Convolutional Neural Networks (CNN) in my understanding: //source//

When we see an image of a dog, certain neurons in our brain are stimulated, sending signals to other neurons which send signals to even more neurons, ultimately resulting in certain neurons being fired that "tell" us that we see a dog. Neural networks attempt to simulate that process, building a "mini-brain" that can complete simple tasks such as distinguishing cats from dogs.

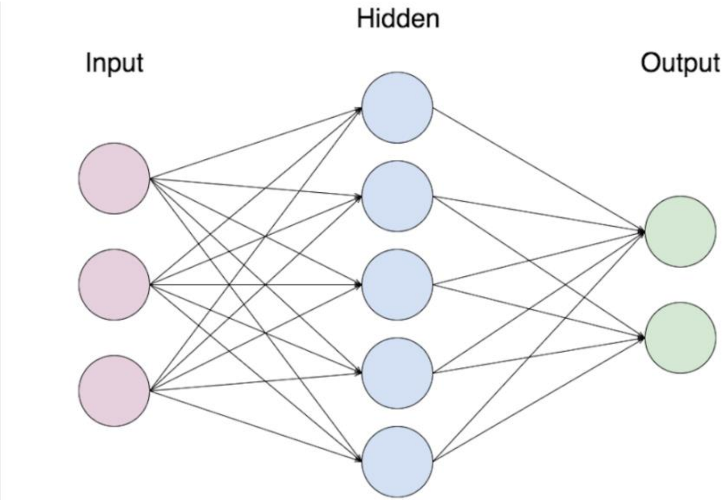
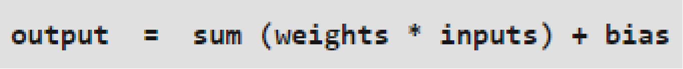


Image 1: Basic Neural Network

The most basic neural network looks something like this. We start out with an input layer of neurons, which activate neurons in the hidden layers, which then activate neurons in the output layer. Think of each circle in the diagram above as a neuron. Each neuron contains a number, knows as its activation.

Basic Components of CNN: //source//

- Weight: Coefficient numbers of the respective coordinates.
- Bias: Bias is a certain fixed value that is added to the final output to generalize the data. The addition of bias reduces the variance and hence introduces flexibility and better generalisation to the neural network.



```
output  =  sum (weights * inputs) + bias
```

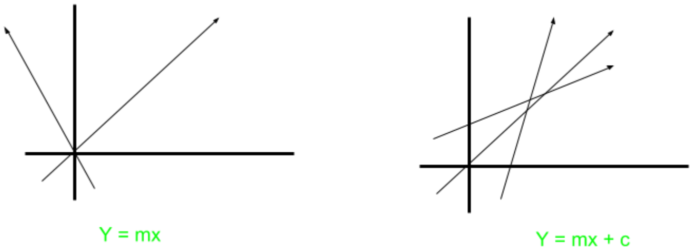


Image 2: Need of Bias

Here,

m = weight

c = bias

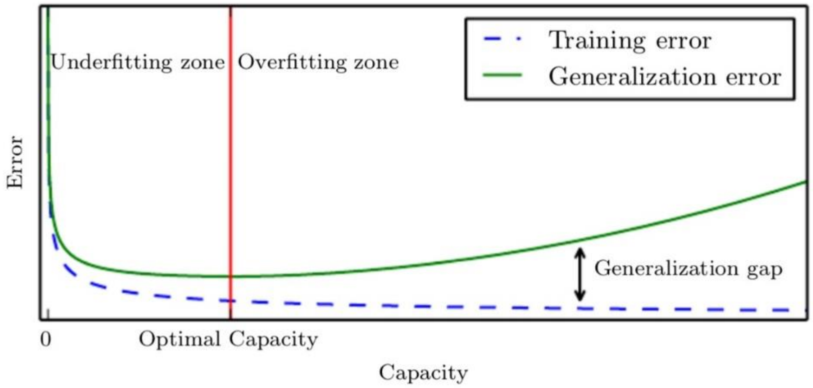


Image 2: Underfitting and Overfitting Concept

You can see that that the training error (blue dotted line) keeps on decreasing. In the initial phase, it's too high (High Bias). Later, it decreases (Low Bias).

High Bias means the model is not even fitting on the training data. So, we have to make the bias low.

- How to lower the bias?
  - ➢ Increase the epochs (iterations)
  - ➢ Try a Bigger network

❖ Variance: The Variance of a model is the difference between validation error and training error. In the figure, you can see that the gap between validation error and training error is increasing. That is, the variance is increasing (Overfitting).

- What is the importance of variance?

  Variance gives us the information about the generalization power of our model.
  If the Variance is high, the model is not performing well on the validation set. We always want a low variance.

- How to lower the variance?

  1. Increase the training set data
  2. Try Regularisation
  3. Try a different Neural Network Architecture

We always want a low bias and low variance.

- How do we take decision whether we need to higher or lower the bias/variance? Let's look for the following examples:

- Human error ~ 0%
- Train set error: 1%
- Validation set error: 11%

  Variance = Validation set error - Train set error = 11 - 1 =10%
  Bias = Train set error - Human error = 1%

Low Bias and High Variance (Overfitting). Since the Variance is greater than bias, this is a Variance problem. We have to lower the variance.

- Let's look at another example:

- Human error~ 0%
- Train set error: 15%
- Validation set error: 16%

Variance = Validation set error - Train set error = 16 – 15 = 1%

Bias = Train set error - Human error = 15%

High Bias and Low Variance (Underfitting). Since the Bias is greater than Variance, this is a Bias problem. We have to lower the Bias.

- Training Set: To fit the parameters [i.e., Weights]
- Validation Set: To tune the parameters [i.e., Architecture]
- Test Set: To assess the performance [i.e., Generalization and predictive power]

Articles I’m exploring:

1. A Newbie’s Introduction to Convolutional Neural Networks
2. Bias and Variance in Neural Network
3. How Does A Face Detection Program Work? (Using Neural Networks)
4. What Does A Face Detection Neural Network Look Like?
5. I Implemented a Face Detection Model. Here’s How I Did It.