Neural Network and Deep Learning

Qingyun Li

VISION@OUC

July 9, 2018

1 First Week

1.1 Deep Learning

First, we should know what is deep learning, Wu tell us the term deep learning refers to training neural networks and sometimes it is very large networks.

1.2 Neural Network

We start with a housing price prediction example. We know the size of the house and the price of the house. And we want to fit a function to predict the price of the house as a function of the size. And we can think of this function as a neural network, and this nerual network is very simple, we can think it as a single neuron. All the neuron does is input the size, computes the linear function, takes max of zero, and then outputs the estimated price.

In addition, this function which goes at zero for some time and then takes off as a straight line. It is called ReLU function which stands for rectified linear unit.

So this is a single neural network and a tiny little neural network, a larger neural network is formed by taking many of these single neurons and stacking them together. Instead of predicting the price of the house just from the size, we have other features, such as the number of bedrooms and this two features decide whether or not a house can fit your family's family size. And the zip code or wealth always effect the price of the house. And according to this features, we can build a neural network, as is shown at Figure. 1. And the first row represented by x, which called input layer and the y is called output layer. If given enough data about x and y, given enough training examples with bith x and y, neural networks are remarkably good at figuring out functions that accurately map from x to y.

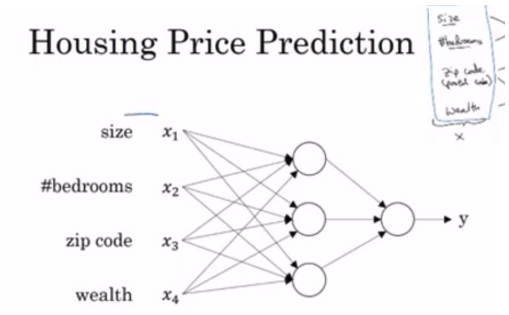

Figure 1: A Neural Network

1.3 Supervised Learning

It turns out that so far, almost all the economic value created by neural networks has been through one of machine learing, called supervised learning. In supervised learning, we have some input x and we want to learn a function mapping to some output y. And the supervised learning have many applications, such as online advertising, photo tagging, speech recognition, machine translation and autonomous driving. For real estate or online advertising, we always use the universally neural network architecture. For image applications we'll often use convolutional neural networks often abbreviated CNN. For sequence data, for example, audio has a temporal component and it is played out over time, so audio is most naturally represented as a one-dimensional time series. So for sequence data, we often use an RNN, a recurrent neural network. And language, English and Chinese, the alphabets or the words come one at a time, so language is also most naturally represented as sequence data. We can see the three neural networks at the Figure. 2.

Machine learning is always apply to both Structured Data and Unstructured Data. Structure data means basically database of data and each of the features have a very well defined meaning. In contrast, unstructured data refers to things like audio, raw audio, or images where you might want to recognize

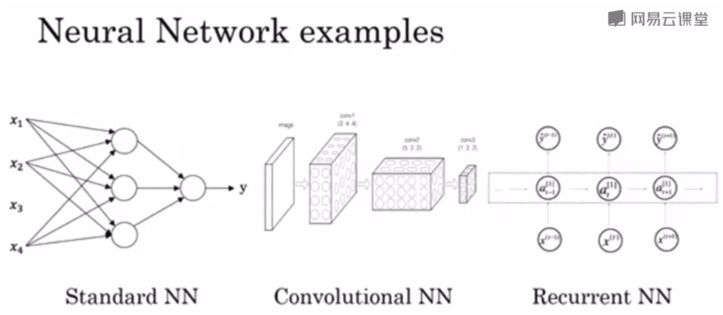

Figure 2: Three Neural Networks

what's in the image or text. And here the features might be the pixel values in an image or the individual words in a piece of text.

Historically, it has been much harder for computers to make sense of unstructured data compared to structured data. But now, people just really good at interpreting unstructured data. Due to the development of deep learning and neural network, computers are now much better at interpreting unstructured data.

1.4 Why is deep learning taking off?

Deep learning is taking off due to a large amount of data available through the digitization of the society, faster computation and innovation in the development of neural network algorithm. And the development of the hardware, whatever CPU or GPU, made the speed of computation faster, so we can train very large neural netwoks and enabled us to make a lot of progress. And in the last several years, we've seen tremendous algorithmic innovation as well. And interestingly many of the algorithmic innovations have been about trying to make neural networks run much faster. And so faster computation has really helped in terms of speeding up the rate at which you can get an experimental result back.

2 Second Week

2.1 Binary Classification

We have a example of a binary classification problem, an image is an input and we output a label to recognize this image as being a cat, in which case we output 1, or not-cat which case you output 0, we're going to use y to denote the output label.

2.2 Logistic Regression

This is a learning algorithm that we use when the output labels y in a supervised learning problem are all either zero or one, so far binary classification problems.

Given x, we want

$$\hat{y} = \sigma(w^T x + b) \tag{1}$$

where $w \in R^{n_x}$, $b \in R$, $\sigma(z) = \frac{1}{1+e^{-z}}$.

When we implement logistic regression, our job is to try to learn parameters w and b, so that $\hat{y}$ becomes a good estimate.

2.3 Logistic Regression cost function

First, we defined a loss function to measure how good our output $\hat{y}$ is when the true label is y

$$L(\hat{y}, y) = \frac{(\hat{y} - y)^2}{2} \tag{2}$$

But it makes gradient descent not work well, so we define a new loss function as follows:

$$L(\hat{y}, y) = -(ylog\hat{y} + (1-y)log(1-\hat{y})) \tag{3}$$

The loss function was defined with respect to a single training example, it measures how well you're doing on a single training example.

The cost function measures how well you're doing an entire training set:

$$J(w, b) = \frac{1}{m}\sum_{i-1}^{m} L(\hat{y}^{(i)}, y^{(i)}) \tag{4}$$

The cost function is the cost of our parameters, so in training logistic regression model, we're going to try to find parameters w and b that minimize the overall cost function J written at the bottom.

2.4 Gradient Descent

In the last segment, we want to find w, b that minimize J(w,b).Here is an illustration of gradient descent, as is shown at Figure. 3, in this diagram the horizontal axes represent spatial parameters w and b. And in practice, w can be much higher dimensional, but for the purpose of plotting, we illustrate w as a single real number and be as a single real number. It turns out that this cost function J is a convex function, and this is one of the huge reasons why we use this particular cost function J for logistic regression. So to find a good value for the parameters what we'll do is initialize w and b to some initial value.

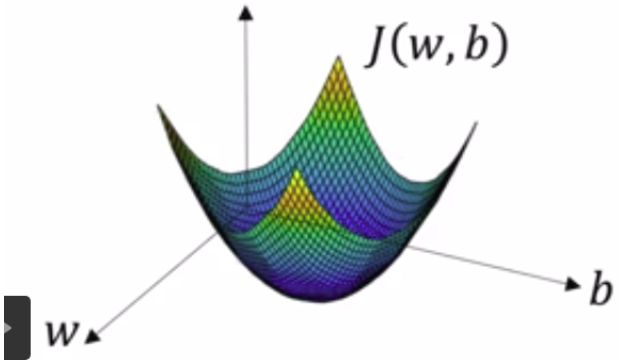

Figure 3: Cost Function