Classifying Cats and Dogs

An Introduction to Neural Networks and Auto-Encoders

Adam Oken[1]

Department of Mathamatics, Whitman College

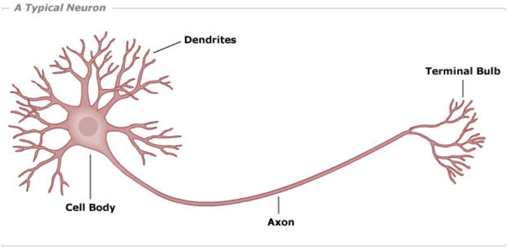
Senior Projects, April 2017

Table of Contents

The Applications

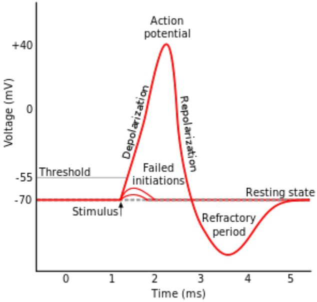
- Stock market prediction
- Pattern recognition
- Data recognition
- A bunch of other things

A Biological View of a Neuron



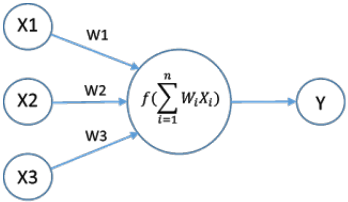
- Neurons are made up of a cell body, axon, and dendrites
- Dendrites propagate an electrochemical signal to the cell body
- The cell body sums the signal and can send an action potential down the axon
- The axon transmits information to neurons, muscles and glands.

Summing and Activation Function



- The signals from the dendrites are summed
- After the summation, the signal is passed through an activation function
- The activation function corresponds to the activation state of the node

A Neuron Through a Mathematical Perspective



- The model for one neuron
- We can think of a neural network as many individual neurons. Thus by knowing how to model one, we can see how a neural network functions.

Network Architecture

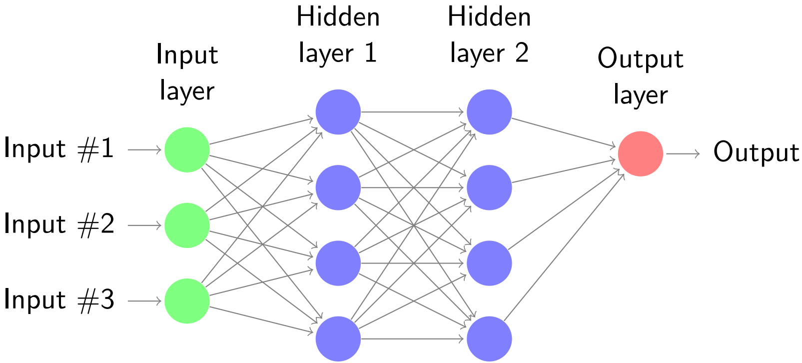
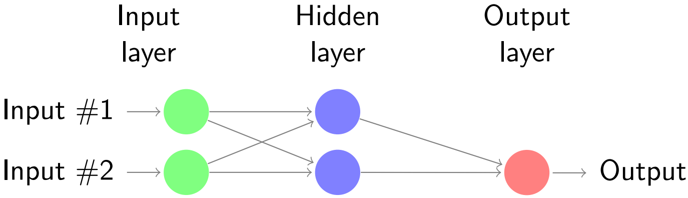


Figure: A 3-4-4-1 neural network

Overview of Feed-Forward Neural Nets



- A feed-forward neural network creates a mapping from $\mathbb{R}^n \rightarrow \mathbb{R}^m$
- They are considered supervised learning
- Our goal is to minimize the error between the outputs of the network and the targets

Calculating the Output to a Feed-Forward Neural Net

Relationship between the states of two adjacent layers

$$S_{i-1} \to P_i = W_i S_{i-1} + \mathrm{b}_i \to S_i = \sigma(P_i)$$

where b_i corresponds to the vector of biases for layer i, W_i is the matrix corresponding to all the weights for layer i, and $S_0 = \mathrm{x}$, the input layer.

- This is the same strategy for calculating the output for one node as we did previously

Generating a Function

- We can write a neural network as function by considering each layer a composition of the one previous.
- This is visualized by:

$$\mathrm{x} \to P_1 = W_1 x + \mathrm{b}_1 \to S_1 = \sigma(P_1)$$
$$S_1 \to P_2 = W_2 S_1 + \mathrm{b}_2 \to S_2 = \sigma(P_2)$$
$$S_2 \to P_3 = W_3 S_2 + \mathrm{b}_3 \to S_3 = \sigma(P_3)$$
$$S_3 \to P_4 = W_4 S_3 + \mathrm{b}_4 \to S_4 = \sigma(P_4)$$
$$\vdots$$
$$S_{n-1} \to P_n = W_n S_{n-1} + b_n \to S_n = \sigma(P_n)$$

- The last state condition is the output of the neural network

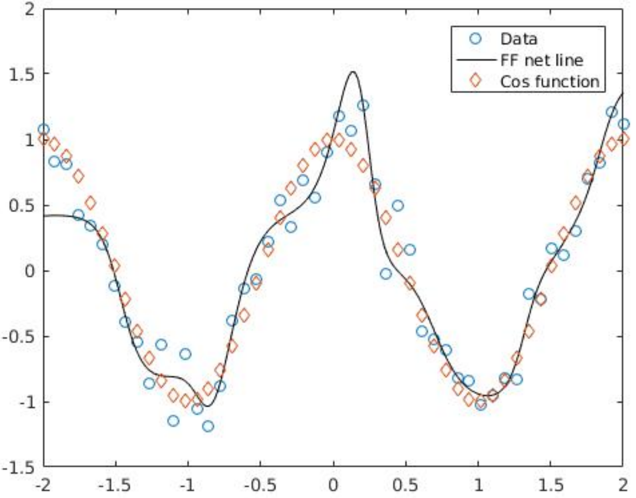
What To Do With The Output

- Our goal is to minimize the error between the output value and our given target
- Our error function will be the sum of squared error between these two terms:

$$E(W_i,\ \mathrm{b}_i) = \frac{1}{2}\sum_{i=1}^{p} ||\mathrm{t}^i - \mathrm{y}^i||^2$$

  where the error function is dependent on the weight matrices W_i and the bias vectors b_i for each layer
- We change the matrix of weights and vector of biases to decrease the error in the opposite direction of the gradient

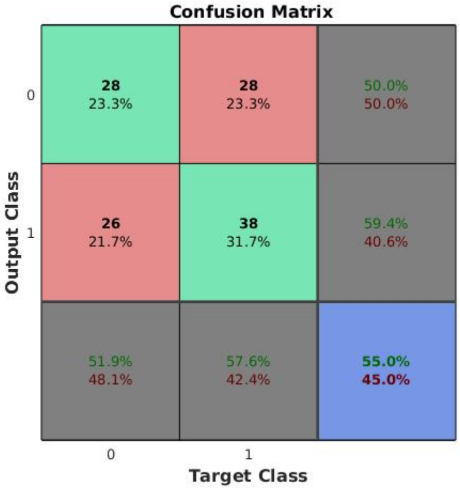
A Quick Example im Matlab



Cats and Dogs Example

- A simple feed forward neural network classifier was used to dtermine what images were cats versus dogs

The Confusion Matrix

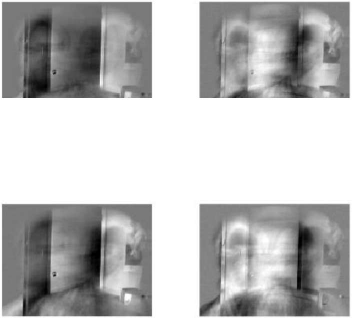


Auto-Encoders and Sparse Auto-Encoders

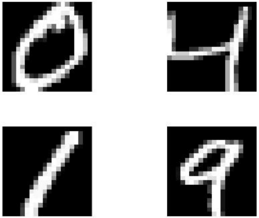
- An auto-encoder network is an unsupervised learning algorithm where the input and output layers are the same size
- The three layer network sets the input values equal to the target values
- It's essentially an identity mapping
- An auto-encoder will take the hidden layer to a smaller dimension than that of the input/output layer.
- A sparse auto-encoder will take the hidden layer to a larger dimension. auto-encoder

Auto-Encoders and the SVD

- The best k-dimension basis of a matrix given from the reduced SVD, is approximately equivalent to the encoding matrix of an auto-encoder with a k-dimensional hidden layer

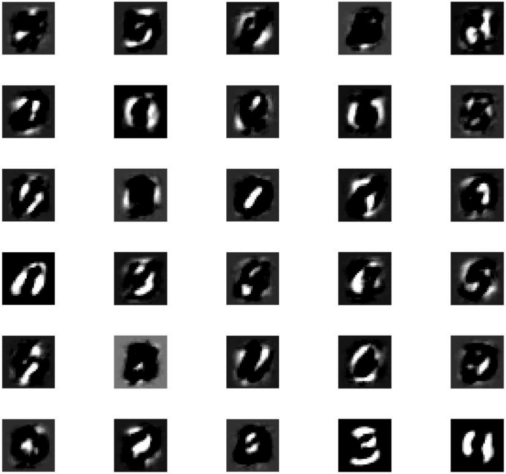


Sparse Auto-Encoders



- MNIST data set is 60,000 handwritten letters, 1/2 from high schoolers and 1/2 from United State Census Bureau employees
- Lowest error rate achieved so far is 0.21 percent on the 10,000 test images
- What we are going to see in the sparse hidden layer is specialization

Sparse Auto-Encoders Continued



Deep Learning and Deep Nets

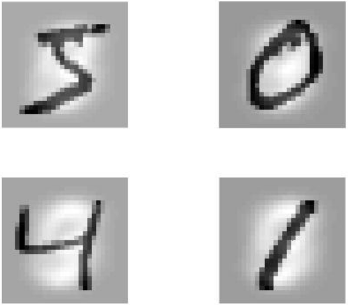
- Deep nets are stacked feed-forward, auto-encoder, and softmax layers with added parameters
- Much more powerful classification tool
- Deep nets are used in advanced technologies including artificial intelligence and self driving cars

The MNIST via a Deep Net

Confusion Matrix

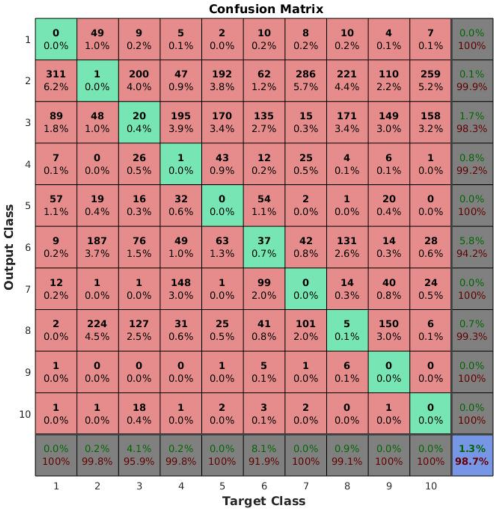
Output Class \ Target Class	1	2	3	4	5	6	7	8	9	10	
1	**338** 6.8%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**0** 0.0%	**3** 0.1%	**0** 0.0%	**0** 0.0%	**0** 0.0%	99.1% 0.9%
2	**0** 0.0%	**381** 7.6%	**0** 0.0%	**0** 0.0%	**1** 0.0%	**0** 0.0%	**0** 0.0%	**1** 0.0%	**0** 0.0%	**0** 0.0%	99.5% 0.5%
3	**2** 0.0%	**110** 2.2%	**368** 7.4%	**5** 0.1%	**2** 0.0%	**5** 0.1%	**18** 0.4%	**2** 0.0%	**0** 0.0%	**1** 0.0%	71.7% 28.3%
4	**0** 0.0%	**19** 0.4%	**7** 0.1%	**256** 5.1%	**5** 0.1%	**12** 0.2%	**0** 0.0%	**12** 0.2%	**12** 0.2%	**8** 0.2%	77.3% 22.7%
5	**3** 0.1%	**0** 0.0%	**7** 0.1%	**0** 0.0%	**385** 7.7%	**1** 0.0%	**12** 0.2%	**6** 0.1%	**1** 0.0%	**105** 2.1%	74.0% 26.0%
6	**133** 2.7%	**0** 0.0%	**24** 0.5%	**177** 3.5%	**36** 0.7%	**419** 8.4%	**22** 0.4%	**18** 0.4%	**103** 2.1%	**48** 1.0%	42.8% 57.2%
7	**6** 0.1%	**0** 0.0%	**28** 0.6%	**0** 0.0%	**7** 0.1%	**6** 0.1%	**411** 8.2%	**0** 0.0%	**0** 0.0%	**0** 0.0%	89.7% 10.3%
8	**1** 0.0%	**0** 0.0%	**1** 0.0%	**0** 0.0%	**1** 0.0%	**0** 0.0%	**0** 0.0%	**400** 8.0%	**1** 0.0%	**3** 0.1%	98.3% 1.7%
9	**5** 0.1%	**19** 0.4%	**58** 1.2%	**69** 1.4%	**36** 0.7%	**14** 0.3%	**16** 0.3%	**15** 0.3%	**374** 7.5%	**20** 0.4%	59.7% 40.3%
10	**1** 0.0%	**1** 0.0%	**0** 0.0%	**2** 0.0%	**26** 0.5%	**1** 0.0%	**0** 0.0%	**109** 2.2%	**3** 0.1%	**298** 6.0%	67.6% 32.4%
	69.1% 30.9%	71.9% 28.1%	74.6% 25.4%	50.3% 49.7%	77.2% 22.8%	91.5% 8.5%	85.3% 14.7%	71.0% 29.0%	75.7% 24.3%	61.7% 38.3%	**72.6%** **27.4%**

An Interesting Result



- A recent article was published regarding testing negative images run through a deep net trained on the same positive images.
- The result was very interesting...

The Result and Implications



Confusion Matrix

Output Class \ Target Class	1	2	3	4	5	6	7	8	9	10	
1	0 0.0%	49 1.0%	9 0.2%	5 0.1%	2 0.0%	10 0.2%	8 0.2%	10 0.2%	4 0.1%	7 0.1%	0.0% 100%
2	311 6.2%	1 0.0%	200 4.0%	47 0.9%	192 3.8%	62 1.2%	286 5.7%	221 4.4%	110 2.2%	259 5.2%	0.1% 99.9%
3	89 1.8%	48 1.0%	20 0.4%	195 3.9%	170 3.4%	135 2.7%	15 0.3%	171 3.4%	149 3.0%	158 3.2%	1.7% 98.3%
4	7 0.1%	0 0.0%	26 0.5%	1 0.0%	43 0.9%	12 0.2%	25 0.5%	4 0.1%	6 0.1%	1 0.0%	0.8% 99.2%
5	57 1.1%	19 0.4%	16 0.3%	32 0.6%	0 0.0%	54 1.1%	2 0.0%	1 0.0%	20 0.4%	0 0.0%	0.0% 100%
6	9 0.2%	187 3.7%	76 1.5%	49 1.0%	63 1.3%	37 0.7%	42 0.8%	131 2.6%	14 0.3%	28 0.6%	5.8% 94.2%
7	12 0.2%	1 0.0%	1 0.0%	148 3.0%	1 0.0%	99 2.0%	0 0.0%	14 0.3%	40 0.8%	24 0.5%	0.0% 100%
8	2 0.0%	224 4.5%	127 2.5%	31 0.6%	25 0.5%	41 0.8%	101 2.0%	5 0.1%	150 3.0%	6 0.1%	0.7% 99.3%
9	1 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.0%	5 0.1%	1 0.0%	6 0.1%	0 0.0%	0 0.0%	0.0% 100%
10	1 0.0%	1 0.0%	18 0.4%	1 0.0%	2 0.0%	3 0.1%	2 0.0%	0 0.0%	1 0.0%	0 0.0%	0.0% 100%
	0.0% 100%	0.2% 99.8%	4.1% 95.9%	0.2% 99.8%	0.0% 100%	8.1% 91.9%	0.0% 100%	0.9% 99.1%	0.0% 100%	0.0% 100%	1.3% 98.7%

Output Class / Target Class

Conclusion

- The universal approximator theorem: states that a feed-froward neural network with one single hidden layer can approximate any continuous function to an arbitrarily small error
- Deep nets are super important for applications in machine learning
- From the last example, deep nets and deep learning are still active areas of research where there is much more to learn
- Go to Google's AI site to see more advanced uses of neural networks in real life: https://aiexperiments.withgoogle.com/

Acknowledgements and References

- Senior project advisor/academic advisor: Douglas Hundley
- Senior seminar professor: Albert Schueller

Acknowledgements and References

- Senior project advisor/academic advisor: Douglas Hundley
- Senior seminar professor: Albert Schueller

- Jochen Frohlich. Supervised and unsupervised learning, jan 2017.
- Hassoun, Mohamad H. Fundamentals of Artificial Neural Networks. MIT Press, 1995.
- Douglas R. Hundley. Backproperror, jan 2017.
- Andrew Ng. sparseautoencoder, jan 2017

Questions?