**Deep Learning for Business**

Basics of Deep Learning Neural Networks

# Neural Network Learning
## (Backpropagation)

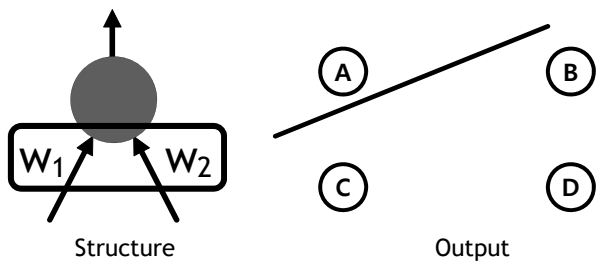**Neural Network Learning**

## NN Structure vs. Level of Intelligence

— One neuron can only make very simple
   one-dimensional decisions

— For more complex intelligence we
   need more neurons working together

# Neural Network Learning

How many neurons and in what structure?

— Weights that need to be trained

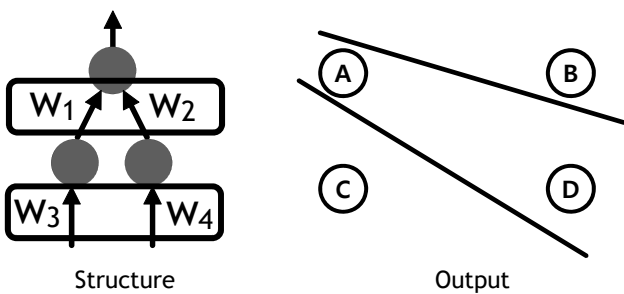- Single layer with weights: $w_1$ $w_2$



Structure          Output

Martin T. Hagan, Howard B. Demuth, Mark H. Beale, Orlando De Jesús, Neural
Network Design, 2nd Ed. ISBN-13: 978-0971732117, Martin Hagan, Sept. 1, 2014.

# Neural Network Learning

How many neurons and in what structure?

— Weights that need to be trained

- Two layers with weights: $w_1$ $w_2$ $w_3$ $w_4$
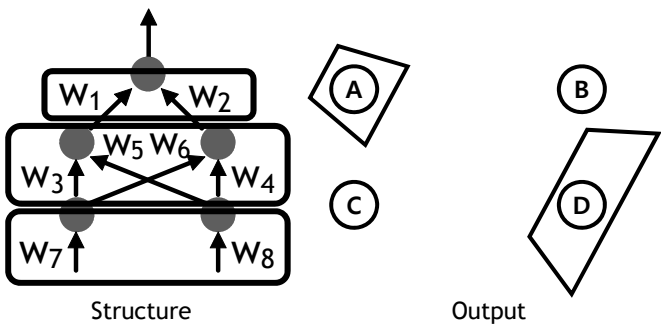


Structure          Output

## Neural Network Learning

How many neurons and in what structure?

— Weights that need to be trained

- Three layers with weights:
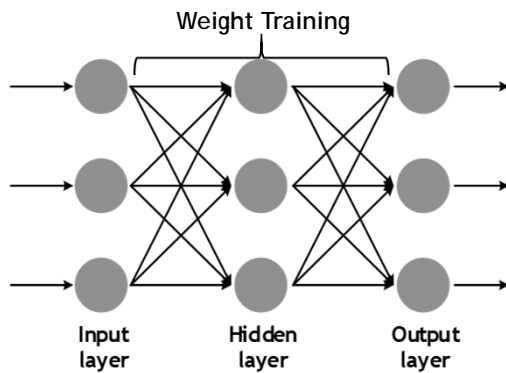
$W_1$ $W_2$ $W_3$ $W_4$ $W_5$ $W_6$ $W_7$ $W_8$
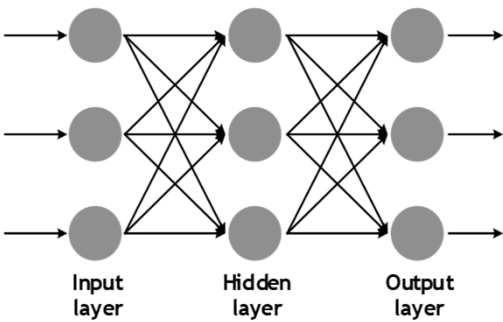


Structure          Output

## Neural Network Learning

Neural Network Layers



Input layer    Hidden layer    Output layer

## Neural Network Learning

### Neural Network Layers

Weight Training

Input
layer

Hidden
layer

Output
layer

## Neural Network Learning

### Neural Network Layers

— Input Layer
  - Layer where the input to the
    NN (Neural Network) comes in

— Output Layer
  - Layer where the output of the NN
    goes out and is used

# Neural Network Learning

## Neural Network Layers

— Hidden Layer
  - Layer that contains the intelligence in a distributed fashion using many neurons, interconnections, weights, biases, activation functions, etc.

— Multiple hidden layers can be used

— Deep NNs have many hidden layers

# Neural Network Learning

## Deep Neural Network

— Deep NNs have many hidden layers
  - How many hidden layers are needed to (qualify to) be called "Deep"?
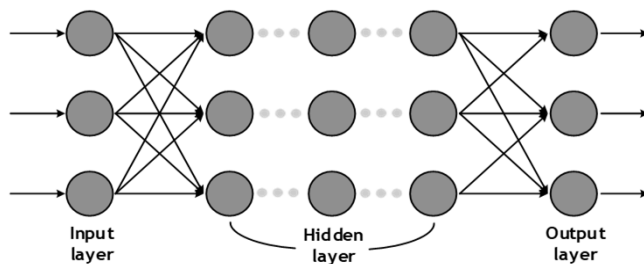
## Learning Methods

Learning Methods

— How do we train the weights of these layers to make the NN Intelligent?

— Supervised Learning
  • Training that uses labeled data (desired outputs)

— Unsupervised Learning
  • Training that uses unlabeled data (no desired outputs are used)

## Learning Methods

Learning Methods

— Semi-supervised Learning
  • Training that uses both labeled data and unlabeled data

— Reinforcement Learning
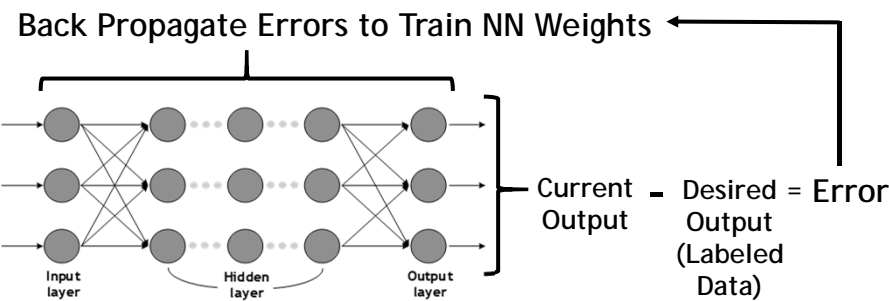  • Feedback is given back into the system, but no labeled data is used

## Learning Methods

**Supervised Learning using BP (Backpropagation) Training**

– BP is used in to train perceptrons and MLPs

– BP uses training iterations where the
Error size as well as the Variation Direction
and Speed are used to determine the
update value of each weight of the NN

## Learning Methods

**Supervised Learning using BP (Backpropagation) Training**

Back Propagate Errors to Train NN Weights

Current – Desired = Error
Output    Output
(Labeled
Data)

Input layer    Hidden layer    Output layer

Learning Methods

## Supervised Learning using BP Training

- Since each weight of the NN is used in the calculation of various other inputs, the size of a weight update has to be small in each training iteration

- This is because a big change in one weight may mess up the weights that were trained to match the other input to output values
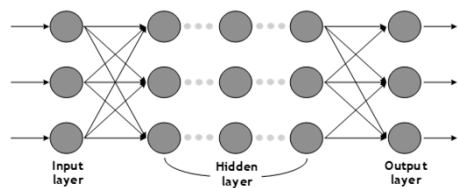
Learning Methods

## Backpropagation tool ➜ Gradient

- Gradient is the derivative of a multi-variable (vector) function

- Gradient points in the direction of the greatest rate of increase of the multi-variable function

- Magnitude of the Gradient represents the slope (rate of change) in that direction

## Learning Methods

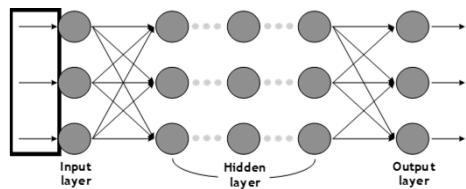## Backpropagation Learning Algorithm

1. Forward propagate the training pattern's input through the NN



## Learning Methods

## Backpropagation Learning Algorithm

1. Forward propagate the training pattern's input through the NN

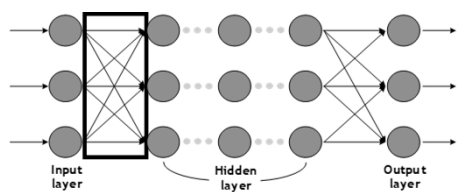## Learning Methods

## Backpropagation Learning Algorithm

1. Forward propagate the training pattern's input through the NN



## Learning Methods

## Backpropagation Learning Algorithm

1. Forward propagate the training pattern's input through the NN

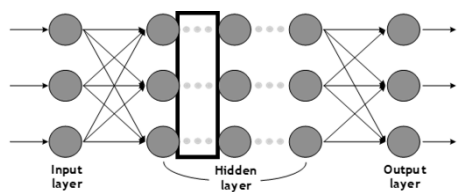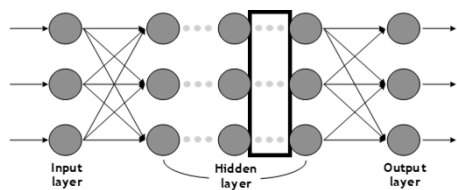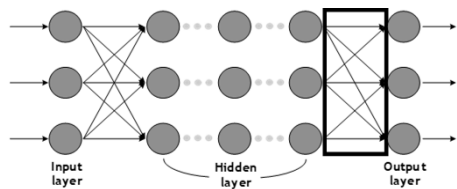## Learning Methods

## Backpropagation Learning Algorithm

1. Forward propagate the training pattern's input through the NN



## Learning Methods

## Backpropagation Learning Algorithm

1. Forward propagate the training pattern's input through the NN



2. NN generates initial output values

### Learning Methods

## Backpropagation Learning Algorithm

1. Forward propagate the training pattern's input through the NN



Input layer    Hidden layer    Output layer
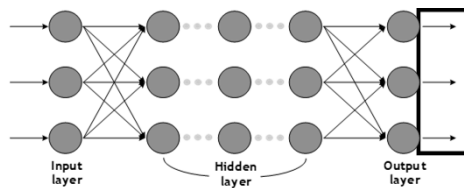
2. NN generates initial output values

### Learning Methods

## Backpropagation Learning Algorithm

3. Use the Difference value and Input value to derive the Gradient of the Weights of the output layer and hidden layer neurons

4. Scale down the Gradient of the Weights (i.e., reduce the Learning Rate)
   - Learning Rate determines the learning speed and resolution

Learning Methods

## Backpropagation Learning Algorithm

5. Update weights in the opposite direction of the +/- sign of the Gradient
   - Because the +/- sign of the Gradient indicates +/- direction of the Error

6. Repeat all steps until the desired Input-to-Output performance is satisfactory

## Deep Learning for Business

Basics of Deep Learning Neural Networks

# References

# References

- I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. book in preparation, MIT Press, www.deeplearningbook.org, 2016.

- D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel & S. Dieleman, "Mastering the game of Go with deep neural networks and tree search," Nature, vol. 529, no. 7587, pp. 484-489, 28 Jan. 2016.

- Jared Hillam, "What is Hadoop?: SQL Comparison," YouTube, https://www.youtube.com/watch?v=MfF750YVDxM

- Wikipedia, http://www.wikipedia.org

- Nvidia, https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/

# References

## Image sources

- Neuron
  By Dhp1080, svg adaptation by Actam (Image:Neuron.svg) [CC BY-SA 3.0 (http://creativecommons.org/licenses/by-sa/3.0) or GFDL (http://www.gnu.org/copyleft/fdl.html)], via Wikimedia Commons

- By Nicolas.Rougier (Own work) [GFDL (http://www.gnu.org/copyleft/fdl.html), CC-BY-SA-3.0 (http://creativecommons.org/licenses/by-sa/3.0/) or CC BY-SA 2.5-2.0-1.0 (http://creativecommons.org/licenses/by-sa/2.5-2.0-1.0)], via Wikimedia Commons

- By Copyright (c) 2003 Nicolas P. Rougier (Nicolas P. Rougier) [GPL (http://www.gnu.org/licenses/gpl.html)], via Wikimedia Commons

## References

### Image sources

- By Donarreiskoffer (Self-photographed) [GFDL (http://www.gnu.org/copyleft/fdl.html) or CC-BY-SA-3.0 (http://creativecommons.org/licenses/by-sa/3.0/)], via Wikimedia Commonsps://commons.wikimedia.org/w/index.php?curid=43383

- By David Lapetina (Own work) [GFDL (http://www.gnu.org/copyleft/fdl.html) or CC BY-SA 3.0 (http://creativecommons.org/licenses/by-sa/3.0)], via Wikimedia Commons