How Does The Neural Networks Work?

Juanling Liu (15320171151901)*

May 19, 2019

1 Theory

We carry in our heads a supercomputer, and superbly adapted to understand the visual world. we humans are good at making sense of what our eyes show us. nearly all that work is done unconsciously. While, In order to recognize some handwritten digits, Neural networks take a large number of handwritten digits, known as training examples, and then develop a system which can learn from those training examples. In other words, the neural network uses the examples to automatically infer rules for recognizing handwritten digits. Furthermore, by increasing the number of training examples, the network can learn more about handwriting, and so improve its accuracy. Perceptron is a type of artificial neuron. A perceptron takes several binary inputs x_1,x_2,..., and produces a single binary output 0 or 1. It can weigh up different kinds of evidence in order to make decisions.

*Email: 854230829@qq.com

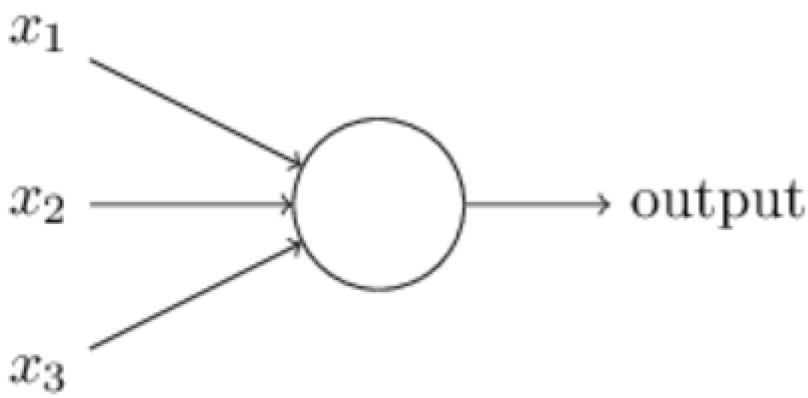

Figure 1: A perceptron

A many-layer network of perceptrons can engage in sophisticated decision making. Each of those perceptrons is making a decision by weighing up the results from the previous layer of decision-making.such multiple layer networks are sometimes called multilayer perceptrons or MLPs. The leftmost layer in this network is called the input layer, and the neurons within the layer are called input neurons. The rightmost or output layer contains the output neurons, or, as in this case, a single output neuron. The middle layer is called a hidden layer.

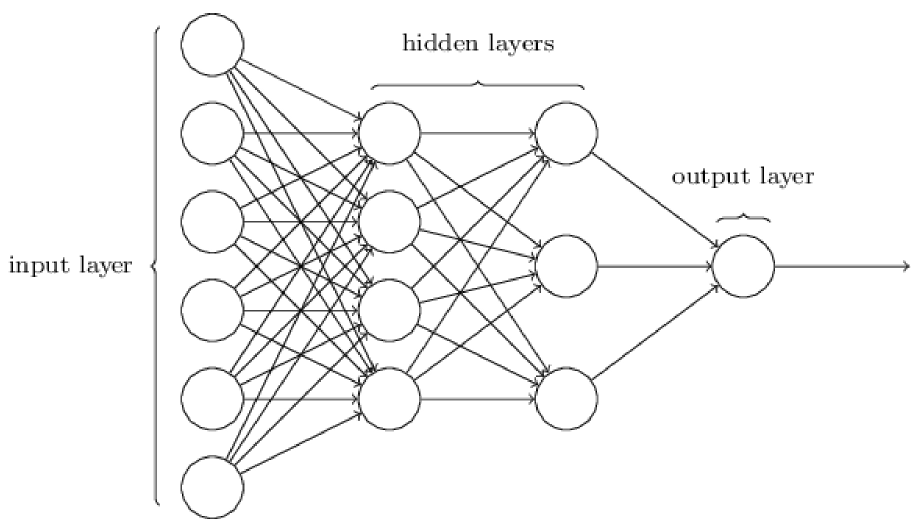

Figure 2: A three-layer network

However, a small change in the weights or bias of any single perceptron in the network can sometimes cause the output of that perceptron to completely flip, say from 0 to 1, then cause the behaviour of the rest of the network. In much modern work on neural networks, the main neuron model used is one called the sigmoid neuron. Sigmoid neurons are similar to perceptrons, but modified so that small changes in their weights and bias cause only a small change in their output. Just like a perceptron, the sigmoid neuron has inputs, but instead of being just 0 or 1, these inputs can also take on any values between 0 and 1, it's σ(w*x+b), where w is weight for each input x, b is an overall bias, σ is called the sigmoid function, defined by: $\sigma(z) \equiv \frac{1}{(1+e^{-z})}$. It's a smoothed out version of a step function.

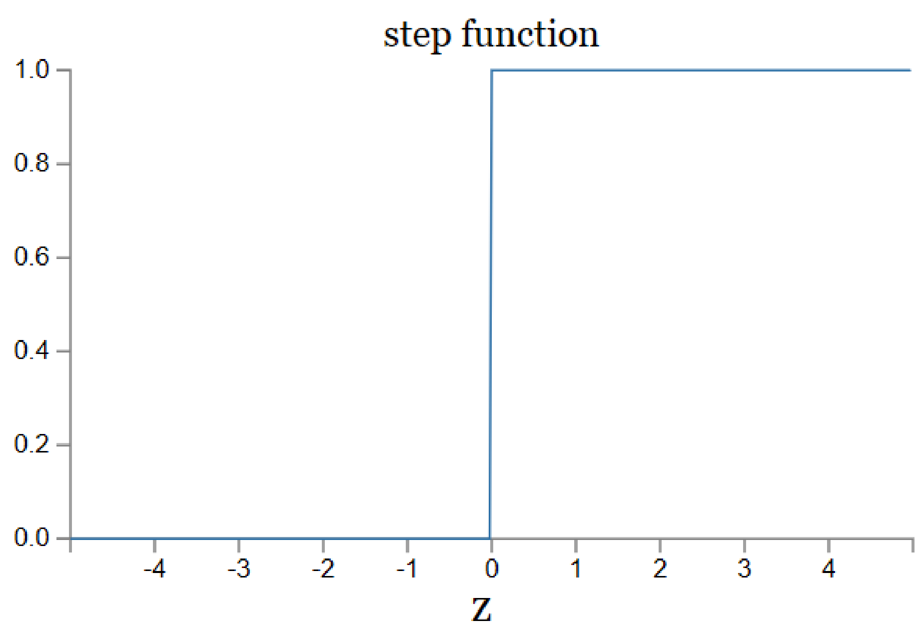

Figure 3: step function

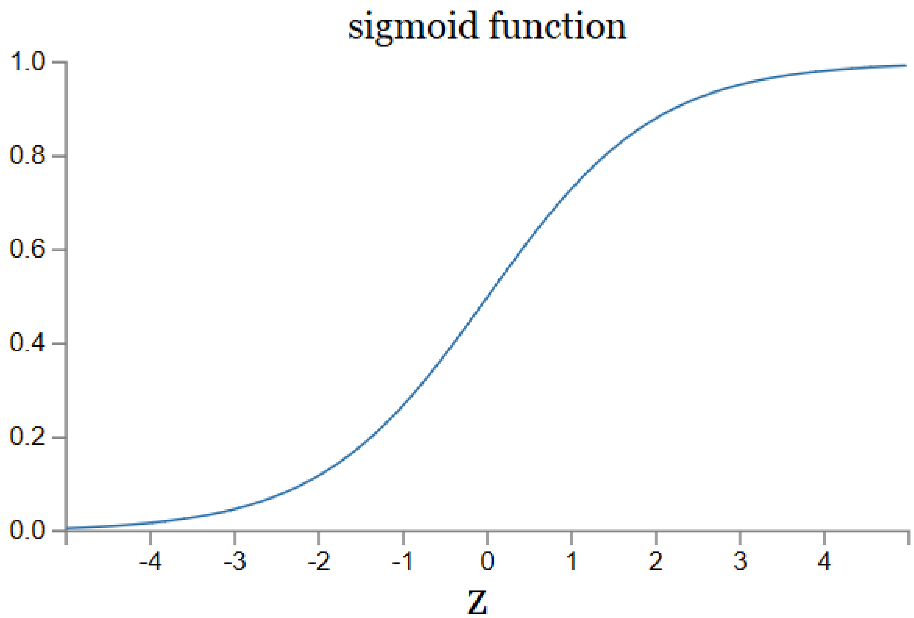

Figure 4: sigmoid function

2 Practice

Let's apply neural networks to the example of Boston Housing Price in class.

First,we choose the weight decay parameter by cross-validation

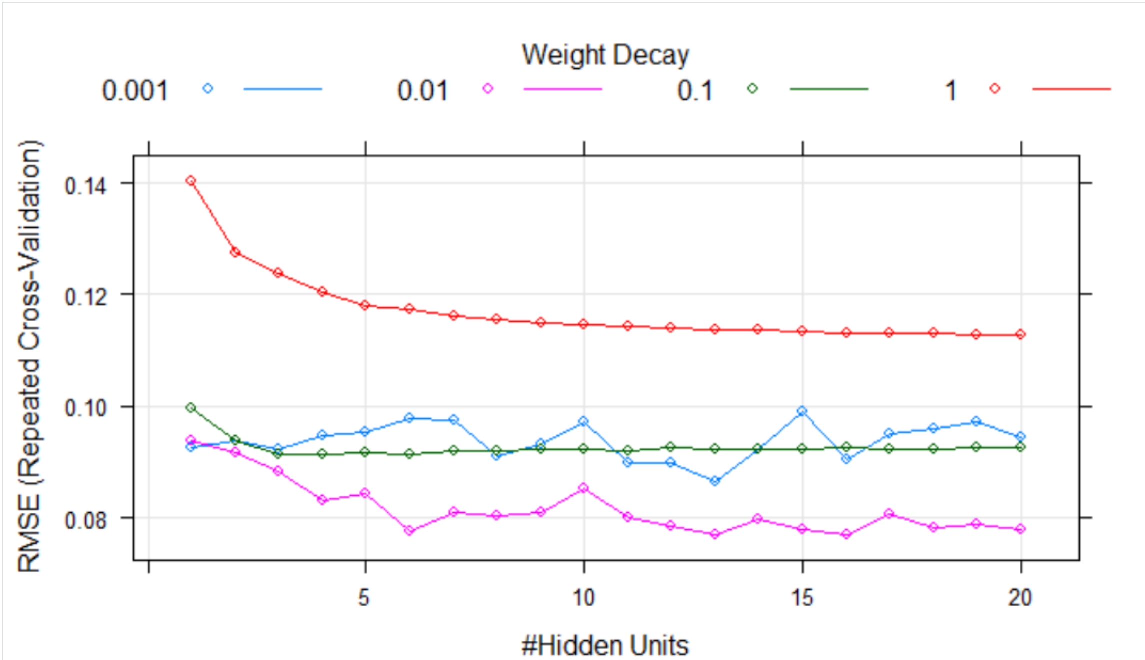

Figure 5: the best weight decay

```
> fit$bestTune
   size decay
50   13  0.01
```

Here we fit a single-hidden-layer network with 13 hidden units

Then we calculat the test error

```
> mean((ytrue-yhat)^2)
[1] 0.006424799
```

Compare with other methods, we can see the neural networks has the smallest MSE.

	$LinearReg$	$RegressionTree$	$Bagging$	$RandomForest$	$NeuralNetworks$
MSE	21.89855	17.34199	10.99368	8.78830	0.00642

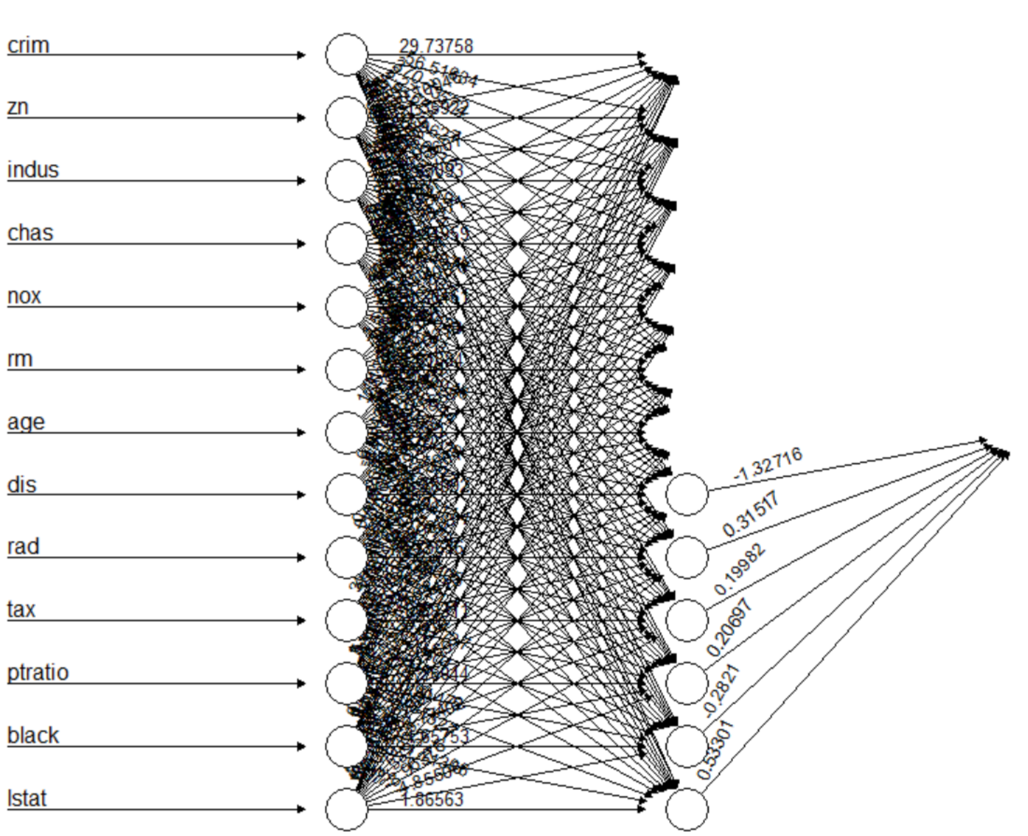

Figure 6: Neural Networks

3 References

Nielsen, Neural Networks and Deep Learning