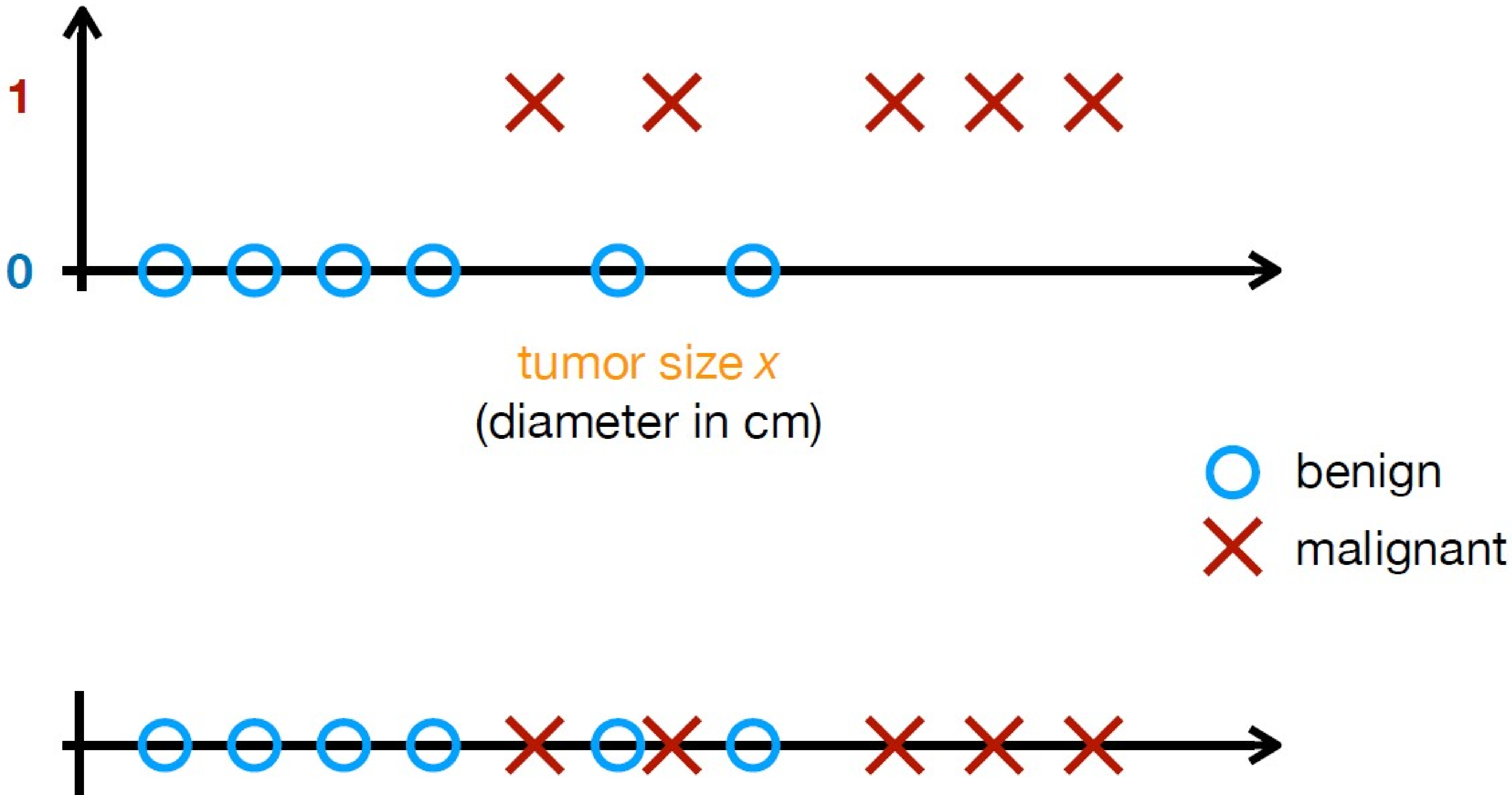
Tema 02. Regresión y clasificación

Autor: Ismael Sagredo Olivenza

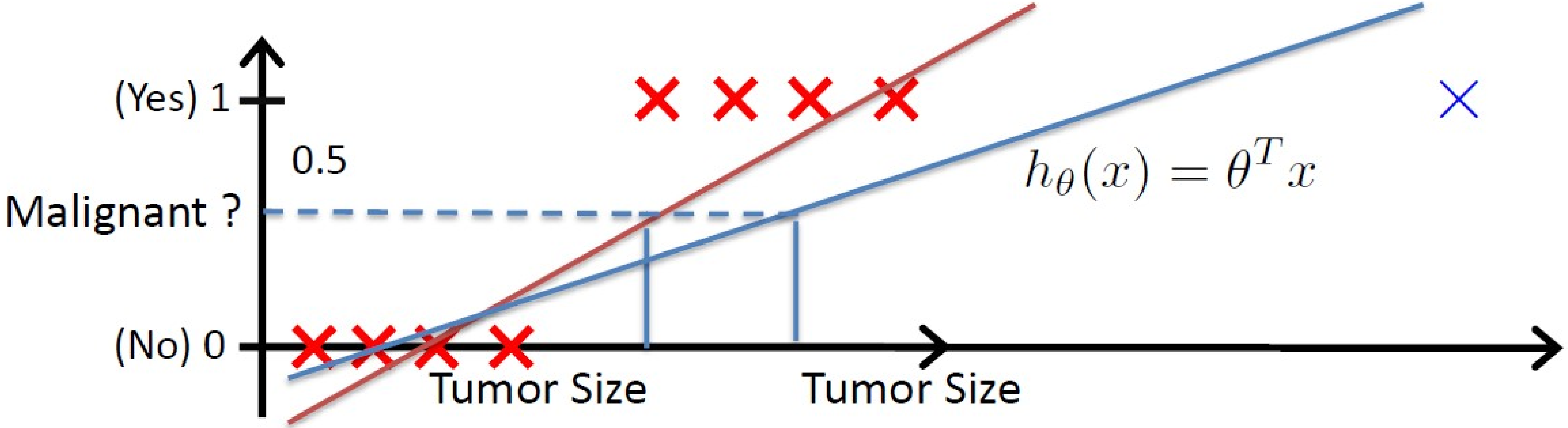
2.3. Classification

- So far, we have estimated the value of a variable based on the value of its features
- But there is another type of problem where we try deduce from input data what type of example it is, for example: wether an image is a cat or a dog, whether an email is spam or not, etc.
- Here, we don't expect a continuos output and we want to approximate it
- We expect a discrete output.

Classification: Breast cancer detection



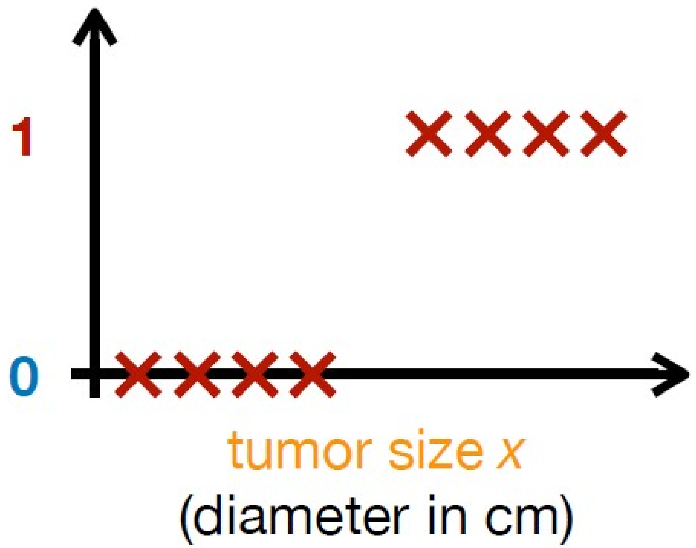
In this type of problems, we don't apply linear regression

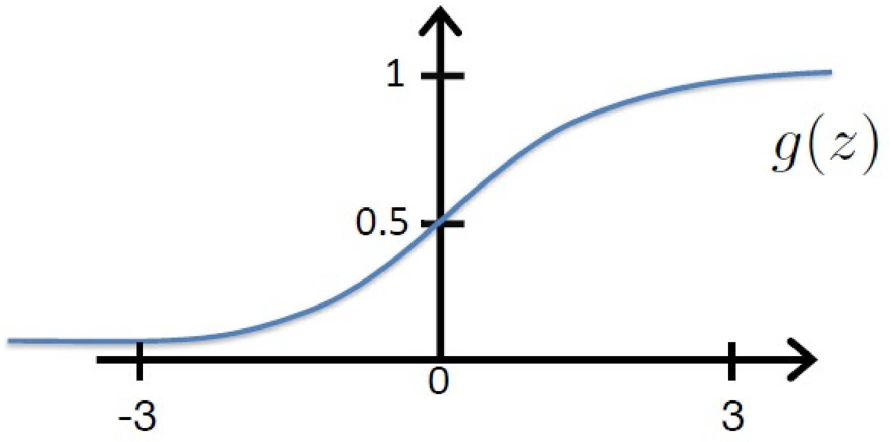


2.3.1 Logistic regression

Logistic functión: $g(z) = \frac{1}{1+e^{-z}} \$$

The output between 0 and 1.





2.3.1 Funtion to optimise

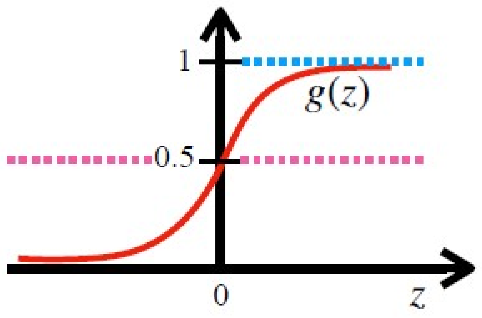
- By combining the logistic and regression functions, we obtain the following equation (In vector forma):

$$f_{\vec{w},b}(\vec{x}) = \frac{1}{1 + e^{-(\vec{w}\vec{x}+b)}}$$

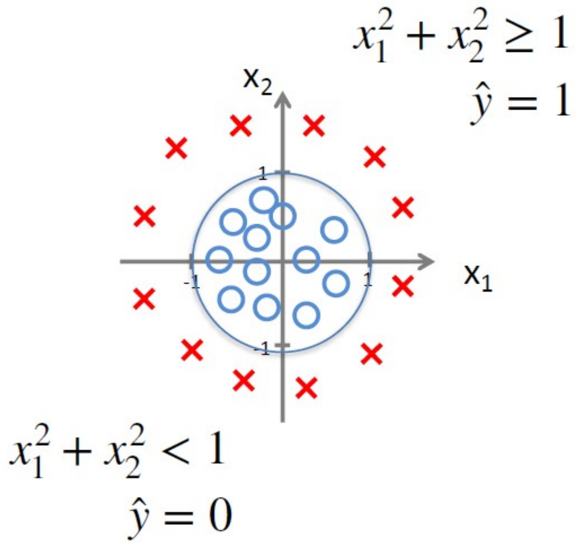
- The output represent the probability of a example belong to one class or the other. F(X) = 0.7 in the breast cancer domine represente a 70% probability to chance of getting cancer.

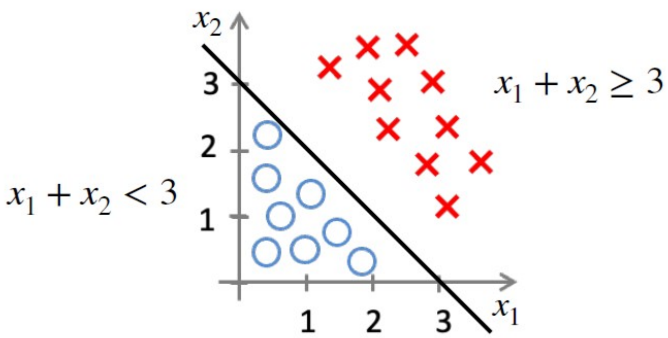
2.3.2 Decision boundary

- the expression we use in the logistic function (Z) determines how we delimit the groups.



$$z = \vec{w} \cdot \vec{x} + b$$





2.3.3 Cost Funtion

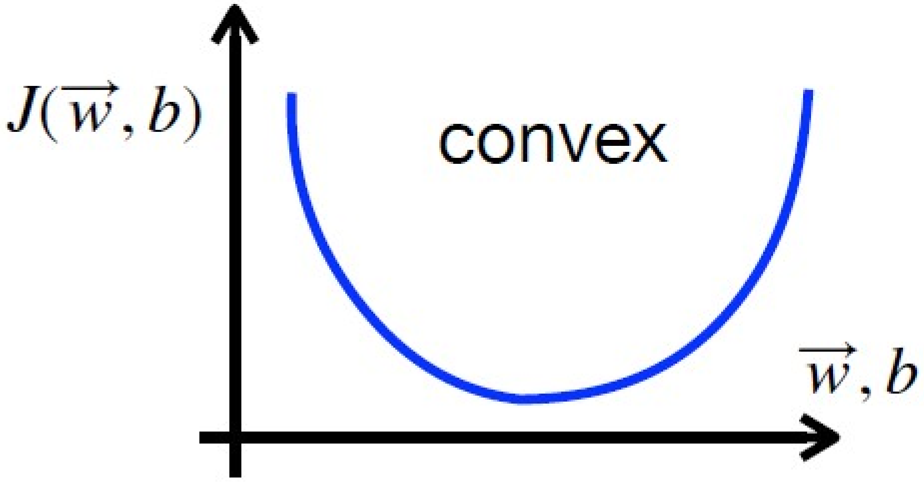
- Using the same error measure.

$$J(\vec{w}, b) = \frac{1}{2m} \cdot \sum_{i=1}^{m} (f_{\vec{w},b}(\vec{x}_i) - y_i)^2$$

- But $f_{\vec{w},b}(\vec{x}_i)$ is different.
- **Loss**: is a value that represents the summation of errors in our model. It measures how well (or bad) our model is doing.

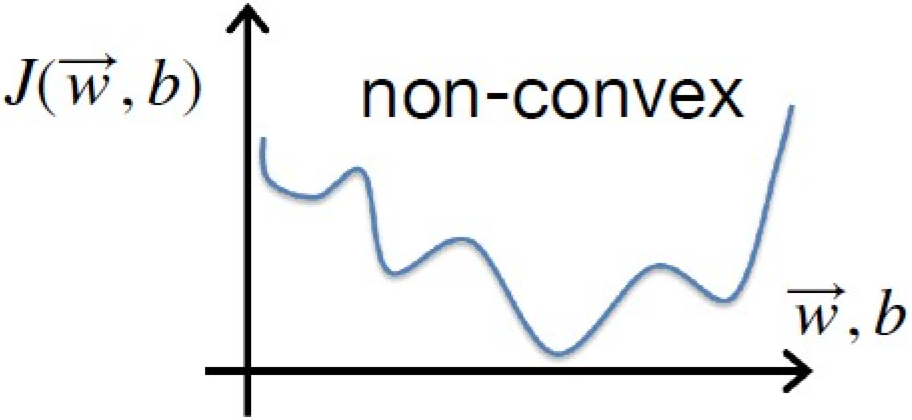
linear regression

$$f_{\overrightarrow{w},b}(\overrightarrow{x}) = \overrightarrow{w} \cdot \overrightarrow{x} + b$$



logistic regression

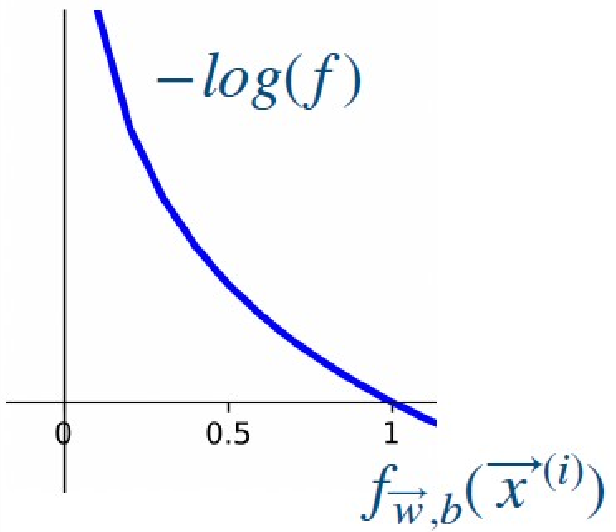
$$f_{\overrightarrow{w},b}(\overrightarrow{x}) = \frac{1}{1 + e^{-(\overrightarrow{w} \cdot \overrightarrow{x} + b)}}$$

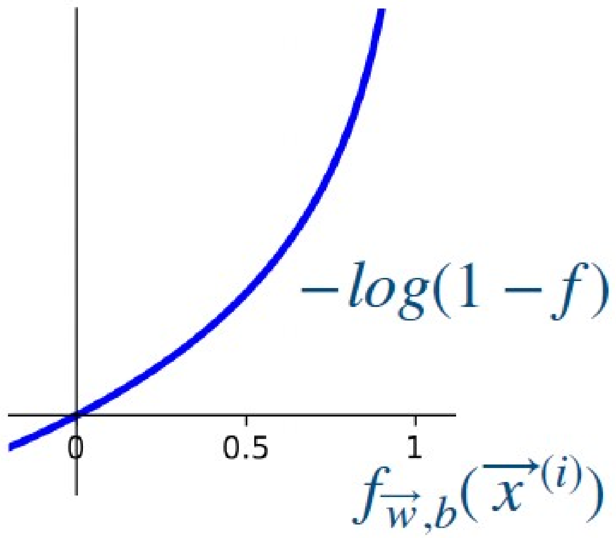


If $J(\vec{w}, b)$ is convex => Can reach global minimun. We will therefore use another cost function

2.3.3.1 Logistic funtion Loss (L)

- If $y_i = 1$ => $L(f_{\vec{w},b}(\vec{x}_i)) = -Log(f_{\vec{w},b}(\vec{x}_i))$
- if $y_i = 0$ => $L(f_{\vec{w},b}(\vec{x}_i)) = -Log(1 - f_{\vec{w},b}(\vec{x}_i))$
- Guaranteed to be convex for all input values, only one minimum





2.3.3.2 Simplified cost fuction

$$J(\vec{w}, b) = \frac{1}{m} \sum_{i=1}^{m} L[(f_{\vec{w},b}(\vec{x}_i), y_i]$$

$$J(\vec{w}, b) = -\frac{1}{m} \sum_{i=1}^{m} [y_i log(y_i') + (1 - y_i)log(y_i')]$$

where

$$y' = f_{\vec{w},b}(\vec{x}_i) = \frac{1}{1 + e^{-(\vec{w}\vec{x}+b)}}$$

2.3.4 Gradient descent

If we derive (J) we see that the result of the derivative is the same as in the normal regression algorithm

$$w_j = w_j - \alpha \frac{1}{m} \sum_{i=1}^{m} y_i' - y_i x_{j,i}$$

$$b = b - \alpha \frac{1}{m} \sum_{i=1}^{m} y_i' - y_i$$

where
$y' = f_{\vec{w},b}(\vec{x}_i) = \frac{1}{1+e^{-(\vec{w}\vec{x}+b)}}$

2.3.5 Multiclass classification

We apply 1 class vs all the rest for each of the classes by calculating their probability.

On a new input x, to make a prediction, pick the class i that maximizes:

$$\max_i f^{(i)}_{\vec{w},b}(\vec{x})$$