An Introductory Guide to Deep Learning and Neural Networks

Introduction

Having a solid grasp on deep learning techniques feels like acquiring a super power these days. From classifying images and translating languages to building a self-driving car, all these tasks are being driven by computers rather than manual human effort. Deep learning has penetrated into multiple and diverse industries, and it continues to break new ground on an almost weekly basis.

Introduction to Deep Learning

The objectives behind this module are below:

- Understand the major trends driving the rise of deep learning
- Be able to explain how deep learning is applied to supervised learning
- Understand what are the major categories of models (such as CNNs and RNNs), and when they should be applied
- Be able to recognize the basics of when deep learning will (or will not) work well
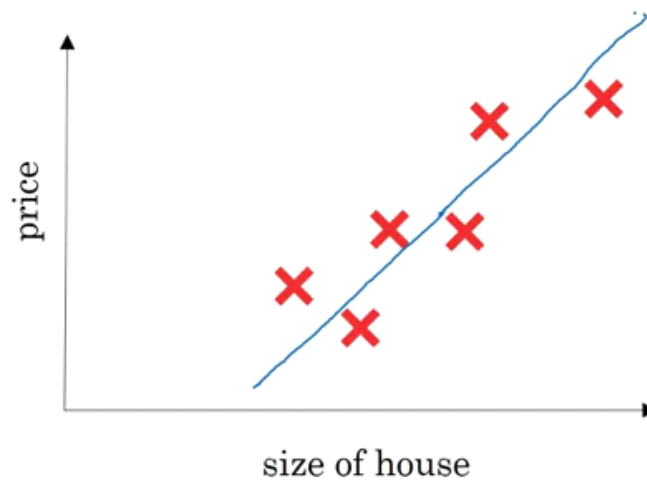
What is a Neural Network?

Let's begin with the crux of the matter and a very critical question. What is a neural network?

Consider an example where we have to predict the price of a house. The variables we are given are the size of the house in square feet (or square meters) and the price of the house. Now assume we have 6 houses. So first let's pull up a plot to visualize what we're looking at:
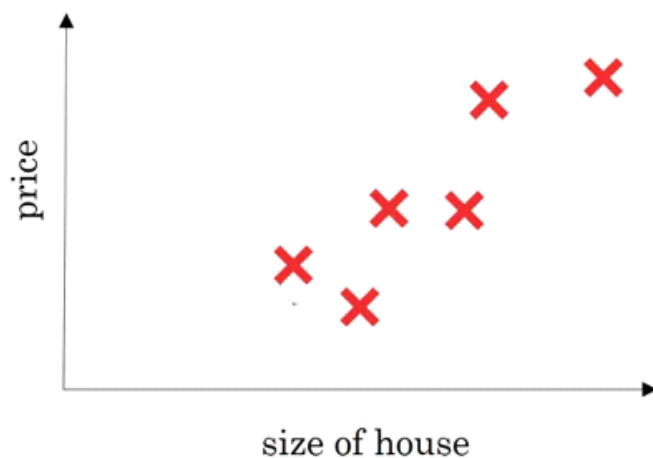
On the x-axis, we have the size of the house and on the y-axis we have it's corresponding price. A linear regression model will try to draw a straight line to fit the data:

size of house

So, the input(x) here is the size of the house and output(y) is the price. Now let's look at how we can solve this using a simple neural network:
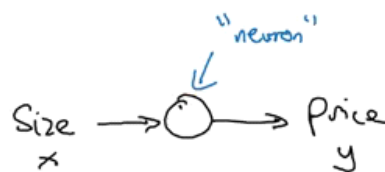

Here, a neuron will take an input, apply some activation function to it, and generate an output. One of the most commonly used activation function is ReLU (Rectified Linear Unit):


ReLU

$$R(z) = max(0,\ z)$$
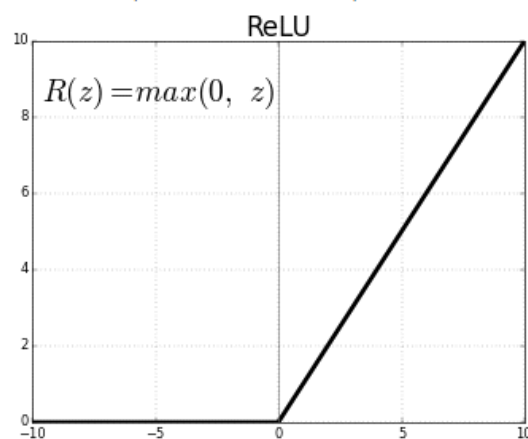
ReLU takes a real number as input and returns the maximum of 0 or that number. So, if we pass 10, the output will be 10, and if the input is -10, the output will be 0. We will discuss activation functions in detail later in this article.

For now let's stick to our example. If we use the ReLU activation function to predict the price of a house based on its size, this is how the predictions may look:
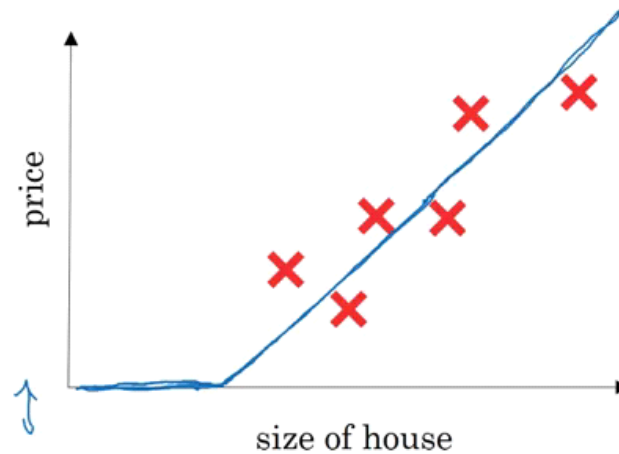
So far, we have seen a neural network with a single neuron, i.e., we only had one feature (size of the house) to predict the house price. But in reality, we'll have to consider multiple features like number of bedrooms, postal code, etc.? House price can also depend on the family size, neighbourhood location or school quality. How can we define a neural network in such cases?
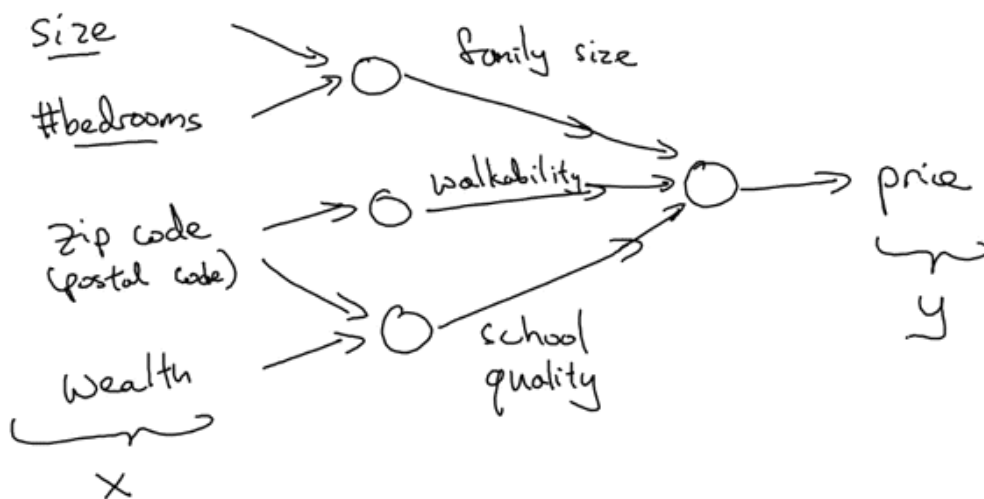
It gets a bit complicated here. Refer to the above image as you read – we pass 4 features as input to the neural network as x, it automatically identifies some hidden features from the input, and finally generates the output y. This is how a neural network with 4 inputs and an output with single hidden layer will look like:
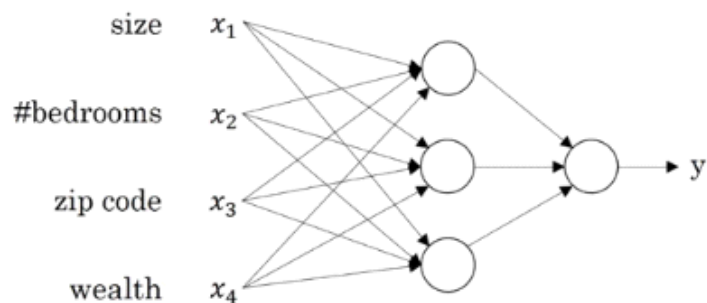

Now that we have an intuition of what neural networks are, let's see how we can use them for supervised learning problems.

Supervised Learning with Neural Networks

Supervised learning refers to a task where we need to find a function that can map input to corresponding outputs (given a set of input-output pairs). We have a defined output for each given input and we train the model on these examples. Below is a pretty handy table that looks at the different applications of supervised learning and the different types of neural networks that can be used to solve those problems:

Input (X)	Output (y)	Application	Type of Neural Network
Home Features	Price	Real Estate	Standard Neural Network
Ad, user info	Click prediction (0/1)	Online Advertising	Standard Neural Network
Image	Image Class	Photo Tagging	CNN
Audio	Text Transcript	Speech Recognition	RNN
English	Chinese	Machine Translation	RNN
Image, Radar info	Position of car	Autonomous Driving	Custom / Hybrid NN

Below is a visual representation of the most common Neural Network types:

Standard NN   Convolutional NN   Recurrent NN
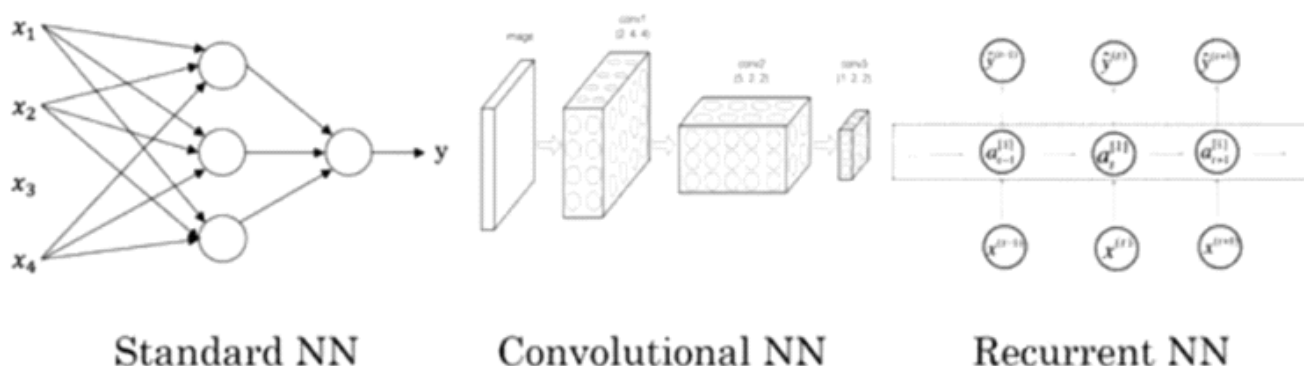
In this article, we will be focusing on the standard neural networks only.

As you might be aware, supervised learning can be used on both structured and unstructured data.

In our house price prediction example, the given data tells us the size and the number of bedrooms. This is **structured data**, meaning that each feature, such as the size of the house, the number of bedrooms, etc. has a very well defined meaning.
In contrast, **unstructured data** refers to things like audio, raw audio, or images where you might want to recognize what's in the image or text (like object detection). Here, the features might be the pixel values in an image, or the individual words in a piece of text. It's not really clear what each pixel of the image represents and therefore this falls under the unstructured data umbrella.

Simple machine learning algorithms work well with structured data. But when it comes to unstructured data, their performance tends to take quite a dip. This is where neural networks have proven to be so effective and useful. They perform exceptionally well on unstructured data. Most of the ground-breaking research these days has neural networks at it's core.