http://bit.ly/idl2020

Day 1 Lecture 3

The Perceptron

Xavier Giro-i-Nieto

@DocXavi

xavier.giro@upc.edu

Associate Professor

Universitat Politècnica de Catalunya

Acknowledgements

Santiago Pascual

Kevin McGuinness
kevin.mcguinness@dcu.ie

Research Fellow
Insight Centre for Data Analytics
Dublin City University

Video lectures

Santiago Pascual, DLSL 2017

Xavier Giro-i-Nieto, DLAI 2018

Outline

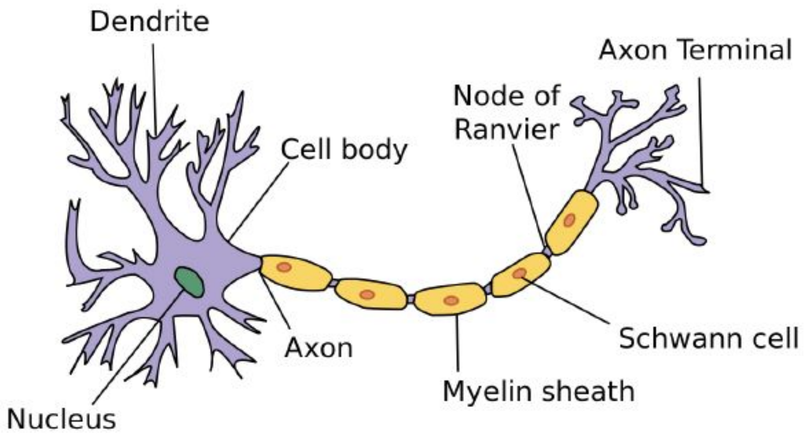
1. **Single neuron model (Perceptron)**
2. Regression
   a. Linear regression
   b. Logistic regression

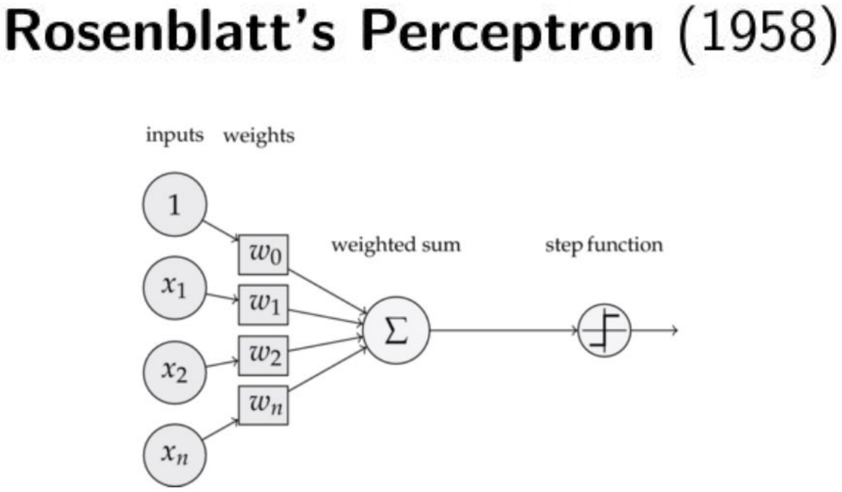
Single neuron model (perceptron)

The Perceptron is seen as an **analogy** to a biological neuron.

Biological neurons fire an impulse once the sum of all inputs is over a threshold.

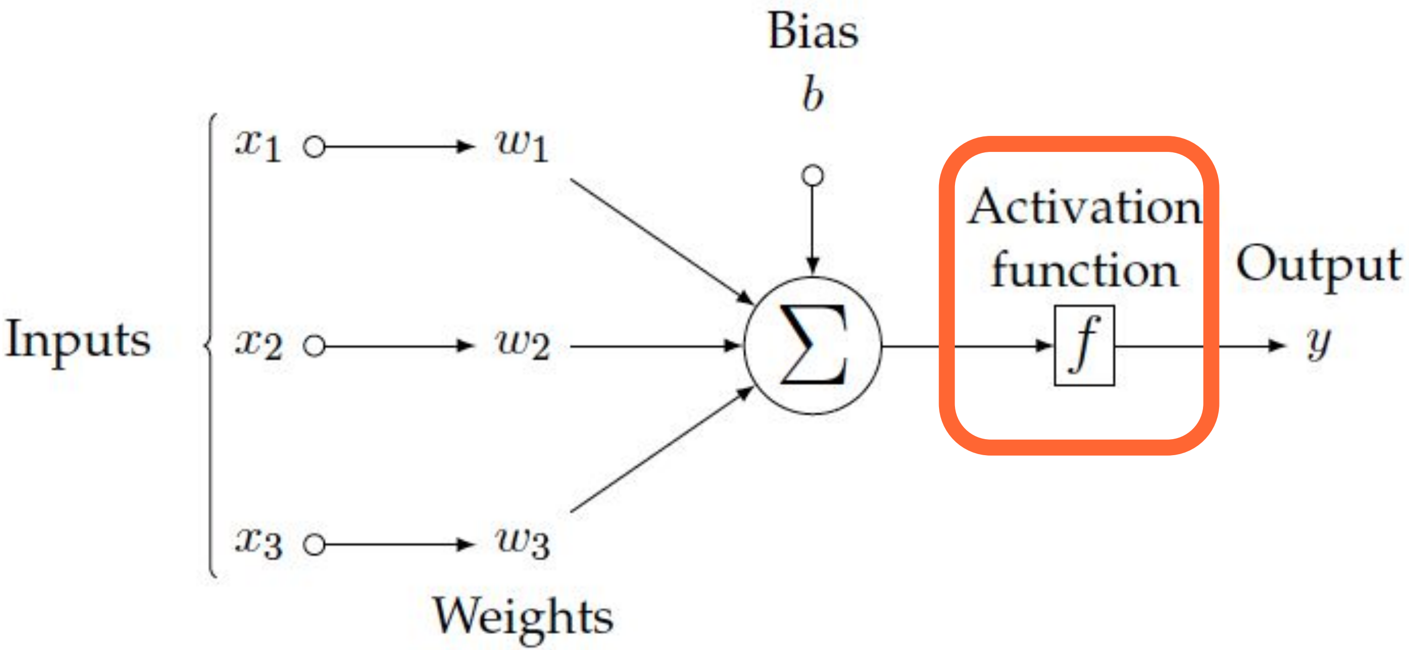
The perceptron acts like a switch (learn how in the next slides...).



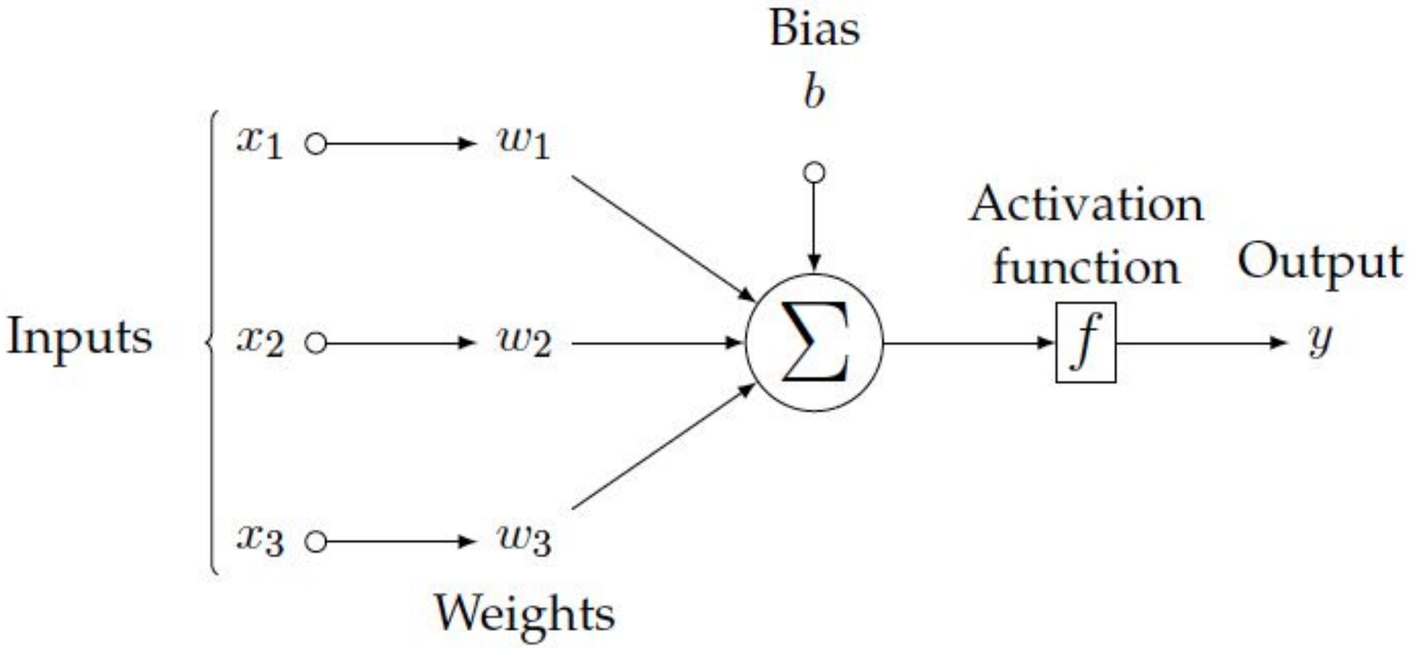


Single Neuron Model (Perceptron)

The perceptron can address both regression or classification problems, depending on the chosen **activation function**.

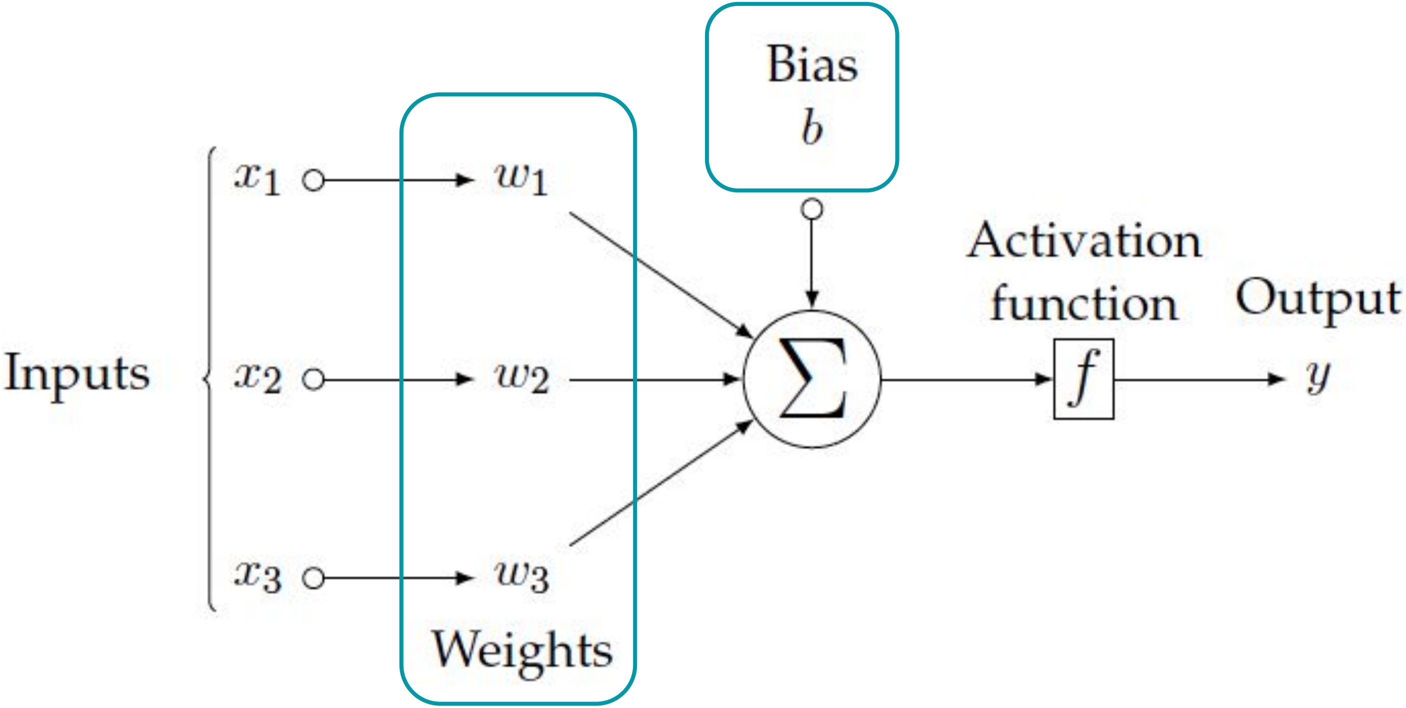


Single Neuron Model (Perceptron)



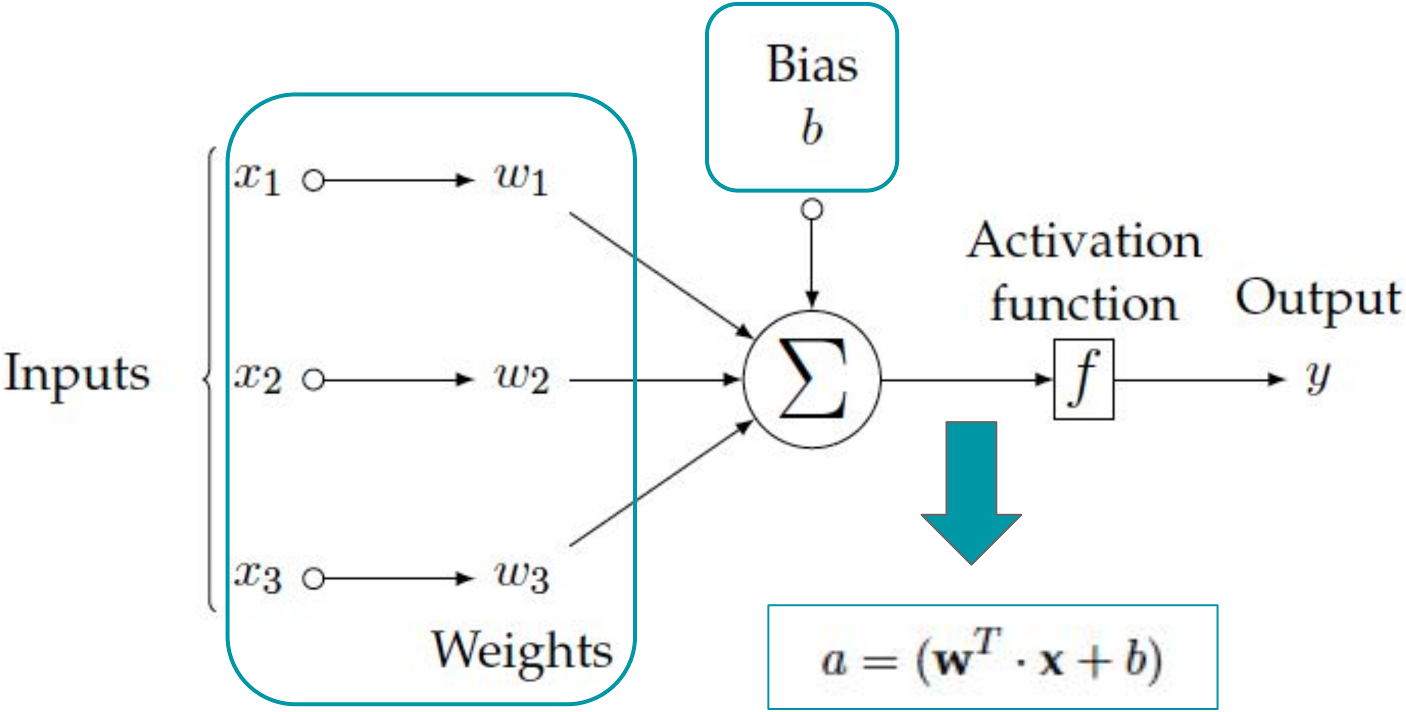
Single Neuron Model (Perceptron)

Weights and bias are the parameters that define the behavior. They must be estimated during training.



Single Neuron Model (Perceptron)

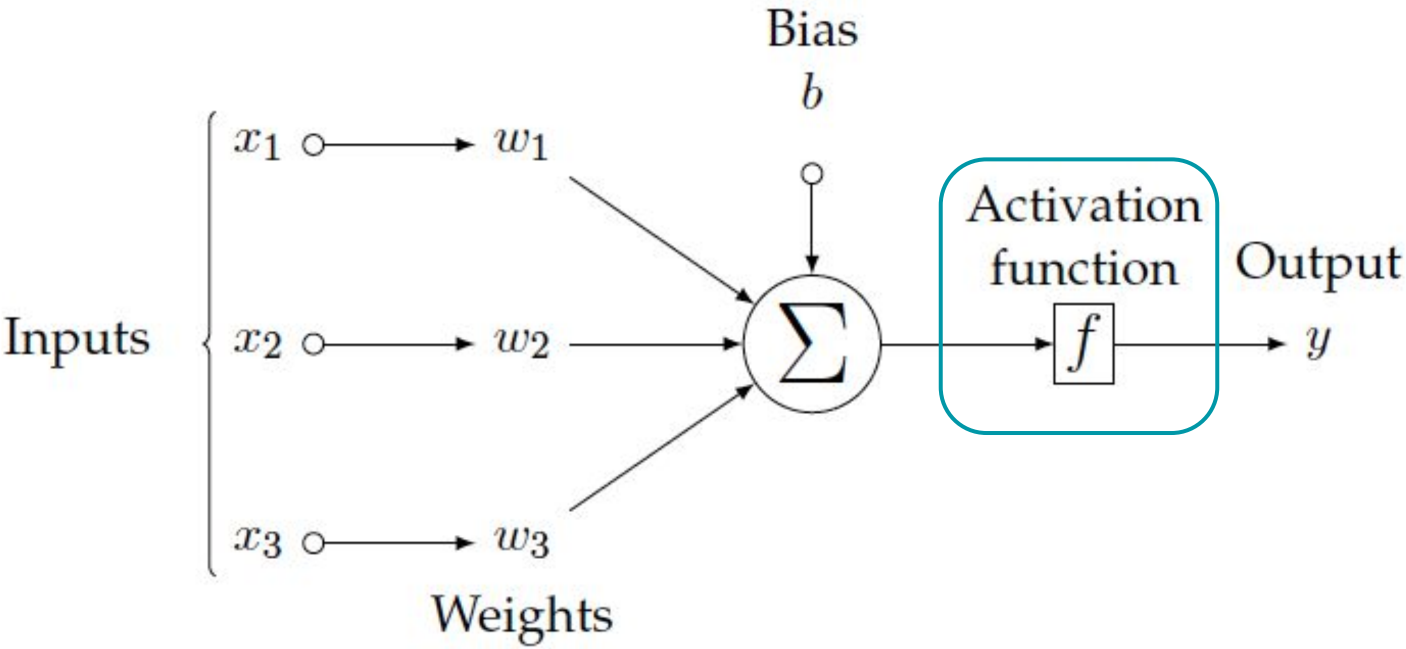
The output y is derived from a sum of the **weighted** inputs plus a **bias** term.



$$a = (\mathbf{w}^T \cdot \mathbf{x} + b)$$

Single Neuron Model (Perceptron)

The **activation function** introduces non-linearities.



Single Neuron Model (Perceptron)

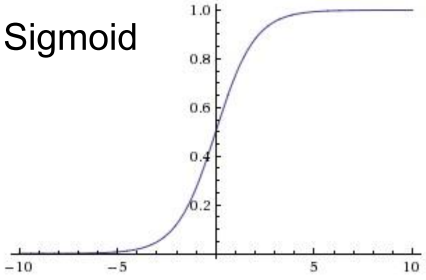
Desirable properties

- Mostly smooth, continuous, differentiable
- Fairly linear

Common nonlinearities

- Sigmoid
- Tanh
- ReLU = max(0, x)

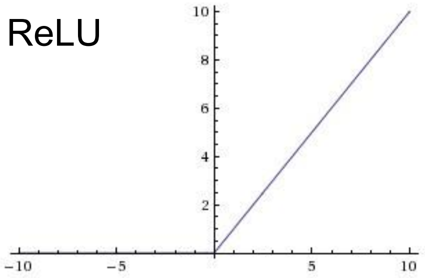
Sigmoid



Why do we need non-linearities ?

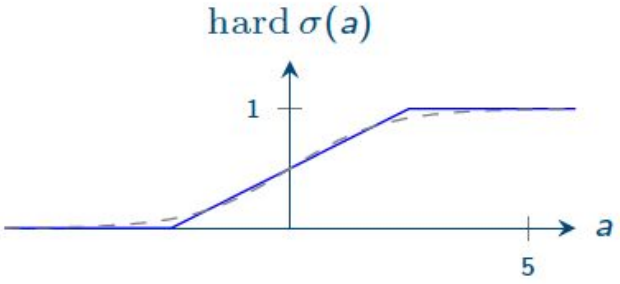
If we only use linear layers we are only able to learn linear transformations of our input.

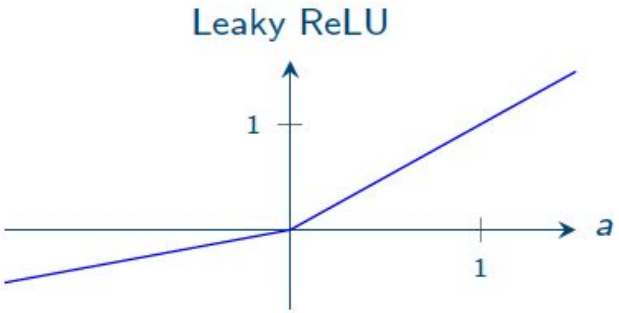
ReLU

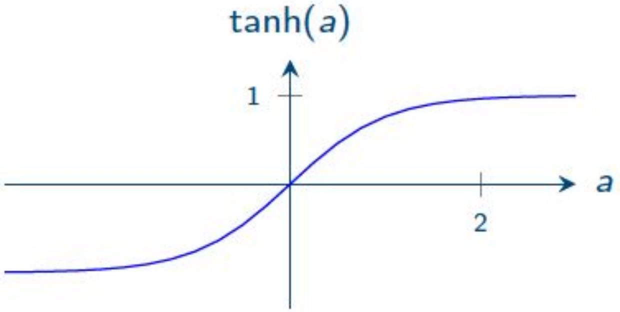


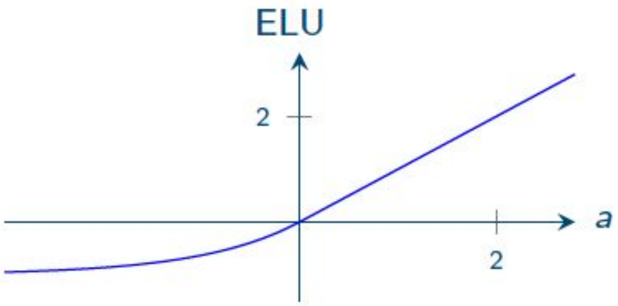
Single neuron model: Regression

Other popular activation functions:









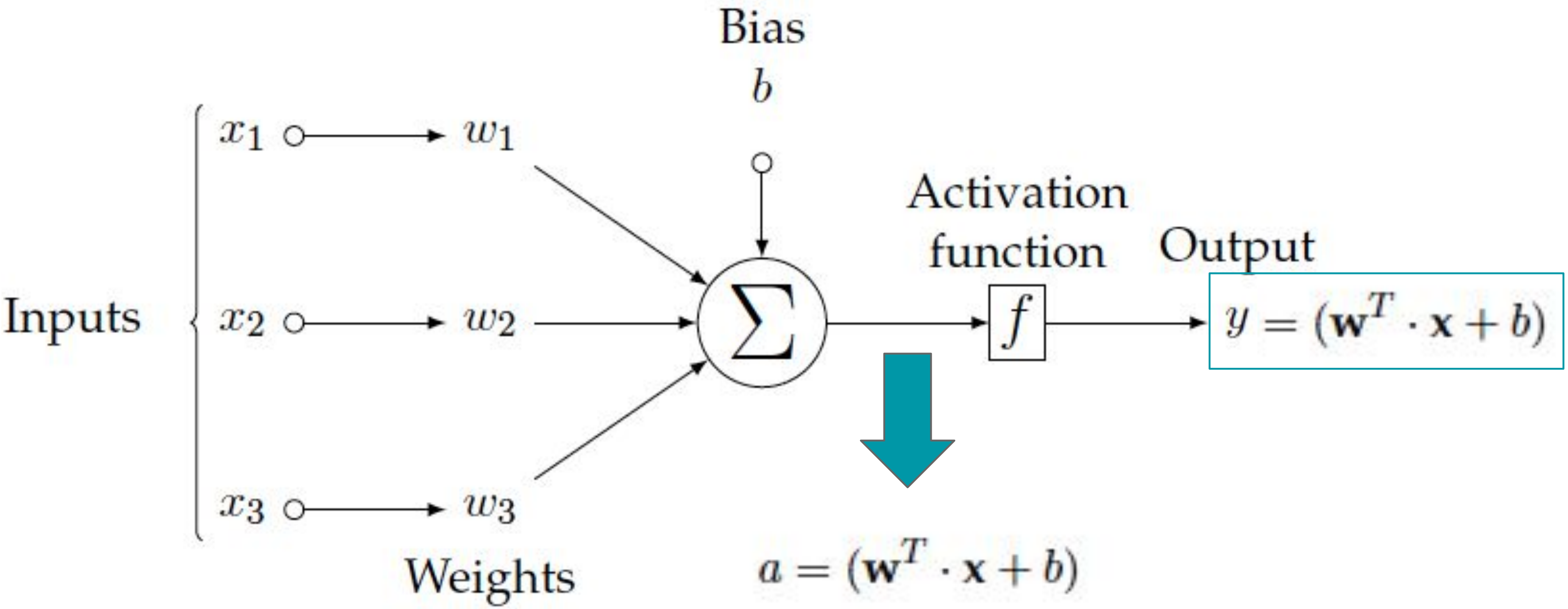
Outline

1. Single neuron model (Perceptron)
2. **Regression**
    a. **Linear regression**
    b. Logistic regression

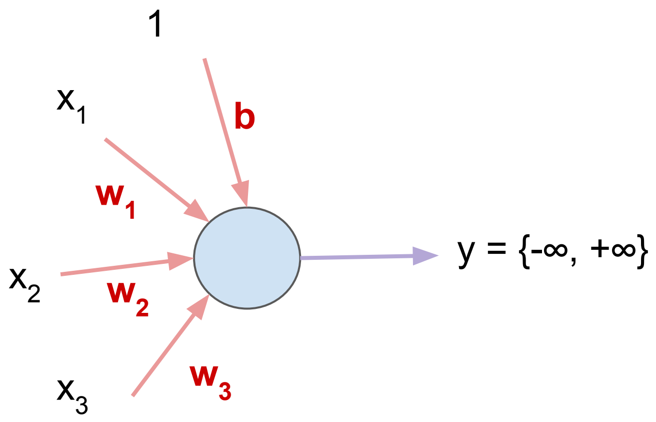
Single neuron model: Linear Regression

A single neuron scheme can solve linear regression problems when f(a)=a.

[identity]



Single neuron model: Linear Regression



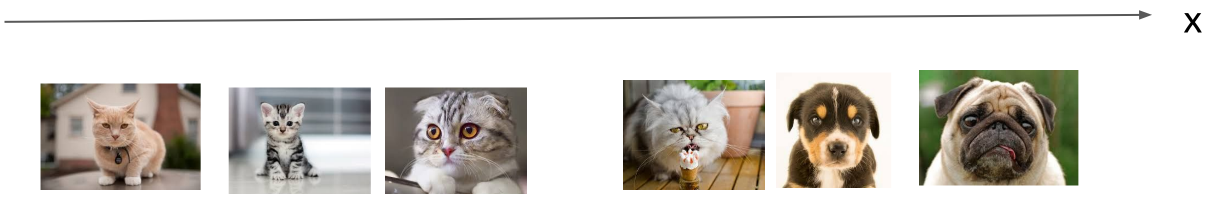

```
lir = nn.Sequential(
    nn.Linear(NUM_INPUTS, 1)
)
```

Outline

1. Single neuron model (Perceptron)
2. **Regression**
    a. Linear regression
    b. **Logistic regression**
3. Limitations of the perceptron

Single neuron model: Binary Classification

Question: Consider a problem of binary classification between cats (class 0) and dogs (class 1). How can a linear regressor be modified to provide 0 or 1 outputs ?

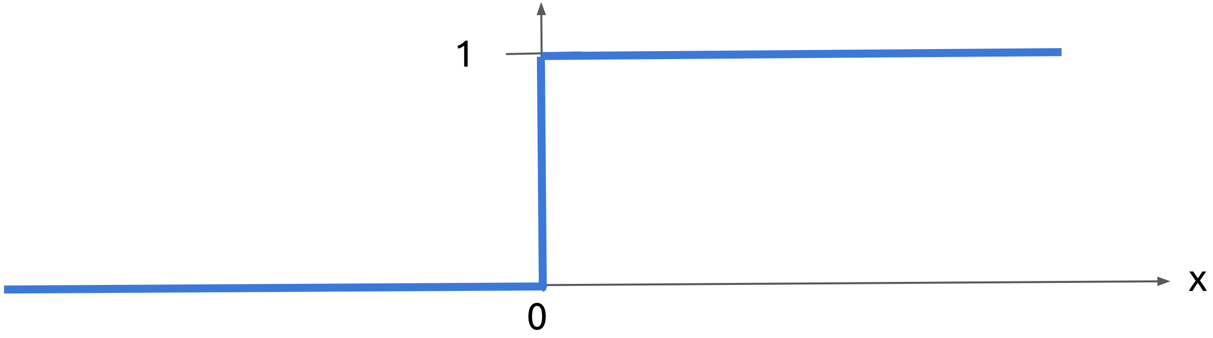


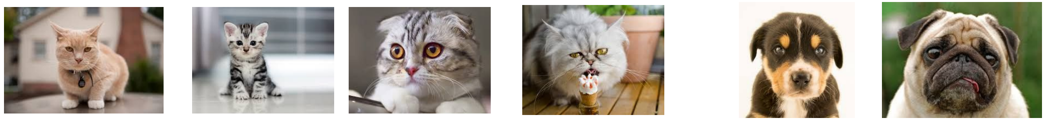
Single neuron model: Binary Classification

<u>Question</u>: Consider a problem of binary classification between cats (class 0) and dogs (class 1). How can a linear regressor be modified to provide 0 or 1 outputs ?

The heaviside (or step) function would generate 0 or 1 outputs only….

y=u(x)

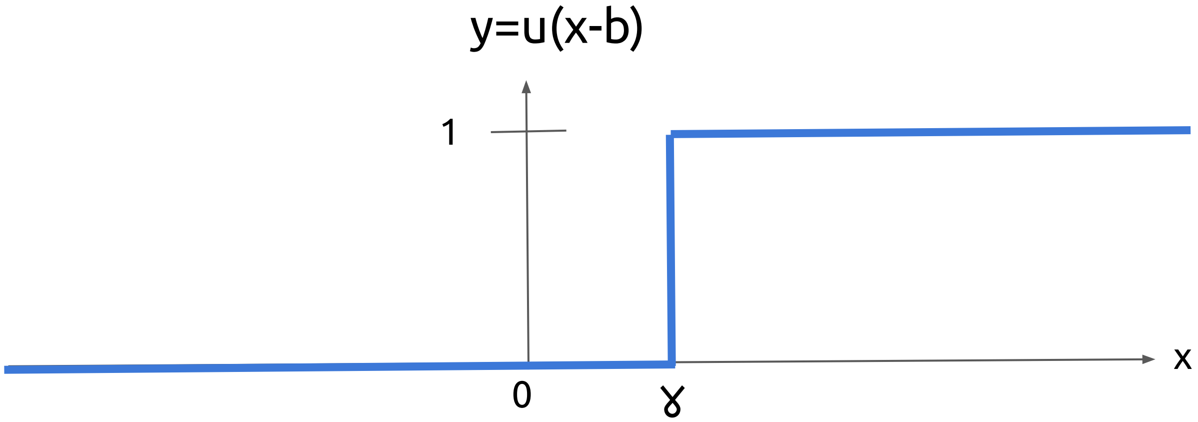


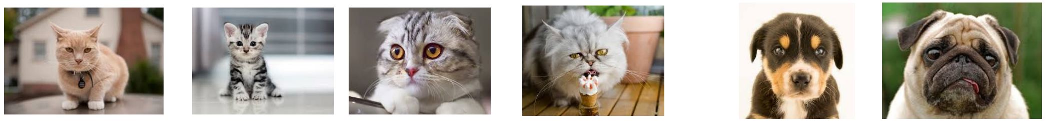


Single neuron model: Binary Classification

Question: Consider a problem of binary classification between dogs (class 0) and cats (class 1). How can a linear regressor be modified to provide 0 or 1 outputs ?

...the decision boundary could be easily set with a simple translation (b)...

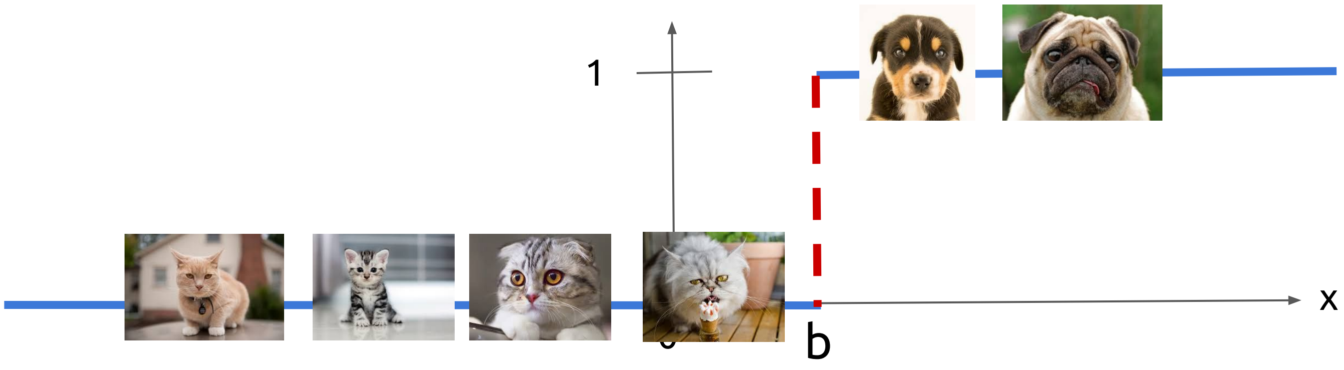




Single neuron model: Binary Classification

Question: Consider a problem of binary classification between dogs (class 0) and cats (class 1). How can a linear regressor be modified to provide 0 or 1 outputs ?

The heaviside (or step) function would generate 0 or 1 outputs only….



...but heavisde is unsuitable for deep learning because it is not differentiable (see later the “Backpropagation” lecture).

Single neuron model: Binary Classification

Question: Consider a problem of binary classification between dogs (class 0) and cats (class 1). How can a linear regressor be modified to provide 0 or 1 outputs ?

A differentiable solution approximation the heaviside function is the **sigmoid function σ(x)**, also referred as **logistic curve**.

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

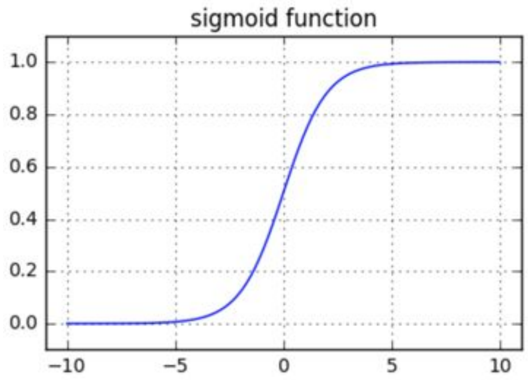
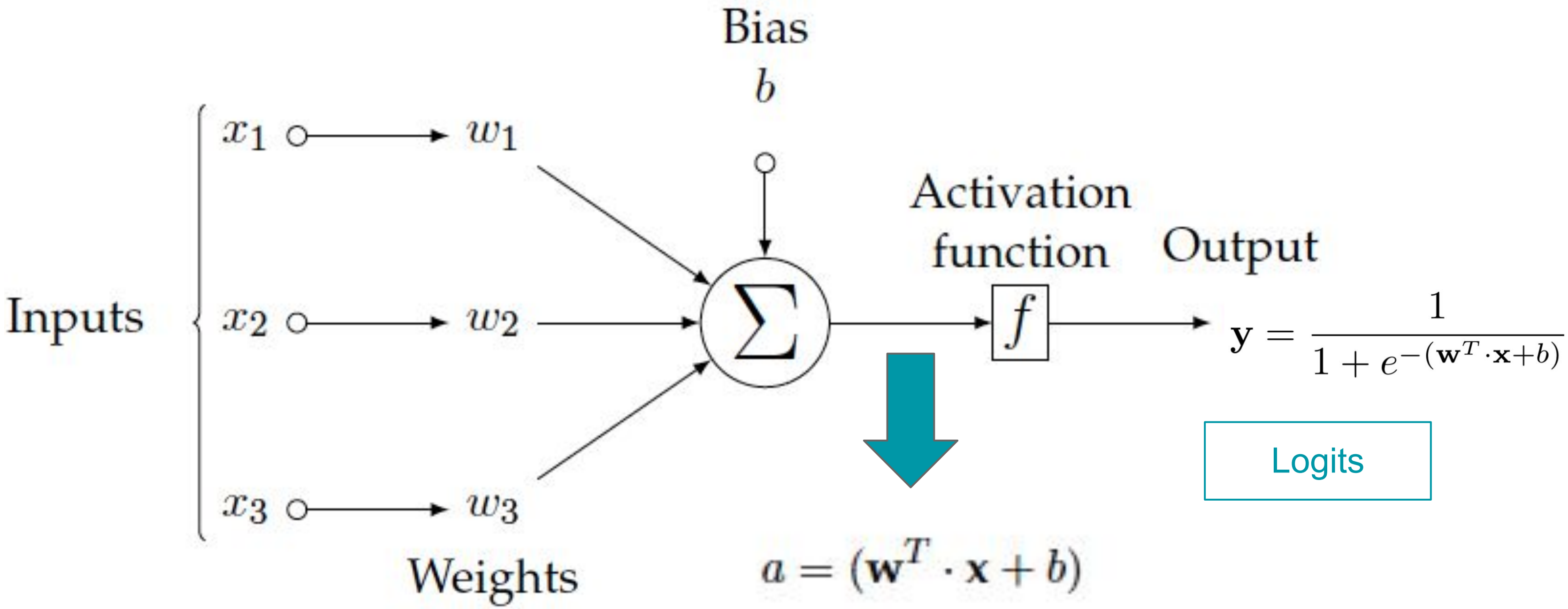


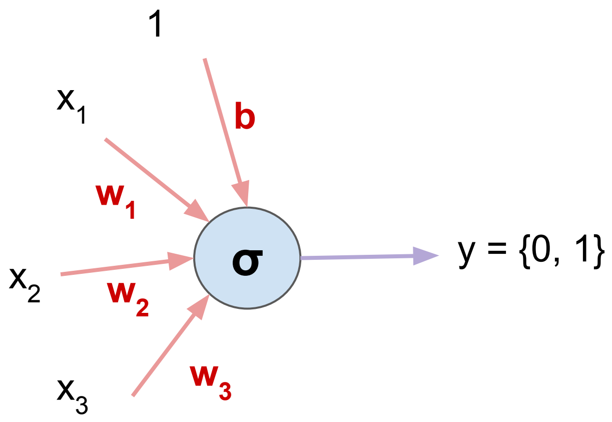
Figure: Andrej Karpathy

Single neuron model: Logistic Regression

The perceptron is suitable for <u>classification</u> problems when f(a)=σ(a). [sigmoid]



Single neuron model: Logistic Regression

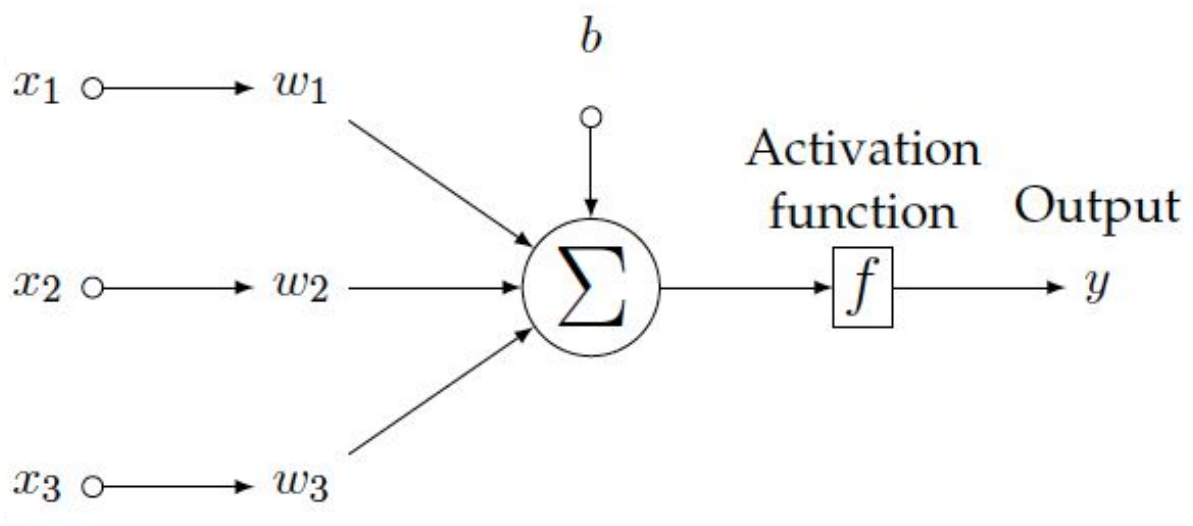


PYTORCH

```
lor = nn.Sequential(
    nn.Linear(NUM_INPUTS, 1),
    nn.Sigmoid()
)
```

Single neuron model: Binary Classification

Setting a **threshold** at the output of the perceptron allows solving classification problems between two classes (binary):



y > ɣ → class 1 (eg. green)

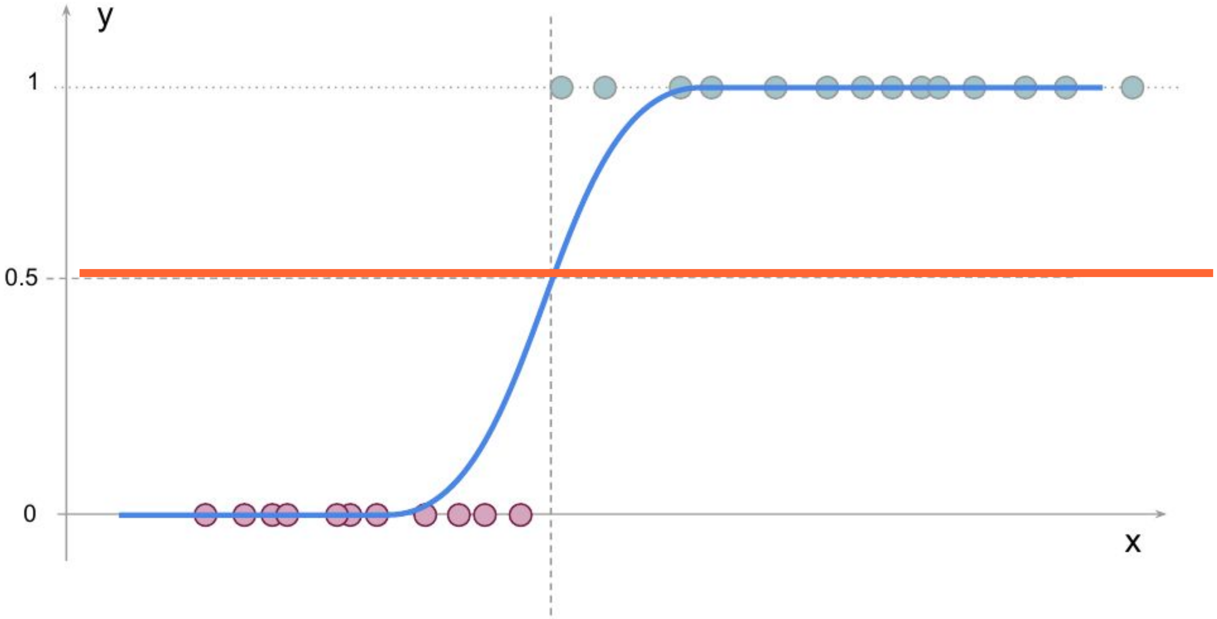
y < ɣ → class 2 (eg. red)

Single neuron model: Binary Classification

For classification, regressed values should be collapsed into 0 and 1 to quantize the confidence of the predictions ("probabilities").

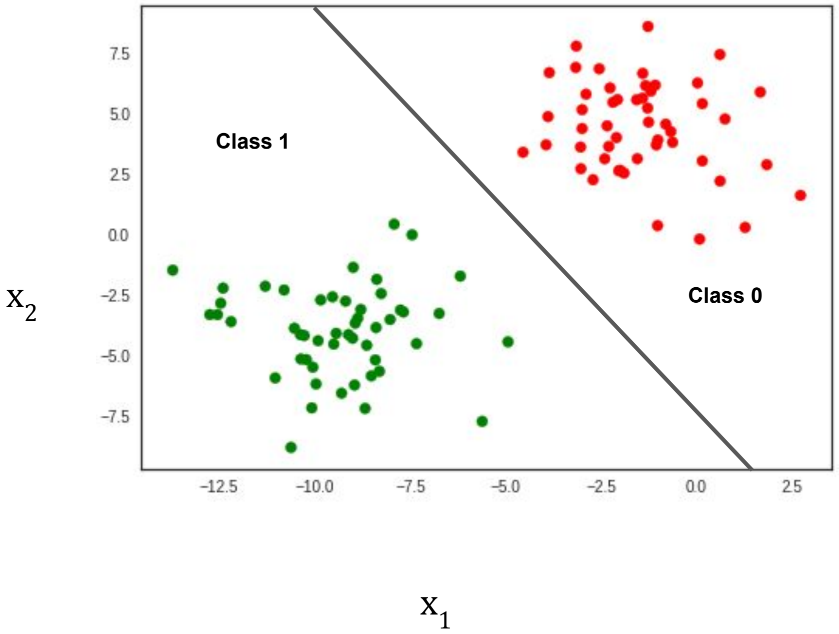
$$\sigma(\mathbf{x}) = \frac{1}{1 + e^{-(\mathbf{w}^T \cdot \mathbf{x} + b)}}$$

Threshold (γ)



Single neuron model: Binary Classification

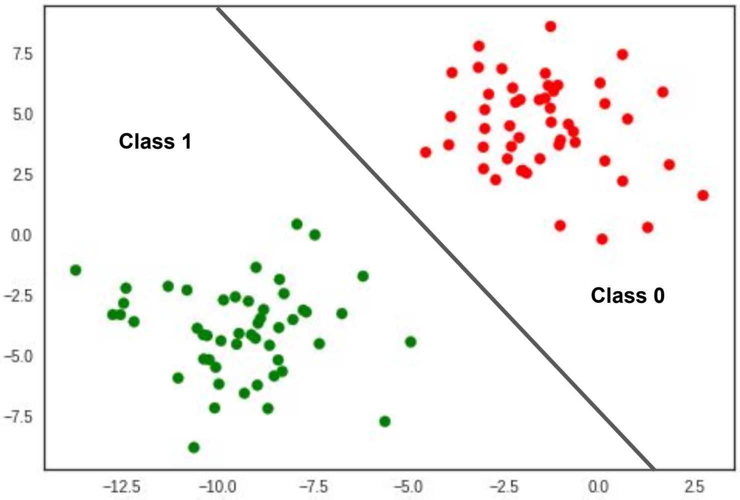
2D input space data



$$f(x) = \begin{cases} 1 & \text{if } w \cdot x + b > \ 0.5 \\ 0 & \text{otherwise} \end{cases}$$

Single neuron model: Binary Classification

2D input space data



x_2

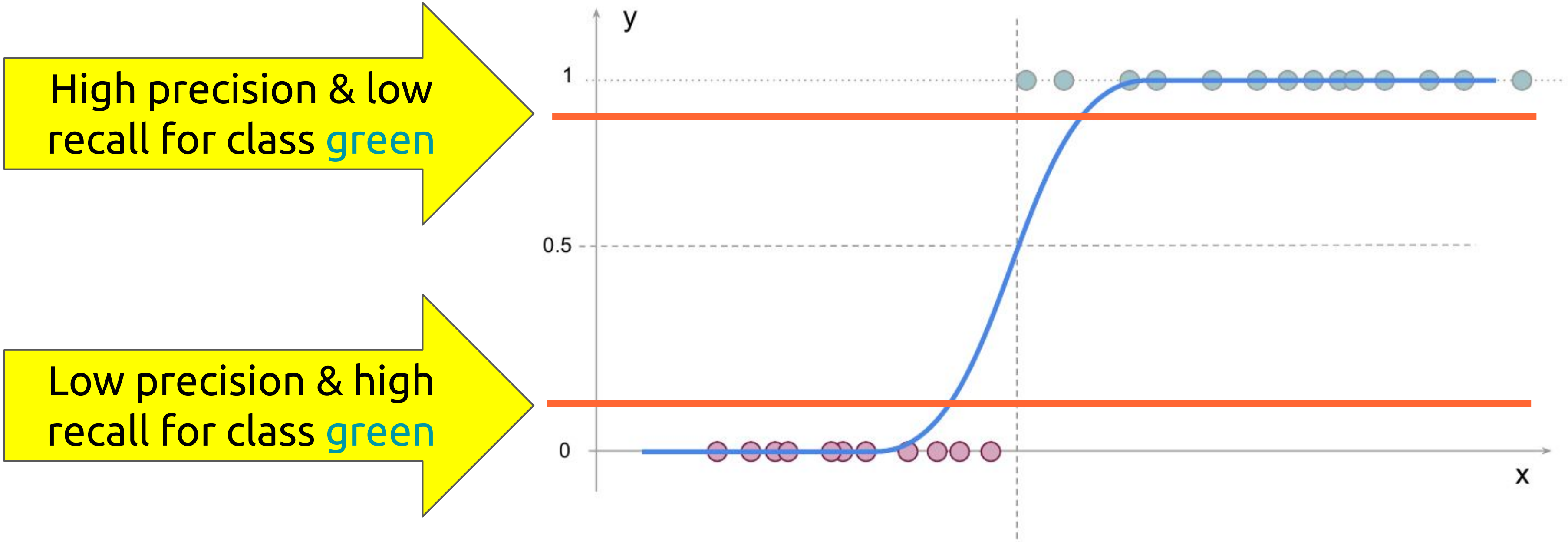
x_1

Parameters of the line.
They are estimated based on training data - *Learning Stage*.

$$f(x) = \begin{cases} 1 & \text{if } w \cdot x + b > 0.5 \\ 0 & \text{otherwise} \end{cases}$$

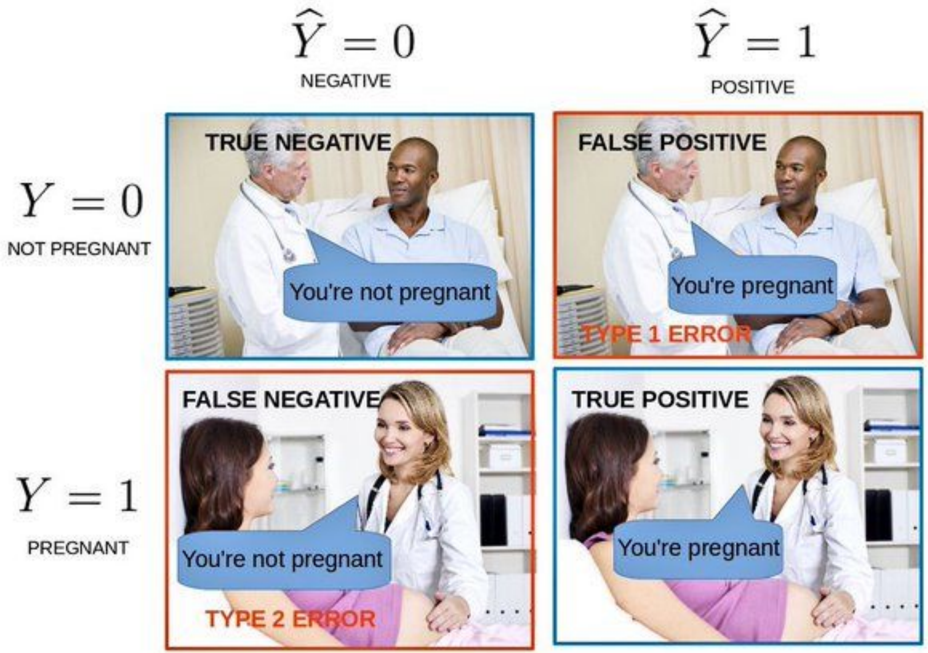
Single neuron model: Binary Classification

The classification threshold can be adjusted based on the desired precision - recall trade-off:



Single neuron model: Binary Classification

Adjusting the decision threshold allows defining whether false positives or false negatives are more critical for our application.



Single neuron model: Binary Classification

Exercise: Consider a binary classifier implemented with a single neuron modelled by two weights $w_1=0.2$ and $w_2=0.8$ and a bias $b=-1$. Consider the activation function to be a sigmoid $f(x) = 1 / (1+e^{-x})$.

a) Draw a scheme of the model.

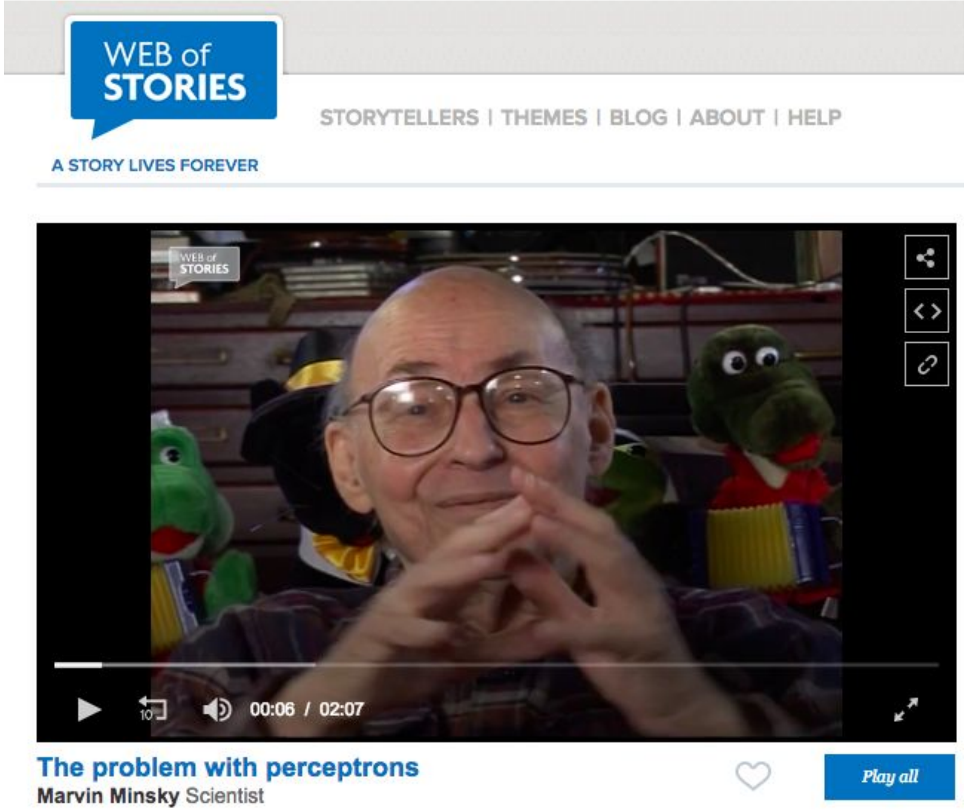
b) Compute the output of the logistic regressor for a given input $x=[1,1]$.

c) Considering a classification threshold of $y_{th}=0.9$ ($y_{th}>0.9$ for class A, and $y_{th}<0.9$ for class B), which class would be predicted for the considered input $x=[1,1]$?

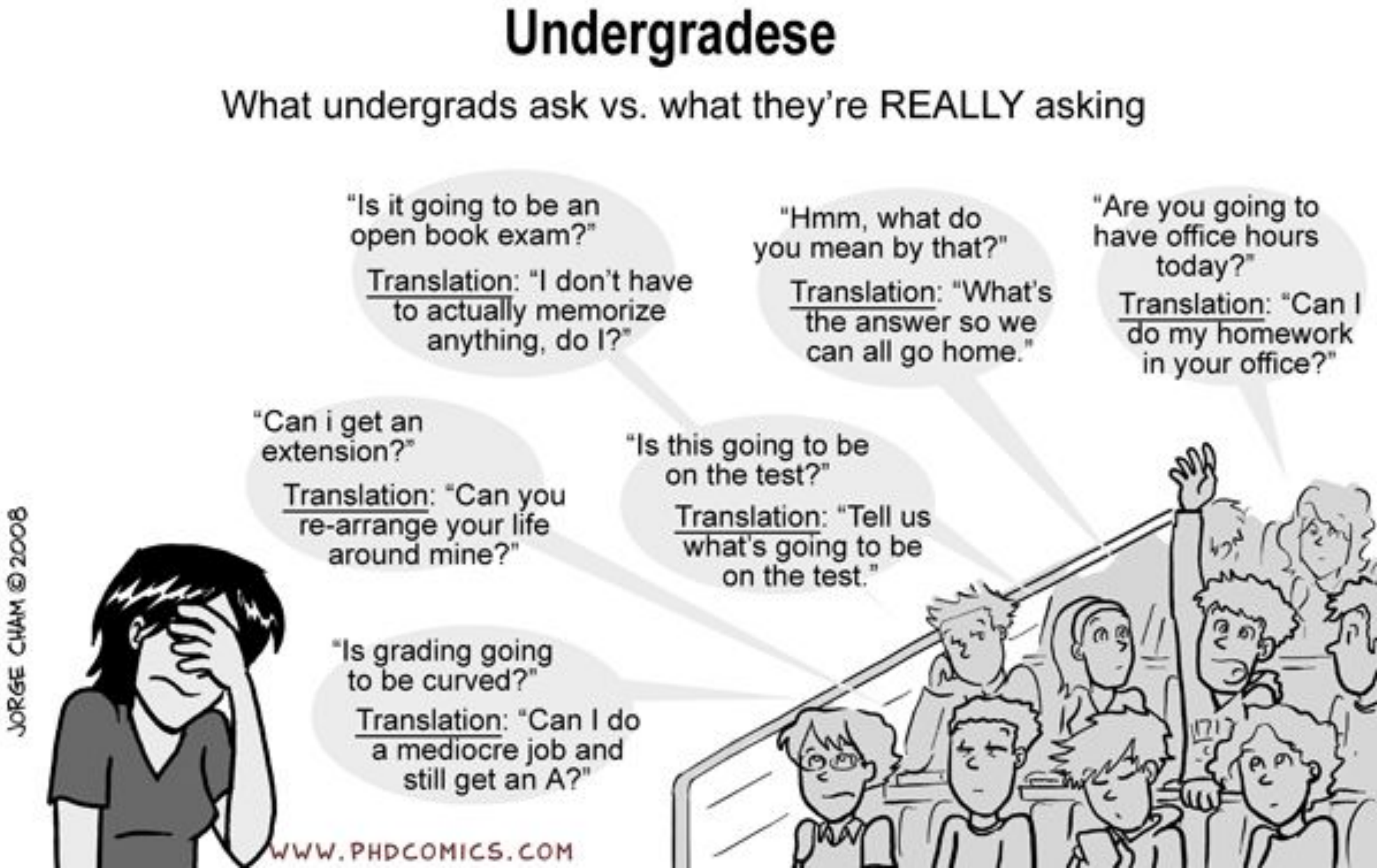
Outline

1. Single neuron model (Perceptron)
2. Regression
   a. Linear regression
   b. Logistic regression

Learn more



Minsky, Marvin, and Seymour A. Papert. Perceptrons: An introduction to computational geometry. 1969


Undergradese
What undergrads ask vs. what they're REALLY asking
"Is it going to be an open book exam?" Translation: "I don't have to actually memorize anything, do I?"
"Hmm, what do you mean by that?" Translation: "What's the answer so we can all go home."
"Are you going to have office hours today?" Translation: "Can I do my homework in your office?"
"Can i get an extension?" Translation: "Can you re-arrange your life around mine?"
"Is this going to be on the test?" Translation: "Tell us what's going to be on the test."
"Is grading going to be curved?" Translation: "Can I do a mediocre job and still get an A?"
JORGE CHAM © 2008
WWW.PHDCOMICS.COM