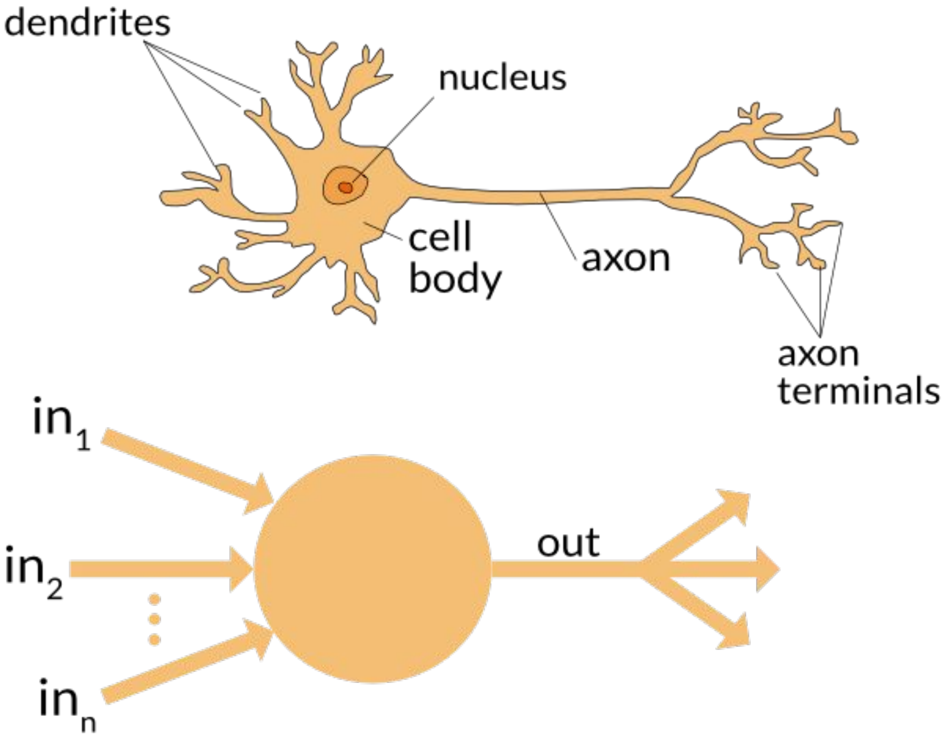
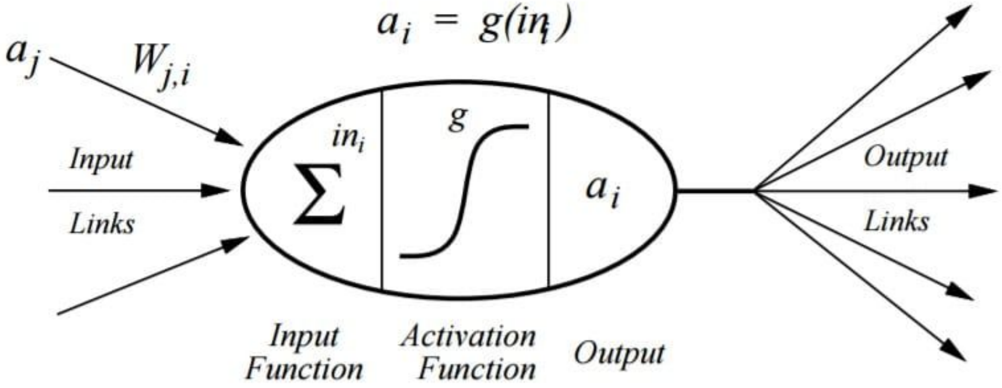
MSSP 608 Recitation 5

Deep Learning Review

Early Deep Learning (Perceptron)



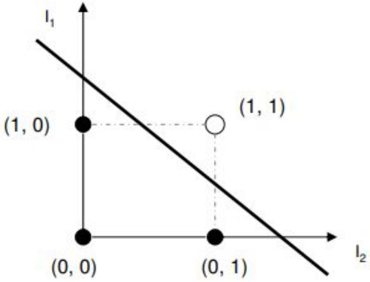
Perceptron Math



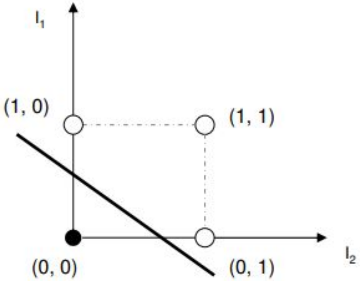
$$a_i = g(\sum_j W_{j,i} a_j)$$

First AI Winter

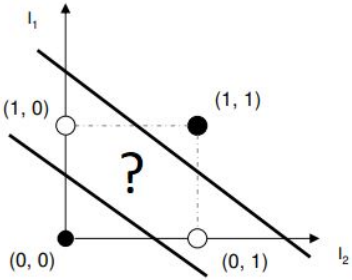
AND		
I_1	I_2	out
0	0	0
0	1	0
1	0	0
1	1	1



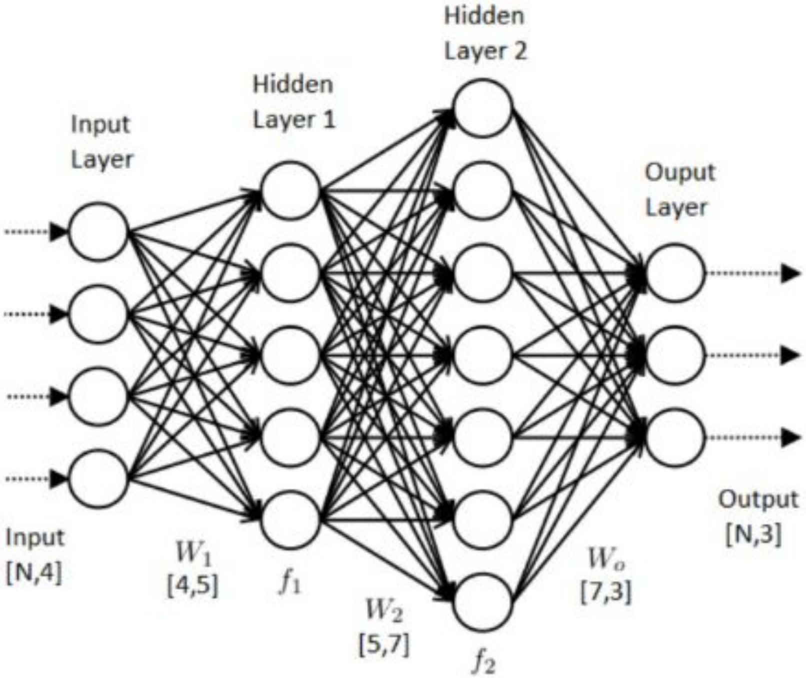
OR		
I_1	I_2	out
0	0	0
0	1	1
1	0	1
1	1	1



XOR		
I_1	I_2	out
0	0	0
0	1	1
1	0	1
1	1	0



Solution - Multilayer Perceptrons



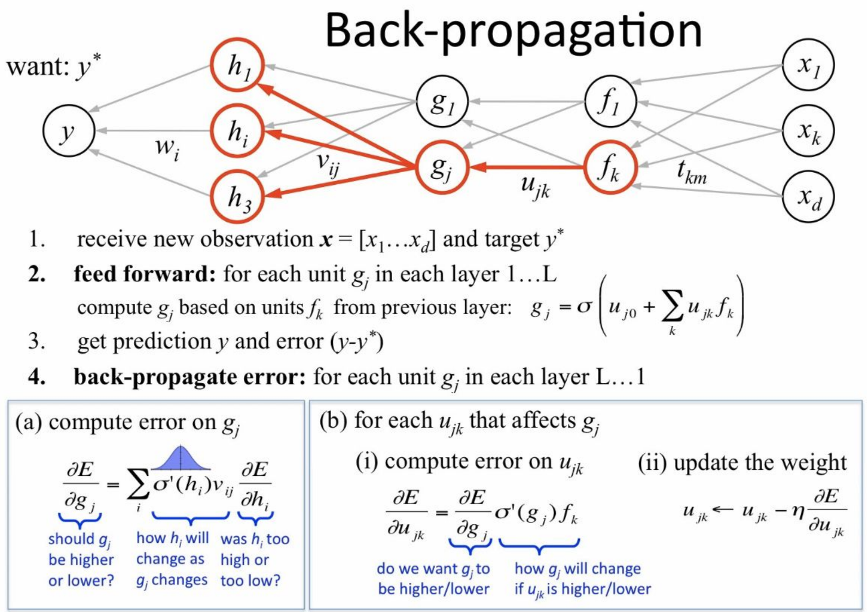
How to train MLP quickly?

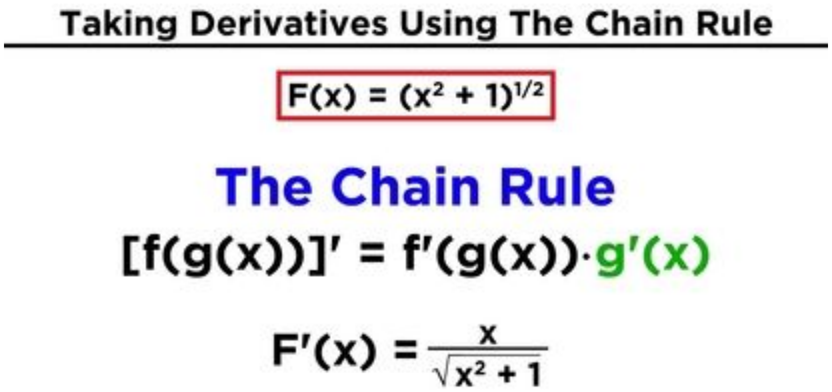
- Step 1: Initialize the weights of the network
- Step 2: For epoch in epochs:
    - (a): Get a batch of data
    - (b): Put the batch of data through the network
    - (c): Compute the value of the loss function
    - (d): Use gradient descent + **backpropagation** to compute gradients
    - (e): Update the weights to be W - c * dW where c is the learning rate
- Step 3: Save the model

- Modern Deep Learning works well because it also can take good advantage of **powerful GPU hardware**

What is backpropagation

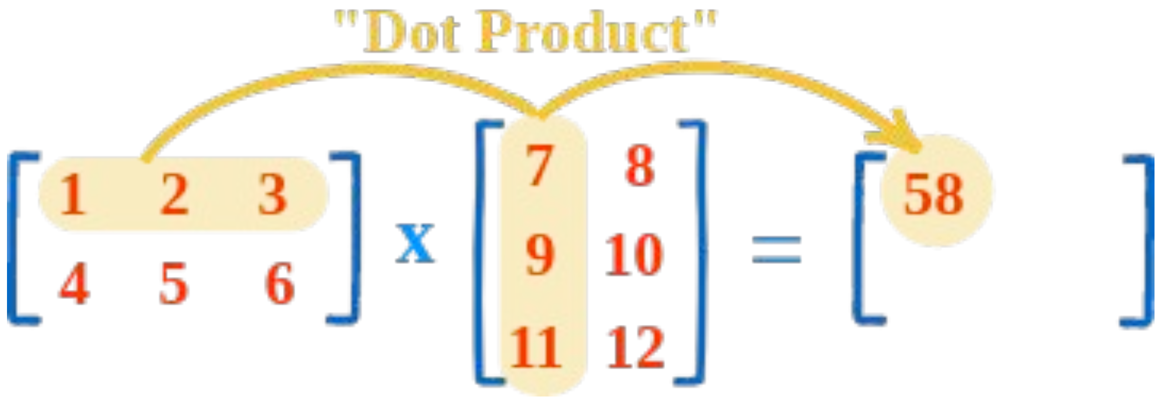
Basically, a clever way to compute the chain rule using dynamic programming





Why GPU?

- The GPU is very good at parallelizable operations
- Can have thousands of cores rather than just 2-64

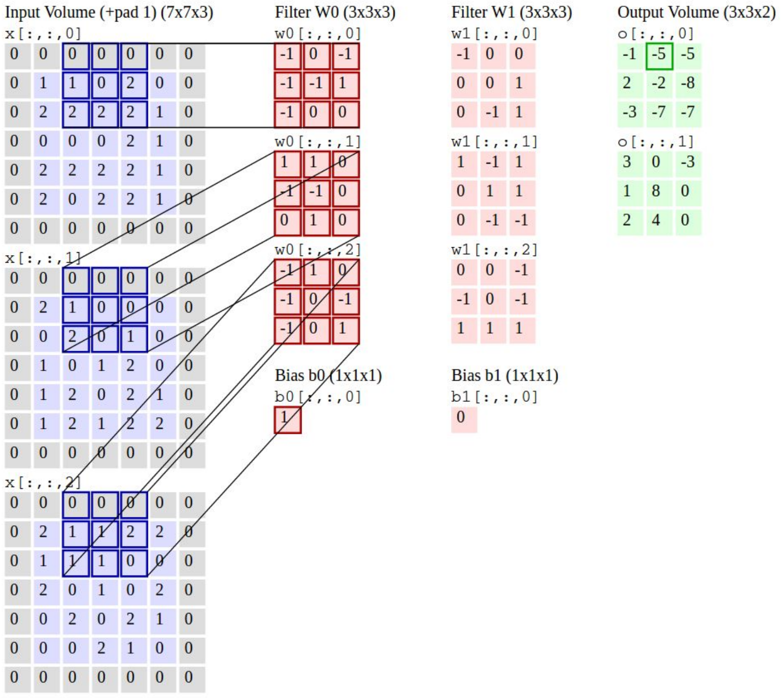


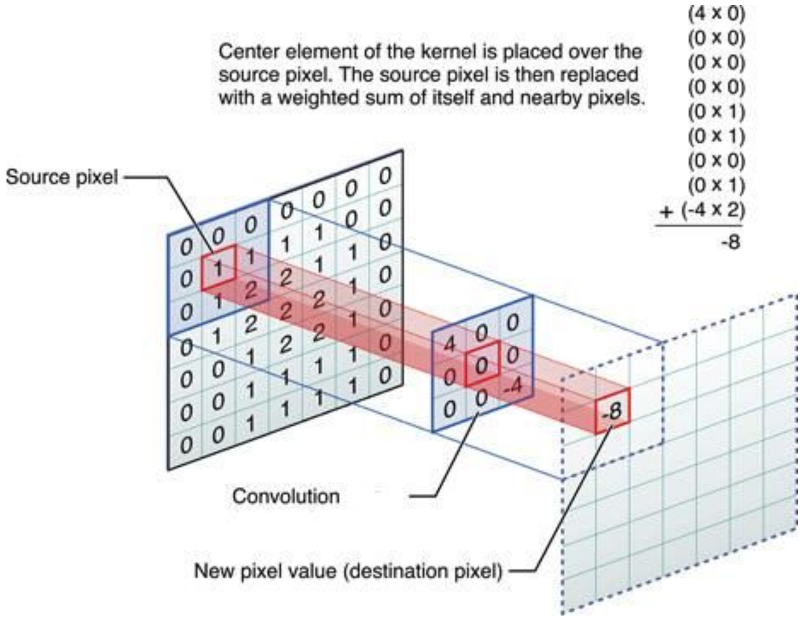
More advanced Networks (Convolution)

We can incorporate knowledge we know to be true about the data through the network architecture

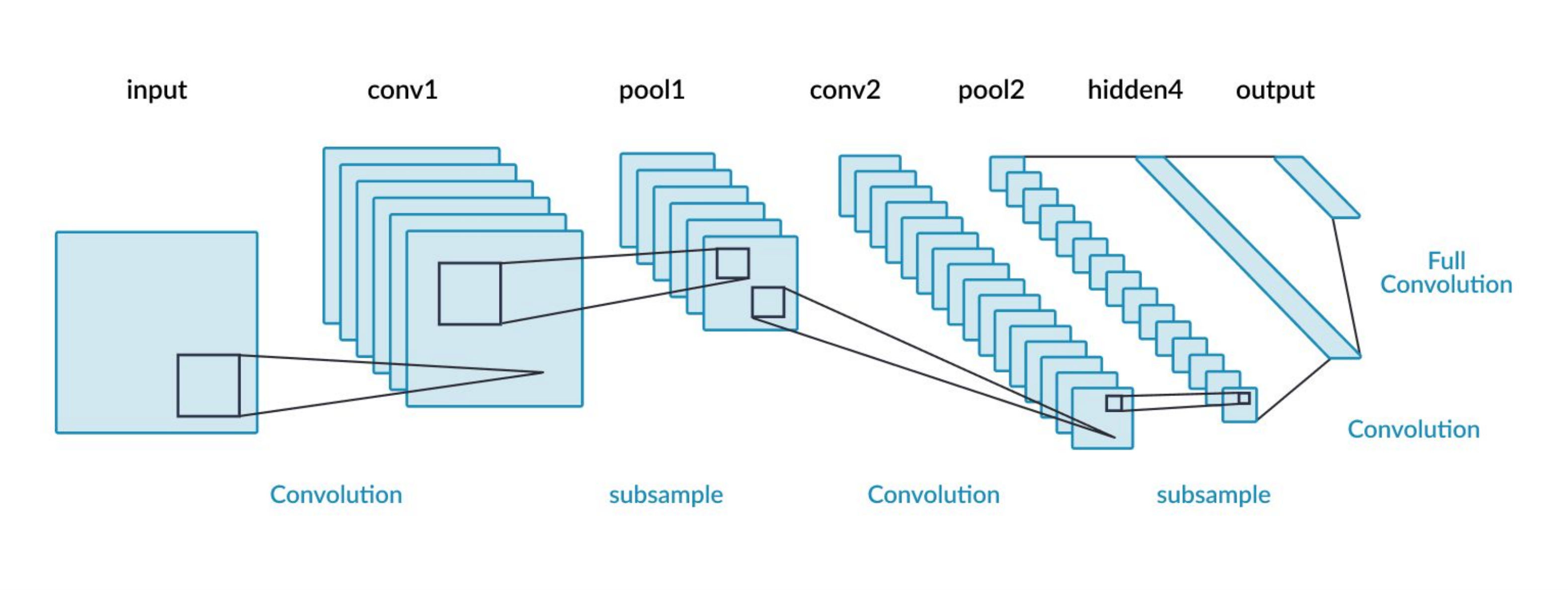
One piece of knowledge we know for images -- If you shift an image left or right it still has the same object in it

Convolutional Networks (CNNs)





Convolutional Networks (CNNs)



More Deep Learning

- There is a lot more to deep learning
  - Other network architectures and elements than MLP or Convolutional networks
  - Other (better) variants of gradient descent for optimization
  - Other activation functions
  - Advanced methods for initialization of networks
  - Data augmentation
  - Much more
- Here are some resources I like:
  - 3Blue1Brown visual introduction course on YouTube
  - Fast.ai deep learning for coders course

Why I recommend fast.ai

5 lines of code gets you

- State of the art architecture
- Good optimizer and initialization
- Transfer learning
- Training loop already programmed
- Also has a full course on how to get more out of the library and as an introduction + deep dive on deep learning

```
from fastai.vision import *
path = untar_data(URLs.MNIST_SAMPLE)
data = ImageDataBunch.from_folder(path)
learn = cnn_learner(data, models.resnet50, metrics=accuracy)
learn.fit(5)
```