Understanding Neural Networks by Playing Games

What are we doing today?

A bit about me

Head of Data Science

- Multi cultural and multi talented Global Data Science Team
- Appx. 30 data scientists and machine learning engineers
- Focused on building innovative products for GfK and our clients
- GfK is a leading German market research company whose DNA lies in Data Science

Agenda

- Brief overview of Neural Networks
- What are CNNs?
- Working step by step through our game
- Understanding what the model actually learnt

Intermediate steps:

- Please clone the repository at https://github.com/sidhusmart/pyconde2018.git
- Install Anaconda

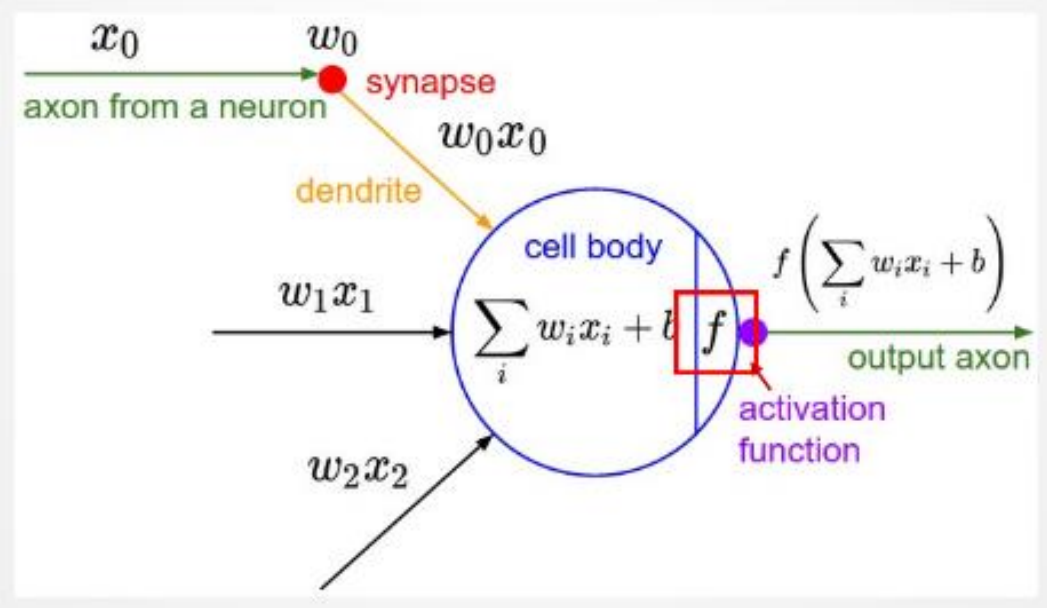
What we won’t be covering

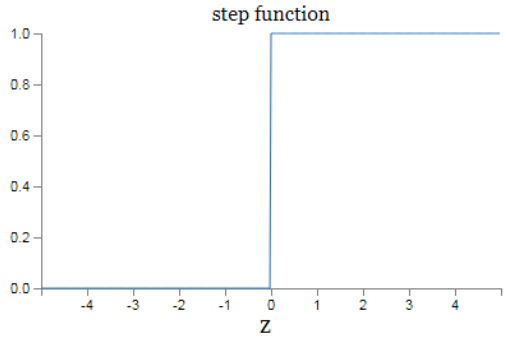
- Frameworks and their details

Introduction to Neural Networks

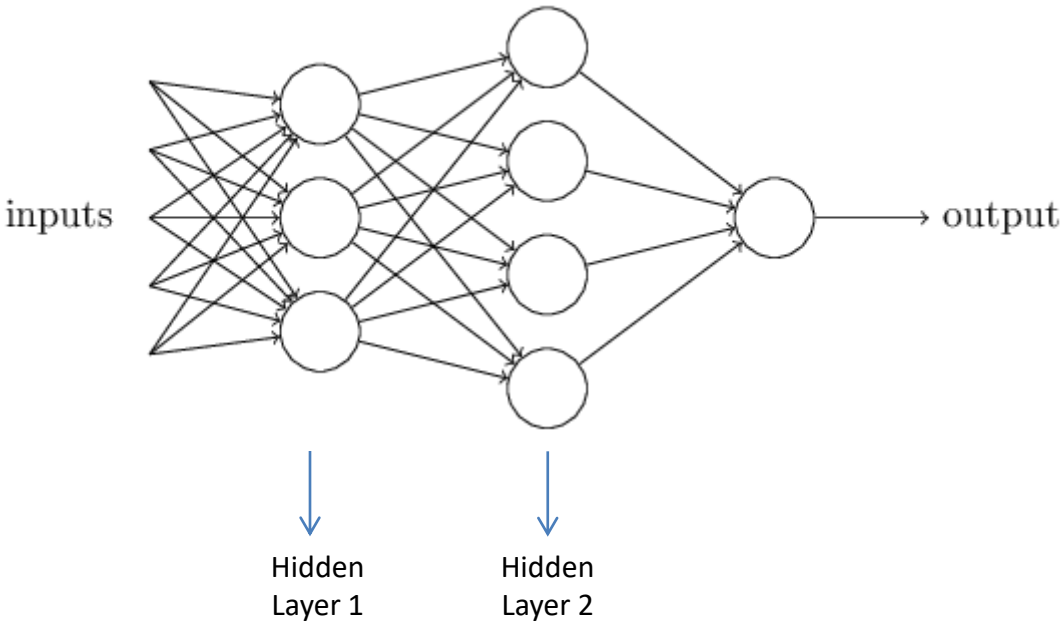
Introduction to Neural Networks

- Perceptron – a basic building block



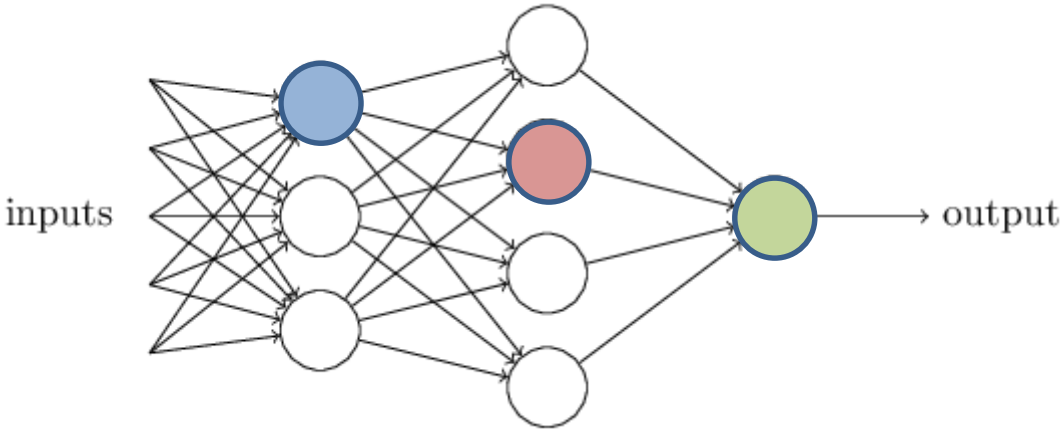


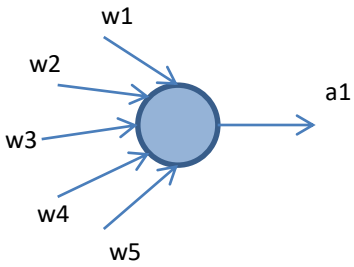
A fully connected neural network



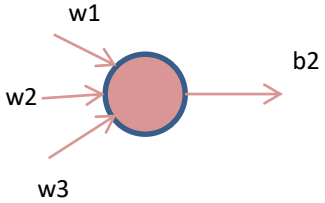
- A single layer can be created by stacking a number of perceptrons together
- Each layer that is not an input or output layer is called a hidden layer
- The output of every perceptron in a layer is used as the input for every perceptron in the next layer

What happens in a fully connected network?

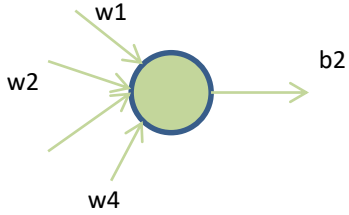




a1 = f(w1*x1 + w2*x2 + w3*x3 + w4*x4 + w5*x5)

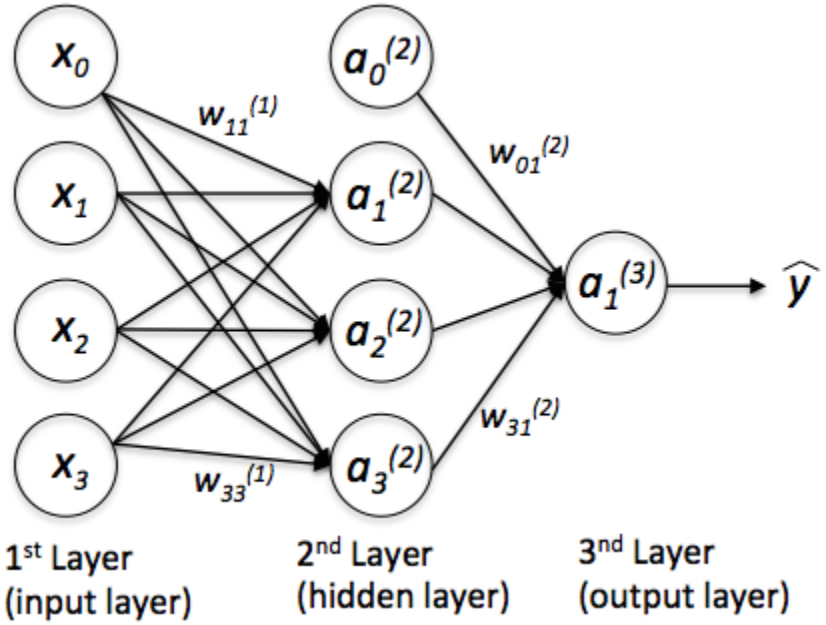


b2 = f(w1*(f(w1*x1 + w2*x2 + w3*x3 + w4*x4 + w5*x5)
+ w2*a2 + w3*a3



y = f(w1*b1 + w2*(f(w1*(f(w1*x1 + w2*x2 + w3*x3 +
w4*x4 + w5*x5) +
w2*a2 + w3*a3)) + w3*b3 + w4*b4

1. Feed Forward



Process of traversing through the network starting from the input, calculating outputs and applying activation function for each neuron right till the output layer

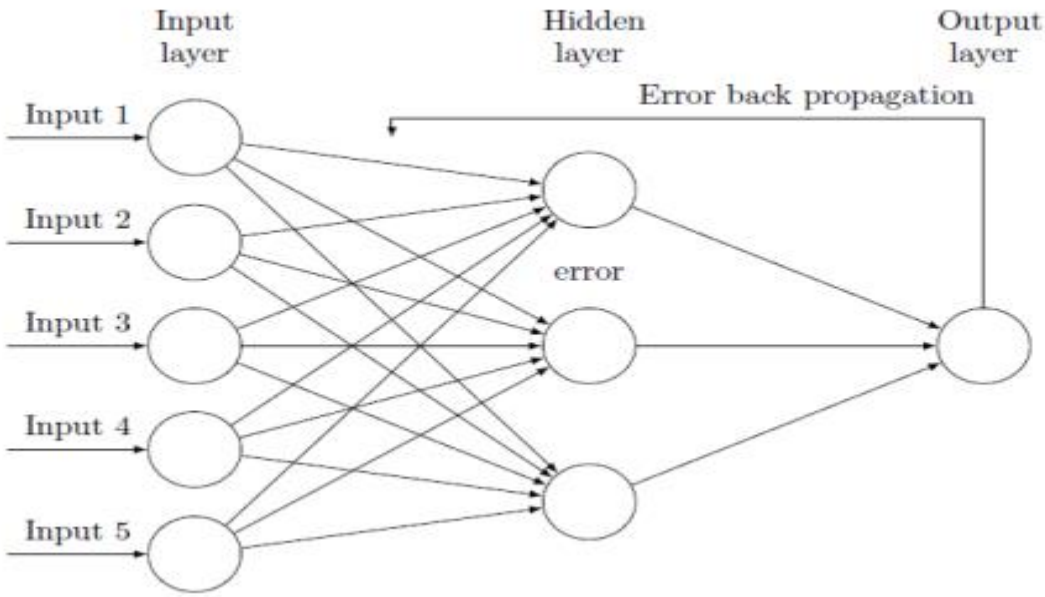
2. Error and Cost Function

Cost Function	• Define a cost function that quantifies error between the expected output and the calculated output • Measure of deviation from the expected solution • Modify parameter values to minimize error i.e. cost function • Examples include quadratic error, cross-entropy error etc.

3. Backpropagation

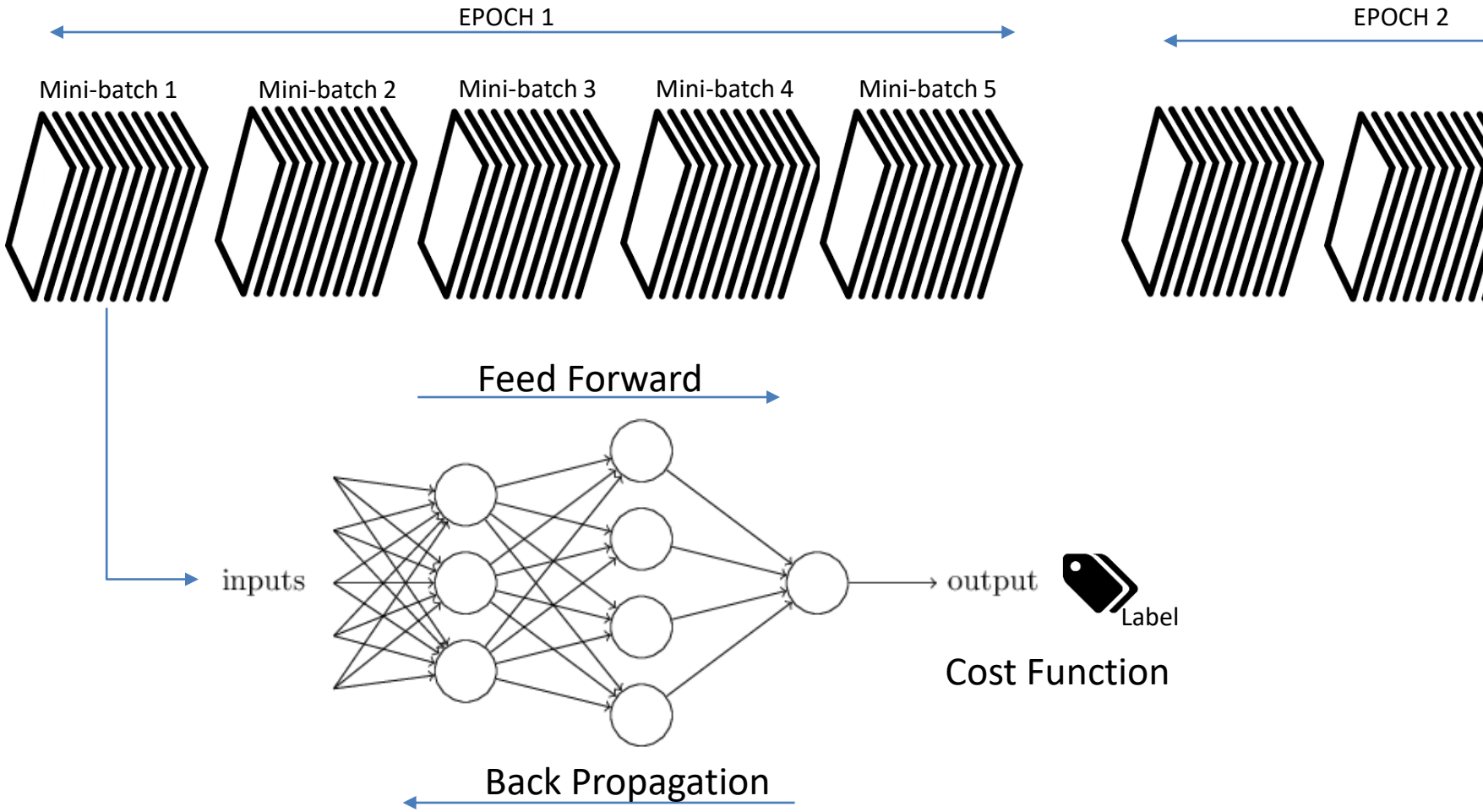
TECHNIQUE THAT ALLOWS NEURAL NETWORKS LEARN PARAMETER VALUES USING GRADIENT DESCENT

Backpropagate this error in reverse through the network to obtain a gradient with respect to each weight and bias parameter at each neuron



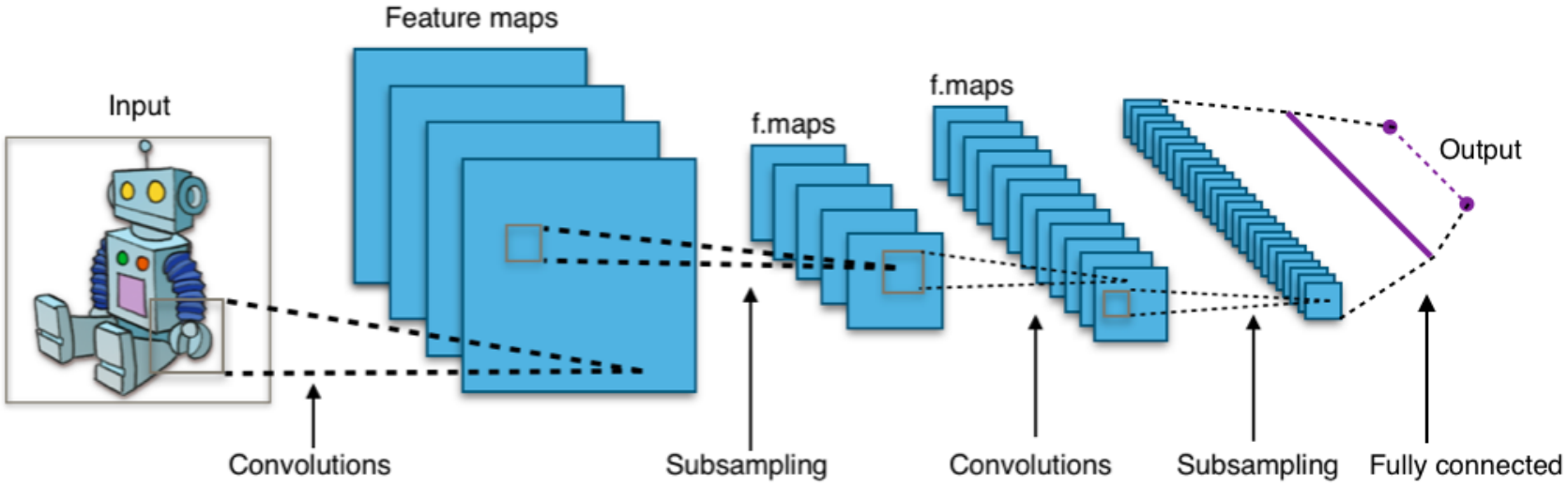
The error at each neuron is a factor of error at previous neurons – using this and some standard equations, we can arrive at the gradient descent for each parameter and adjust those values for the next pass.

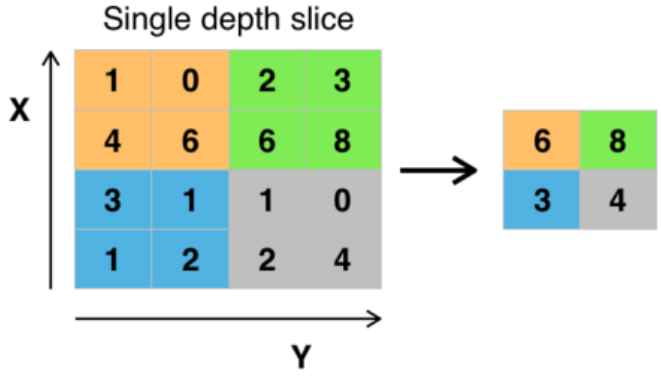
4. Training & Optimization



What are CNNs?

Architecture of CNN





Sub-sampling is normally achieved by Max pooling

Source - Wikipedia

Let’s get some intuition what it does

Beautiful visualization of what a convolutional layer does -

https://www.youtube.com/watch?v=Oqm9vsf_hvU&feature=youtu.be

Get it to work!

- Jupyter Notebook