networks with cycles are commonly called recurrent

Sigmoid Function

Why would I chose the following formula for the unit calculation?

$$\frac{1}{1+e^{-\nu}}$$



It wasn’t my idea.

This formula is known as the sigmoid function or logistic function and is often chosen as the the activation function or the unit calculation used in Neural Networks. The activation function does not need to be the sigmoid function.

This formula is chosen as it approximates a step function (discrete activation of zero or one) with a continuous and differentiable function. Why this is important is covered later.

http://en.wikipedia.org/wiki/Sigmoid_function

Learning

- Neural Networks have the possibility of learning.
  - Given a task to solve and a class of functions, learning means using a set of observations to find the optimal function in the class of functions to solve the task.
- Learning entails the definition of a cost function,
  - The learning algorithm choses the function that has the smallest possible cost.
  - Online machine learning is frequently employed, where the cost is partially minimized as each new solution is seen.

http://en.wikipedia.org/wiki/Artificial_neural_network

How Neural Networks Learn

Neural Networks learn by updating the weights used in the calculations. The algorithm chooses optimal results by minimizing a cost function.

The cost function normally chosen as the total sum of squared errors:

$$E(w) = \frac{1}{2}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

$\hat{y}$ *is the predicted value*

y *is the training value*

w *is the weight*

E(w) is a function of the weights of the model because the predicted value of y is a function of the weights. Because the function to predict y is continuous and differentiable (remember the sigmoid) the gradient descent method can be used to move toward the optimal values of w. This process of moving the values of w toward the optimal values is learning.

Introduction to Data Mining: Tan, Steinbach, Kumar

Gradient Descent

- Take the gradient of the cost function.
- Multiply by -1 to point to the direction of quickest descent.
- Multiply by the learning rate λ.
- Adjust weights in direction of steepest descent.

$$w_j \leftarrow w_j - \lambda \frac{\delta E(w)}{\delta w_j}$$

It is possible this method could result in the convergence of the cost function to a local minimum.

Introduction to Data Mining: Tan, Steinbach, Kumar
http://andrew.gibiansky.com/blog/machine-learning/machine-learning-neural-networks/

Gradient Descent

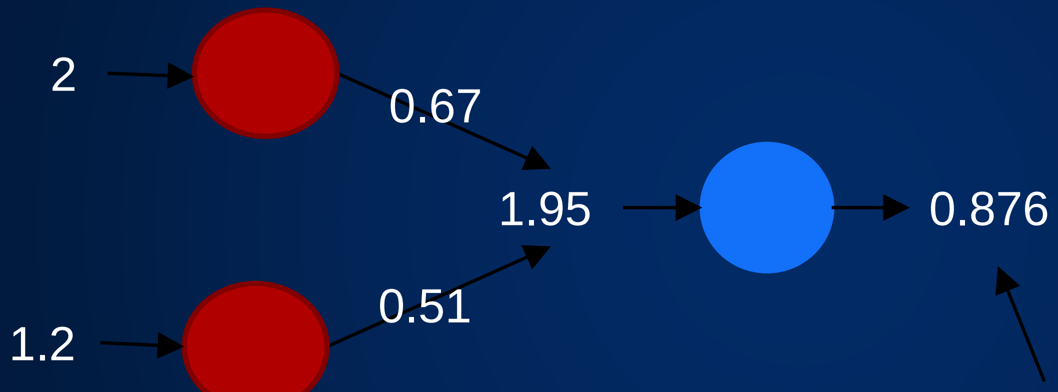
Gradient Descent with Decay

- Parameter alpha causes the weight to decay in proportion to its size.

$$w_j \leftarrow w_j - \lambda \frac{\delta E(w)}{\delta w_j} - \alpha \lambda w_j$$

Back Propagation

If my cost function is the sum of squared errors, and I have a hidden layer what value of y do I use for the training value?



What do I compare this value to in the hidden layer?

A method known as back-propagation has been developed for this problem.

Introduction to Data Mining: Tan, Steinbach, Kumar
http://andrew.gibiansky.com/blog/machine-learning/machine-learning-neural-networks/

Back Propagation

Two phase process

- Predict an output based on random weights
- Phase 1
  - Calculate the outputs and error, with the current weights.
- Phase 2
  - Use the gradient descent method to update the weights for level k+1.
  - Use the weights and the errors for level k+1 to update the weights for level k.
  - Go back to phase 1 or terminate.

The mathematical derivation of this method involves the use of derivatives and partial derivatives which is a reason for the choice of a differentiable activation function.

https://www4.rgu.ac.uk/files/chapter3%20-%20bp.pdf
http://en.wikipedia.org/wiki/Backpropagation

Back Propagation

Remember the cost function:

$$E(w) = \frac{1}{2}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

For back propagation the $(y_i - \hat{y}_i)$ is replaced with weighted sum of all the errors from the k+1 nodes connected to the hidden node and the weights between the hidden node and the k+1 nodes.

Uses of Artificial Neural Networks

- Supervised Learning
  - Pattern recognition
  - Regression
- Unsupervised Learning
  - Clustering
  - Compression
  - filtering
- Reinforcement Learning
  - Games
  - Coupled with dynamic programming
    - Vehicle Routing

Some Examples of Neural Network Usage

- Computer Vision
- Handwriting Recognition
- Autonomously flying aircraft
- Forecasting energy consumption
- Image Compression
- Lung cancer triage

http://www.nasa.gov/centers/dryden/news/NewsReleases/2003/03-49.html#.VOU2fi7cvEY

http://www.ncbi.nlm.nih.gov/pubmed/11718427

http://neuron.eng.wayne.edu/bpImageCompression9PLUS/bp9PLUS.html

Positive Features of Neural Networks

Artificial Neural Networks can solve some problems that are hard to solve with ordinary rule-based programming.

Possibility of learning – An ANN can find the best function from a class of functions for solving a task via utilizing a cost function.

Multilayer Neural networks with at least one hidden layer are universal approximators: i.e., they can be used to approximate any target function.

Can handle redundant features.

Once trained, data can be classified rapidly.

Introduction to Data Mining: Tan, Steinbach, Kumar
http://en.wikipedia.org/wiki/Artificial_neural_network

Negative Features of Neural Networks

The industry has favored support vector machines and linear classifiers starting in the 1990s, although there is renewed interest in Neural Networks starting in the 2000s due to the advent of deep learning

Sensitive to input scaling

Too many weights can lead to overfitting.

Require a large diversity of training for real-world operation

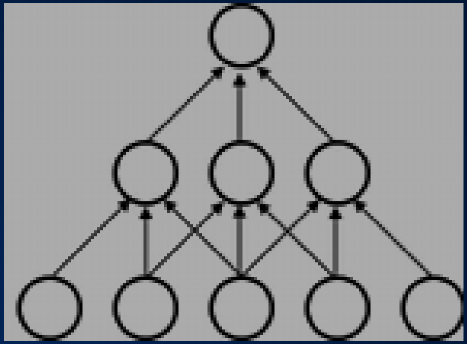
Training a Neural Network is a time consuming process

Sensitive to the presence of noise

Not good where the goal is to describe the physical process and the roles of the individual inputs

Introduction to Data Mining: Tan, Steinbach, Kumar
The Elements of Statistical Learning: Hastie, Tibshirani, Friedman
http://en.wikipedia.org/wiki/Artificial_neural_network

Deep Learning



- No formal definition of Deep Learning
- Set of algorithms in machine learning that attempt to model high-level abstractions in data by using model architectures composed of multiple non-linear transformations
- Supervised, unsupervised, feedforward, or recurrent (deep by nature) NNs with many nonlinear processing stages
- Mostly unsupervised
- Deep NN – Multiple hidden layers
- Convoluted deep NN – Semi overlapping and weight sharing
- Has been called a rebranding of neural networks
- Example is a Deep NN that was able to recognize the high level concept of a cat by watching youtube videos

References

Books:

Introduction to Data Mining: Tan, Steinbach, Kumar

The Elements of Statistical Learning: Hastie, Tibshirani, Friedman

https://www4.rgu.ac.uk/files/chapter3%20-%20bp.pdf

Tutorials/Papers:

http://www.doc.ic.ac.uk/~sgc/teaching/pre2012/v231/lecture12.html

http://andrew.gibiansky.com/blog/machine-learning/machine-learning-neural-networks/

http://papers.nips.cc/paper/563-a-simple-weight-decay-can-improve-generalization.pdf

http://en.wikipedia.org/wiki/Artificial_neural_network

http://en.wikipedia.org/wiki/Deep_learning#cite_note-BOOK2014-1

http://en.wikipedia.org/wiki/Backpropagation

http://en.wikipedia.org/wiki/Sigmoid_function

http://en.wikipedia.org/wiki/Convolutional_neural_network

http://deeplearning.net/tutorial/lenet.html

References

Articles:

http://www.pcworld.com/article/2889432/google-ai-program-masters-classic-atari-video-games.html

http://spectrum.ieee.org/robotics/artificial-intelligence/machinelearning-maestro-michael-jordan-on-the-delusions-of-big-data-and-other-huge-engineering-efforts

http://www.fastcolabs.com/3026423/why-google-is-investing-in-deep-learning

http://www.nasa.gov/centers/dryden/news/NewsReleases/2003/03-49.html#.VOU2fi7cvEY

http://www.ncbi.nlm.nih.gov/pubmed/11718427

http://neuron.eng.wayne.edu/bpImageCompression9PLUS/bp9PLUS.html