PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO GRANDE DO SUL
FACULDADE DE INFORMÁTICA
PROGRAMA DE PÓS GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

MLP and Convolutional Neural Nets

Gabriel Simões

BUSINESS INTELLIGENCE AND
MACHINE LEARNING RESEARCH GROUP

Agenda

- Some words of Machine Learning
- Neural Networks (NN) Overview
  - Multi-layer Perceptron (MLP) training issues
- Convolutional Neural Networks (CNN)
  - CNN training
- Frameworks
  - What kind of DL frameworks we have?
  - Samples

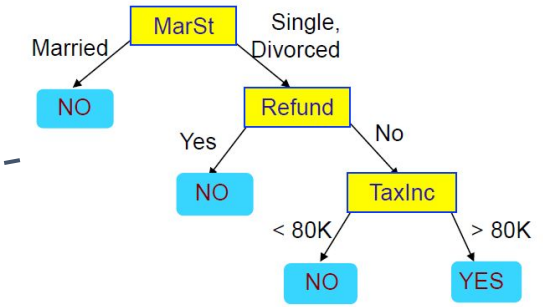
Machine Learning: some simple words

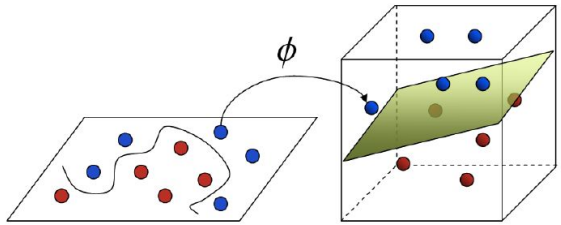
A kind of magic

Machine Learning

Supervised Learning

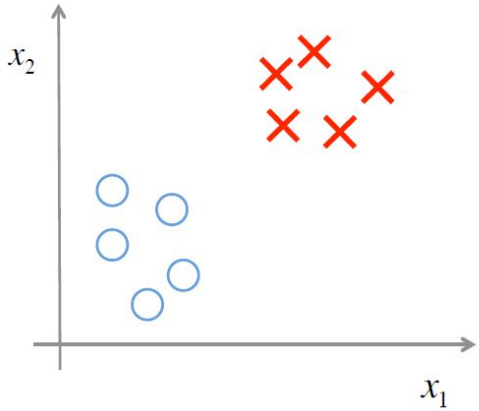
- Decision Tree
- SVM
- Naive Bayes
- Neural Network





Unsupervised Learning

- EM - Expectation Maximization
- K-means



Reinforcement Learning

- Monte Carlo Methods

Traditional Machine Learning

Is good for:

- Predictions
  - Predict sales
  - Predict prices
  - Pattern recognition
- Clustering
  - Customer profile
  - Anomaly detection

Not so good for **high dimensionality** problems!

Such image, video, audio...

Deep Learning

"A family of methods that use **deep architectures** to learn **high-level features** abstractions."

Kevin Duh (Nara Institute of Science and Technology)

Deep Learning

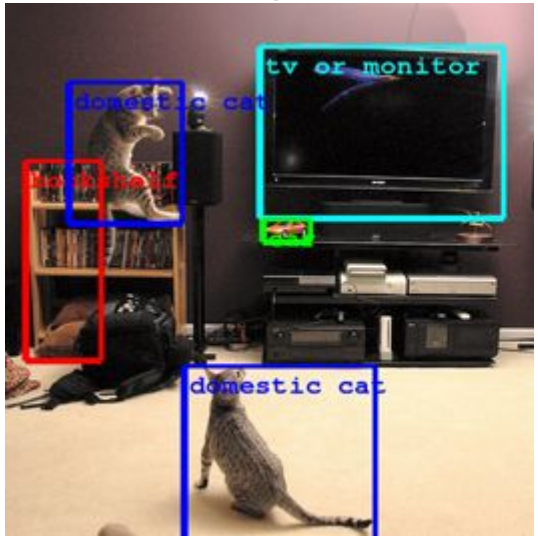
"A family of methods that use **deep architectures** to learn **high-level features** abstractions."

Kevin Duh (Nara Institute of Science and Technology)

Deep Learning

Is good for:

Image



Video

Audio

Introduction to Neural Networks

How the magic happens

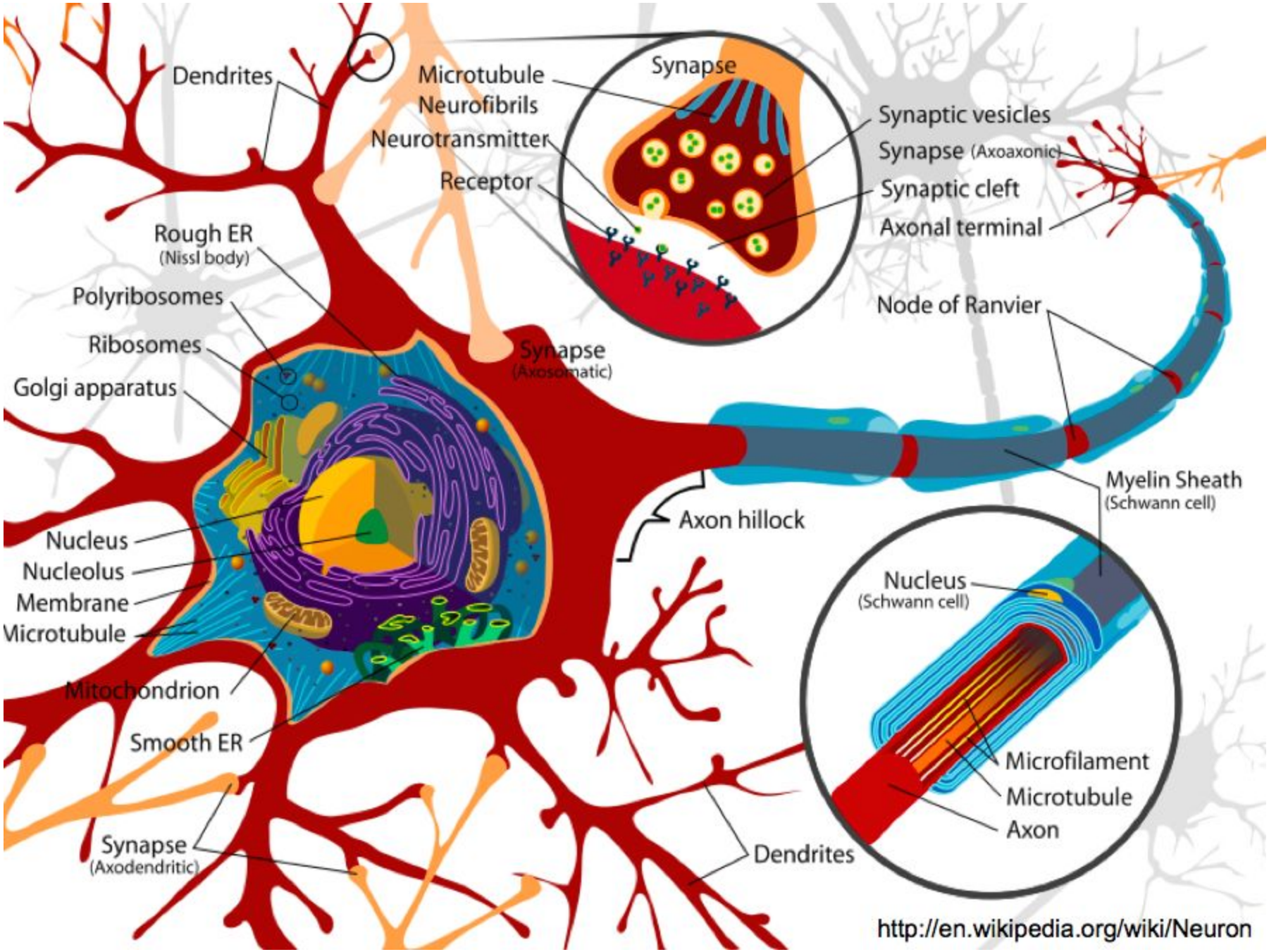
Neural Networks

A neural network is a massively parallel distributed processor made up of simple processing units that has a natural propensity for storing experiential knowledge and making it available for use. It resembles the brain in two respects:

1. Knowledge is acquired by the network from its environment through a learning process.

2. Interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge.

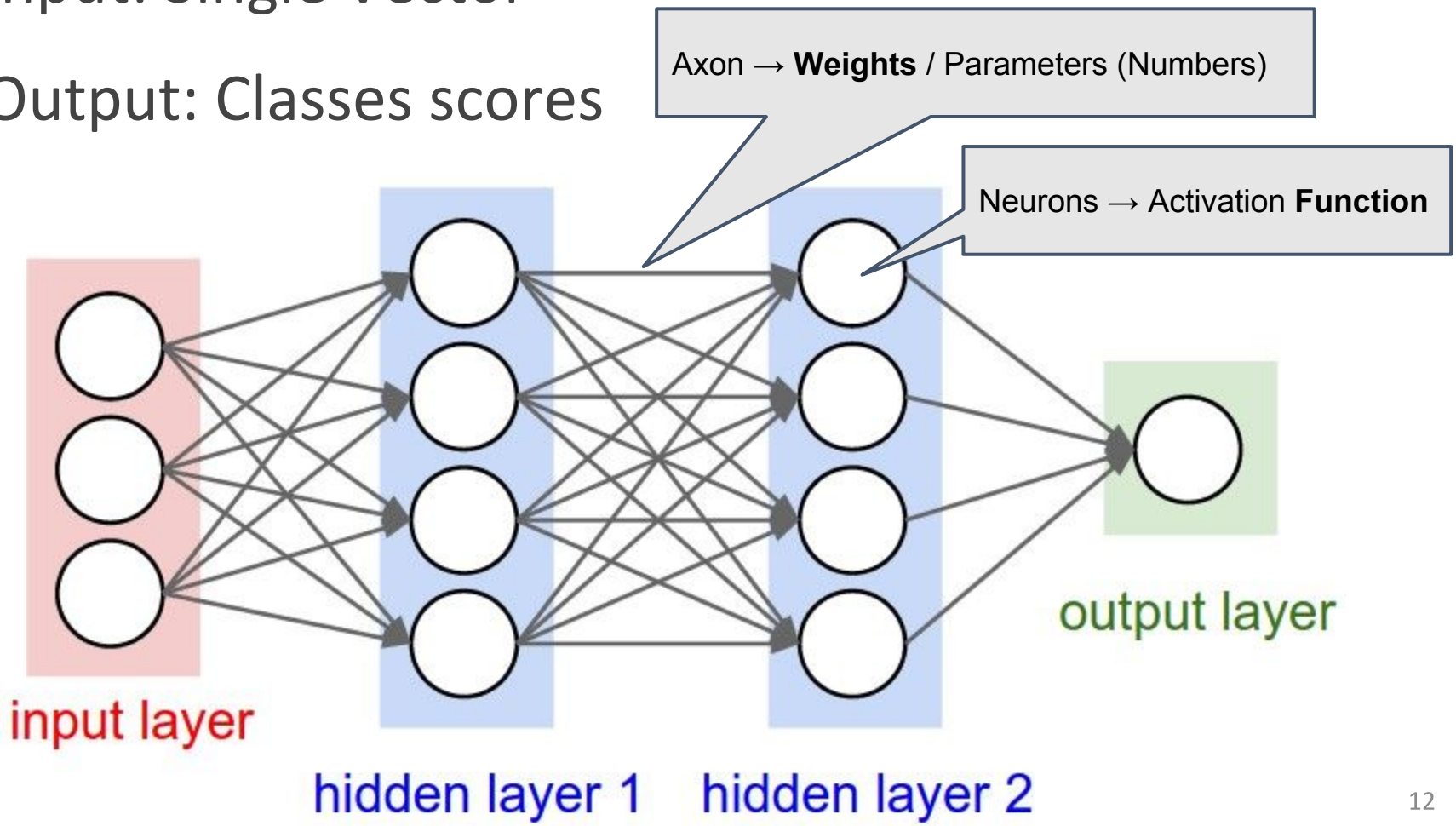
Haykin, Simon. Neural networks and learning machines. 3rd ed.


Dendrites
Microtubule
Neurofibrils
Neurotransmitter
Receptor
Synapse
Synaptic vesicles
Synapse (Axoaxonic)
Synaptic cleft
Axonal terminal
Rough ER
(Nissl body)
Polyribosomes
Ribosomes
Golgi apparatus
Synapse
(Axosomatic)
Node of Ranvier
Myelin Sheath
(Schwann cell)
Axon hillock
Nucleus
Nucleolus
Membrane
Microtubule
Mitochondrion
Smooth ER
Nucleus
(Schwann cell)
Microfilament
Microtubule
Axon
Synapse
(Axodendritic)
Dendrites
http://en.wikipedia.org/wiki/Neuron

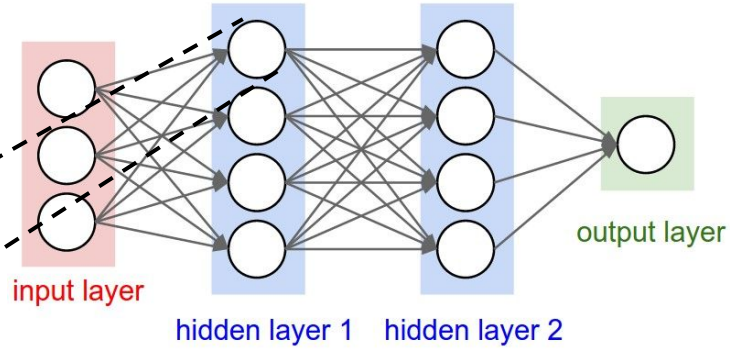
Neural Networks

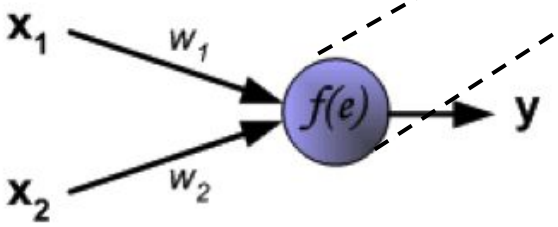
Input: Single Vector

Output: Classes scores



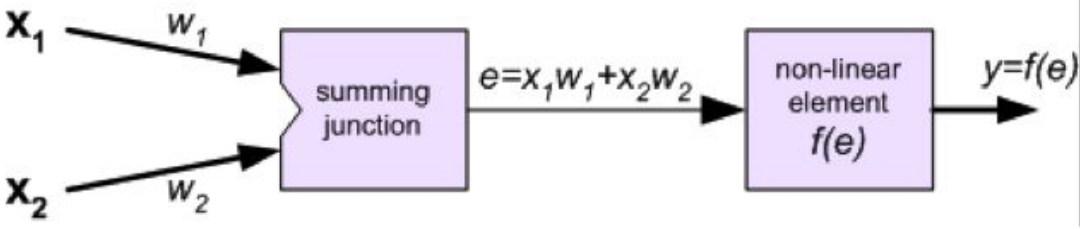
Neural Networks





$$f(x) = \frac{1}{1 + e^{-x}}$$

Activation Function

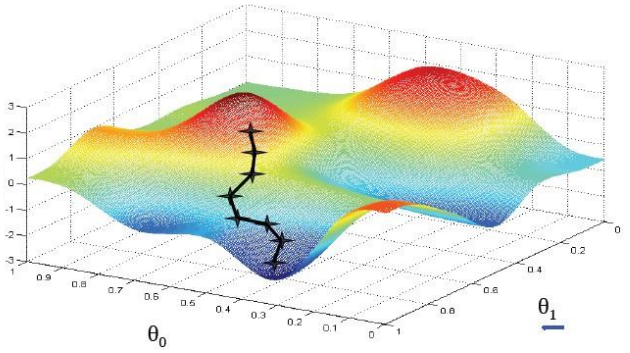


Backpropagation

A technique for implementing gradient descent in weight space for a multilayer perceptron.

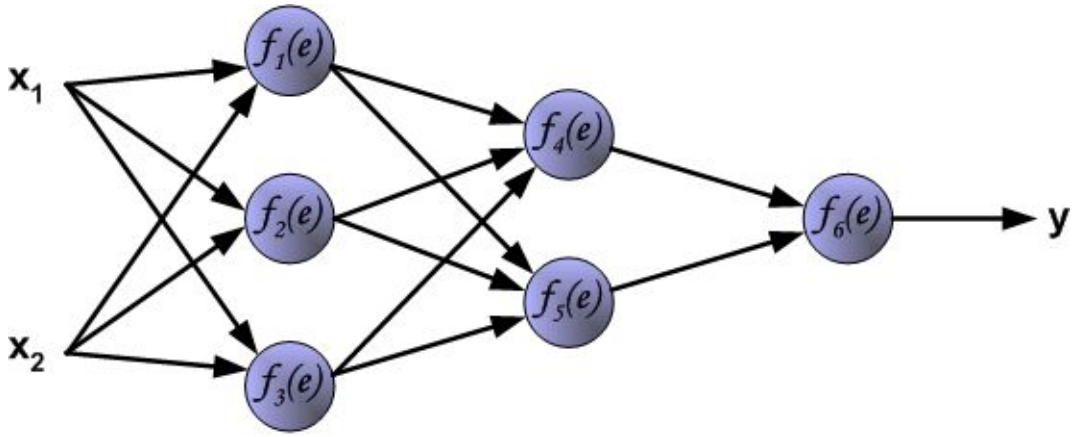
Training steps:

- For each training instance
  - Input Data
  - Feed-Forward
  - Output error calculation
  - Error Backpropagation
  - Weights adjust



Haykin, Simon. Neural networks and learning machines. 3rd ed.

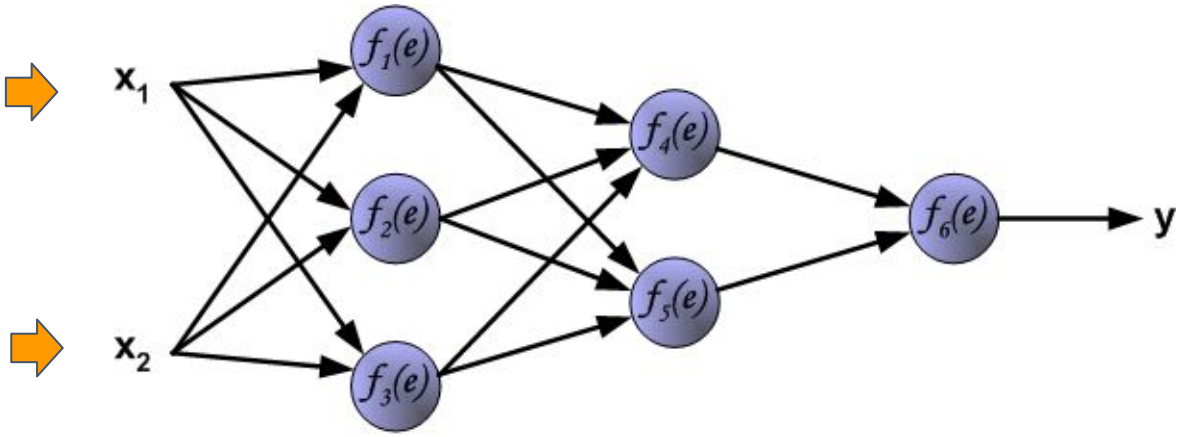
Backpropagation



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

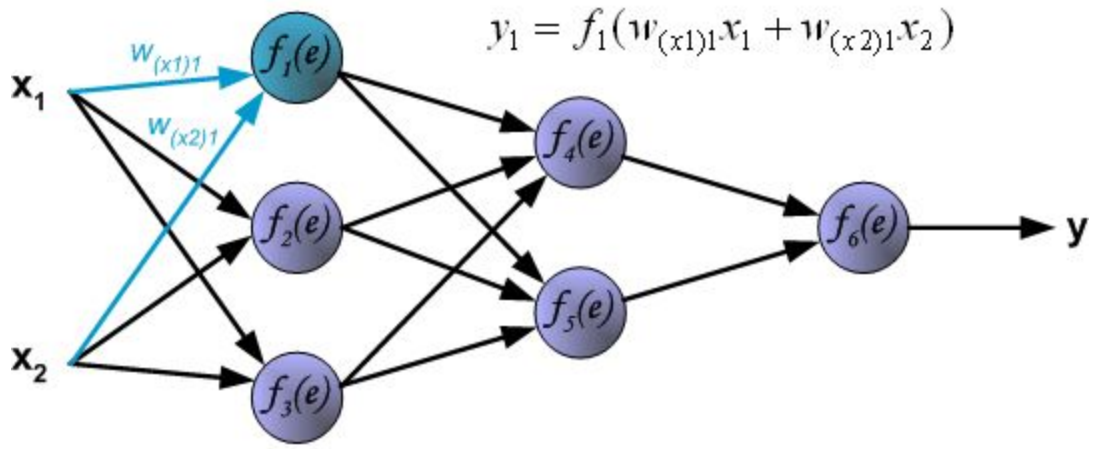
Input Data



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

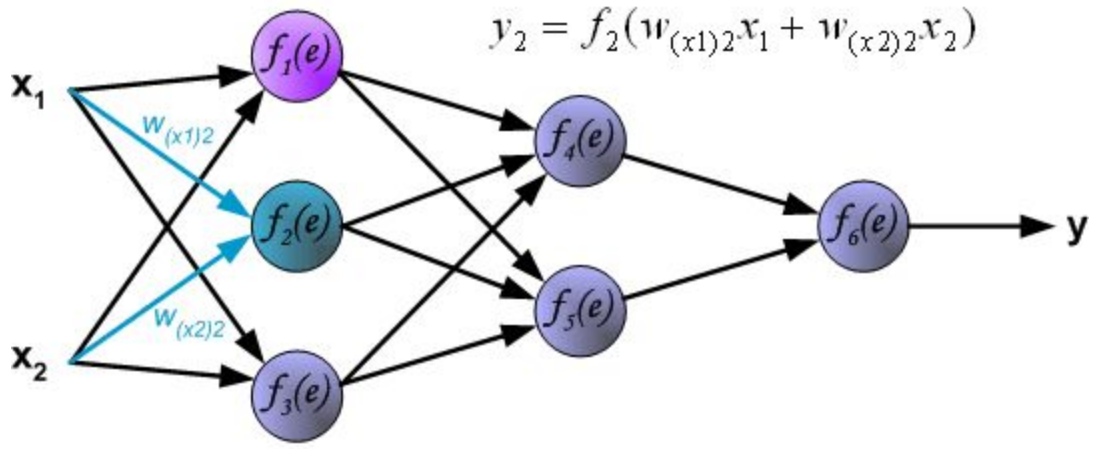
Feed-forward



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

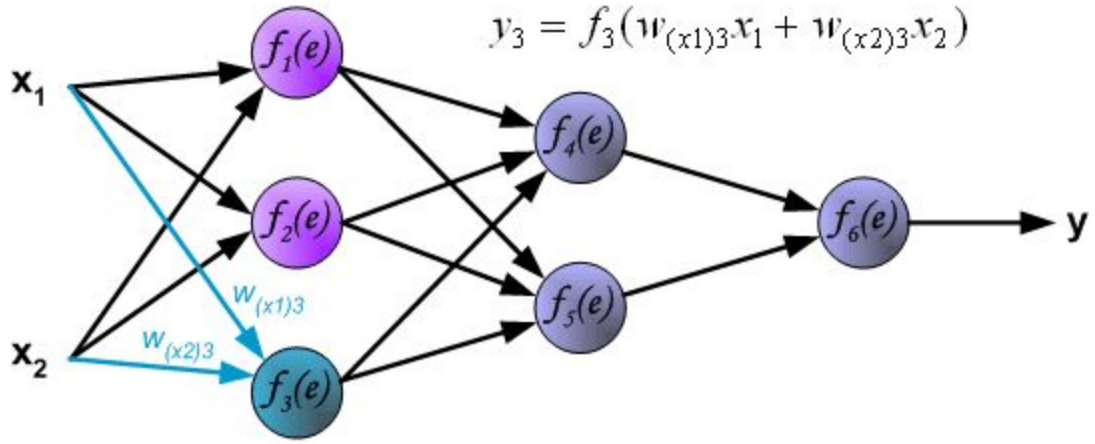
Feed-forward



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

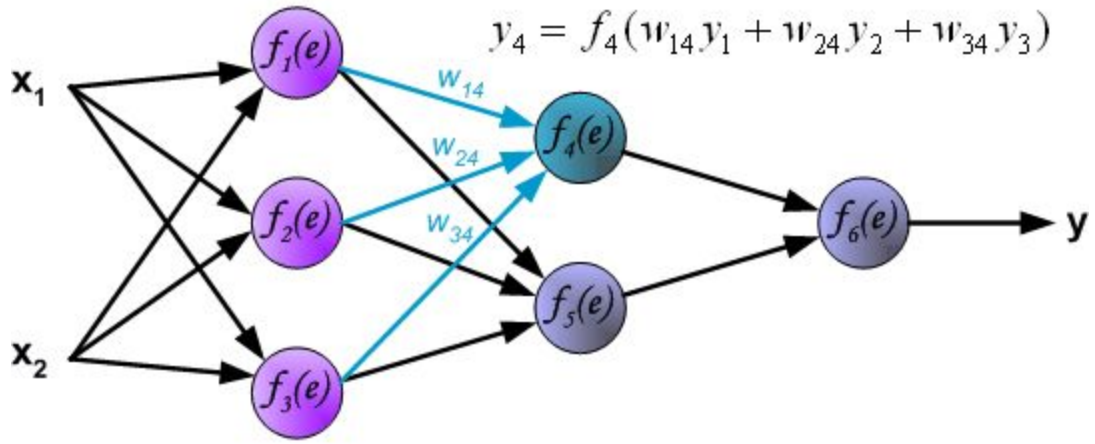
Feed-forward



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

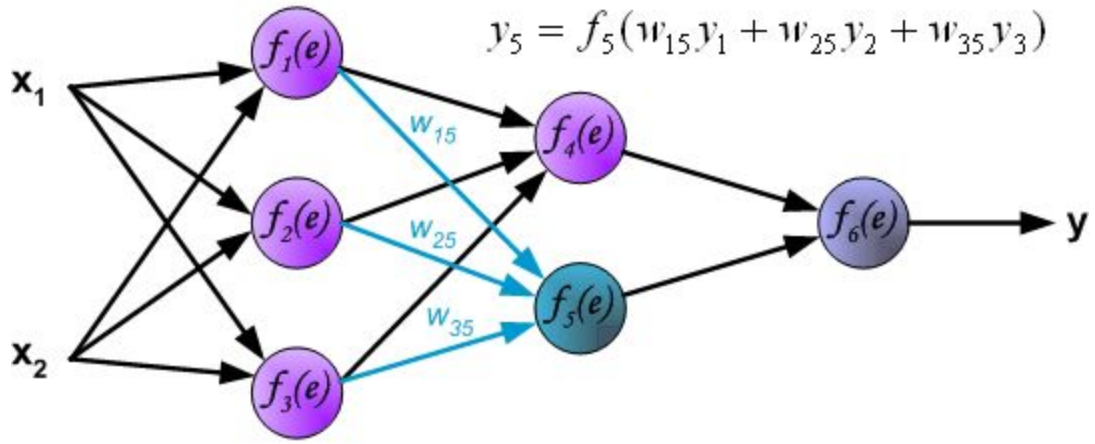
Feed-forward



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

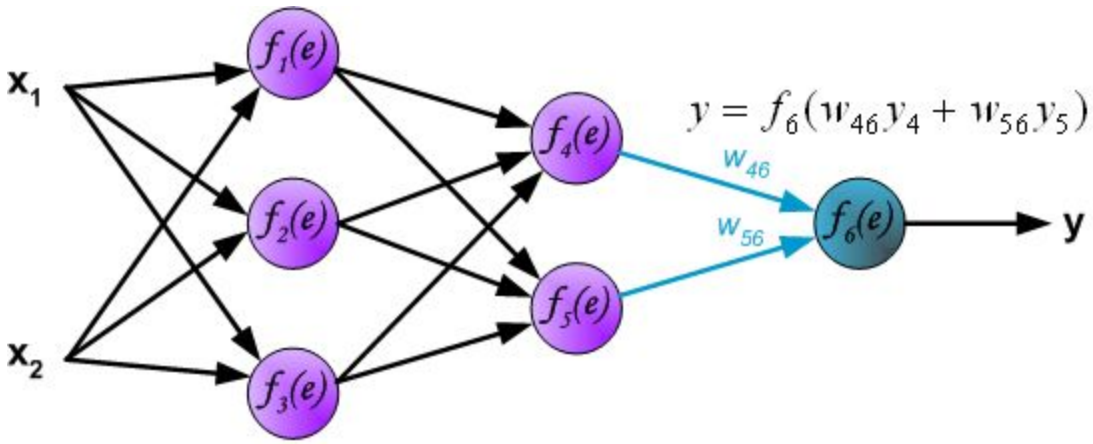
Feed-forward



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

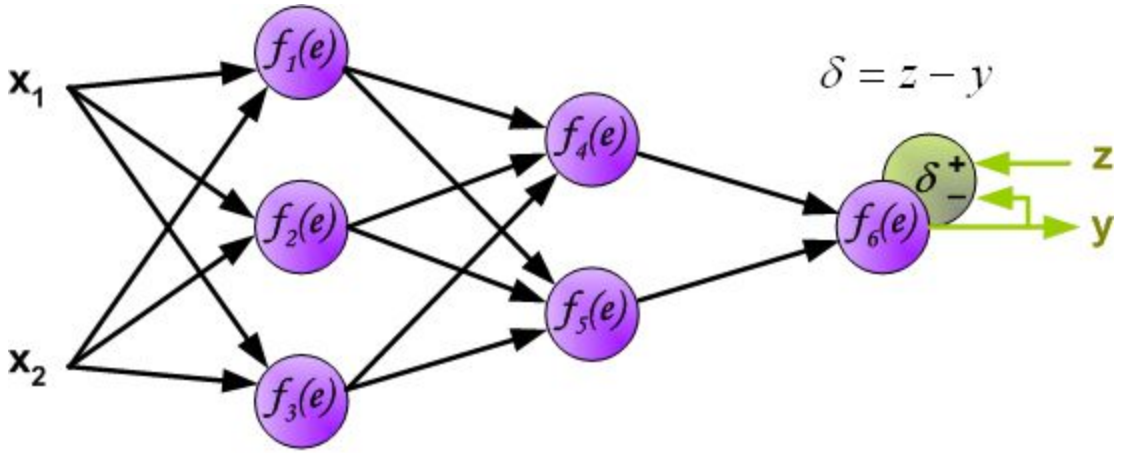
Feed-forward



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

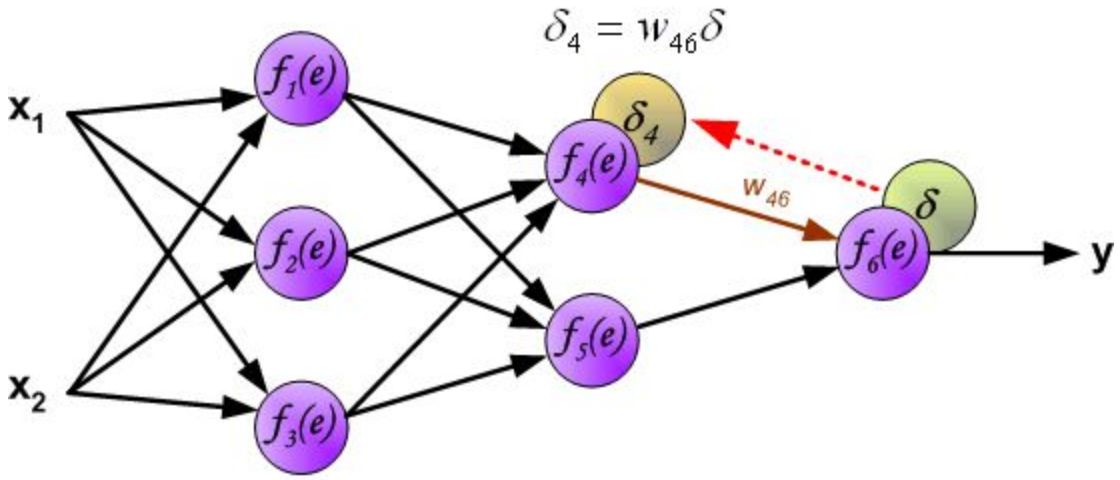
Output error calculation



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

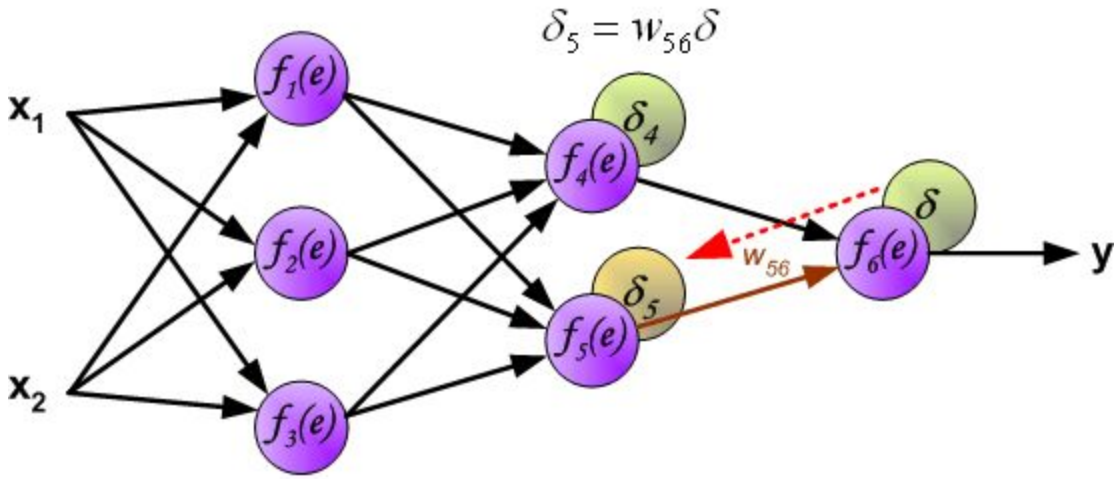
Backpropagate error



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

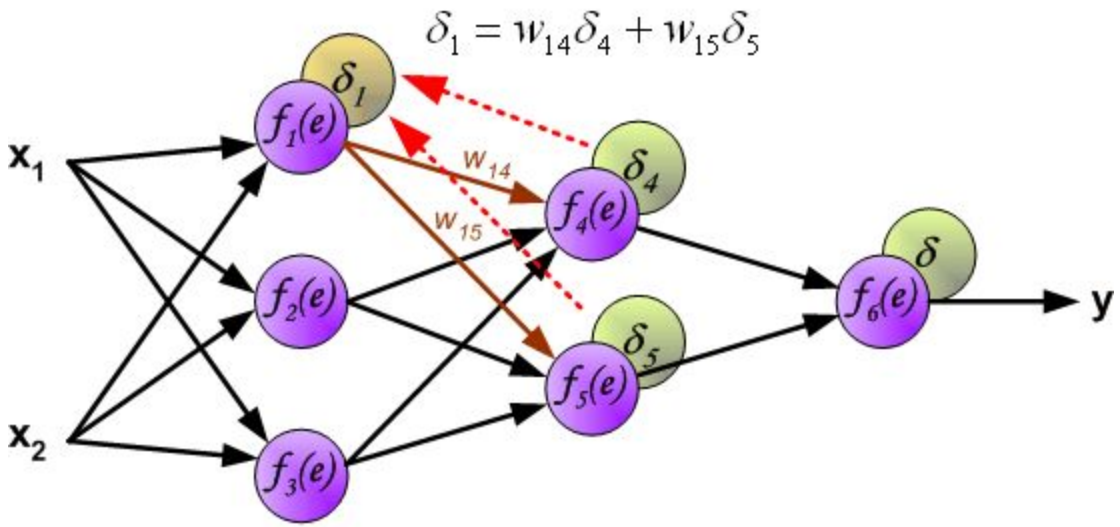
Backpropagate error



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

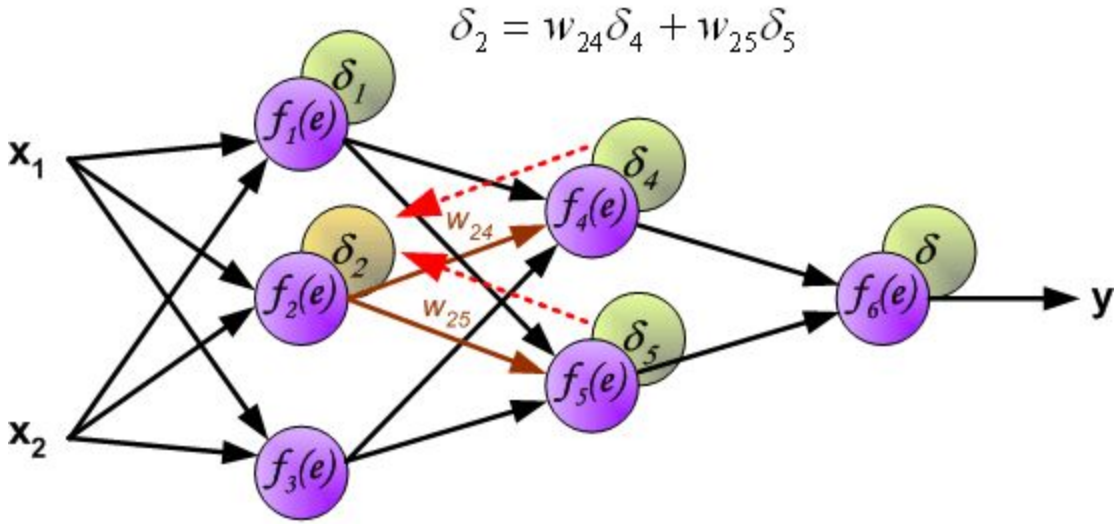
Backpropagate error



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

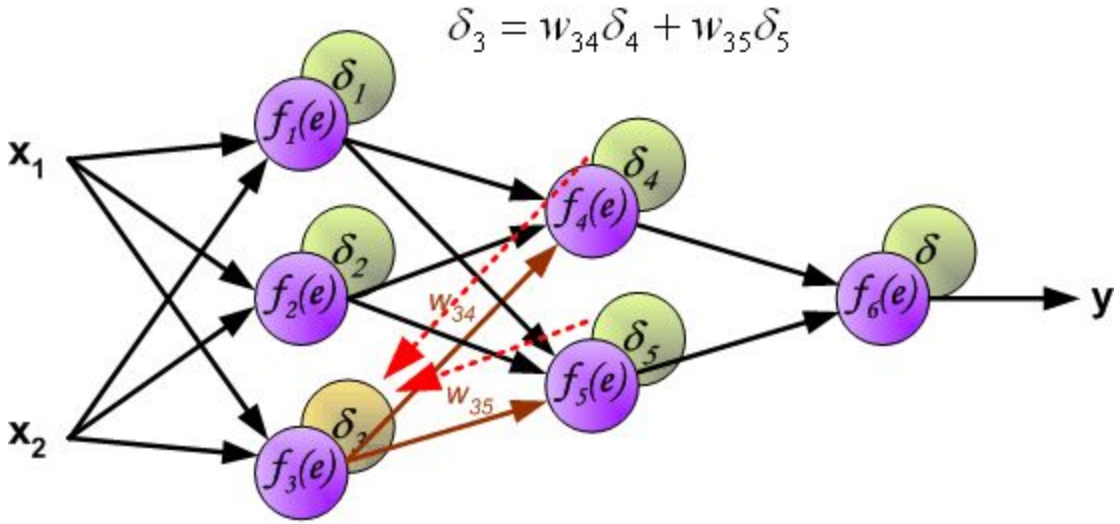
Backpropagate error



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

Backpropagate error



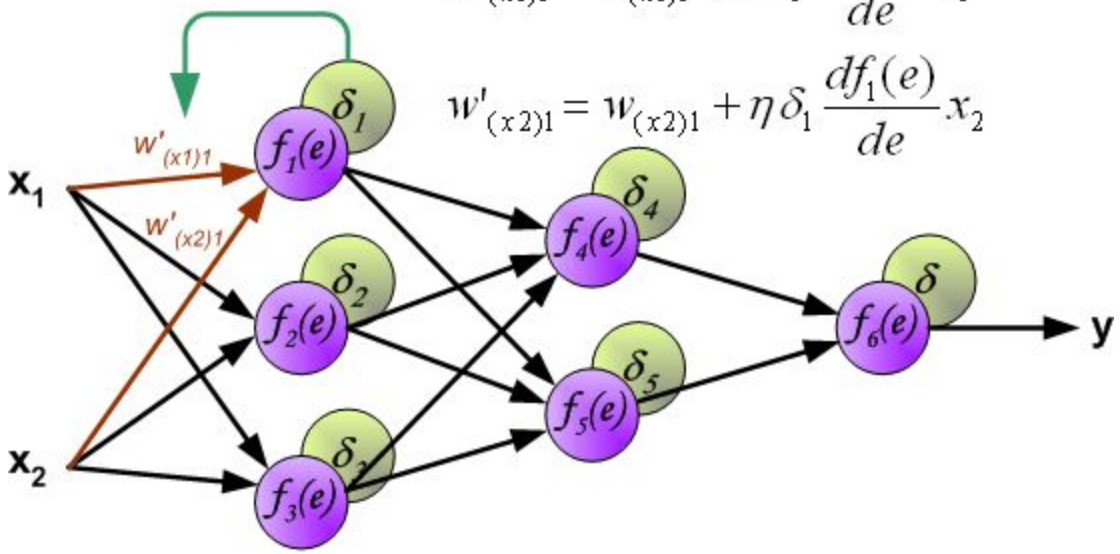
http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

Weights update

$$w'_{(x1)1} = w_{(x1)1} + \eta \delta_1 \frac{df_1(e)}{de} x_1$$

$$w'_{(x2)1} = w_{(x2)1} + \eta \delta_1 \frac{df_1(e)}{de} x_2$$



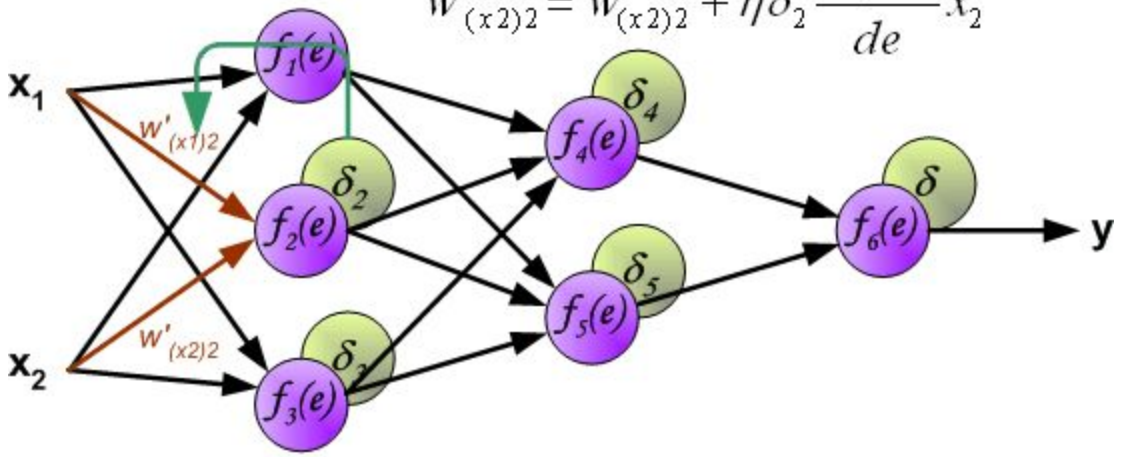
http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

Weights update

$$w'_{(x1)2} = w_{(x1)2} + \eta \delta_2 \frac{df_2(e)}{de} x_1$$

$$w'_{(x2)2} = w_{(x2)2} + \eta \delta_2 \frac{df_2(e)}{de} x_2$$



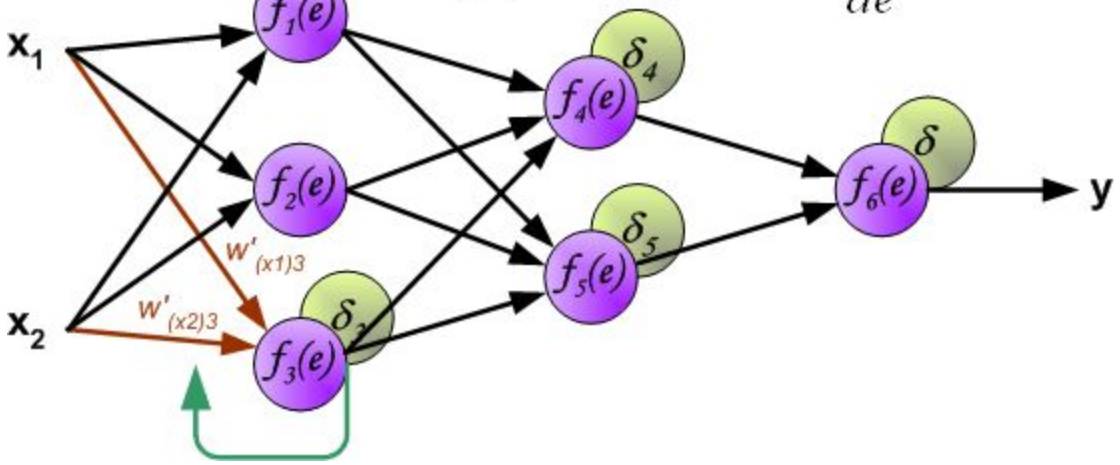
http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

Weights update

$$w'_{(x1)3} = w_{(x1)3} + \eta \delta_3 \frac{df_3(e)}{de} x_1$$

$$w'_{(x2)3} = w_{(x2)3} + \eta \delta_3 \frac{df_3(e)}{de} x_2$$



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

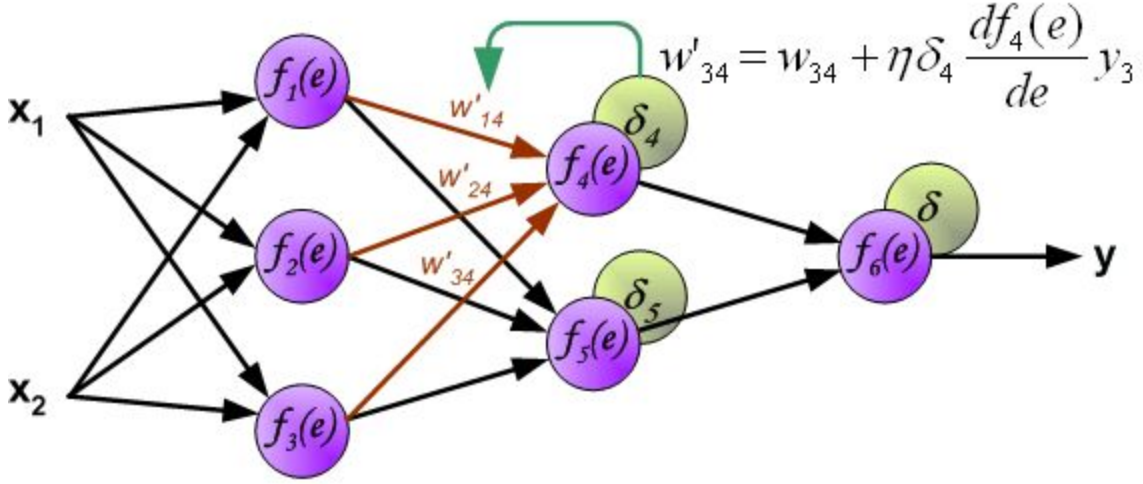
Backpropagation

Weights update

$$w'_{14} = w_{14} + \eta\delta_4 \frac{df_4(e)}{de} y_1$$

$$w'_{24} = w_{24} + \eta\delta_4 \frac{df_4(e)}{de} y_2$$

$$w'_{34} = w_{34} + \eta\delta_4 \frac{df_4(e)}{de} y_3$$



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

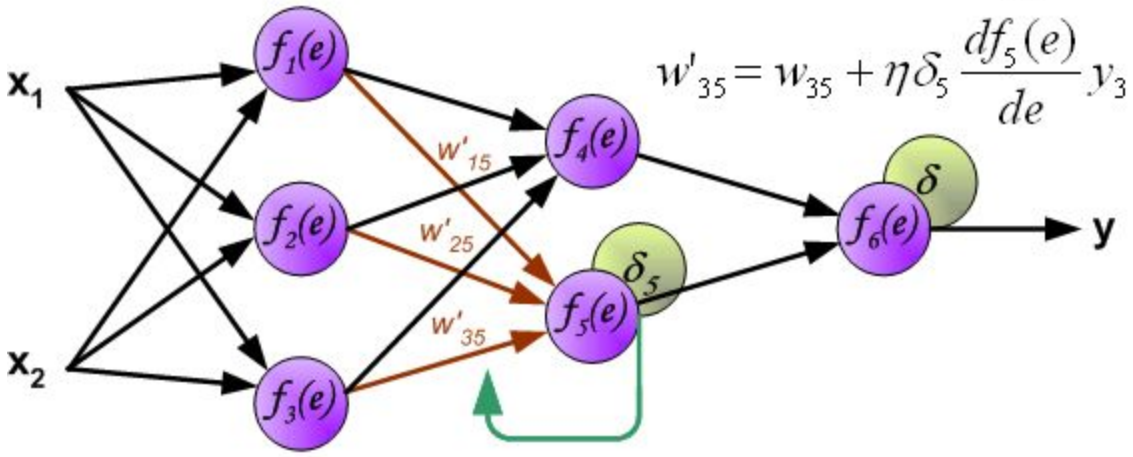
Backpropagation

Weights update

$$w'_{15} = w_{15} + \eta \delta_5 \frac{df_5(e)}{de} y_1$$

$$w'_{25} = w_{25} + \eta \delta_5 \frac{df_5(e)}{de} y_2$$

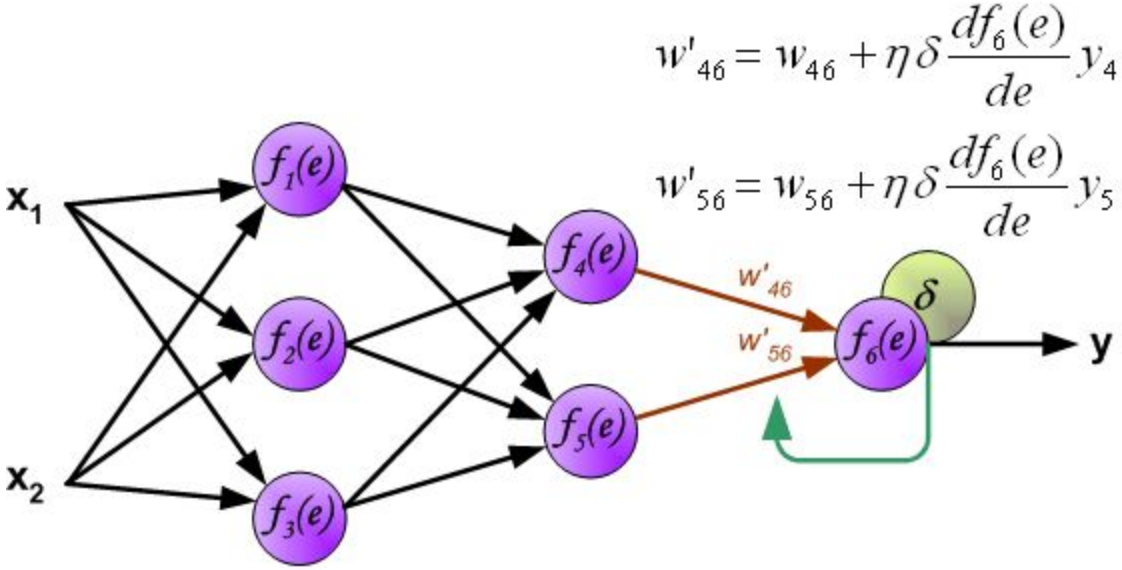
$$w'_{35} = w_{35} + \eta \delta_5 \frac{df_5(e)}{de} y_3$$



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

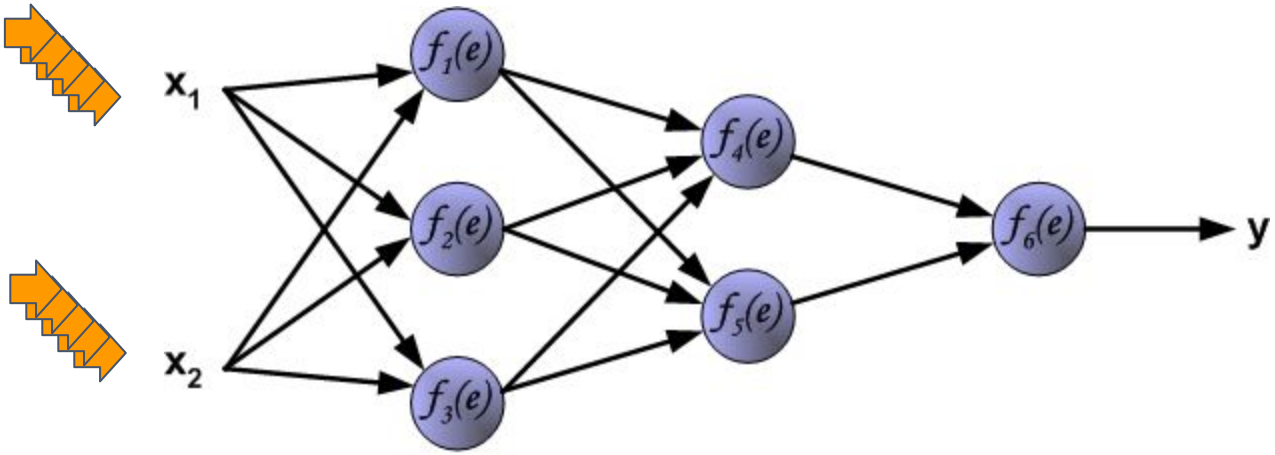
Weights update



http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

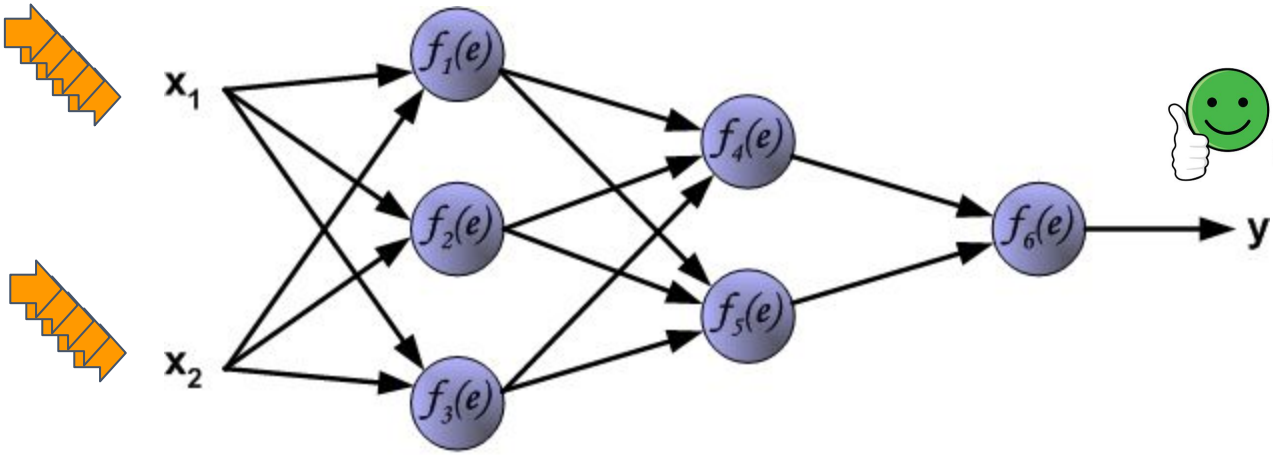
Repeat the process for every training instance...



Fonte: http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Backpropagation

Repeat the process for every training instance…



…*n* times or until the output error be acceptable.

Fonte: http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

Neural Networks

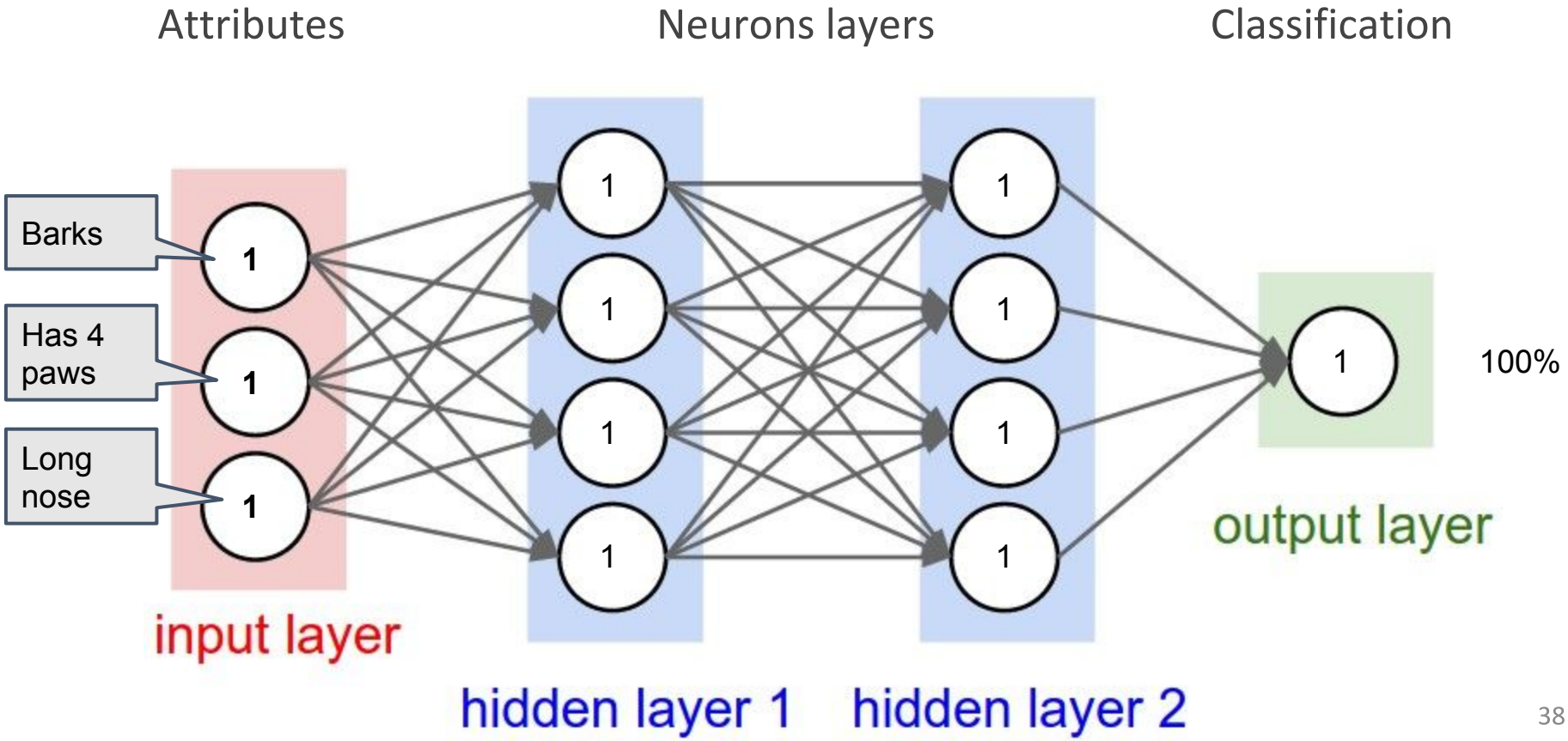
$$\frac{1}{1+e^{-x}}$$

Activation Function Derivative

$$\begin{aligned}
\frac{\partial \sigma}{\partial x} &= \frac{\partial}{\partial x}\left[\frac{1}{1+e^{-x}}\right] \\
&= \frac{\partial}{\partial x}(1+e^{-x})^{-1} \\
&= -(1+e^{-x})^{-2}(-e^{-x}) \\
&= \frac{e^{-x}}{(1+e^{-x})^2} \\
&= \frac{1}{1+e^{-x}} \cdot \frac{e^{-x}}{1+e^{-x}} \\
&= \frac{1}{1+e^{-x}} \cdot \frac{(1+e^{-x})-1}{1+e^{-x}} \\
&= \frac{1}{1+e^{-x}} \cdot \left(1 - \frac{1}{1+e^{-x}}\right) \\
&= \sigma(x) \cdot (1-\sigma(x))
\end{aligned}$$

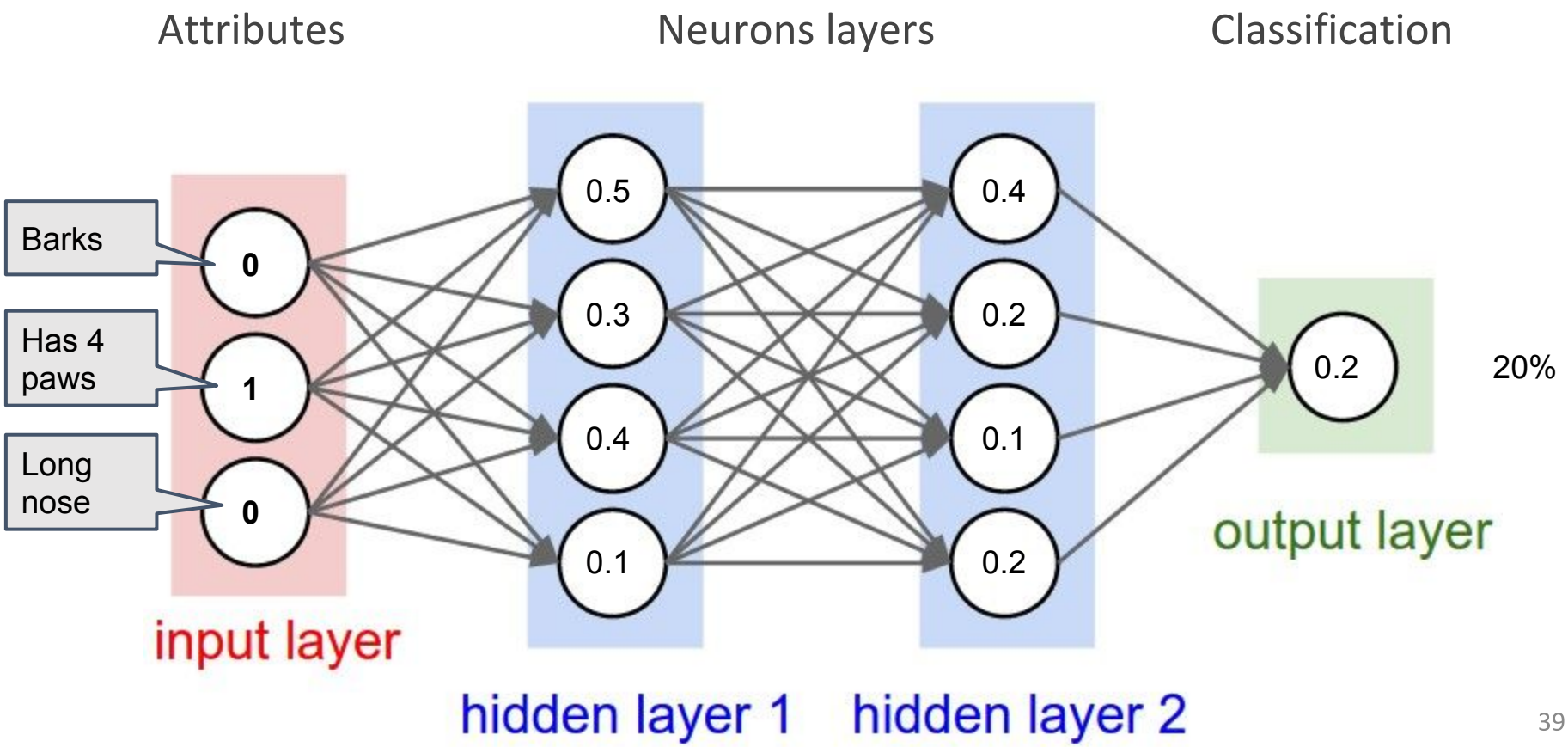
Neural Networks

Toy network for detecting dogs:



Neural Networks

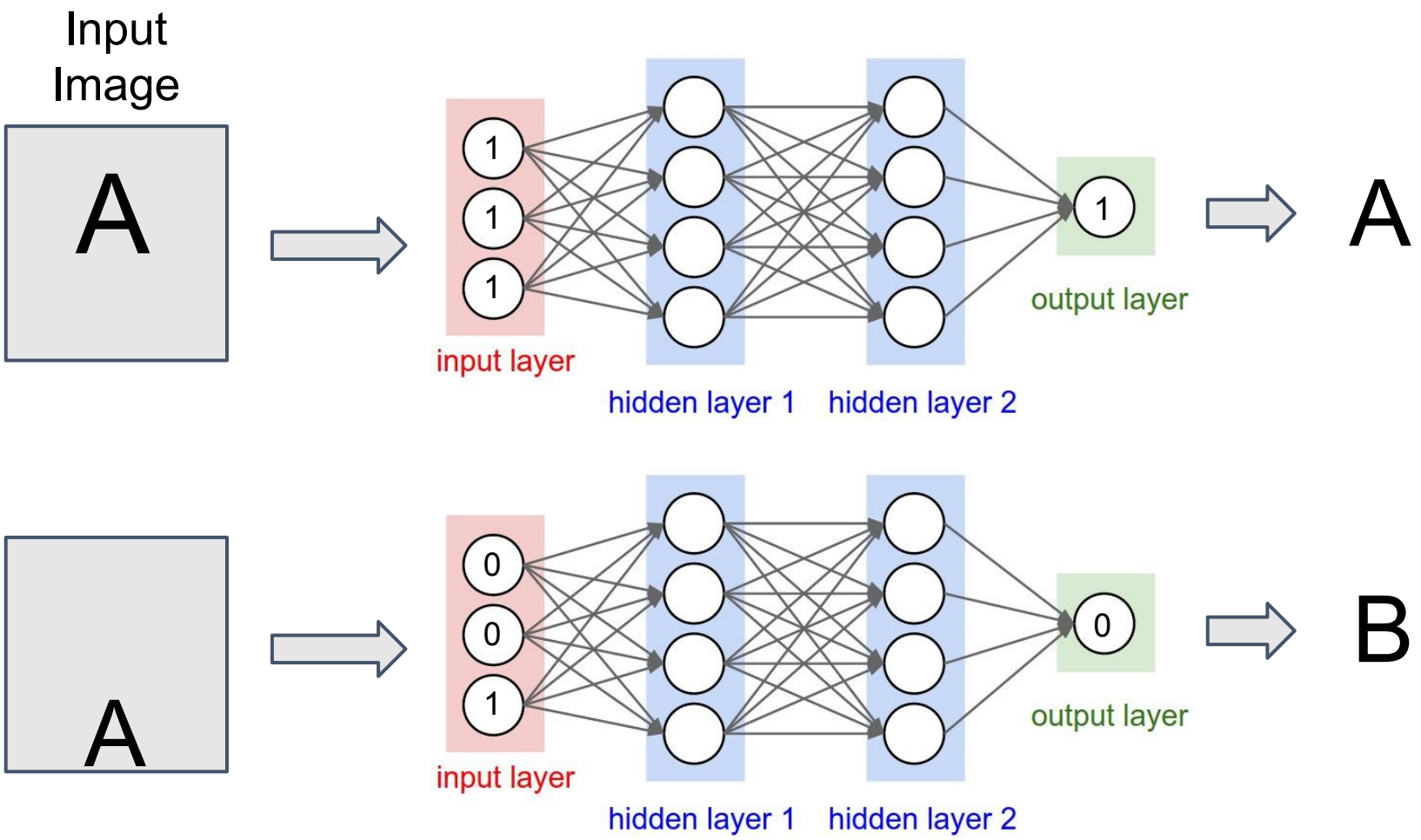
Toy network for detecting dogs:



The image classification challenge

- Image 32 height x 32 width x 3 channels (RGB)
  - 3072 weights (connections)
- Image 200 height x 200 width x 3 channels (RGB)
  - 120,000 weights
- This full connectivity is wasteful and the huge number of parameters would quickly lead to overfitting.
- MLP are not sensitive to input patterns invariance.

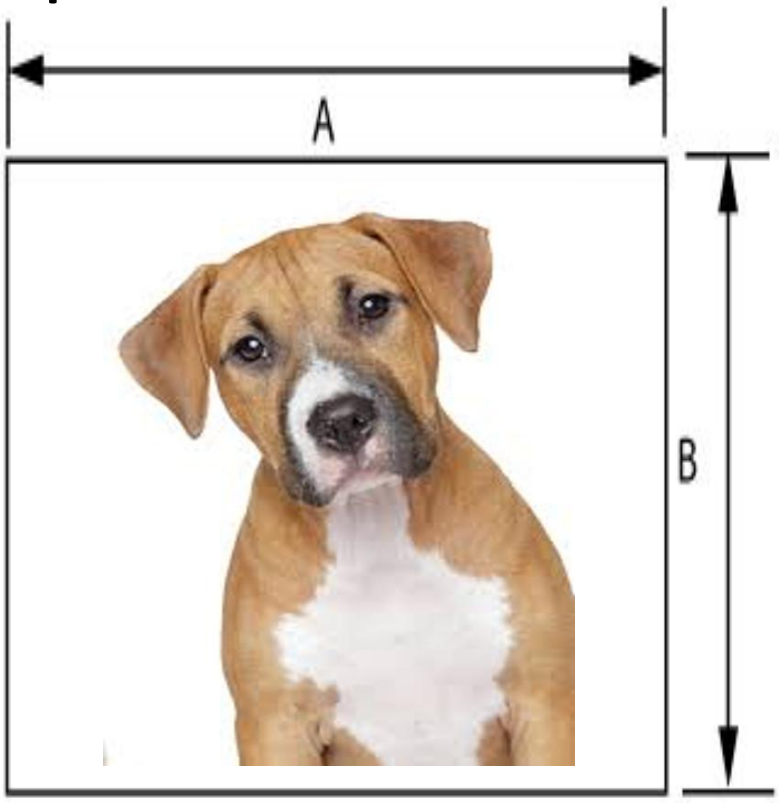
Neural Network Invariance Problem



Fonte: http://cs231n.github.io/neural-networks-1/

The image classification problem

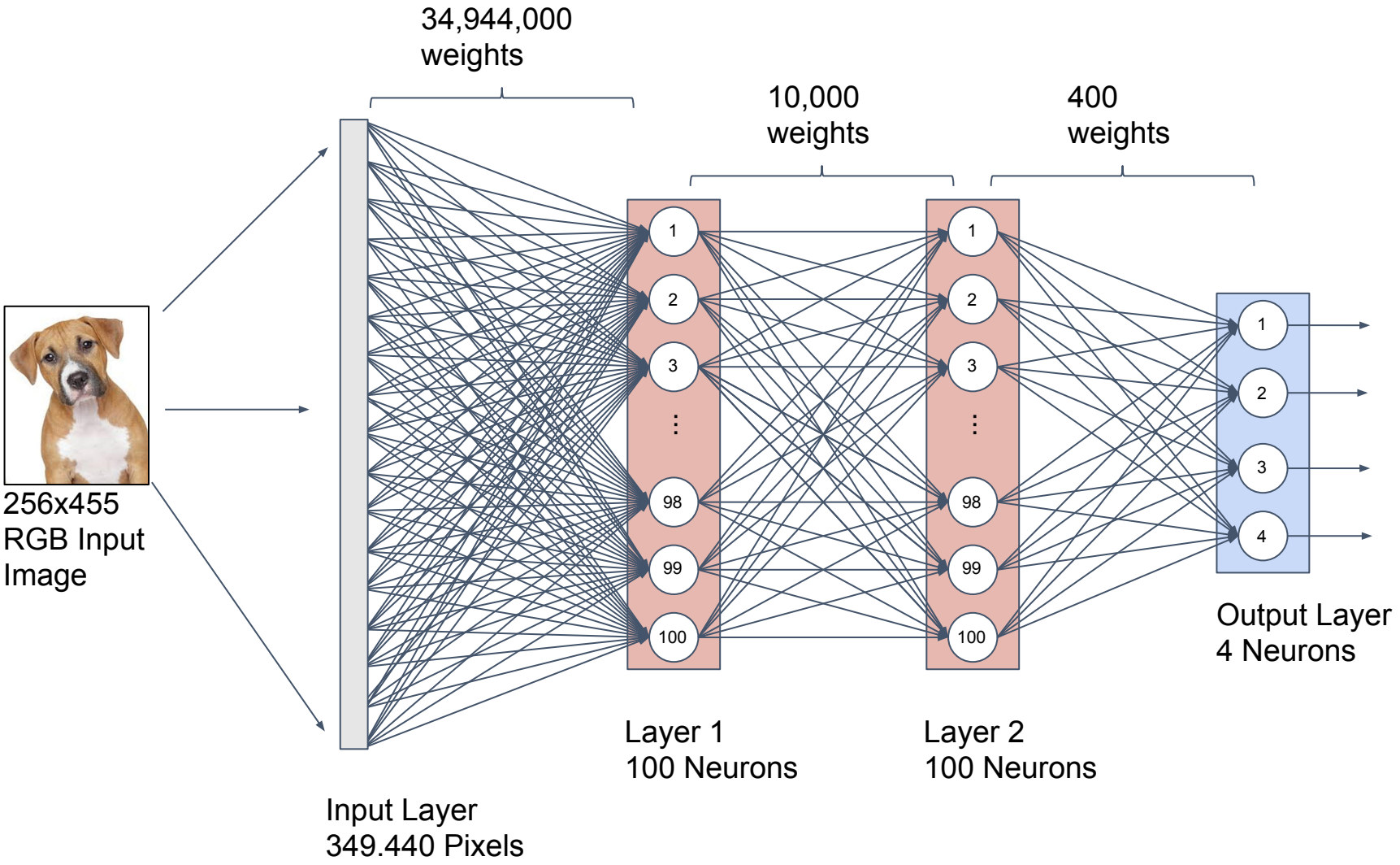
Dimension



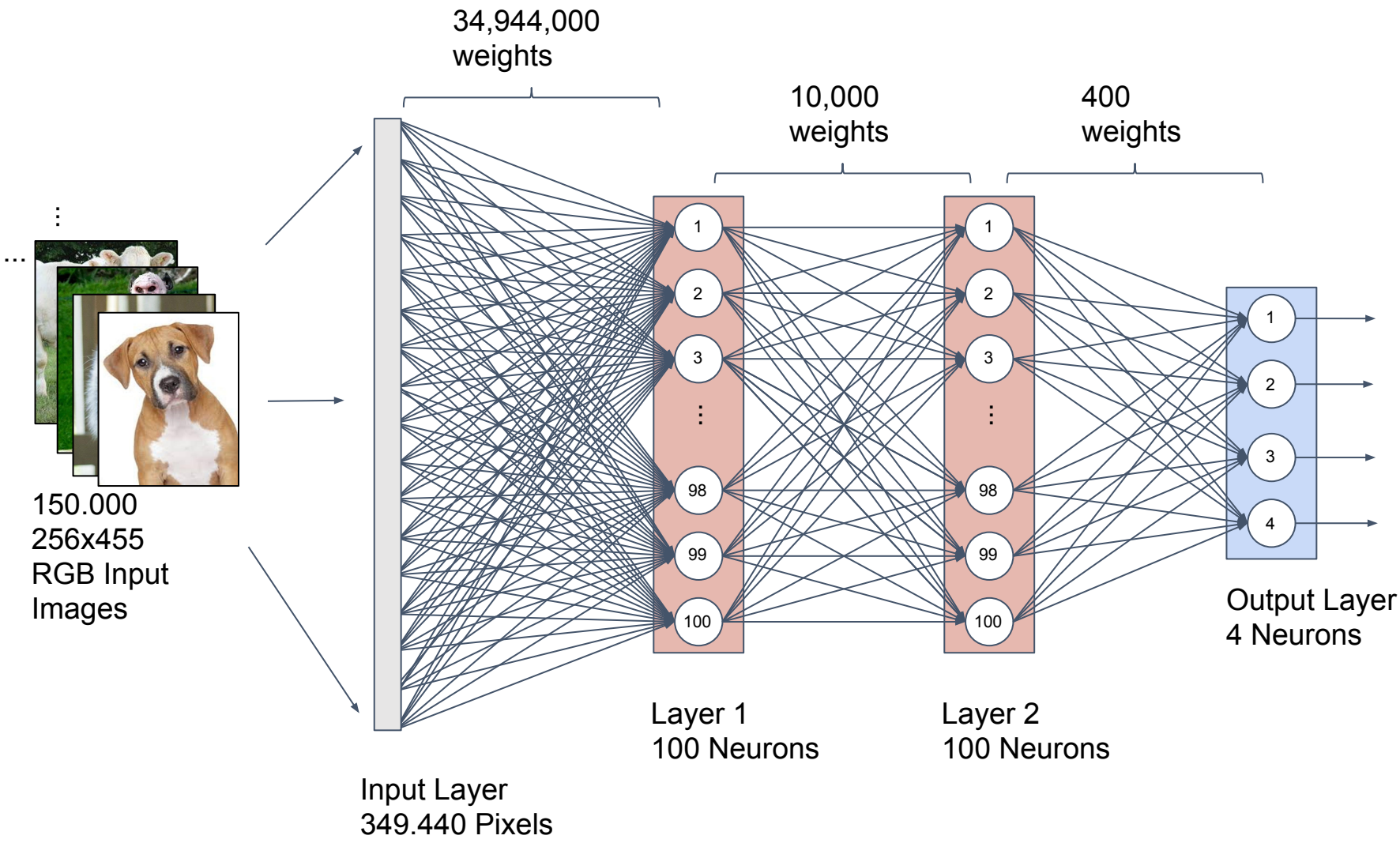
For a 256x455x3 image we have 349,440 inputs.

How to train a neural net with an input layer like that?

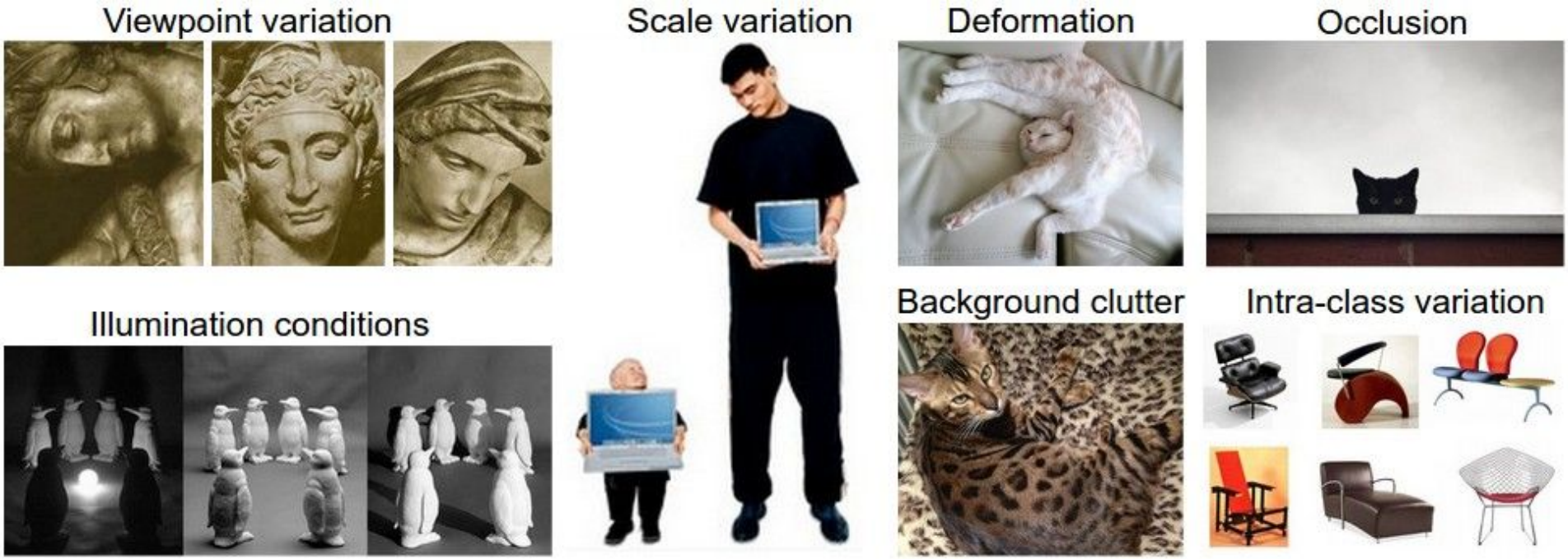
Learning a 256x455 RGB Dog and...



… a Dog in 150k 256x455 RGB Images!



The image classification problem



Fonte: http://cs231n.github.io/classification/

Convolutional Neural Network

When the magic happens again!

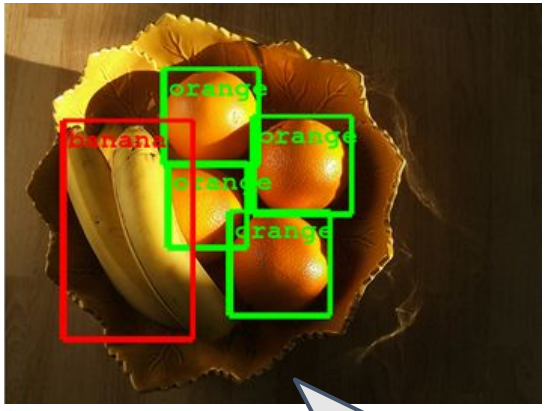
The Convolutional Neural Network

Convolutional neural networks (CNN) are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers.

Fonte: http://www.iro.umontreal.ca/~bengioy/dlbook/convnets.html

Applications

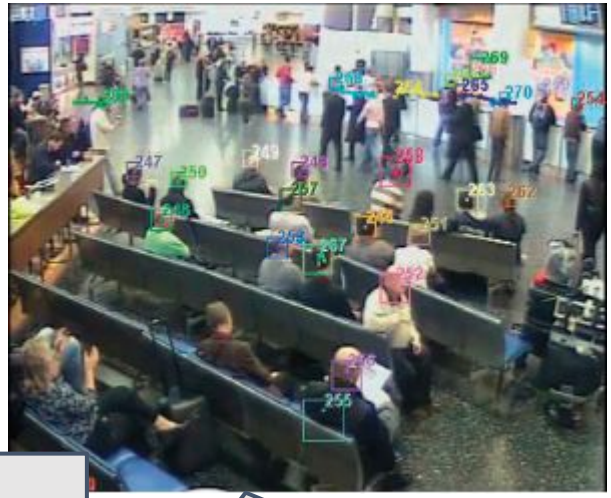
Image Recognition



Scalable Object Detection using Deep Neural Networks
Erhan D. et al (2014)

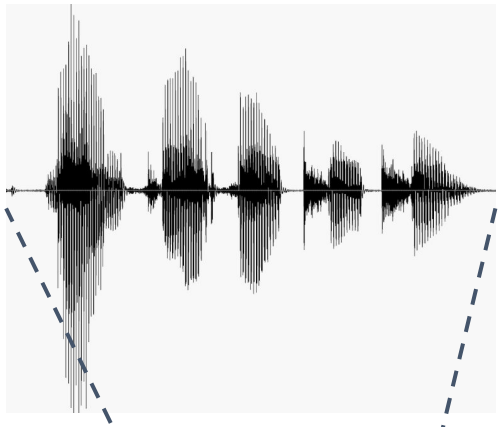
ImageNet Classification with Deep Convolutional Neural Networks.
Alex Krizhevsky et al (2012).

Video Analysis



3D Convolutional Neural Networks for Human Action Recognition
Shuiwang Ji et al (2013)

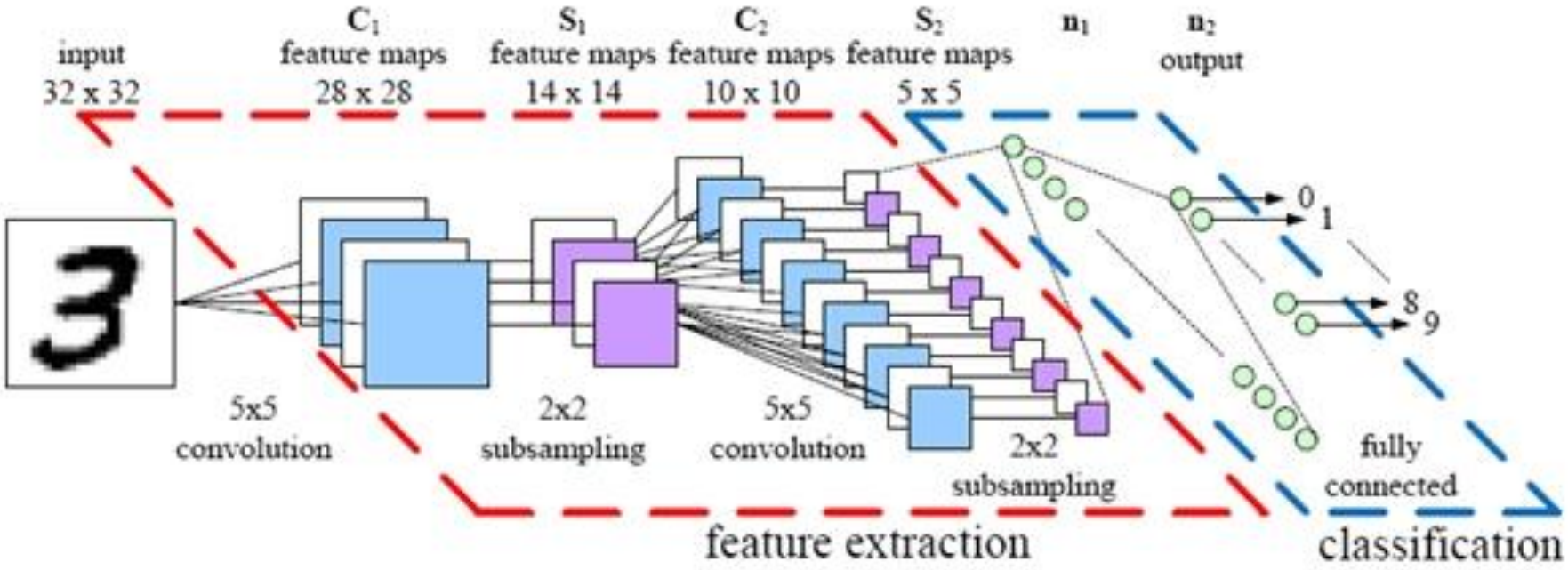
Natural Language Processing



Previsão do tempo para Porto Alegre

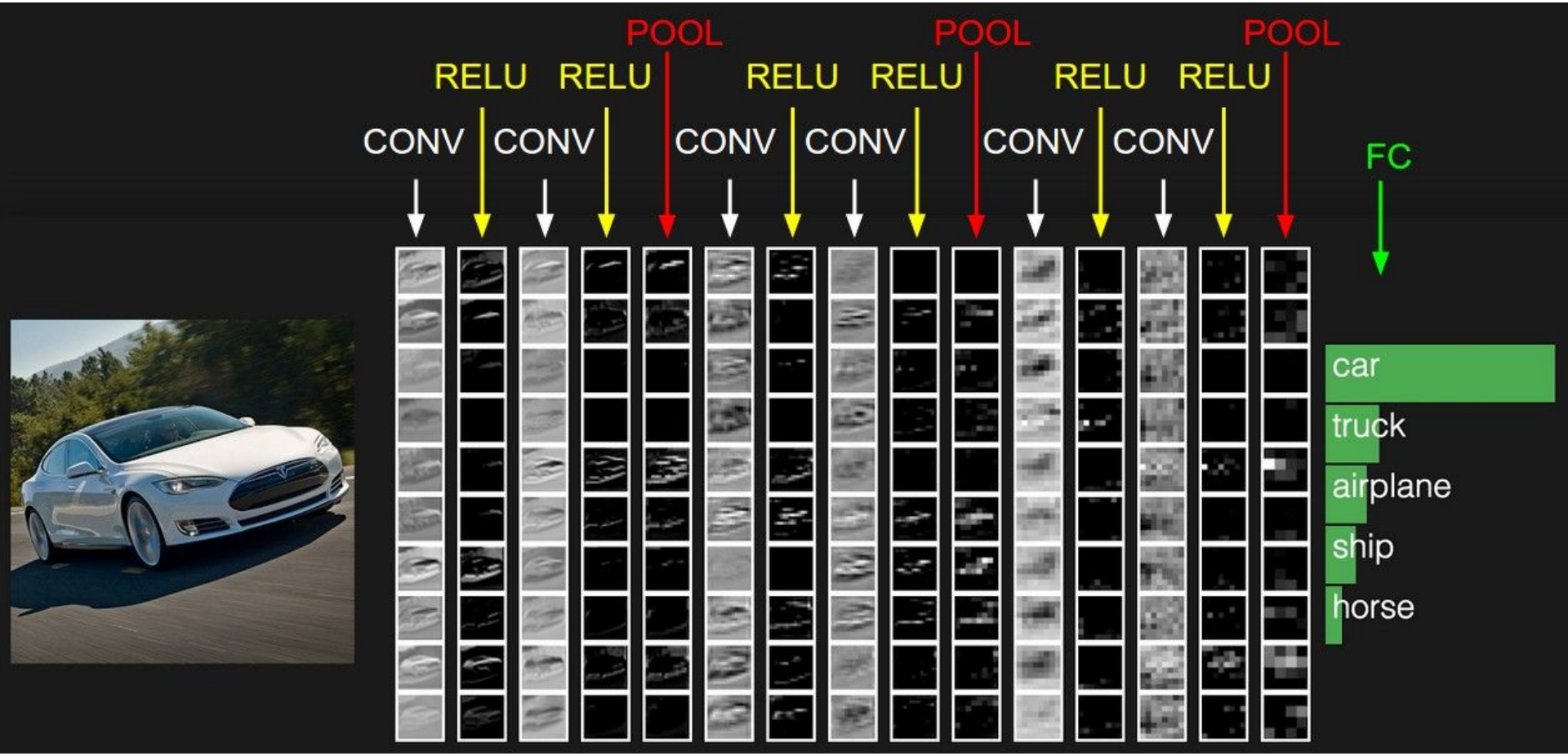
Applying Convolutional Neural Networks Concepts To Hybrid NN Model For Speech Recognition.
Abdel-Hamid et al (2012)

The Convolutional Neural Network



http://www.es.ele.tue.nl/~dshe/Education/Dmm2012

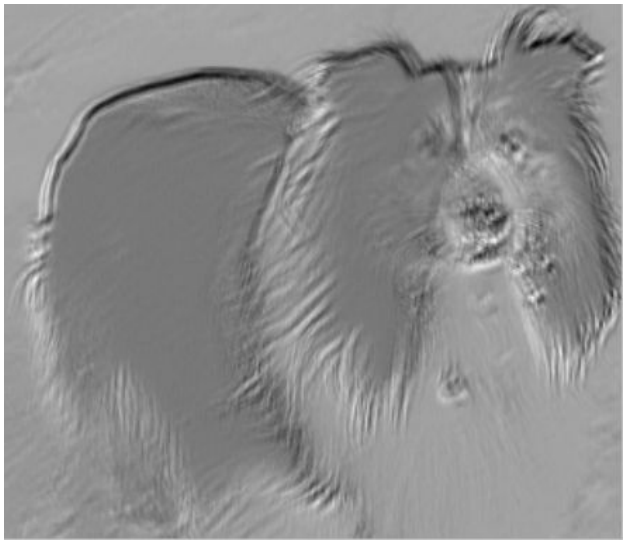
The Convolutional Neural Network



RELU: Rectified Linear Units
Fonte: http://cs231n.stanford.edu/

The Convolutional Neural Network

Convolution



The Convolutional Neural Network

→ What is a convolution (*)?

$$\mathbf{G_x} = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * \mathbf{A} \quad \text{e} \quad \mathbf{G_y} = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * \mathbf{A}$$

Image A

Resulting image: $G_x + G_y$

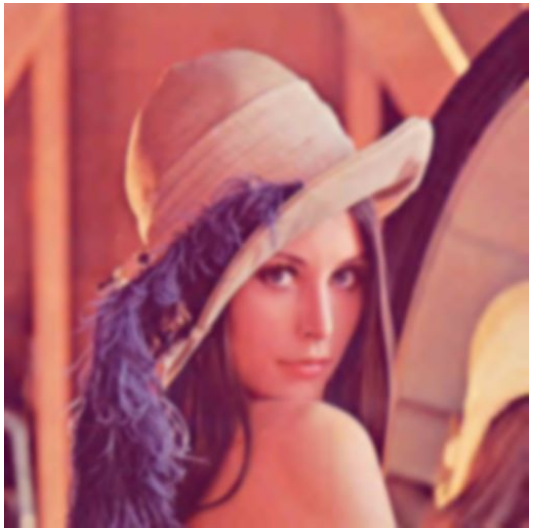
The Convolutional Neural Network

→ What is a convolution (*)?

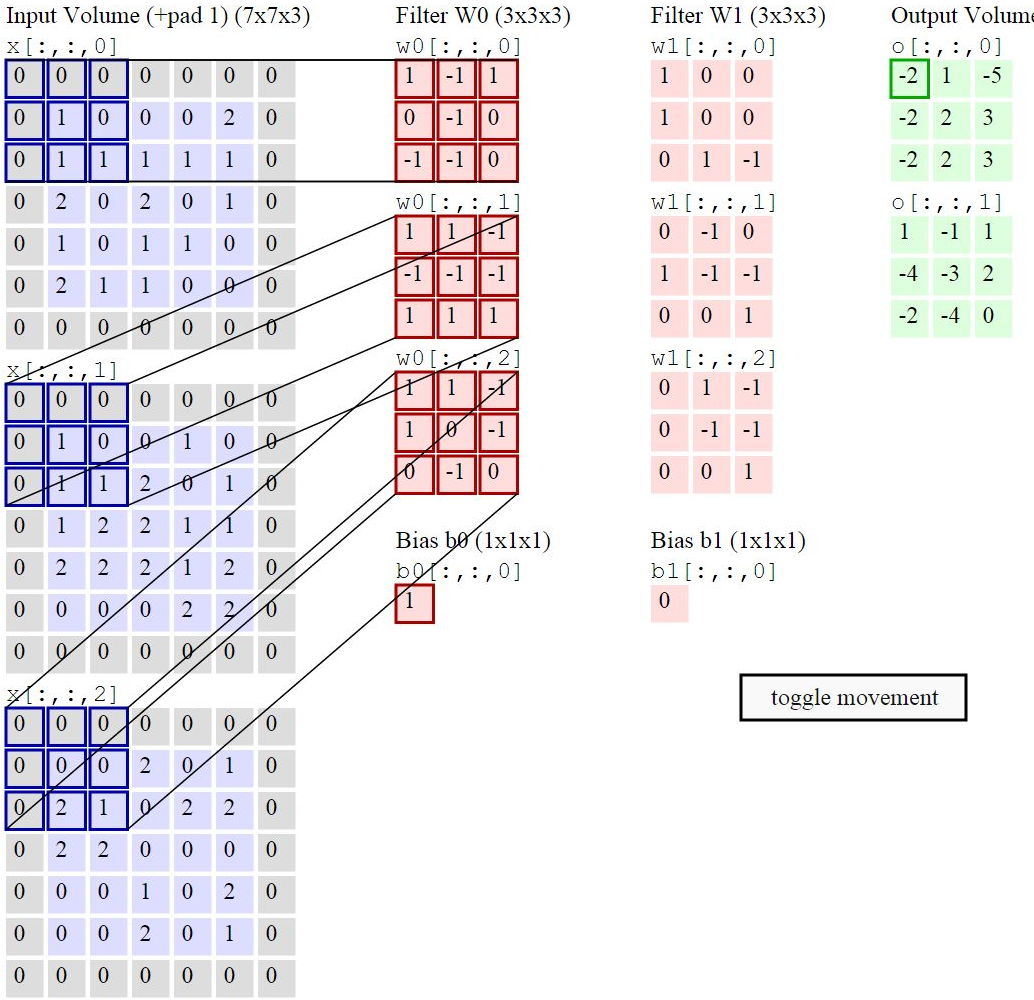
$$G = \frac{1}{273} \cdot \begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix} * A$$

Image A

Resulting image G



The Convolutional Neural Network

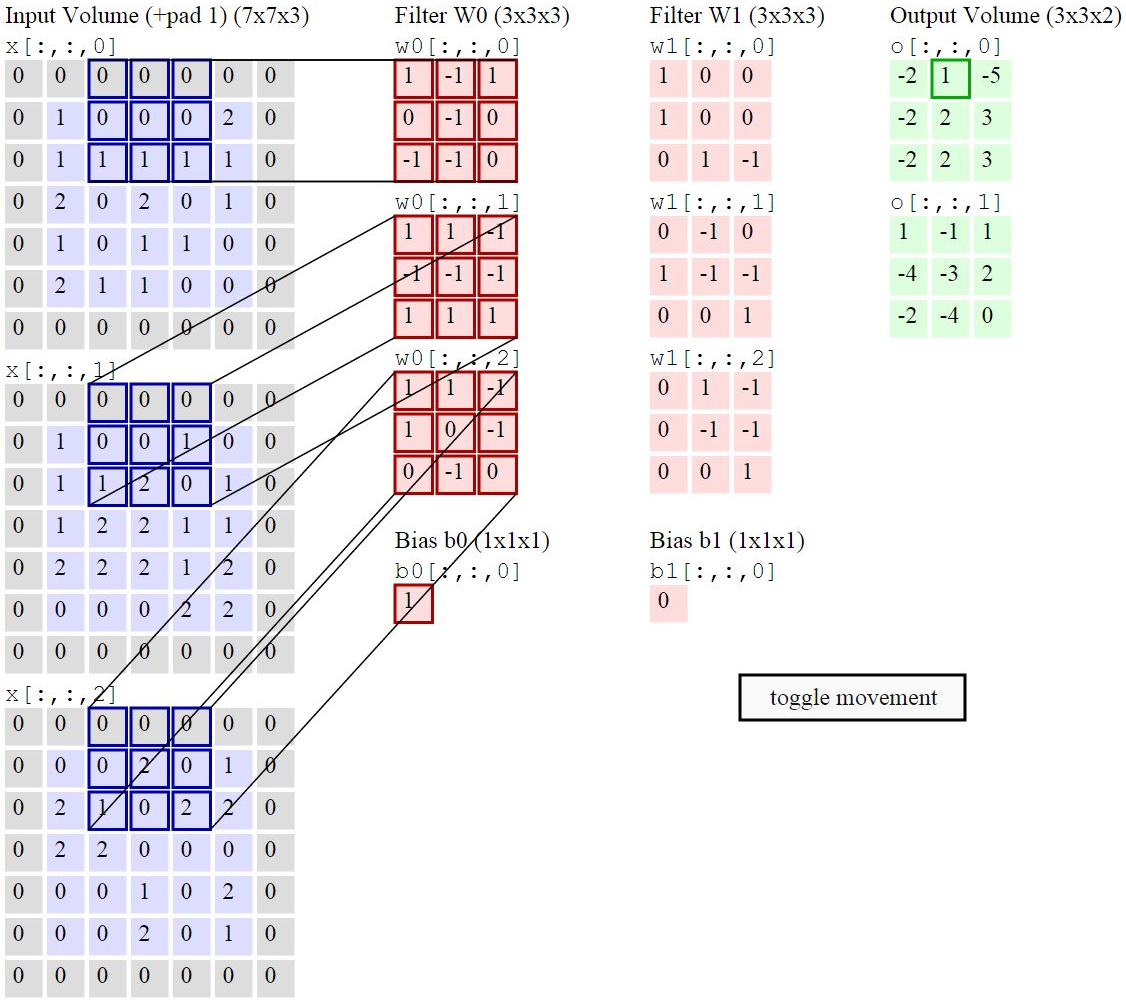


Input Image: 5x5x3
Padding: 1
Input Image (+pad): 7x7x3
Filters: 2
Filters Shape: 3x3
Stride: 2

(5 + 1 * 2 - 3) / 2 + 1 = 3

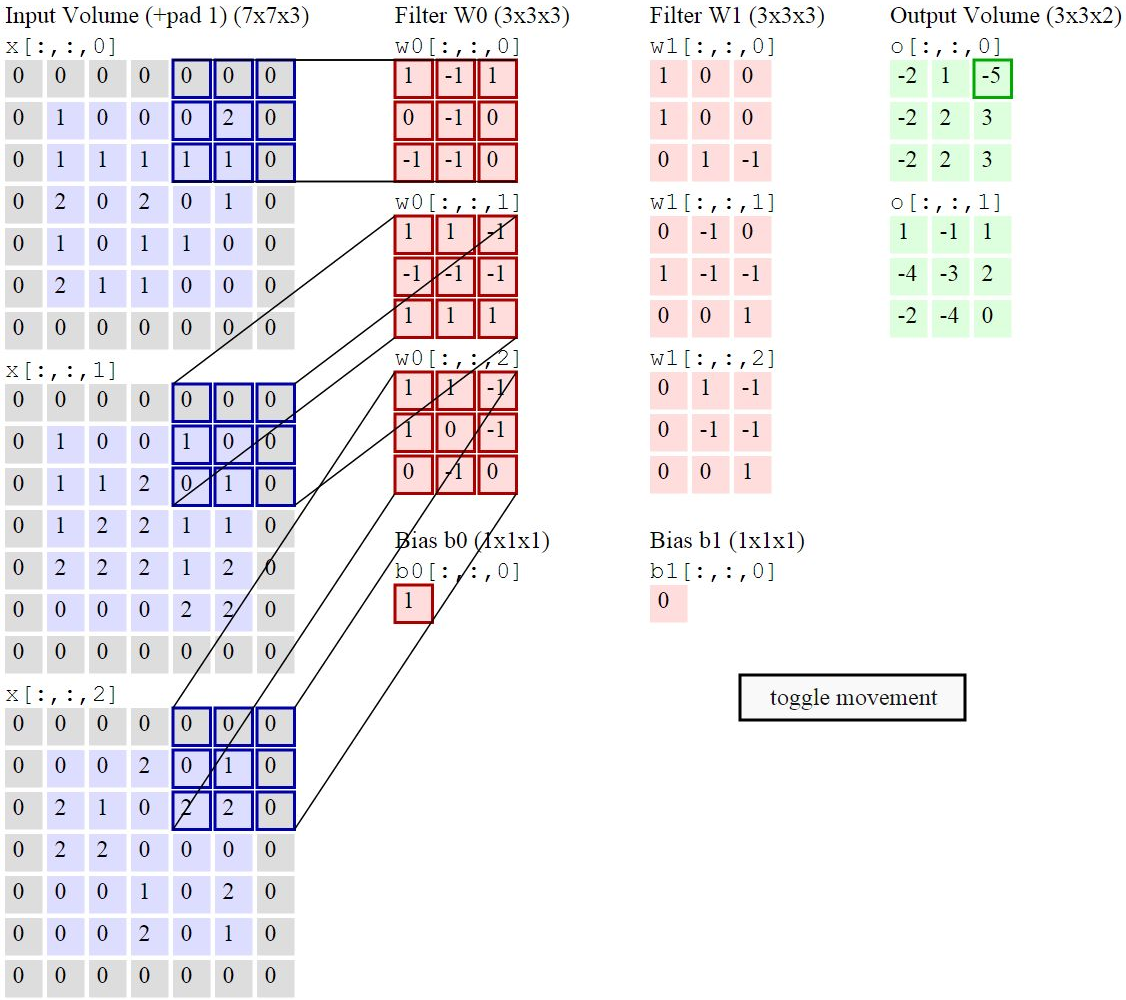
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



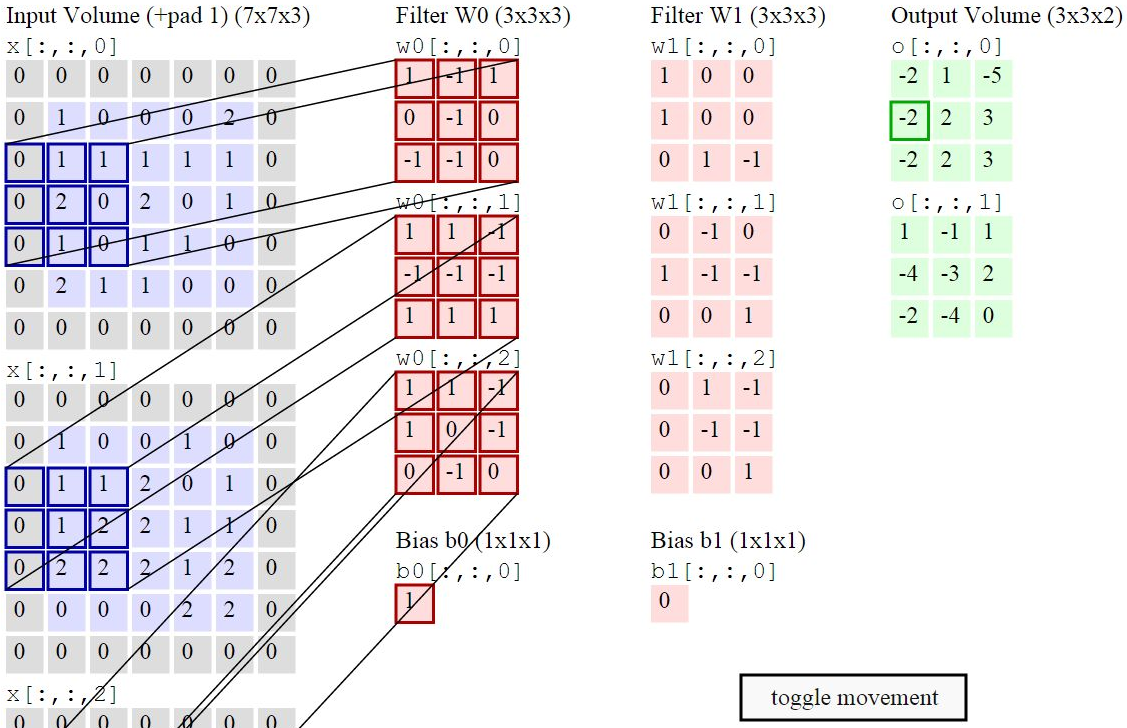
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



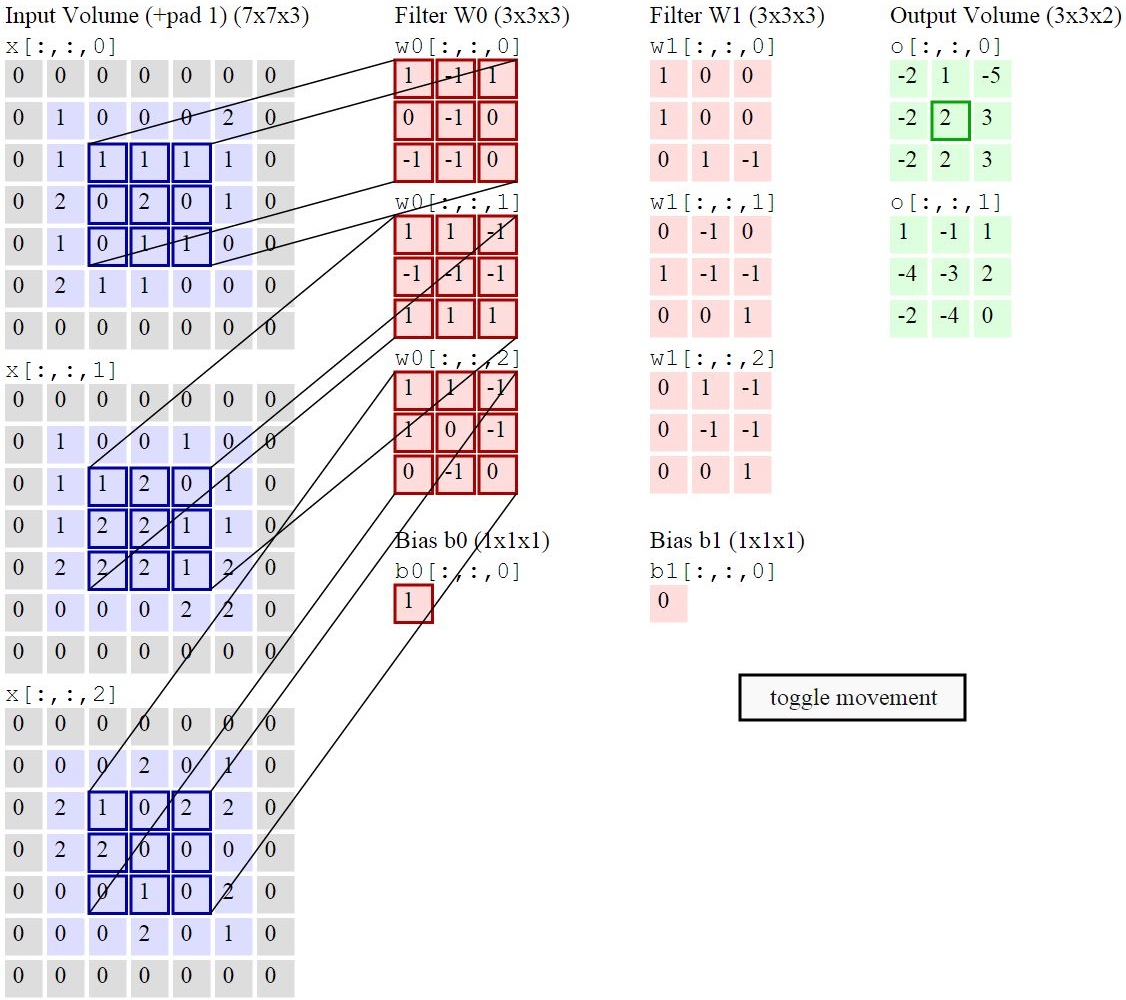
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



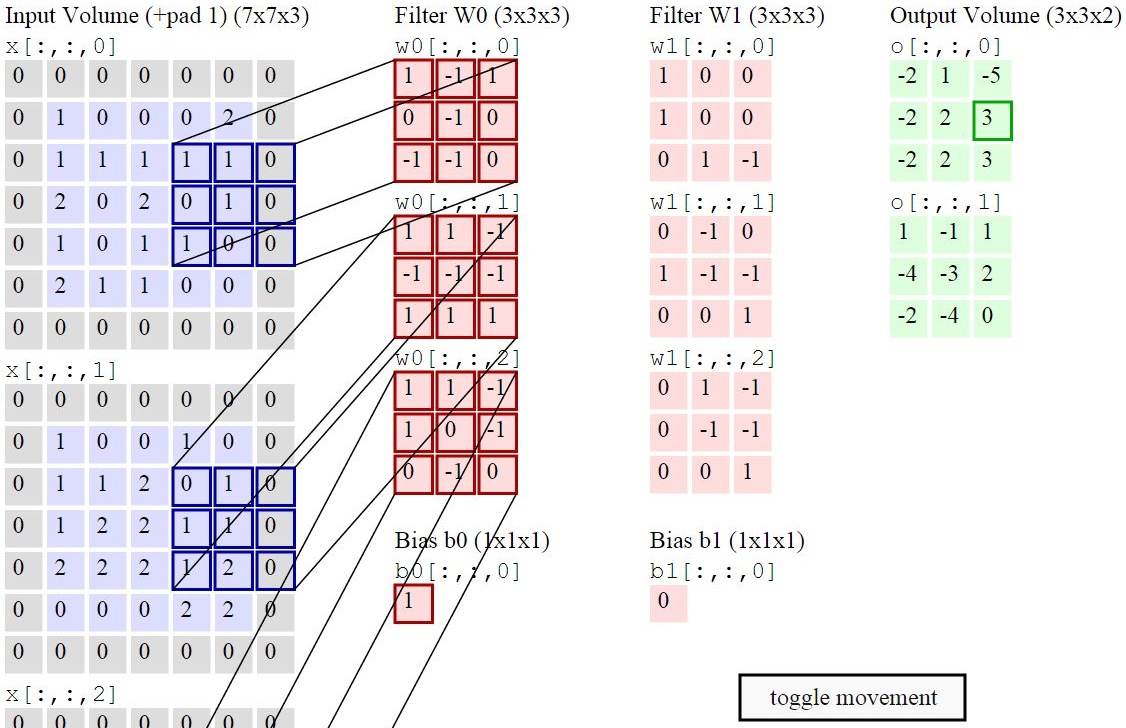
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



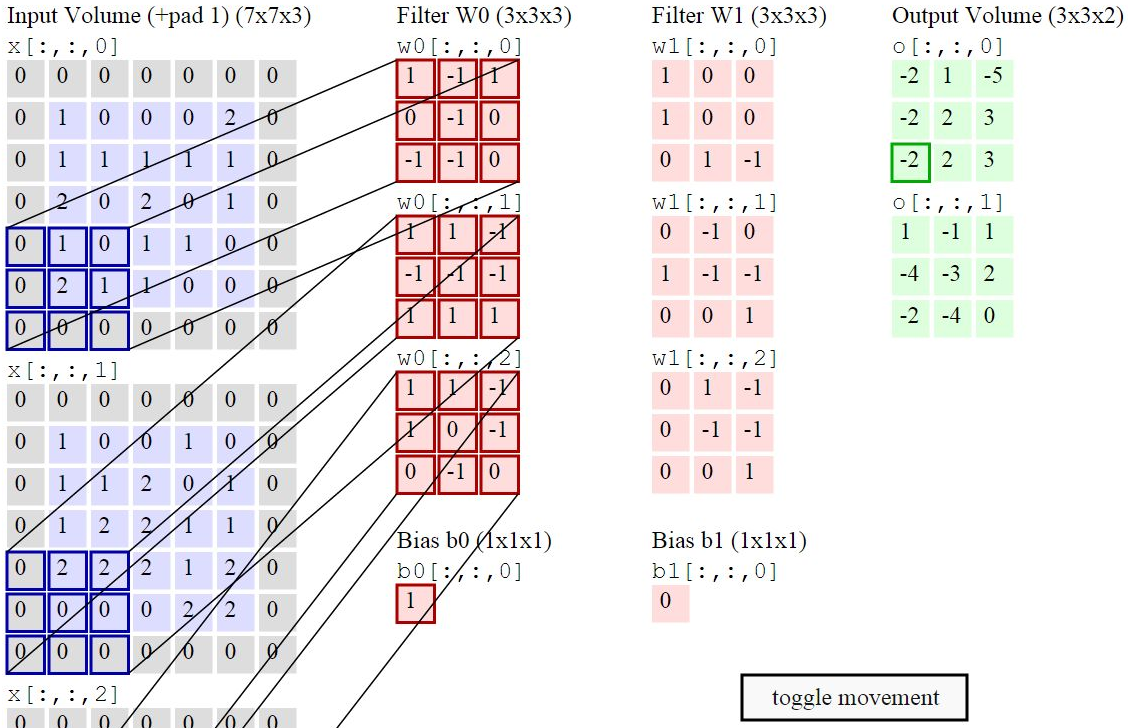
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



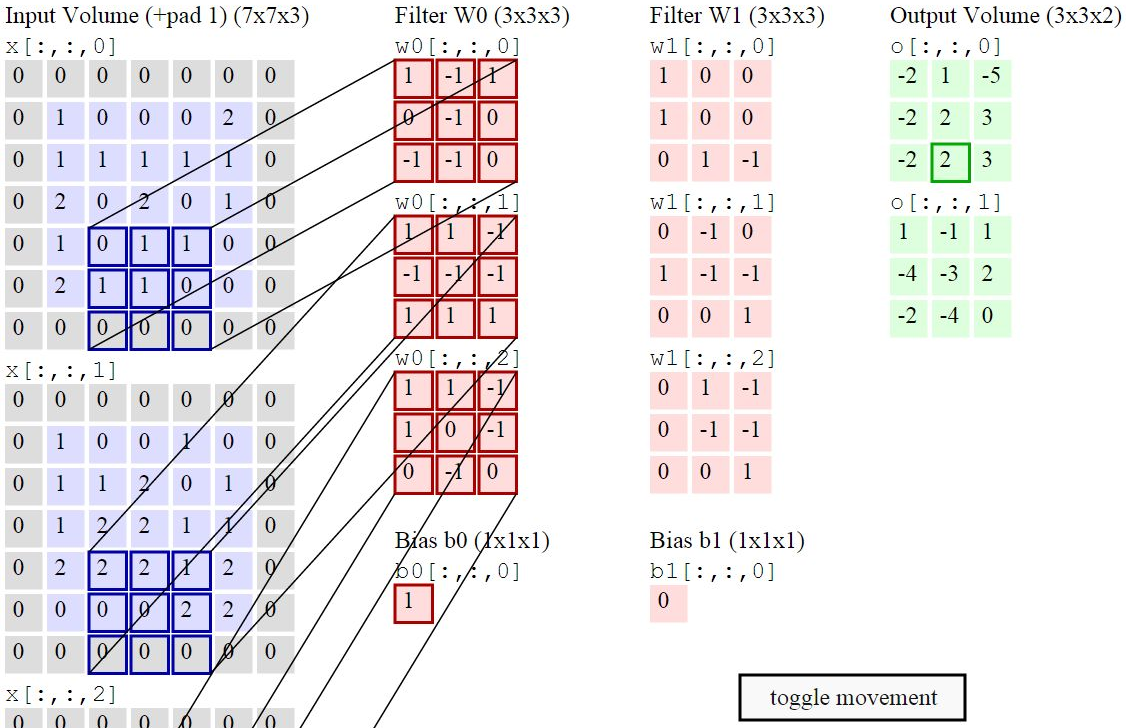
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



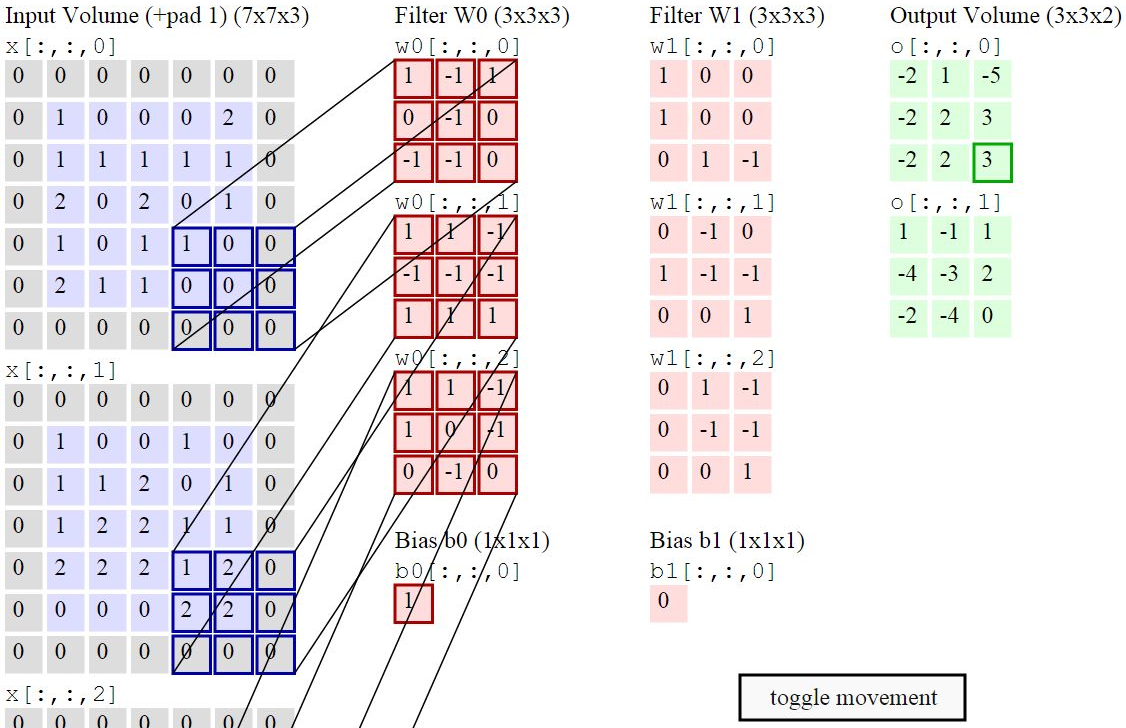
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



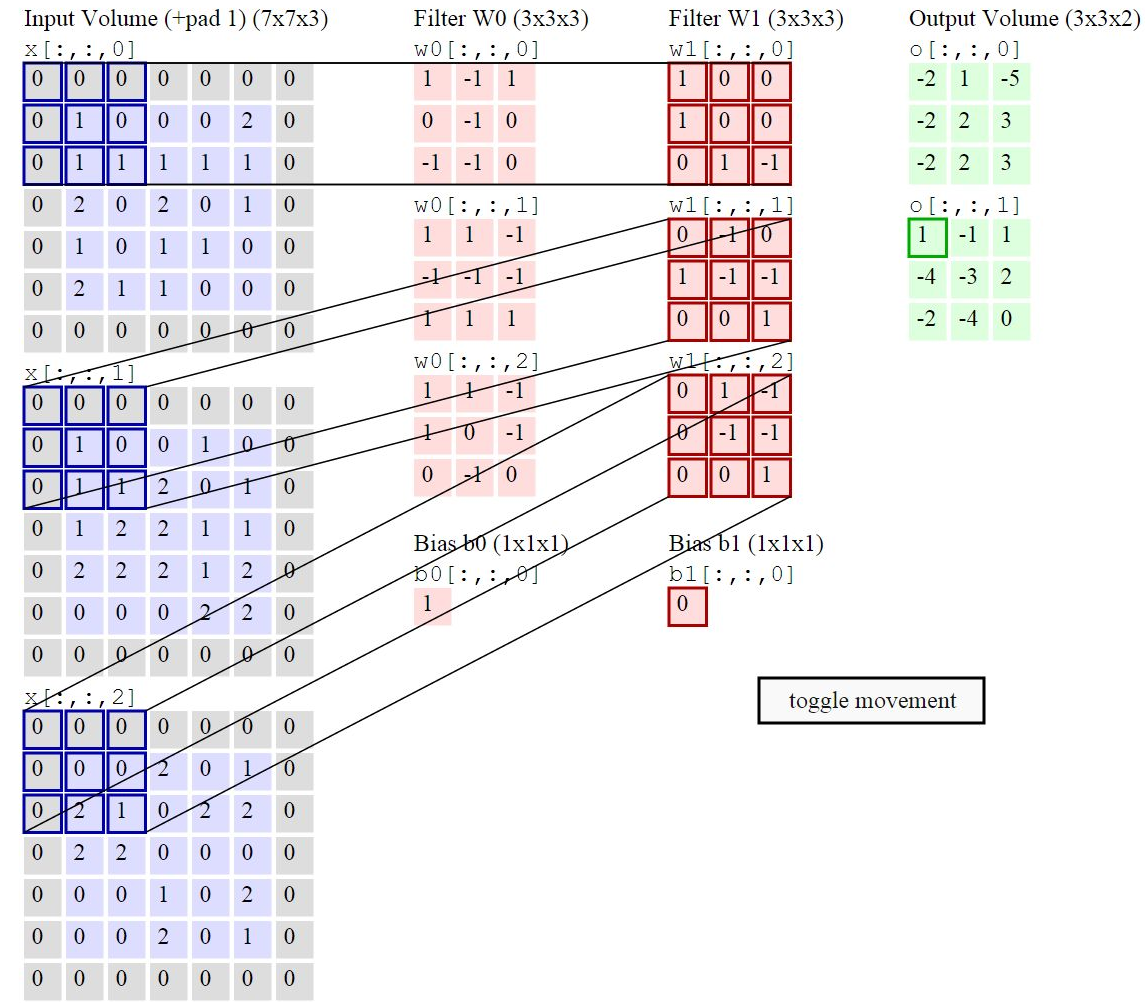
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



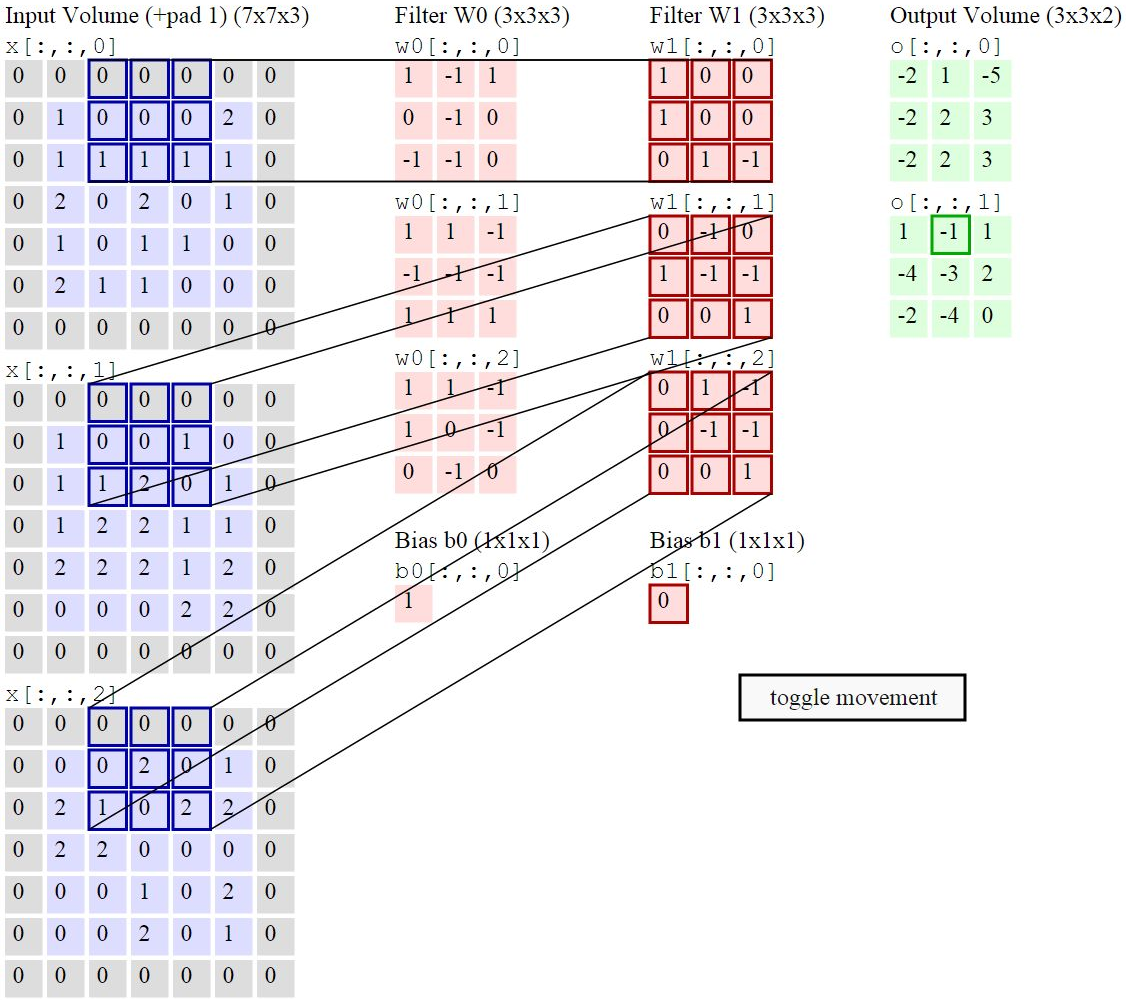
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



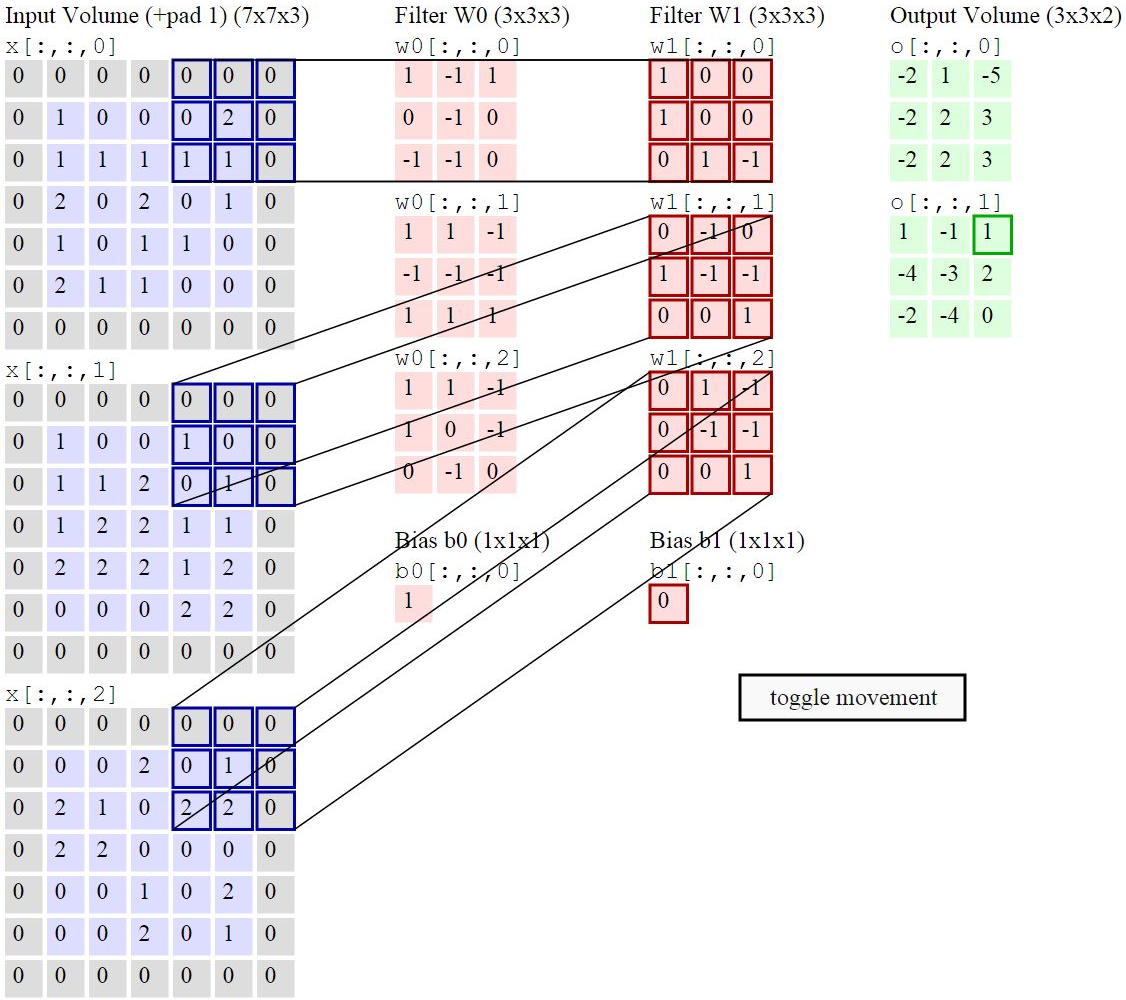
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



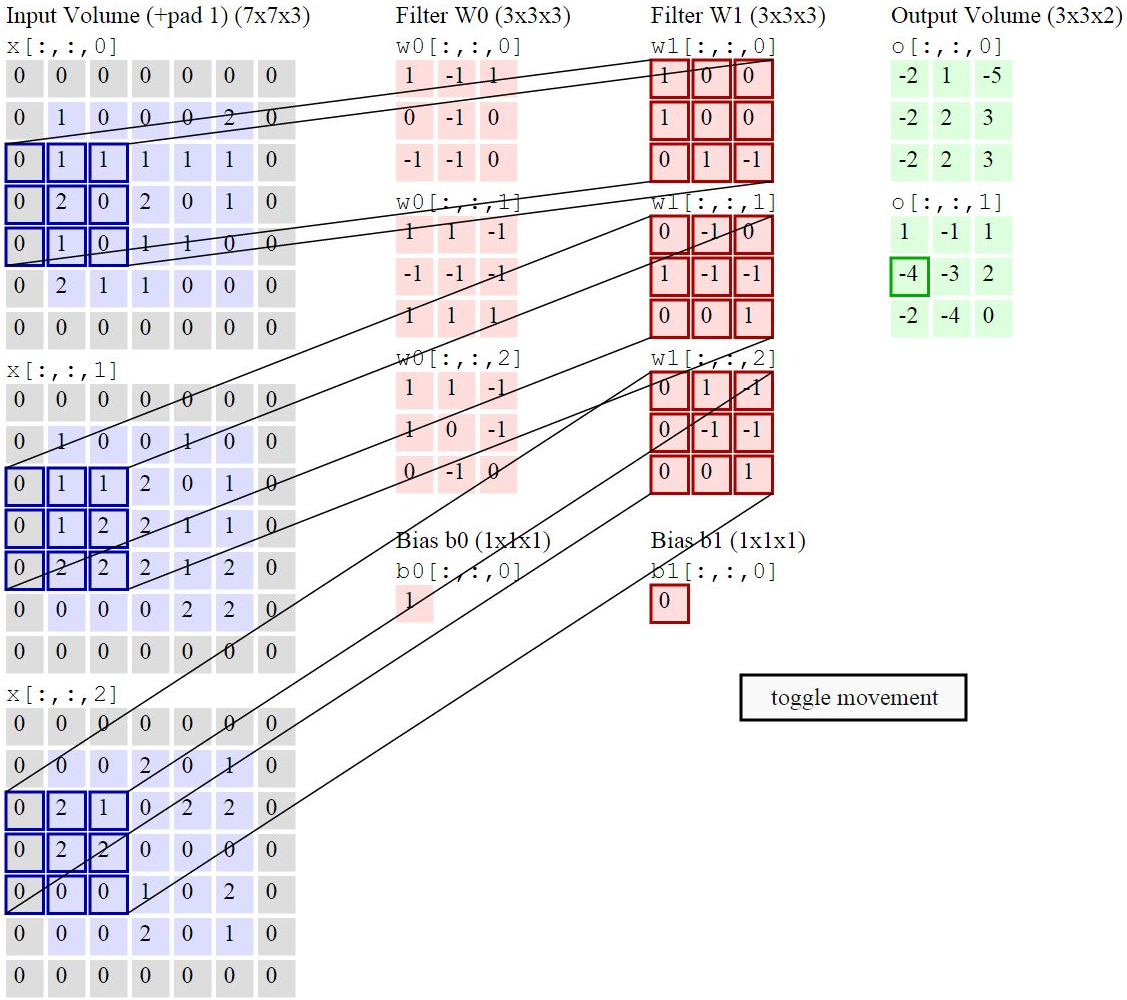
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



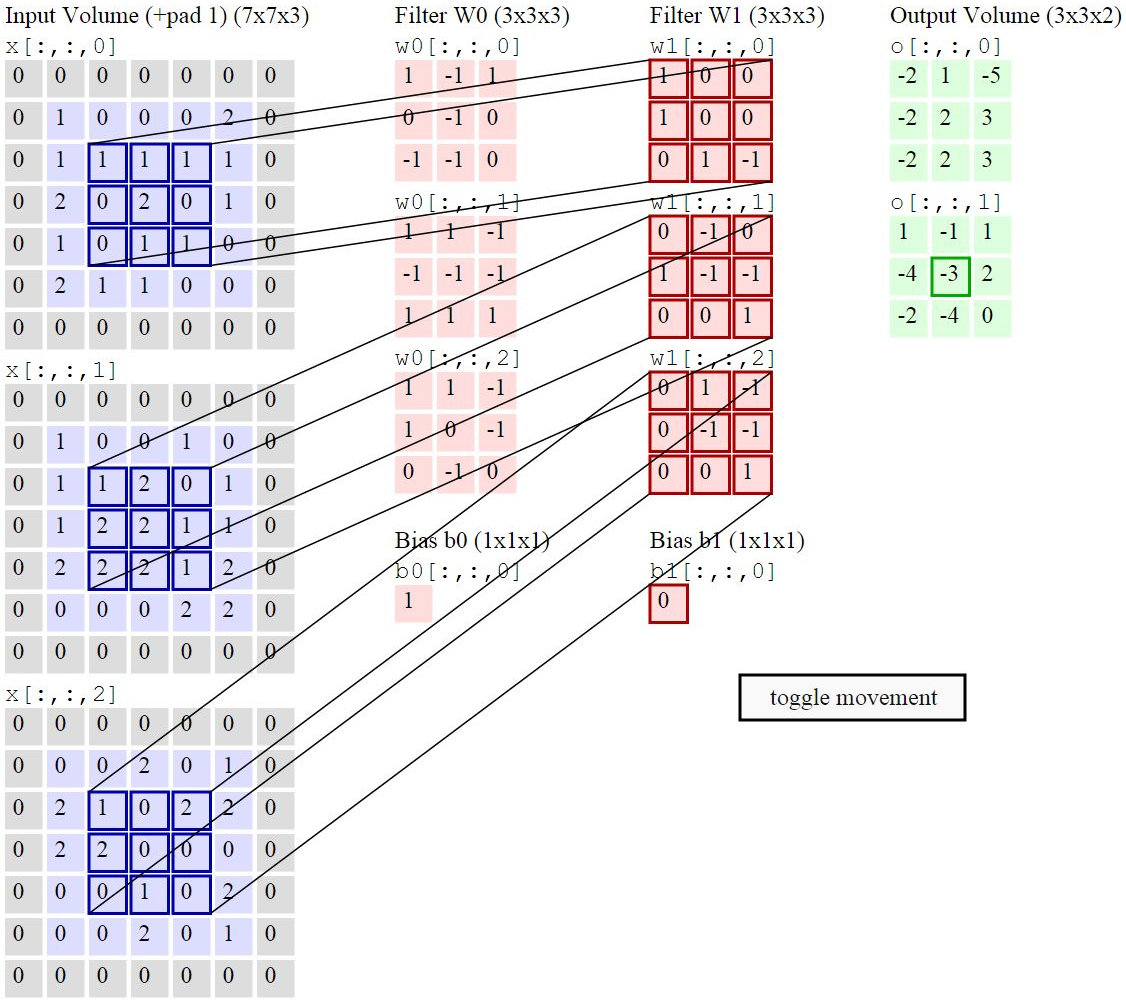
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



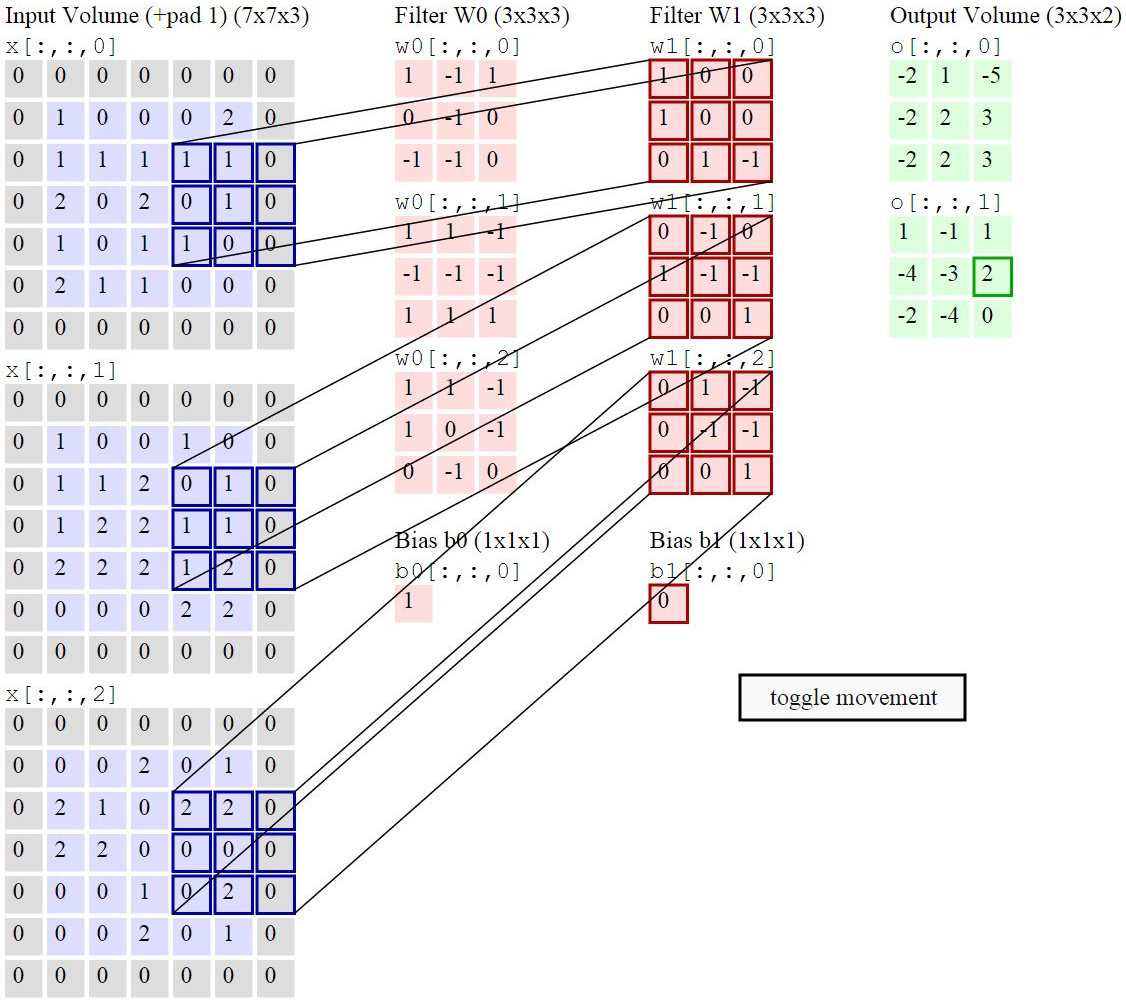
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



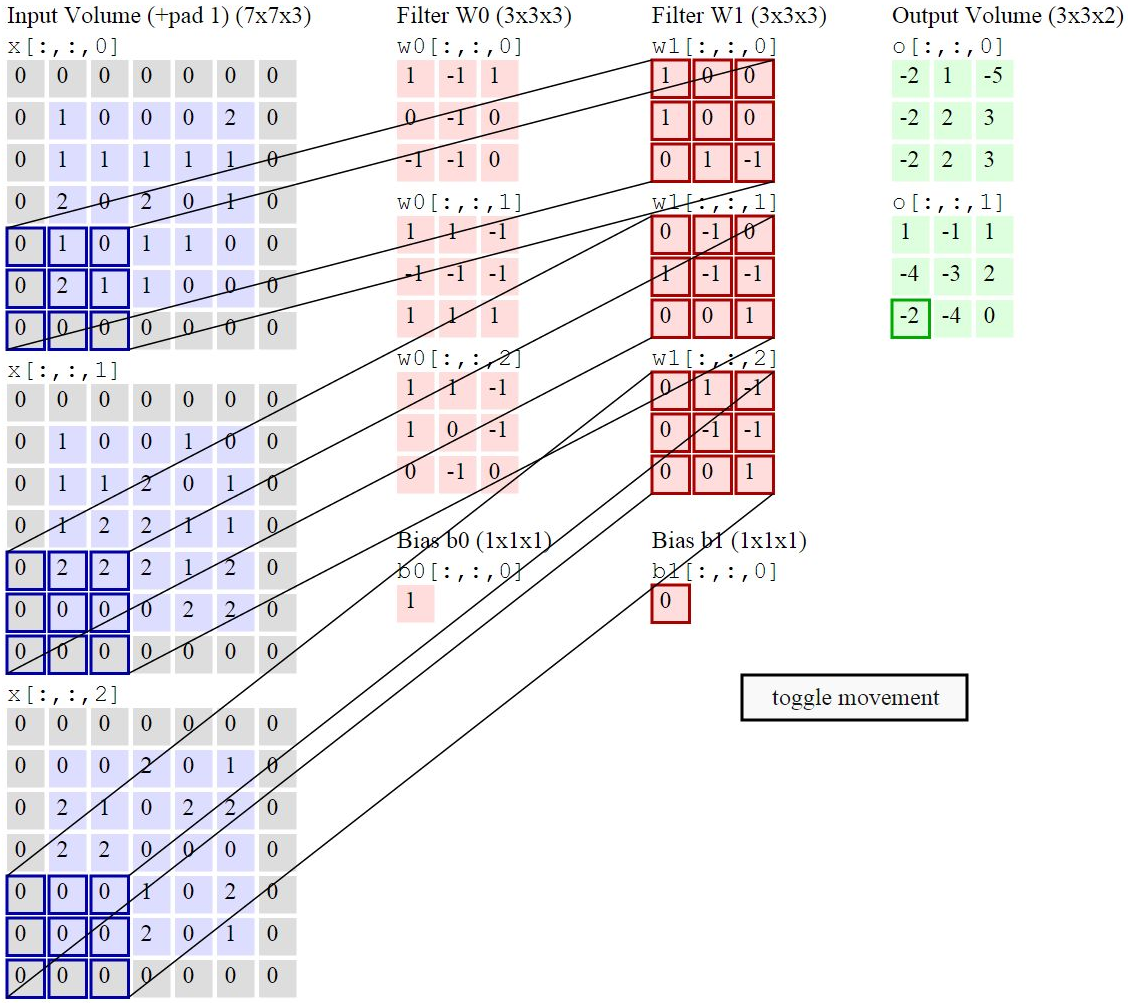
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



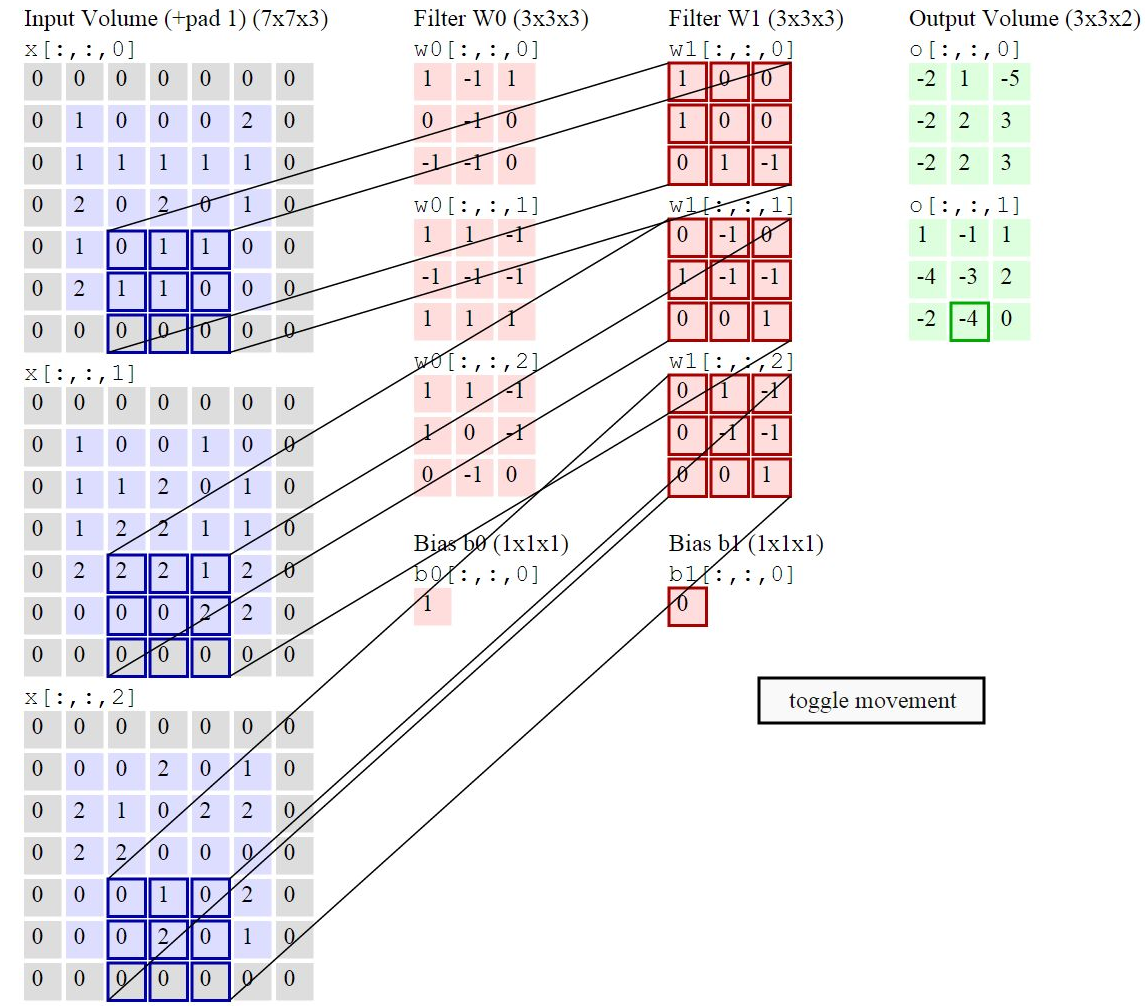
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



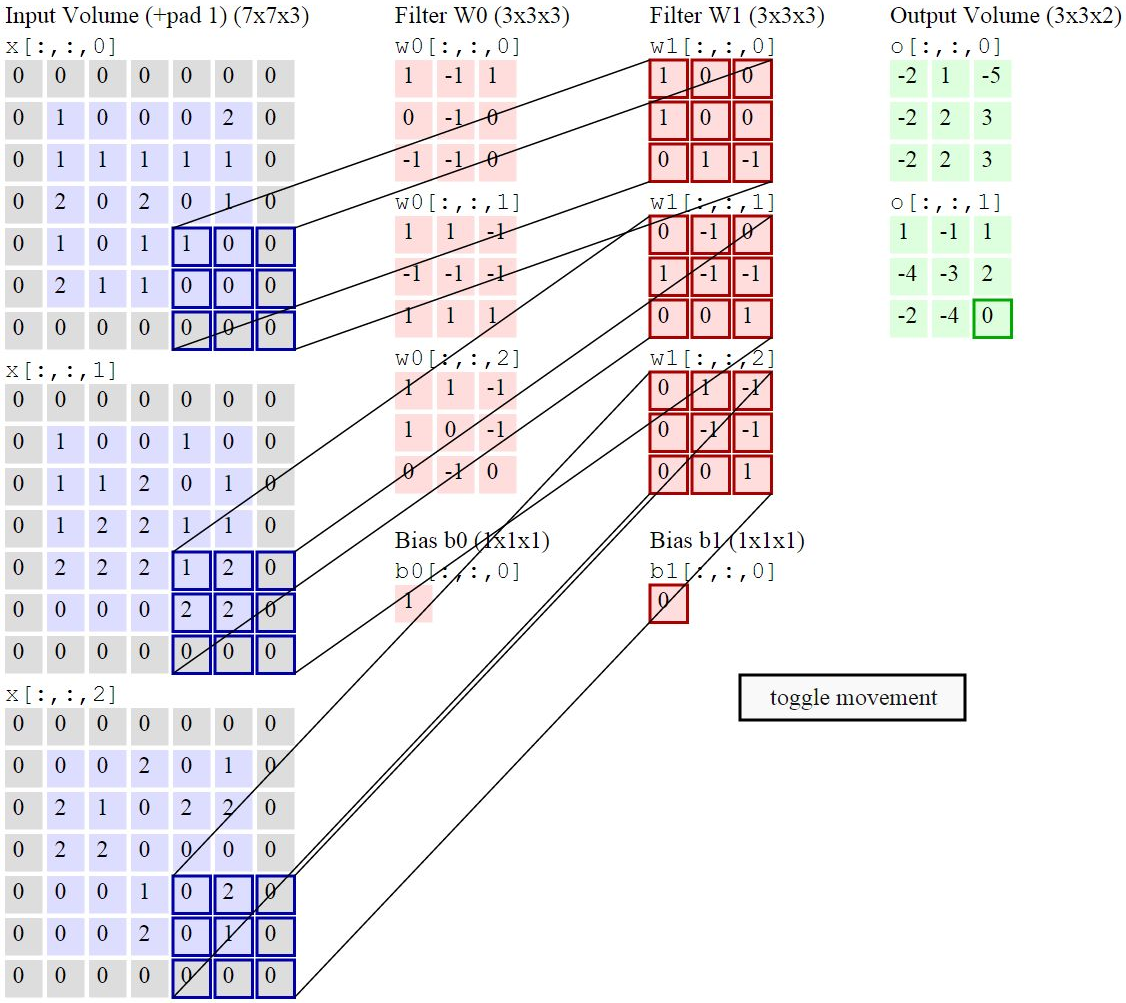
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network



Fonte: http://cs231n.github.io/convolutional-networks/

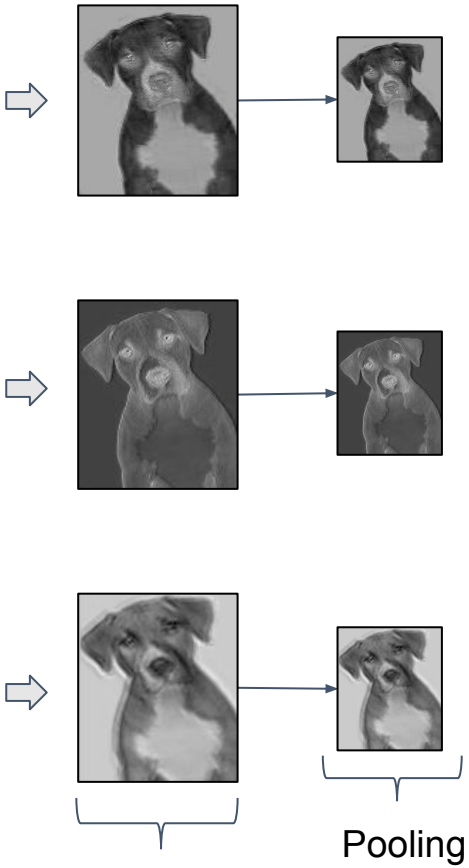
Convolving a Dog

455x256 RGB
Input Image

Convolution
Layer

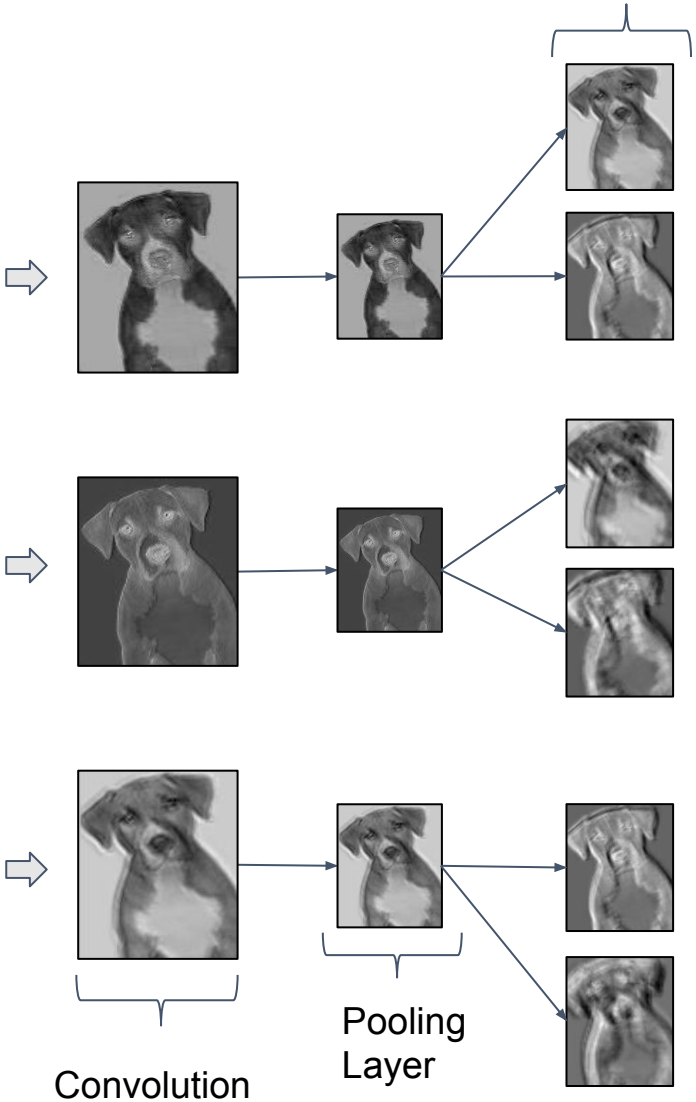
Convolving a Dog

455x256 RGB
Input Image



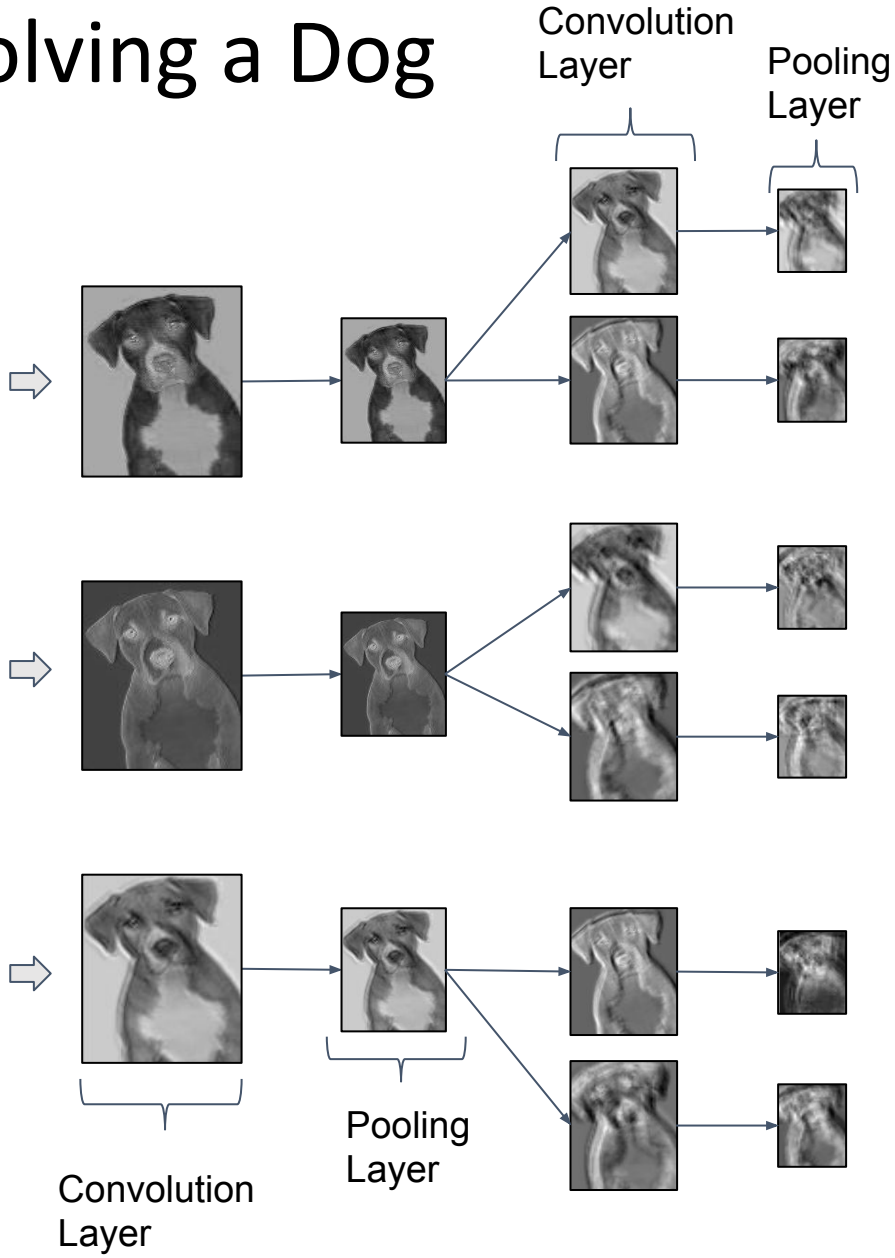
Convolving a Dog

455x256 RGB
Input Image



Convolving a Dog

455x256 RGB
Input Image



Convolving a Dog

455x256 RGB
Input Image

Convolution
Layer 1
223x123

Pooling
Layer 1
111x61

Convolution
Layer 2
50x25

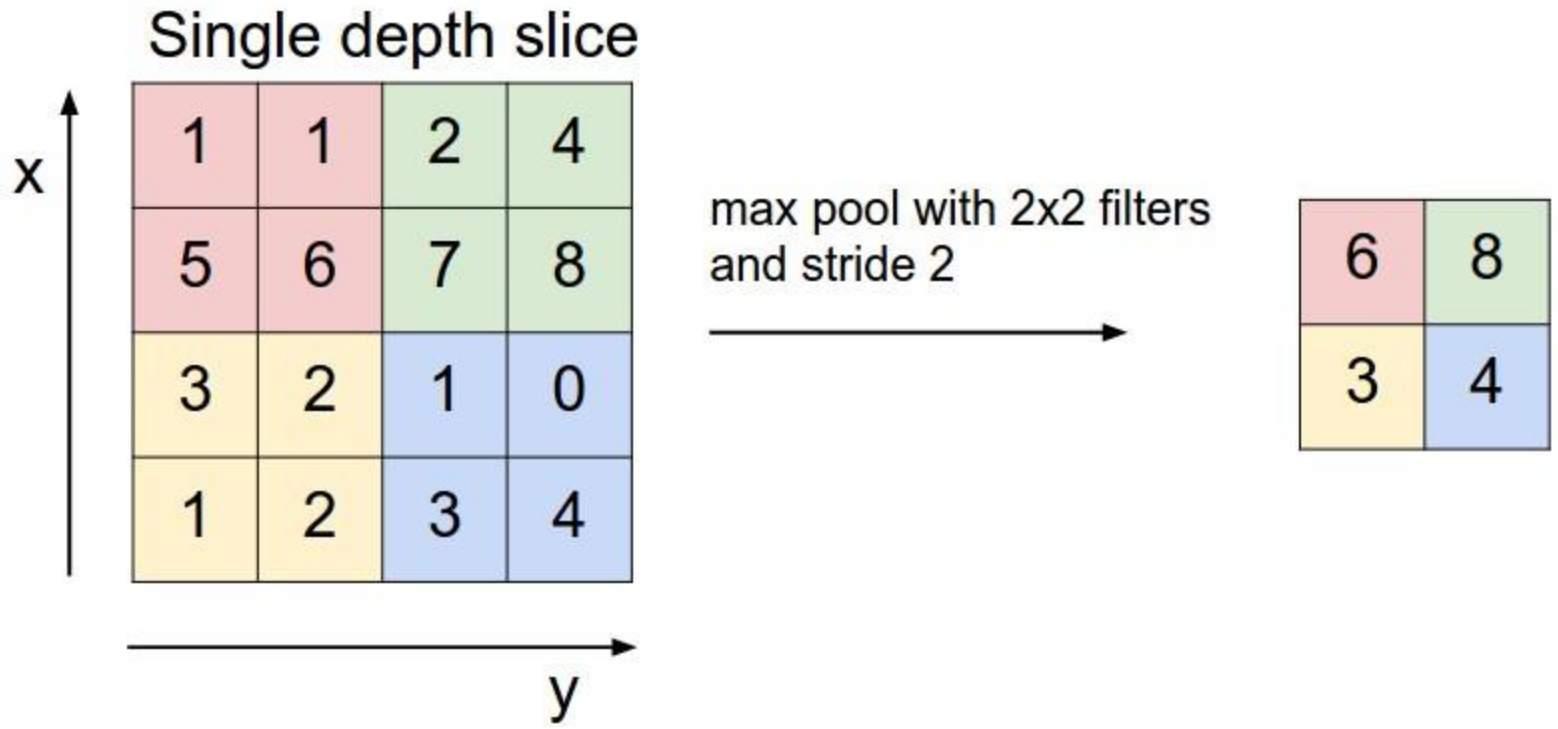
Pooling
Layer 2
24x11

Conv Layer 1: 3 Filters,
Size 11, Stride 2, Padding 1
Conv Layer 2: 2 Filters,
Size 12, Stride 2, Padding 1
Poolings 1 and 2:
Size 2, Stride 2

Fully Connected Input: 1,584
Original: 349,440

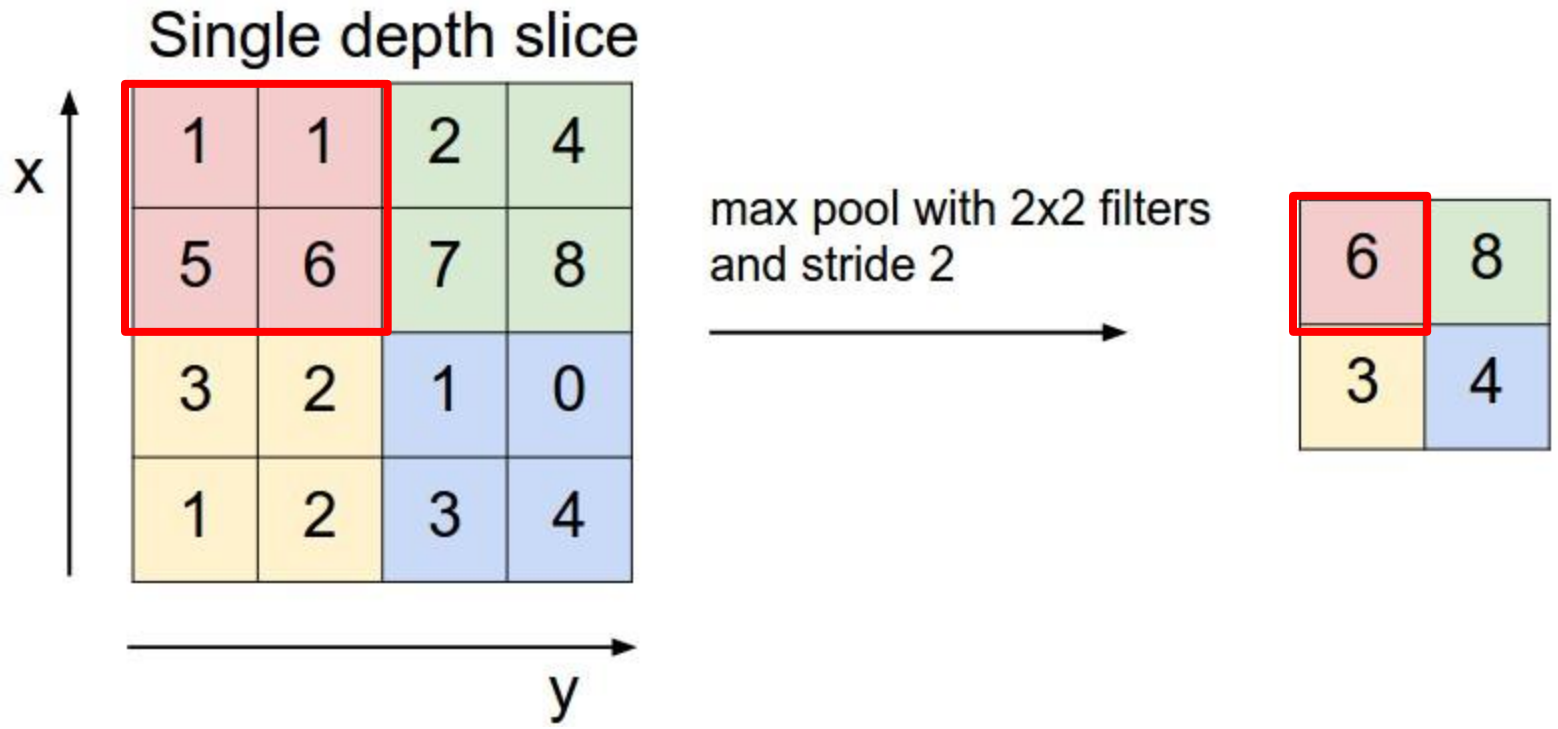
Classifier

The Convolutional Neural Network
Pooling layer



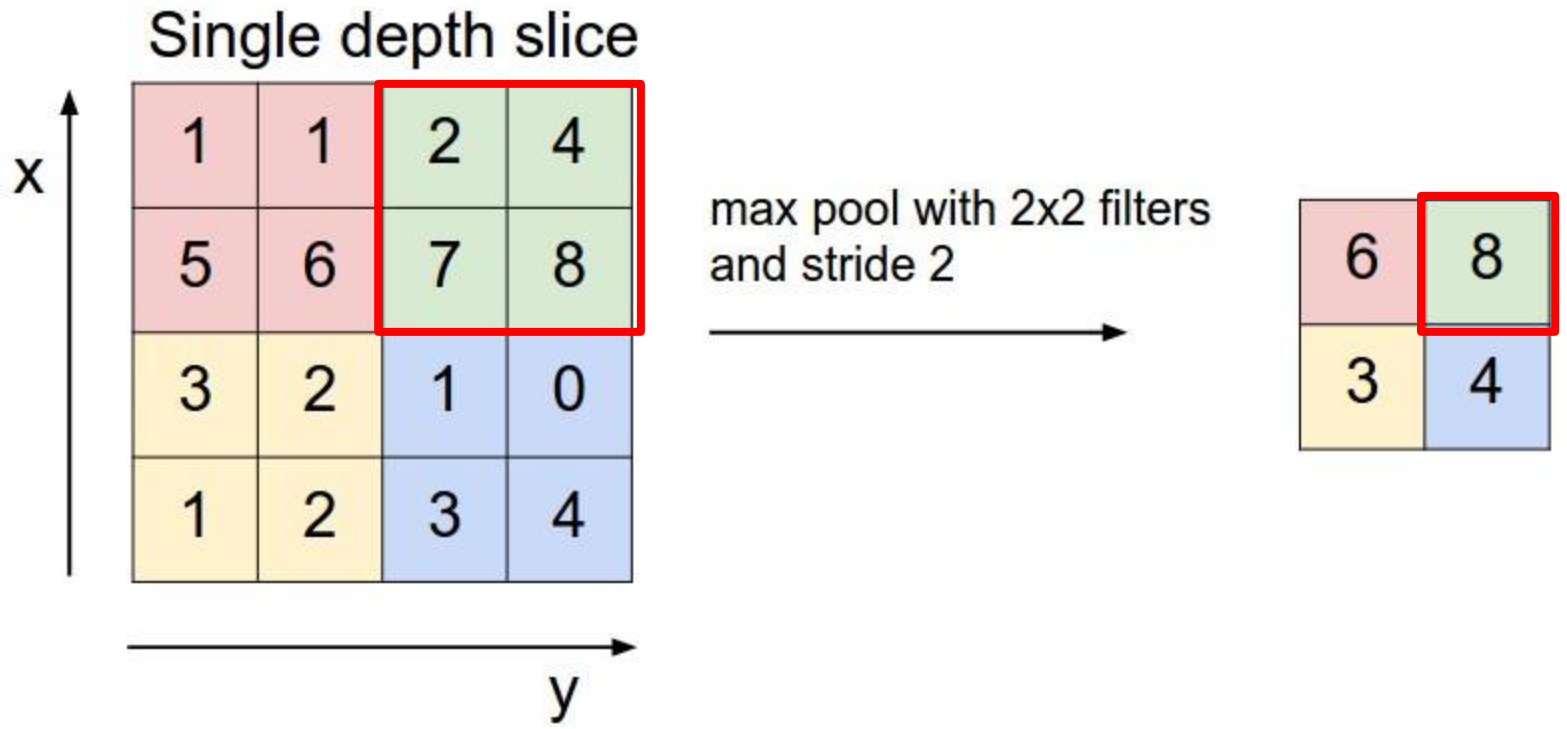
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network
Pooling layer



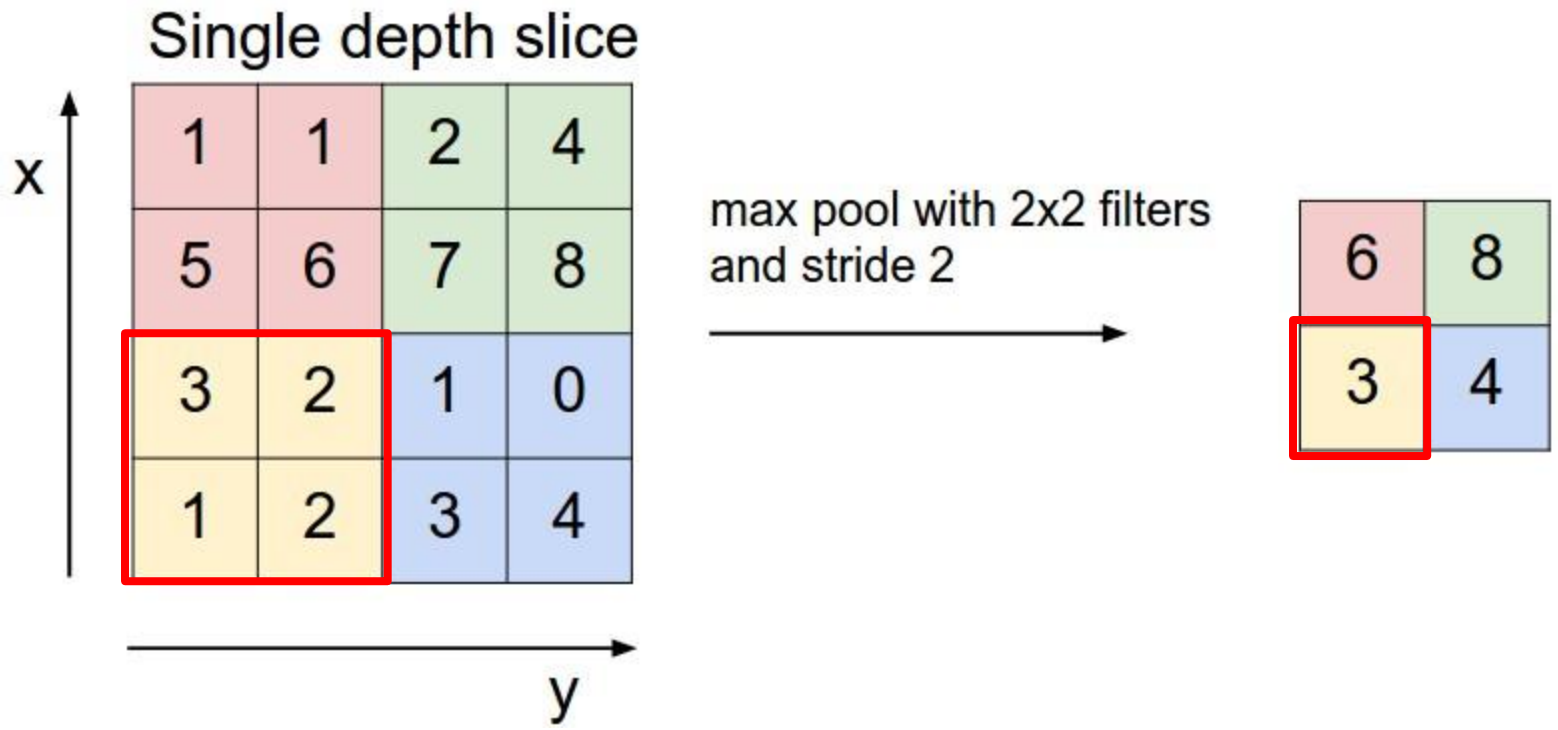
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network
Pooling layer



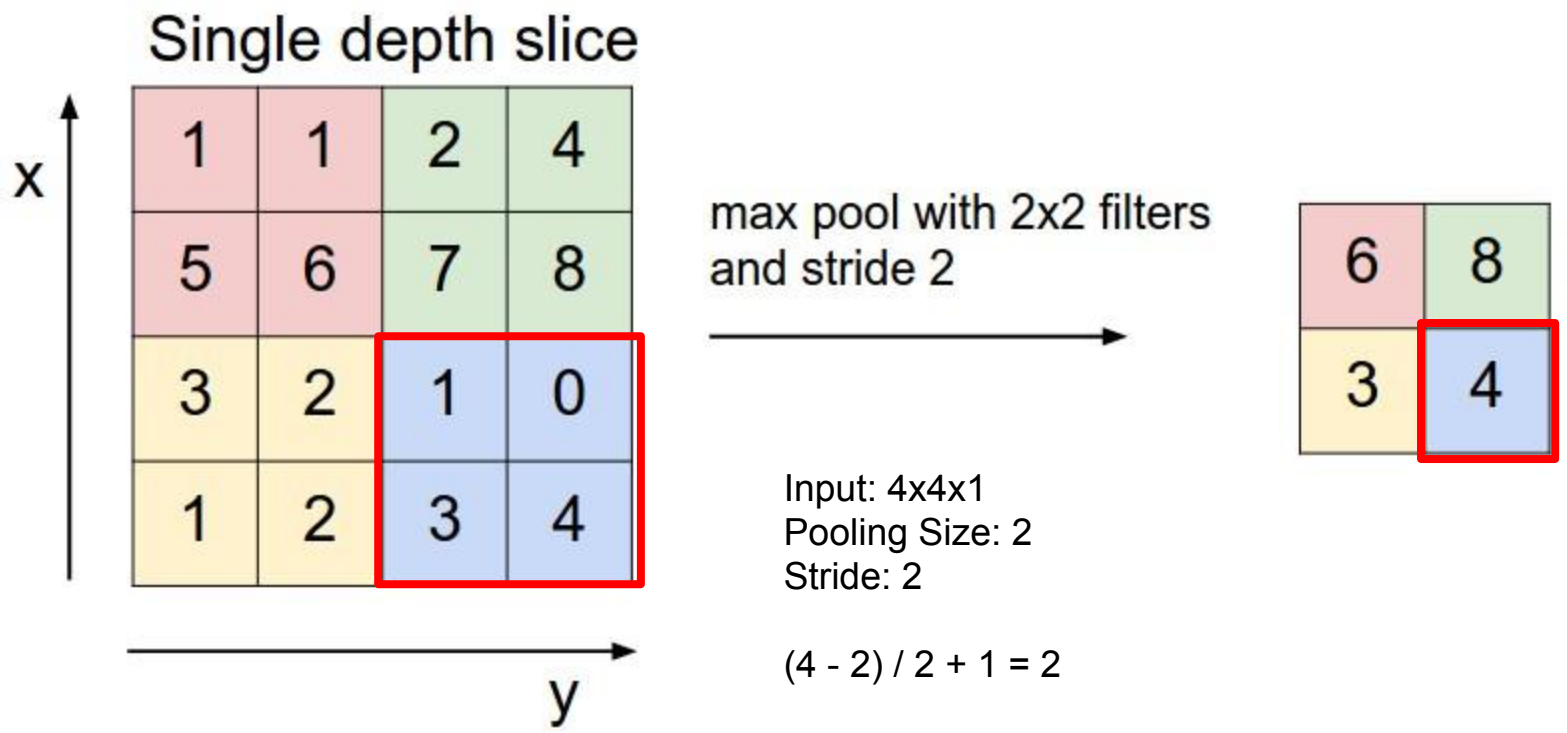
Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network
Pooling layer



Fonte: http://cs231n.github.io/convolutional-networks/

The Convolutional Neural Network
Pooling layer



Input: 4x4x1
Pooling Size: 2
Stride: 2

(4 - 2) / 2 + 1 = 2

Fonte: http://cs231n.github.io/convolutional-networks/

Deep Learning Frameworks

Where the magic turns to be!

What is a deep learning framework?

It is a simple toolbox that implements neural networks concepts in a loose coupled approach. Most of this concepts is presented as building blocks, commonly known as layers.

A deep neural network is **a set of different layers**.

What is a deep learning framework?

Deep learning frameworks also implement algorithms for training neural networks based on backpropagation, including optimizers, parameter initializers, and loss functions.

Most popular frameworks

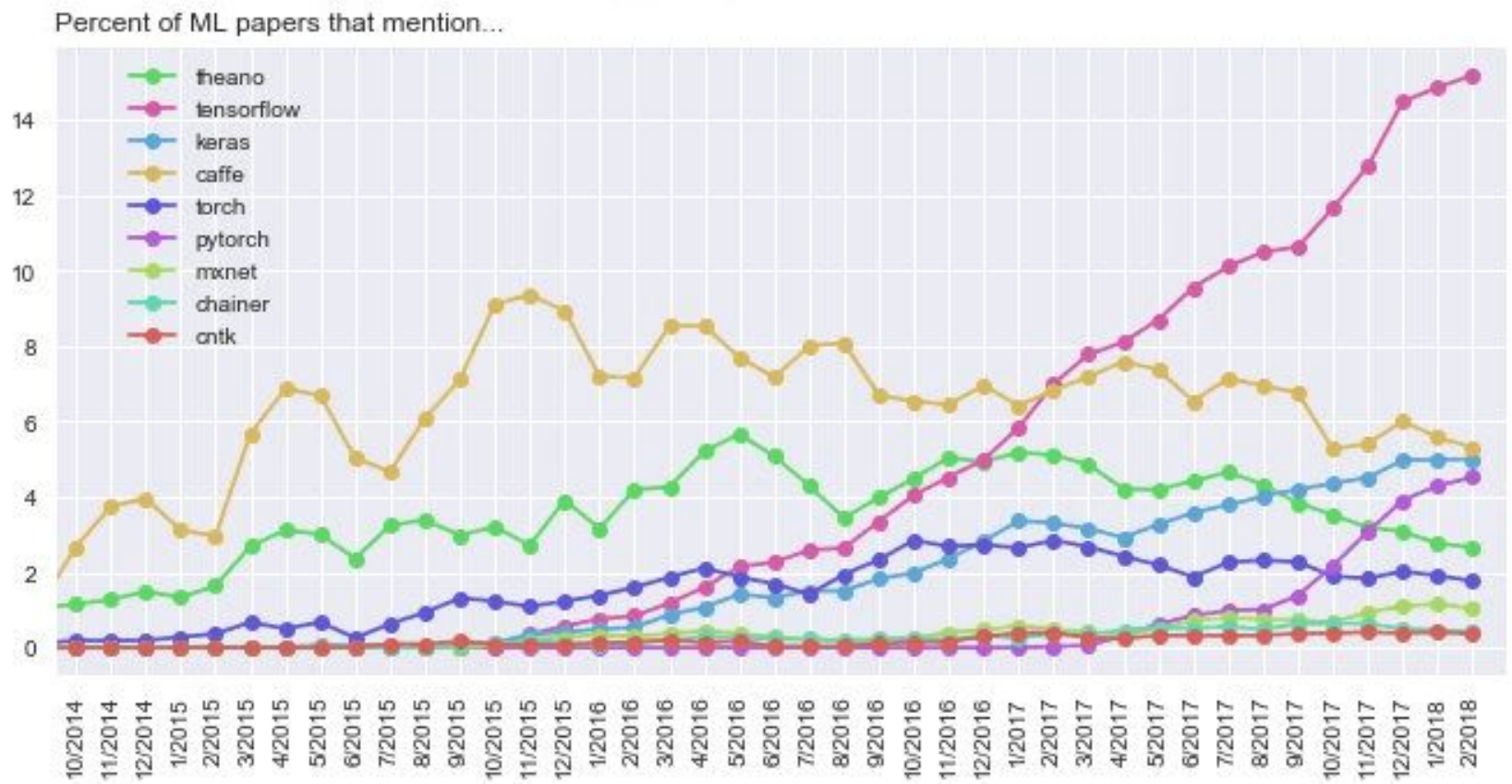
- PyTorch
- Tensorflow
- Keras
- Caffe

Most popular frameworks

So, which framework is the best choice?

There is no answer! But...

Most popular frameworks



https://twitter.com/karpathy/status/972295865187512320/photo/1

Some practice...

To do some practice in our discipline, we will use a virtual machine with some pre-installed frameworks.

This approach is not ideal. We will use this virtual machine due to the lack of specific resources for deep learning, such as GPUs and high performance machines.

Some practice...

Let's take a look at what we have in our sandbox.

Assignment 1 - Car Classifier

Our practice in the sandbox used several pre-trained models to classify contents. One of these models classifies American cars, however, probably this same model will have a fragile performance classifying Brazilian cars.

Assignment 1 - Car Classifier

Our practice in the sandbox used several pre-trained models to classify contents. One of these models classifies American cars, however, probably this same model will have a fragile performance classifying Brazilian cars.

Your challenge is to build a classifier capable of automatically identifying 5 models of Brazilian cars. To do this, use the parking lot of PUCRS and you mobile camera to take photos of different cars belonging to these 5 models. Construct your classifier using neural network features, but remember, you can use any machine learning concepts already seen in the course. Keep in mind that training a new deep model in our sandbox will not be a good idea.

Assignment 1 - Car Classifier

Important dates:
01/12: follow-up
08/12: follow-up
15/12: delivery date

You should deliver your source code, your dataset, and the best model you've gotten in your experiments.

Have Fun!