Introduction to Deep Learning with Keras

Agis Chartsias & Thomas Joyce
IDCOM

Overview

- Introduction to deep learning, focusing on image processing

    1. Brief introduction

    2. High level idea

    3. Example applications

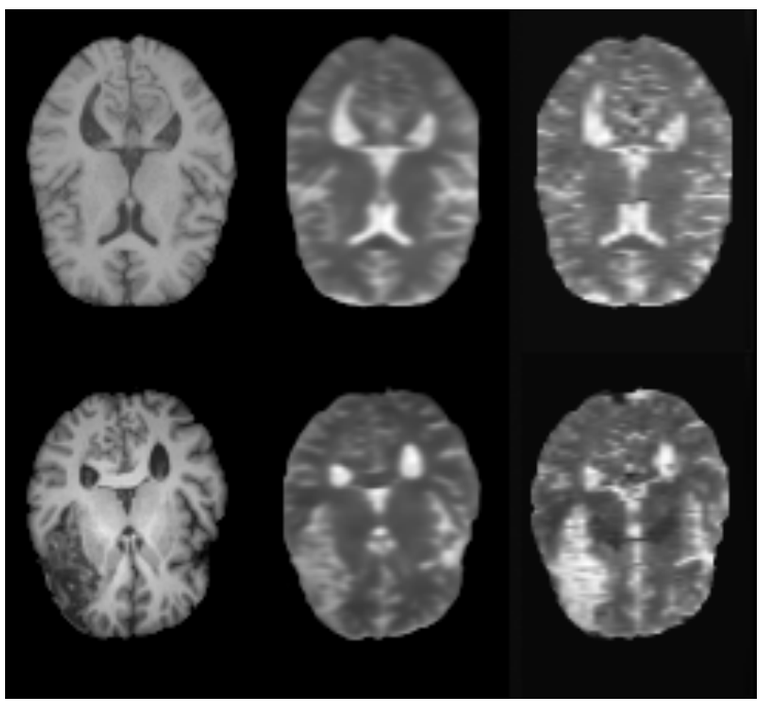
    4. The details

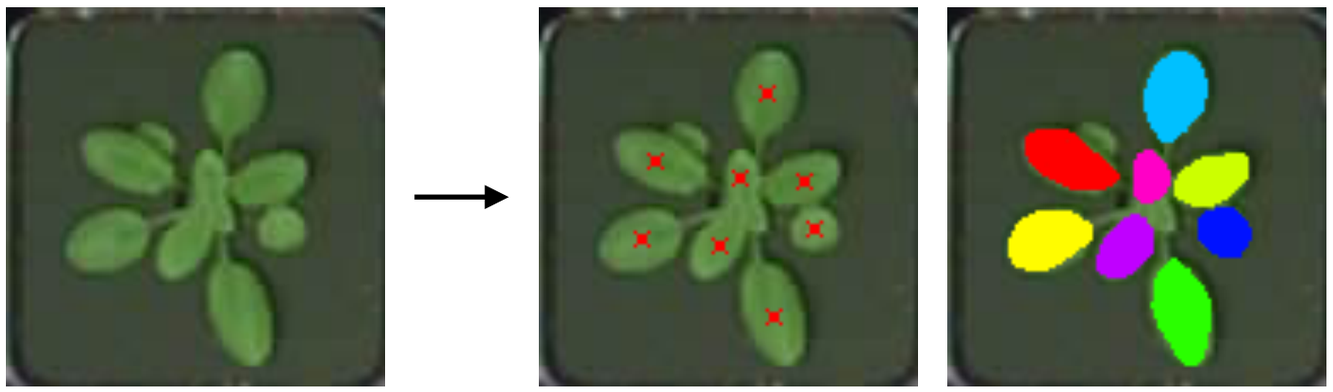
        I. Fully connected layers

        II. Convolutional layers

Who We Are

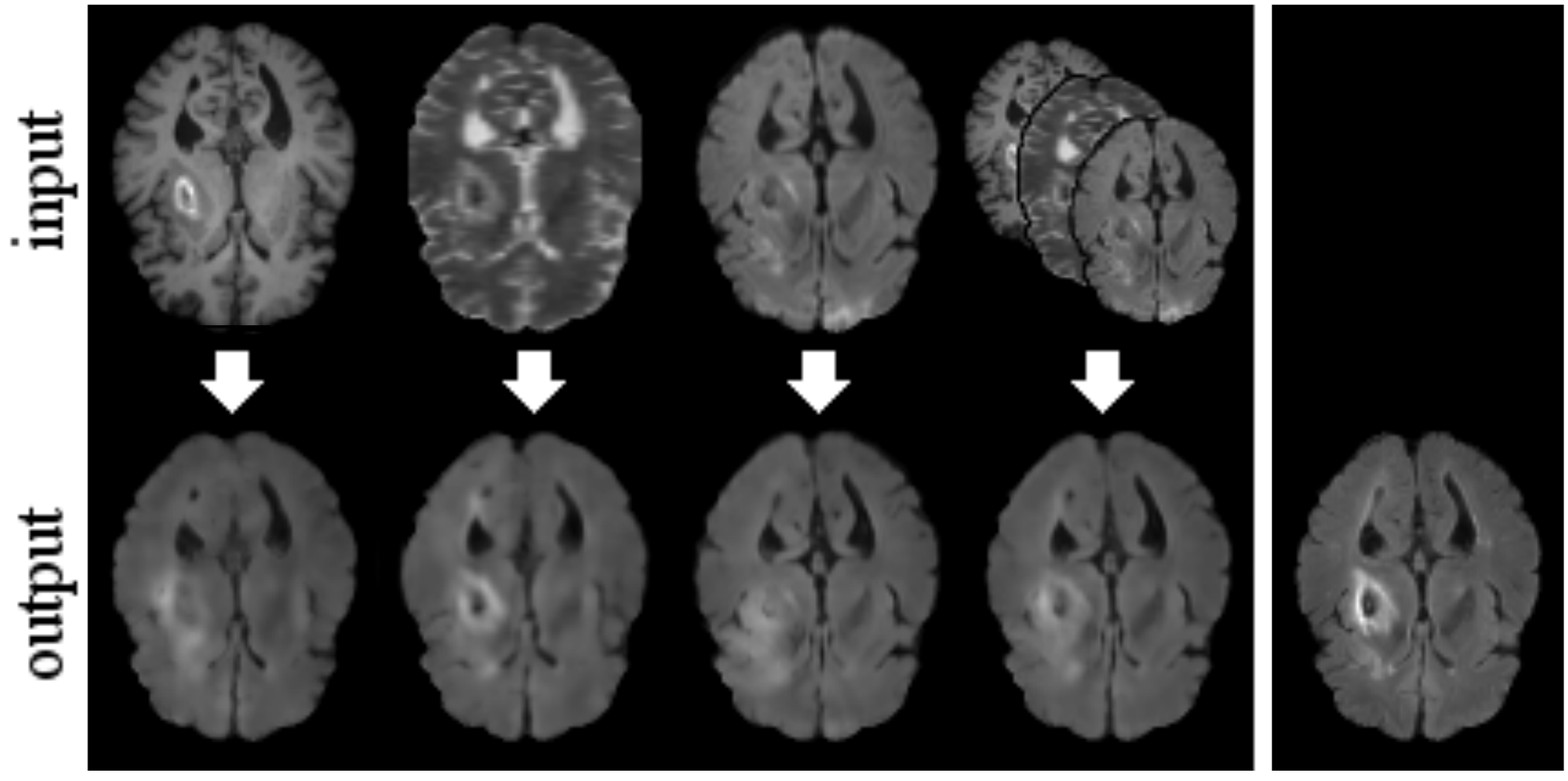
- Group under Dr Sotirios Tsaftaris in IDCOM
- Tom, Agis, Ilkay and Valerio work on medical imaging
- Valerio, Andrei work on plant phenotyping



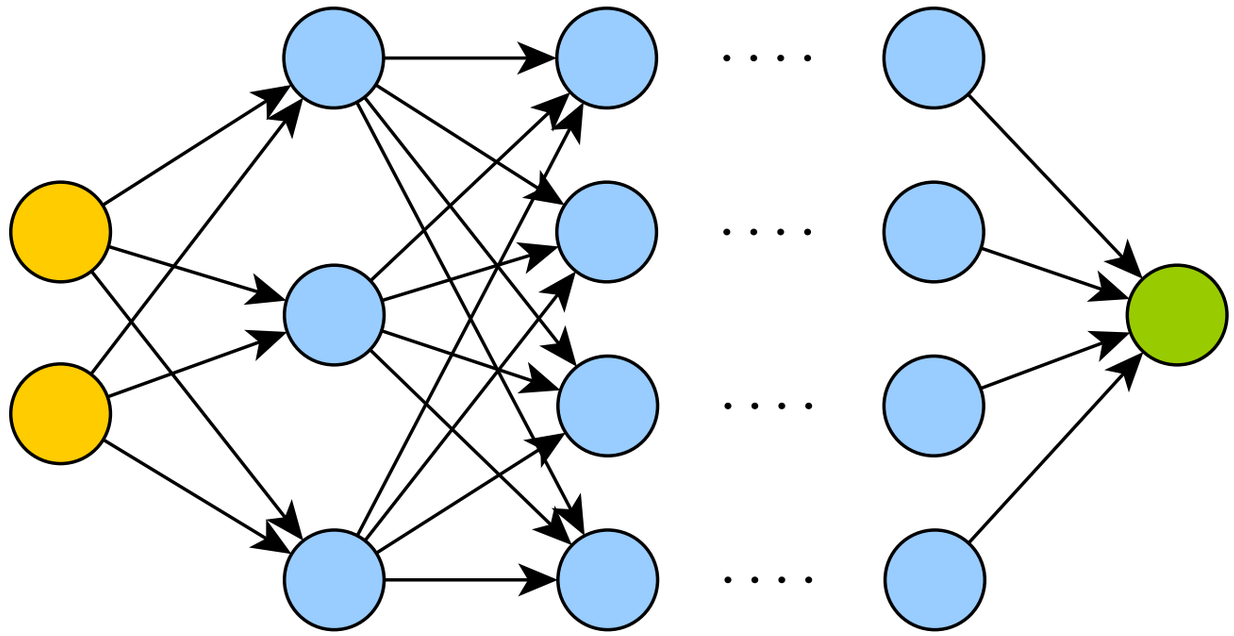


Our Previous Work

- Applying deep learning to medical images

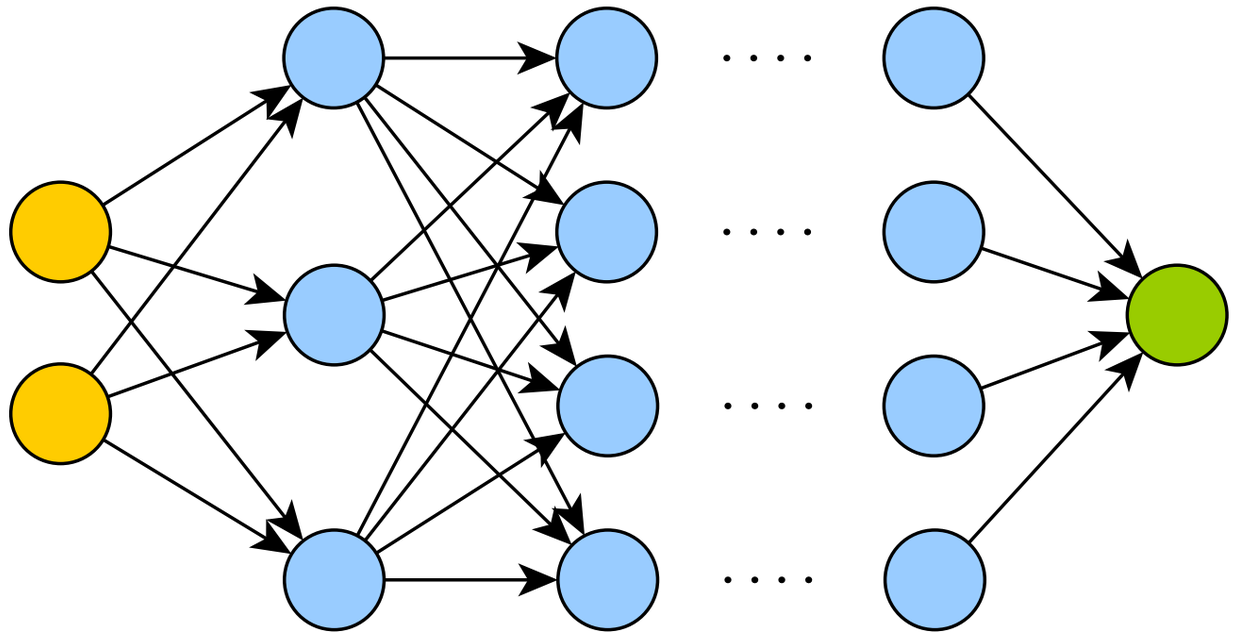


Neural Networks



- A parameterised way to represent functions.
- Universal approximators.
- Deep Learning: neural networks with many layers.

Neural Networks



- Specified by an architecture (the structure of the network, choice of activation functions) and a set of parameters (the weights).

Deep Learning Applications

- What can we do with these universal function approximators?
    - Neural Networks can learn functions by adapting their parameters.
- What can be learnt using neural networks?

MNIST

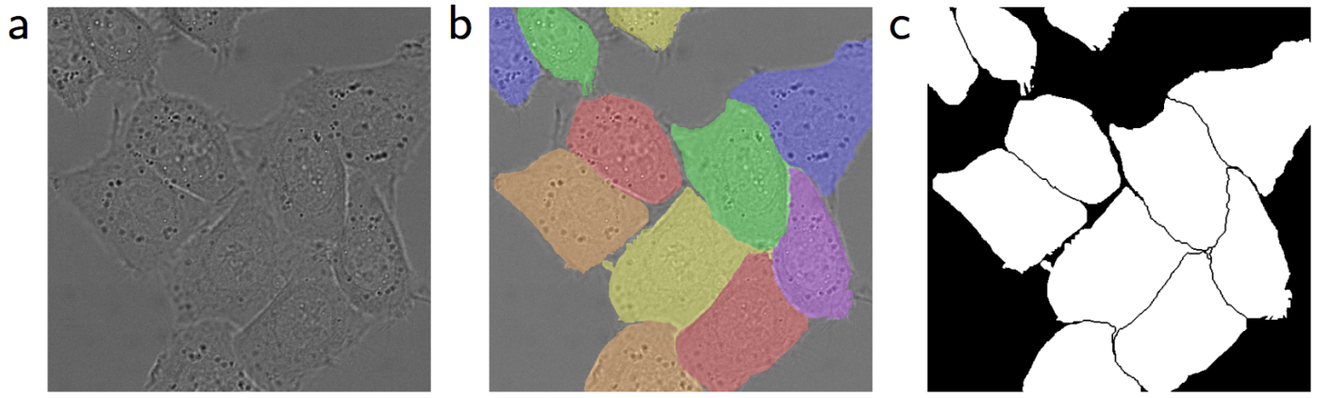
= 3

best error ~ 0.21%

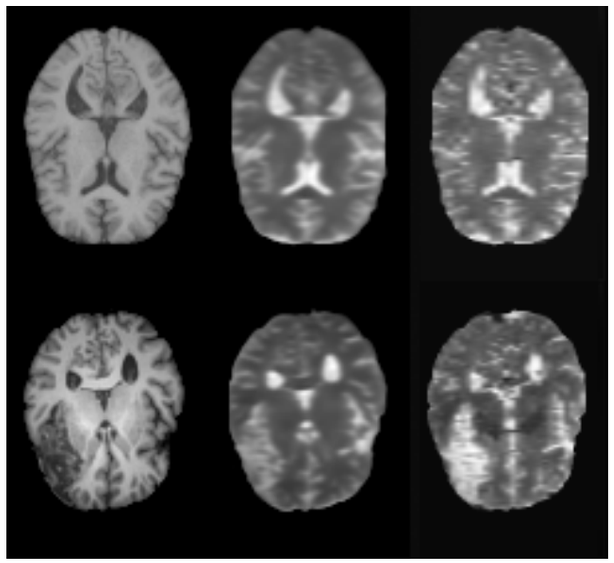
Image Segmentation

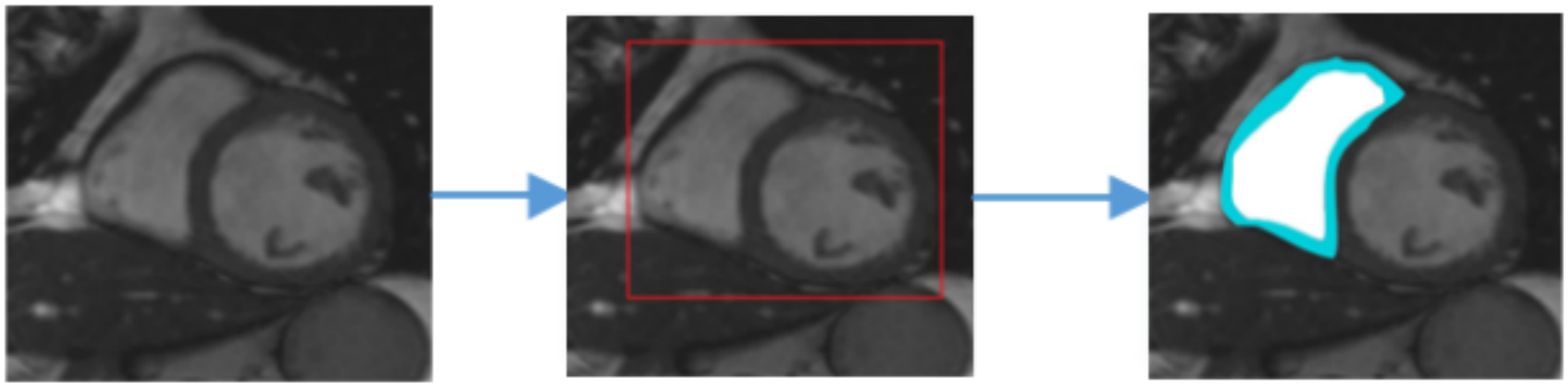
Mask R-CNN, K. He et al.

Biology and Medicine



U-Net - O. Ronneberger et al.





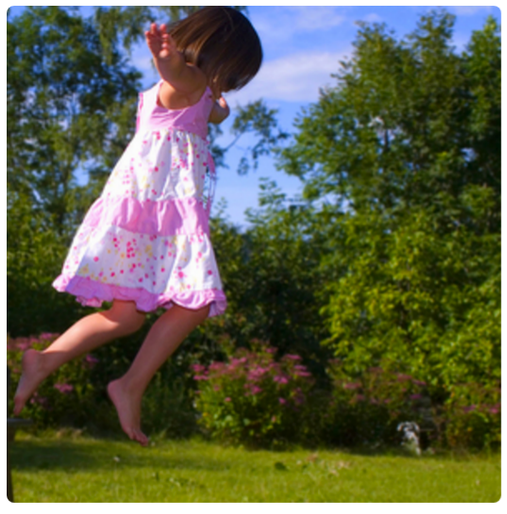
A Deep Learning Network for Right Ventricle Segmentation in Short-Axis MRI - Luo et al.

Image Description

"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."



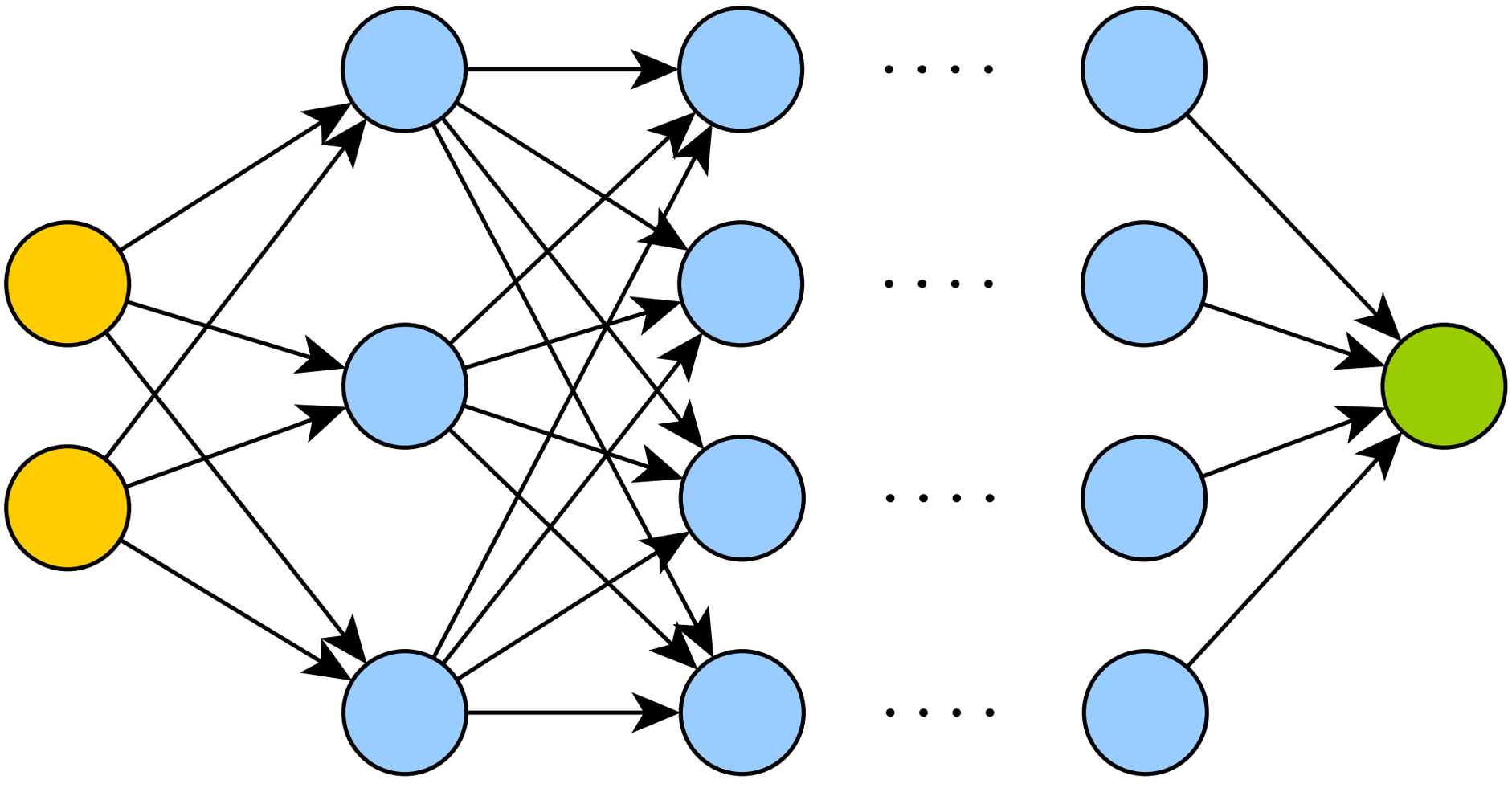
"girl in pink dress is jumping in air."

"black and white dog jumps over bar."

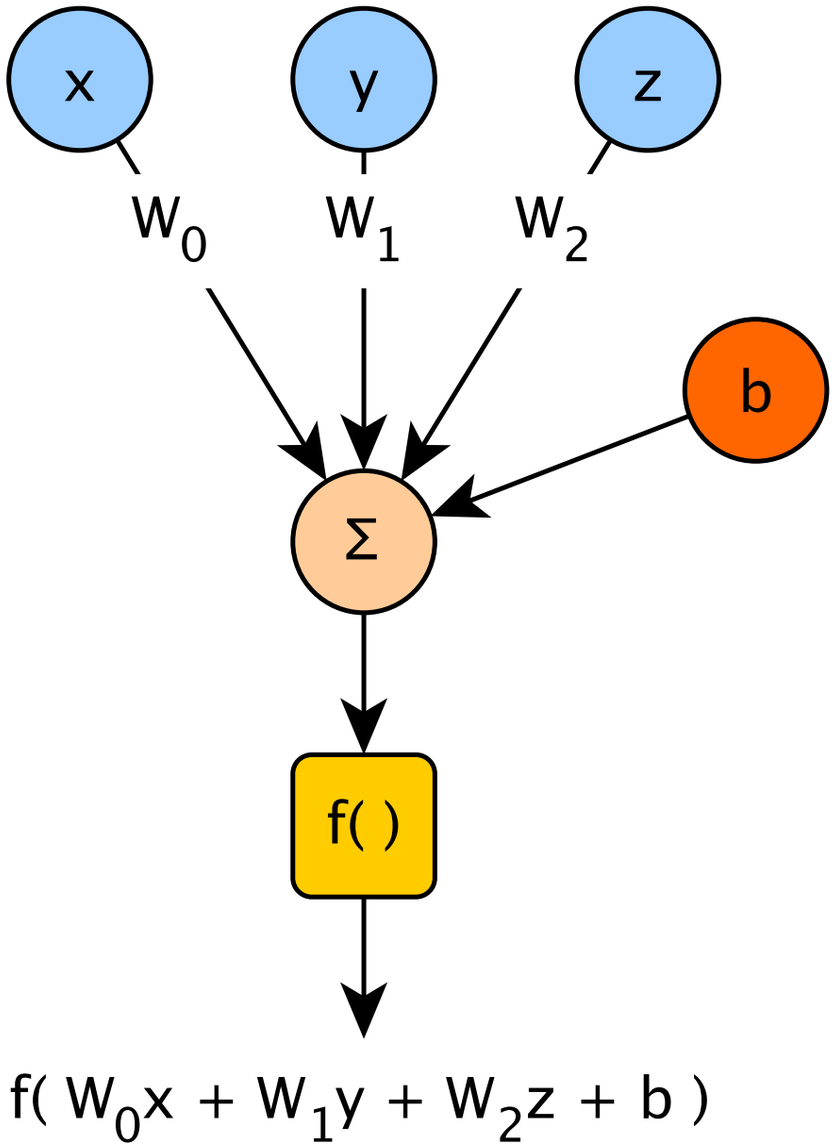
"young girl in pink shirt is swinging on swing."

Deep Visual-Semantic Alignments for Generating Image Descriptions, A. Karpathy & L. Fei-Fei

Neural Network Details

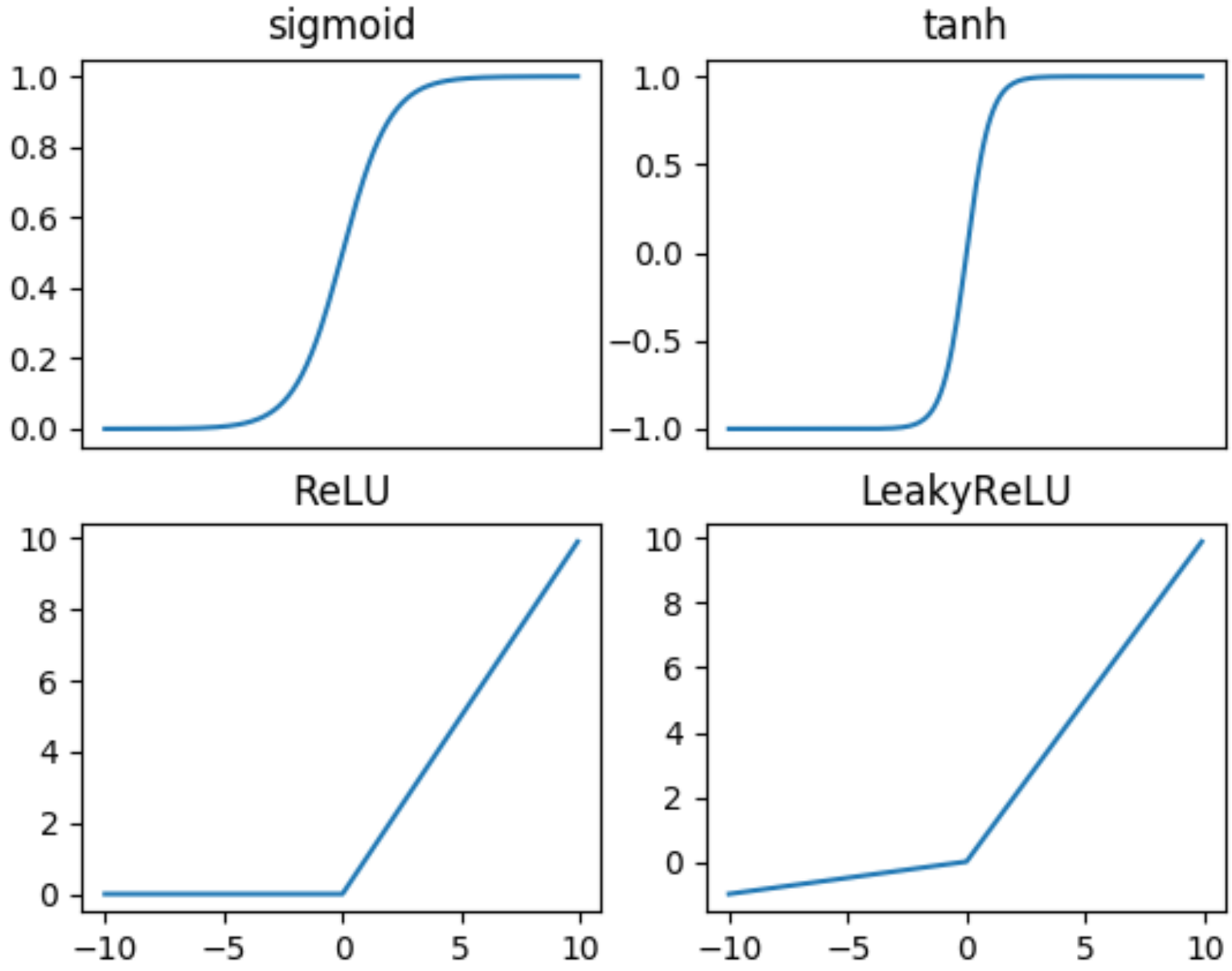


Details of a Neuron

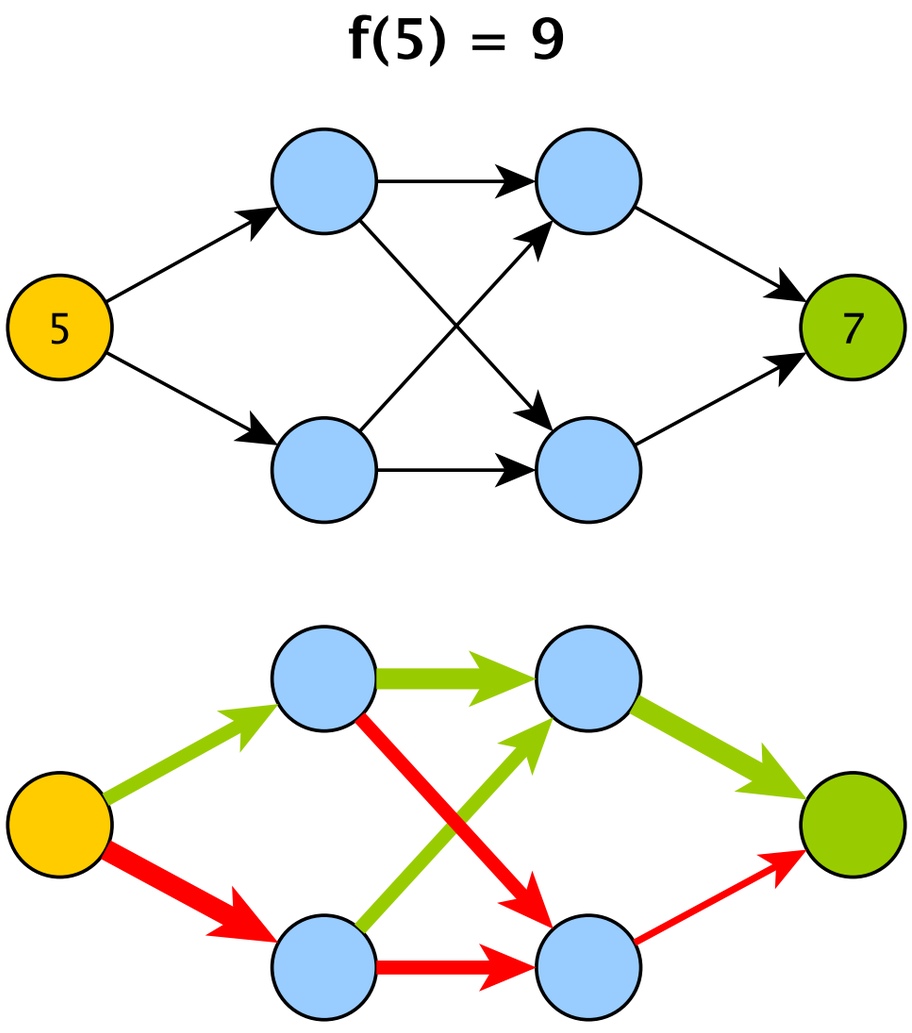


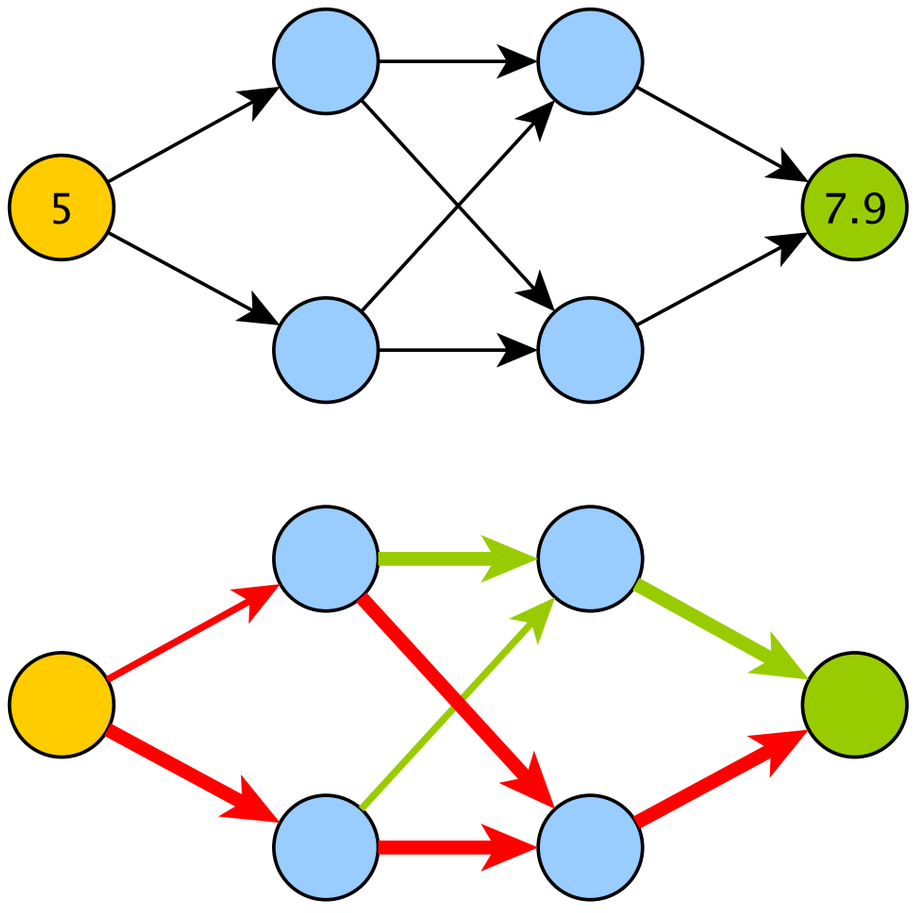
- x,y,z are the inputs
- w_0, w_1, w_2 are the weights
- b is the bias
- f is a non-linear function.

Common Non-Linearities



Training

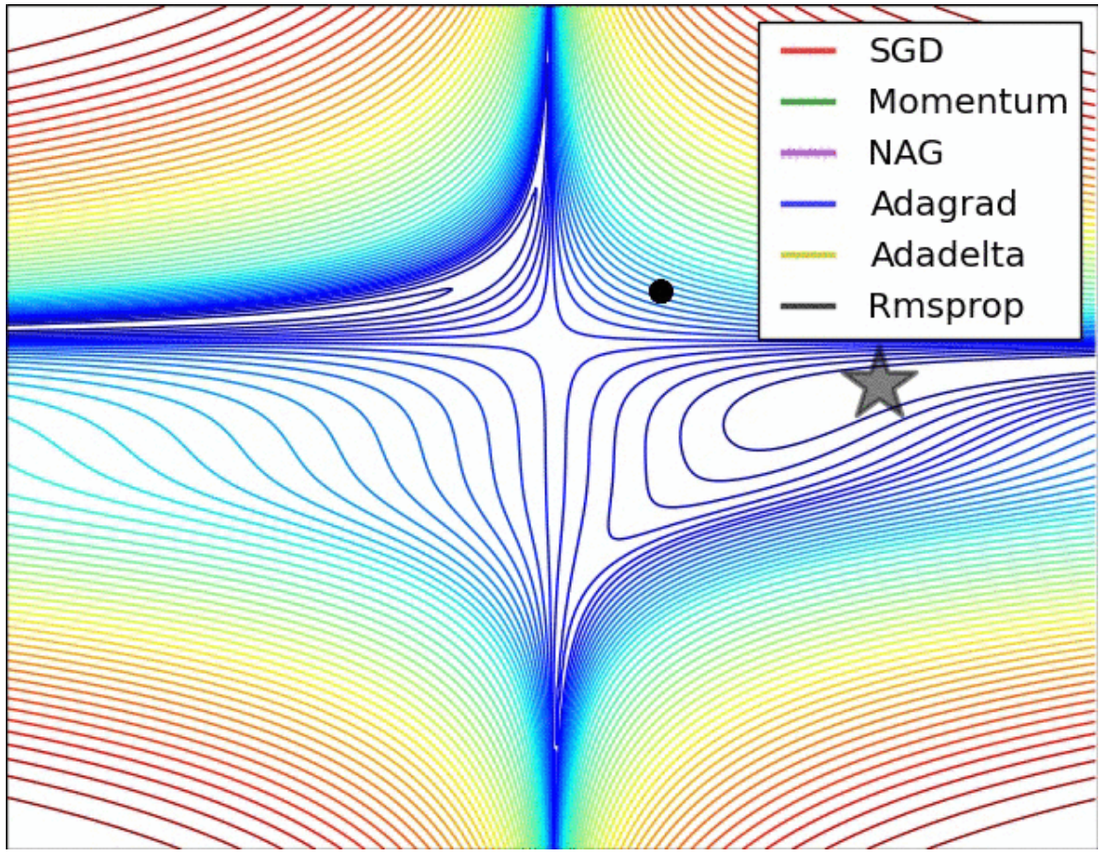




woho!

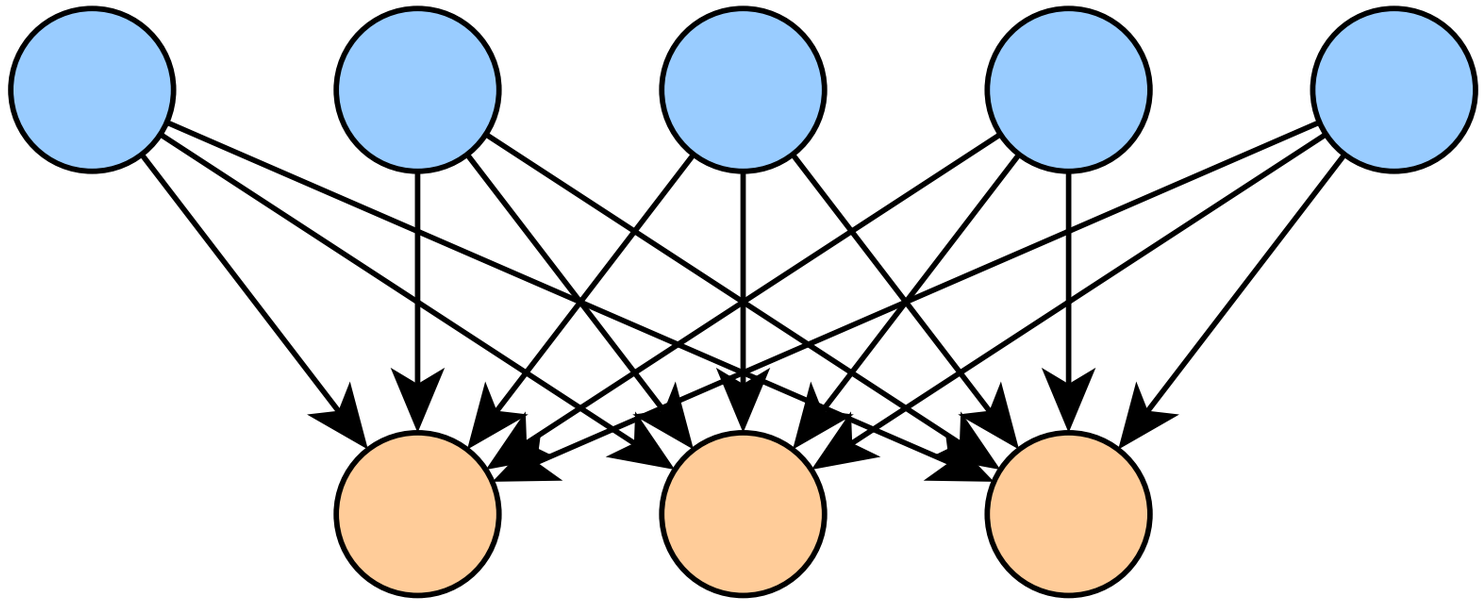
Training

- A neural network can be trained with gradient descent
- Can calculate the gradient analytically
- Very common to use Stochastic Gradient Descent with mini-batches, or an SGD variant.



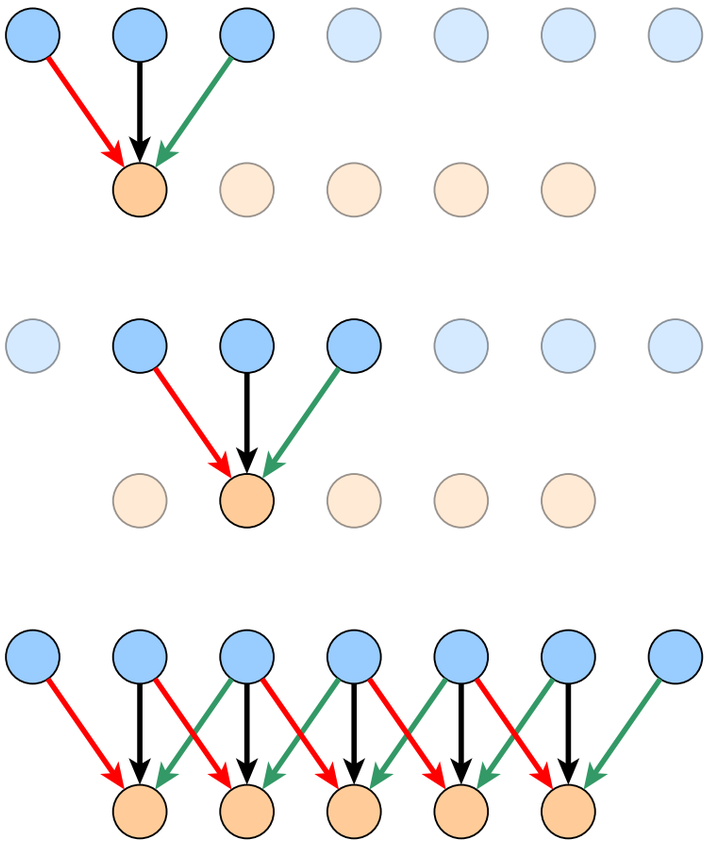
animation by Alec Radford

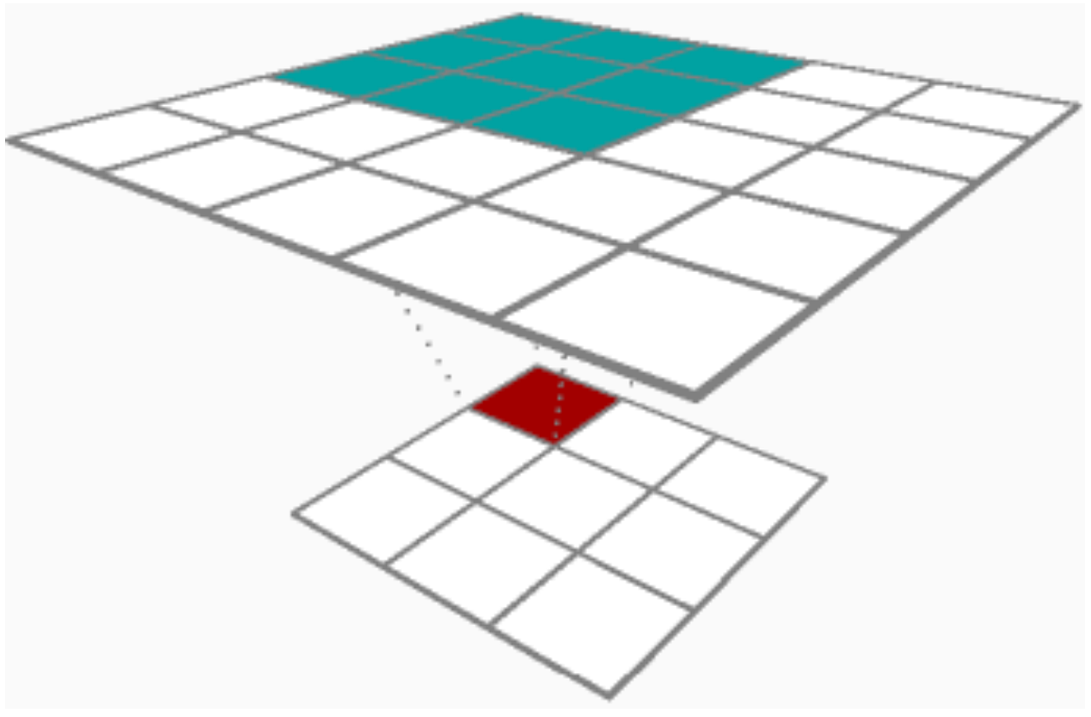
Fully Connected Layers



- Every neuron in one layer connected to every neuron in the next.
- The simplest and most general layer type.
- Every edge has its own weight.

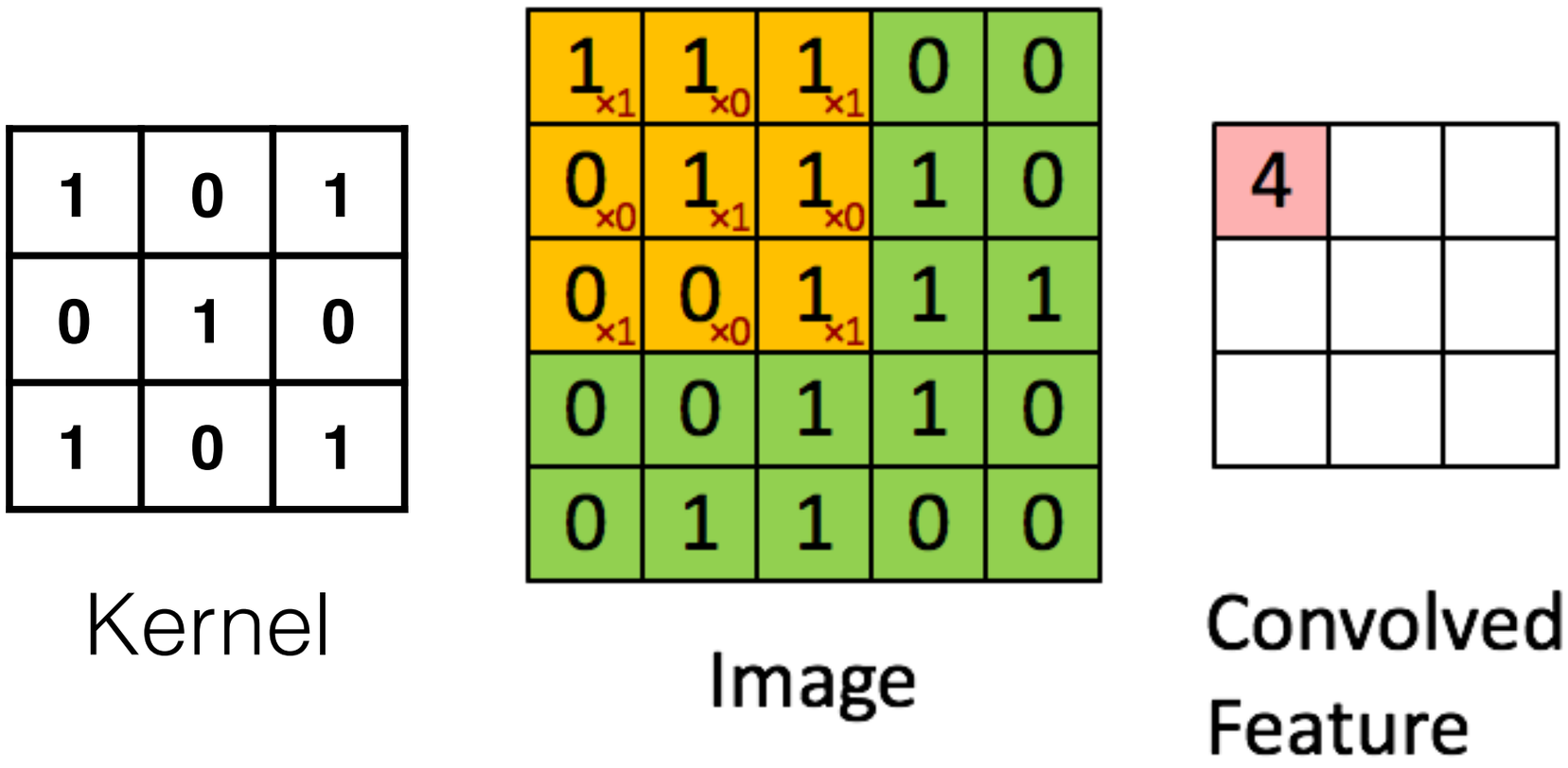
Convolutional Layers





- Apply the same local operation everywhere.
- Can be thought of as a fully connected layer with systematically shared weights and many 0 weights.

Convolutional Layers

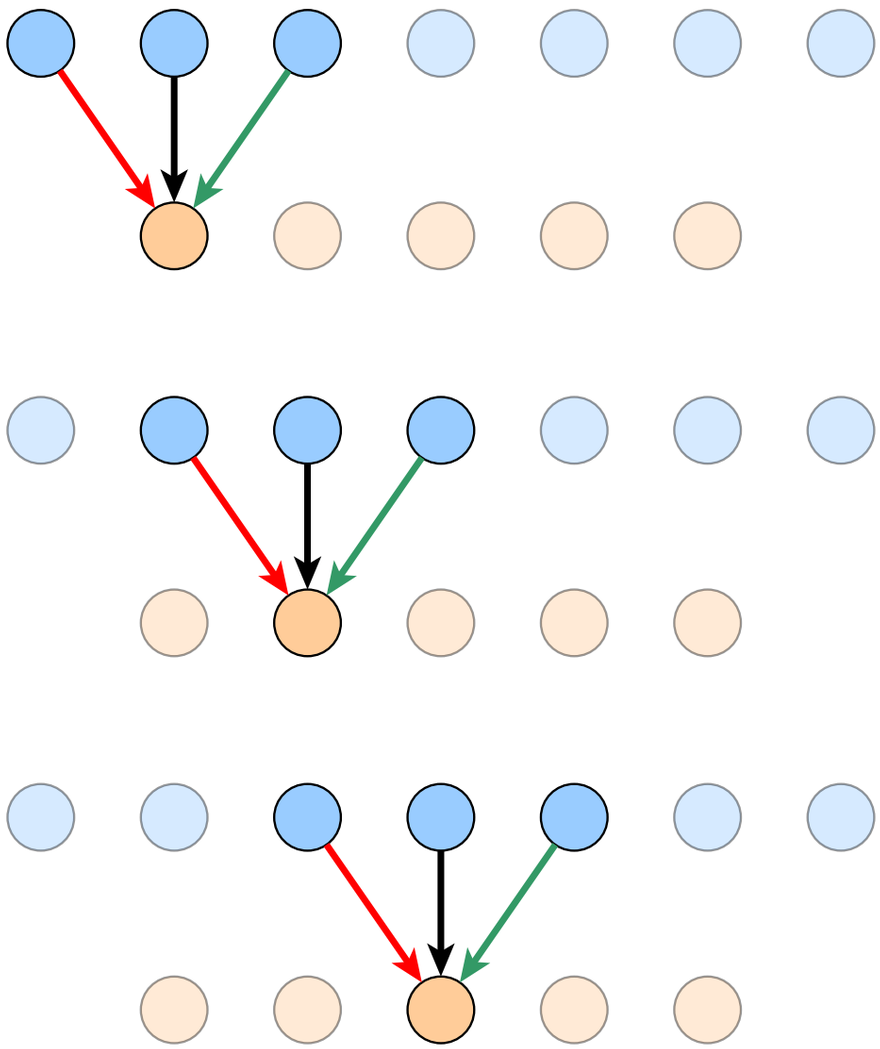


Animation from: https://leonardoaraujosantos.gitbooks.io/artificial-inteligence/content/convolution.html

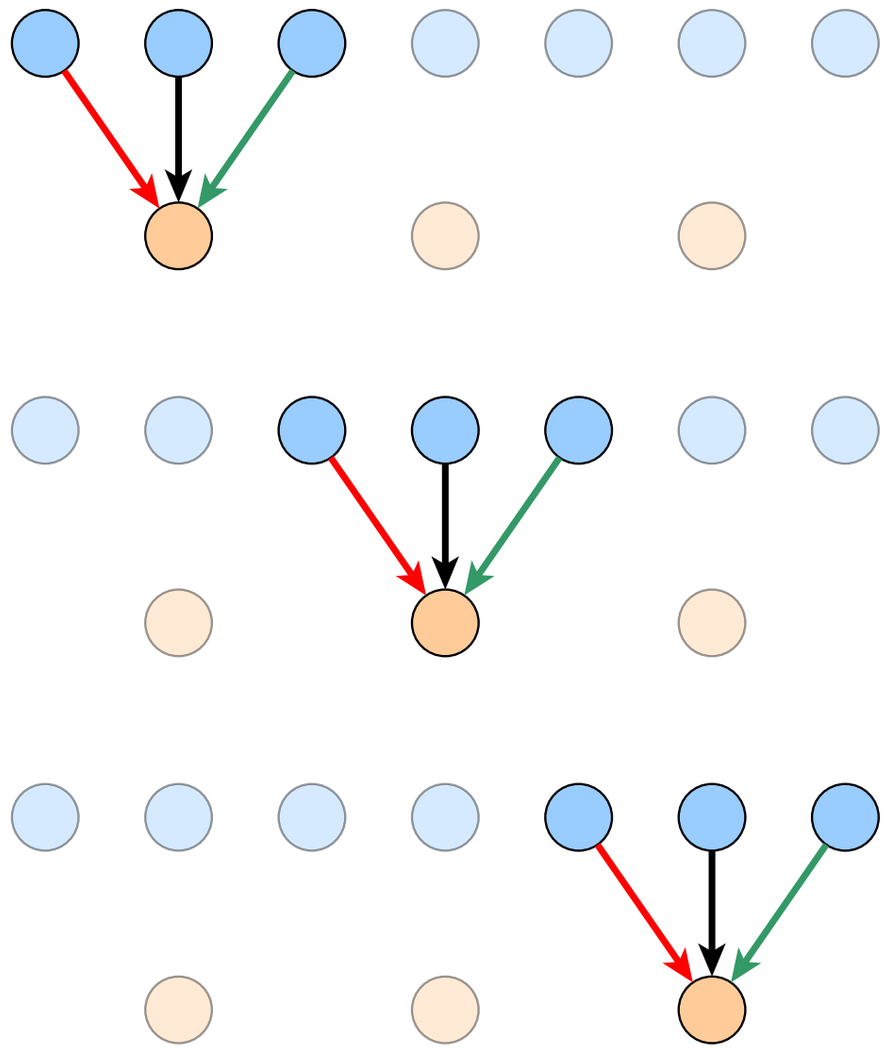
Convolutional Layers

- Convolutions have some additional properties which can initially be confusing:
  - stride
  - padding
  - filter size
  - channels

Convolution - Stride

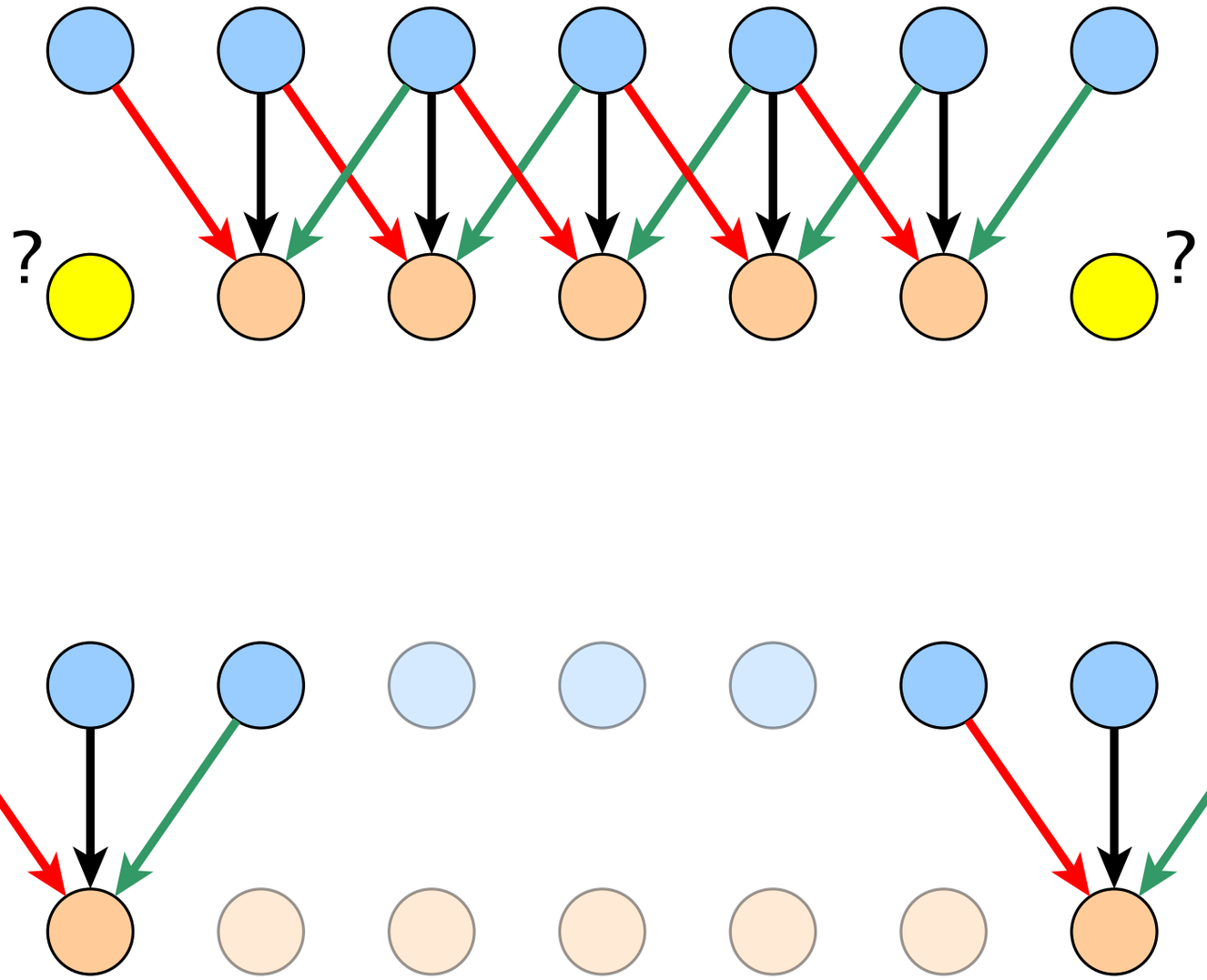


stride 1

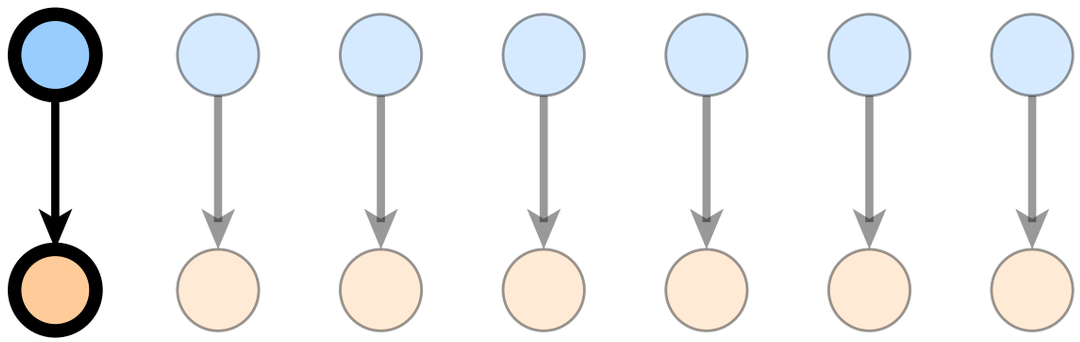


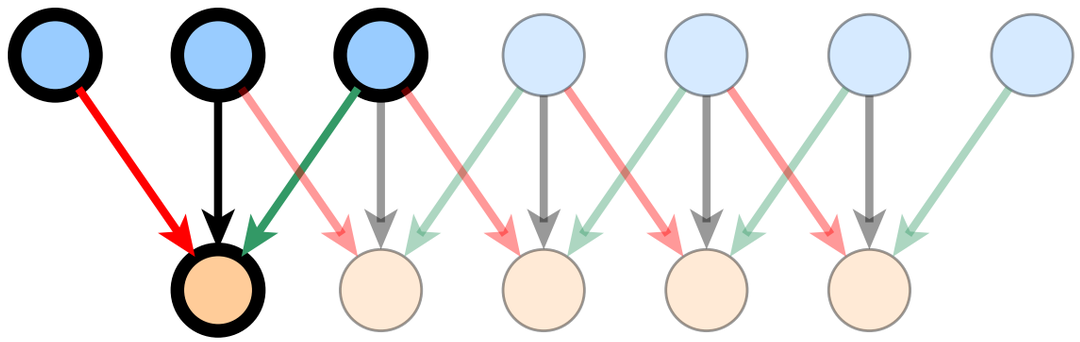
stride 2

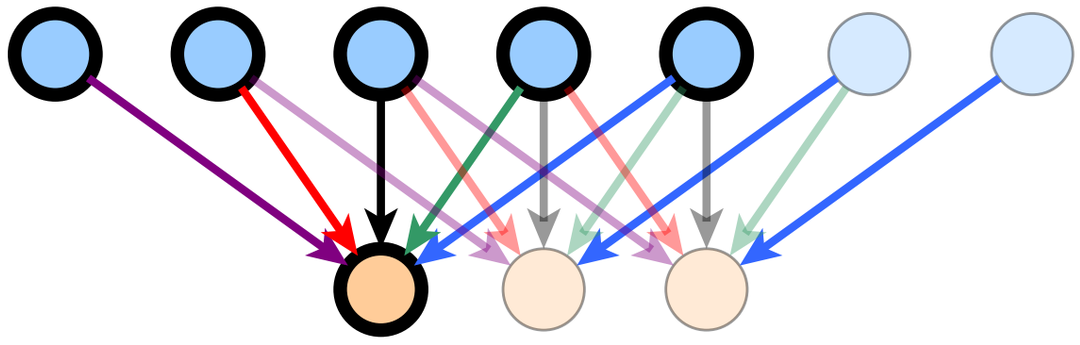
Convolution - Padding



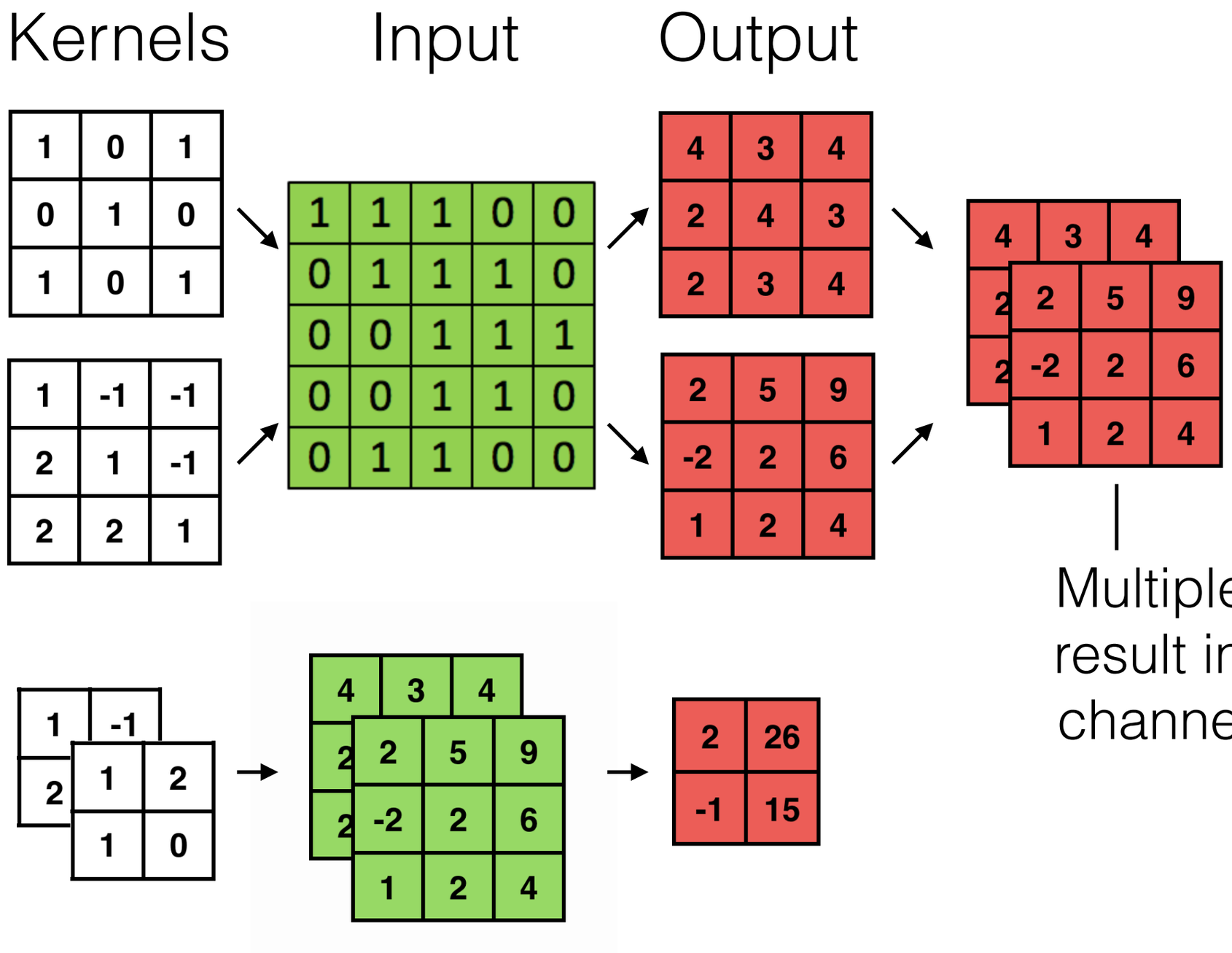
Convolution - Filter Size







Convolution - Channels



Summary

- Introduced neural networks
- Shown some example applications
- Discussed fully connected layers
- Discussed convolutional layers
- Now Agis will show you how to actually use neural networks with Keras.