# Model and Cost function

Not^ns :

$m$ : no. of training ex.s

$x$'s : input var.s/ features

$y$'s : output var.s/ ~~features~~ target var. that we are trying to predict

$(x,y)$ : one training ex. / observ^n

$(x^{(i)}, y^{(i)})$: $i^{th}$ training ex. / observ^n

$(x^{(i)}, y^{(i)})$; $i = 1, 2, \ldots, m$ : training set

$i$ : index into the training set

$X$ : space of I/P val.s

$Y$ : space of O/P val.s

$n$ : no. of features

$$\boxed{\text{Hypothesis} : h_\theta(x) = \theta_0 + \theta_1 x}$$

$\theta_i$'s : para. of model

# $h$ maps from $x$'s to $y$'s

## Classification.

The classification problem is just like the regression problem, except that the values we now want to predict take on a very small no. of discrete val-s.

i.e; we take some input & label it into 1 of ~~our~~ our known classific^ns.

ex : Email : spam/not spam

Online Transaction : fraudulent (y/n)

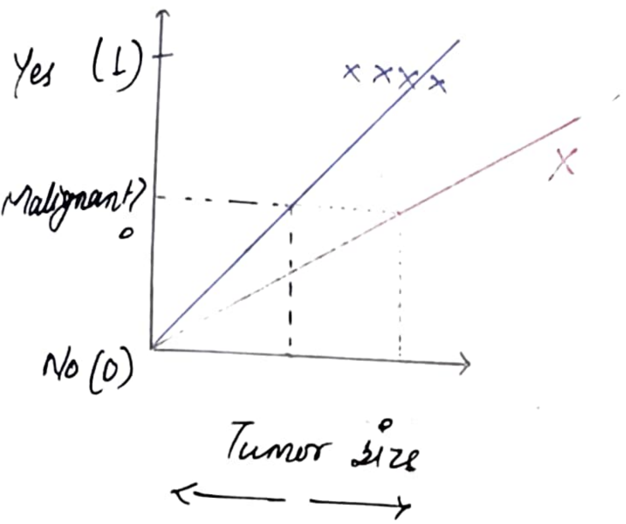
Tumor : Malignant / Benign

# for now, we will focus on binary classification problem.

## Hypothesis Representation

↳ fun^n that we're going to use to represent our hypoth. when we have a classific^n probl.

# Why Linear Regn. won't work at a Classific^n problem:




i.e; on adding 1 irrelevant data set, the linear fit changes & this doesn't work anymore

Threshold classifier o/p $h(\theta)$ at 0.5:

If $h_\theta(x) \geq 0.5$ , predict "$y=1$"

If $h_\theta < 0$ , predict "$y=0$".

.....

$\because$ we k/ that $y \in \{0, 1\}$ (discrete val-s),

we change our hypoth to satisfy $0 \leq h_\theta(x) \leq 1$

i.e.,

Classific^n (binary) : $y = 0$ or $1$

but in Lin. reg., $h_\theta(x)$ can be $>1$ or $<0$

So, Logistic Regr. uses logistic or Sigmoid fun^n

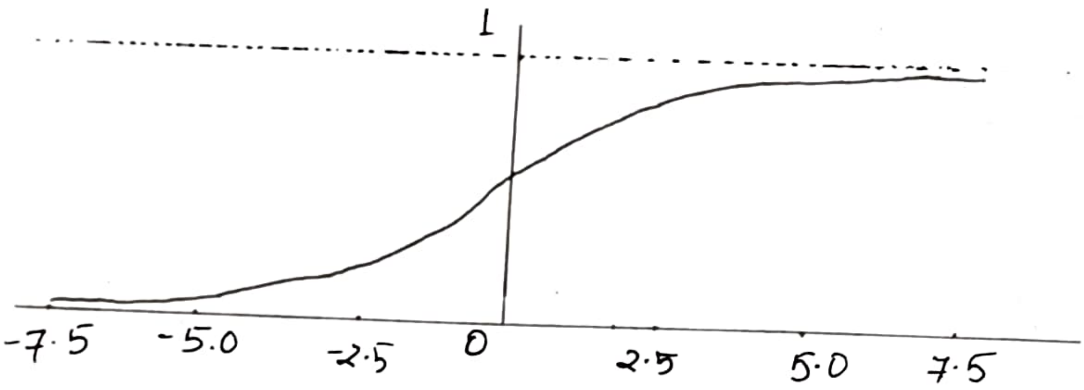
$0 \leq h_\theta(x) \leq 1$

$$h_\theta(x) = g(\theta^T x)$$

$$z = \theta^T x$$

where, $g(z) = \dfrac{1}{1+e^{-z}}$

Sigmoid / Logistic function




The sigmoid funⁿ maps any real no. to the (0,1) interval.

It asymptotes at 0/1 as it approaches $-\infty/\infty$.

continuously approaches but doesn't meet

Interpretⁿ of Hypoth. o/p :

$h_\theta(x)$ = estimated probability that $y=1$ on $x$.

on i/p $x$.

ex: if $x = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 1 \\ \text{tumorsize} \end{bmatrix}$ $x_0$ is always 1

← feature

$h_\theta(x) = 0.7$ for some (patient w some tumor size) input $x$

tell patient that 70 % chance of tumor being malignant.

Mathematically,

$h_\theta(x) = P(y=1 \mid x;\theta)$ "prob that $y=1$ given $x$ parameterized by $\theta$"

# given $x$, we calc. the prob. of $y=1$

# Also,

$$P(y=0 \mid x;\theta) + P(y=1 \mid x;\theta) = 1$$

$$P(y=0 \mid x;\theta) = 1 - P(y=1 \mid x;\theta) = 1 - 0.7$$

$$P(y=0 \mid x;\theta) = 0.3$$ prob that tumor is benign

.....

$z = 0$ , $e^0 = 1$ $\Rightarrow$ $g(z) = \frac{1}{2}$

$z \to \infty$ , $e^{-\infty} \to 0$ $\Rightarrow$ $g(z) = 1$

$z \to -\infty$ , $e^{\infty} \to \infty$ $\Rightarrow$ $g(z) = 0$

# Decision Boundary

↳ is the line that separates the area where $y=0$ and where $y=1$.
It is created by our hypoth funⁿ.

ex: ①

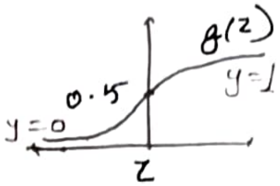
$$h_\theta(x) = g(\theta^T x) = P(y=1 \mid x;\theta)$$

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

Now, $z=0, e^0=1$

$$\Rightarrow g(z) = \frac{1}{2} = 0.5$$

so, predict "$y=1$" if $h_\theta(x) \geq 0.5$
($\rightarrow \theta^T x \geq 0$)




&

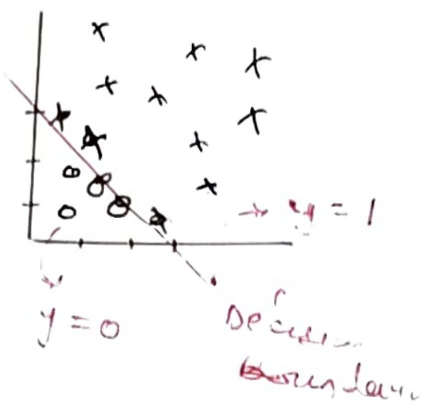
predict "$y=0$" if $h_\theta(x) < 0.5$
($\rightarrow \theta^T x < 0$)

② let, some training set:

$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

let, $\theta_0 = -3, \theta_1 = 1, \theta_2 = 1$

So, $\theta = \begin{bmatrix} -3 \\ 1 \\ -1 \end{bmatrix}$




# we're trying to figure out where a hypoth. would end up predicting $y = 1/0$

Now, predict "y=1" if $-3 + x_1 + x_2 \geq 0$

"y=1" if $x_1 + x_2 \geq 3$

"y=0" if $x_1 + x_2 < 3$

# Dec. Boundary is a prop of hypoth $h_\theta(x)$ & it's paras ($\theta_0, \theta_1, \theta_2, \dots$) & not a prop. of data set.

# So even if we remove the training set (Xs & Os) for this hypoth., the dec. boundary would remain the same.

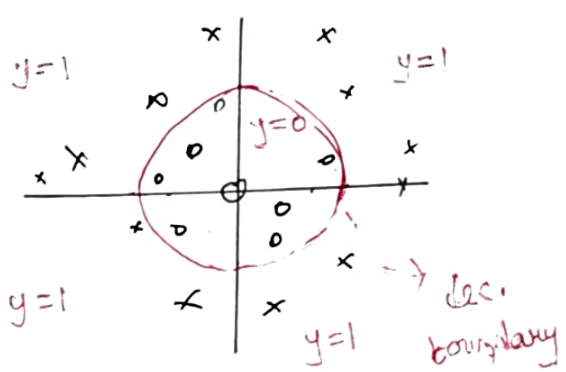
# the training set may be used to fit the paras $\theta$.

## Non-Linear Decision Boundaries

Higher order polynomial features can have complex dec. boundaries.

i.e, for more complex problems, you can get dec. boundaries w much more complex shapes by adding higher order terms ($x_1^2$, $x_1 x_2^3 x_3, \dots$)

ex:




$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2)$

:

predict "y=1" if

$x_1^2 + x_2^2 \geq 1$

# circle eq.

## Cost function.

\# We cannot use the same cost function that we use for lin. regr. bc the Logistic fun^n will cause the o/p to be wavy, causing many local optima.
i.e., it will not be a convex fun^n.

~~Out~~ Cost fun^n for Logistic Regression:

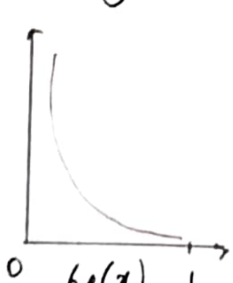
$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} \text{Cost}\left(h_\theta(x^{(i)}), y^{(i)}\right)$$

$$\text{Cost}(h_\theta(x), y) = -\log(h_\theta(x)) \quad \text{if } y = 1$$

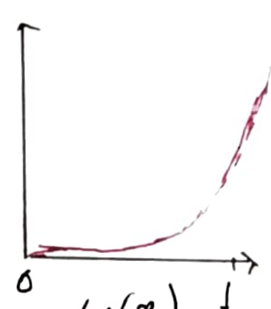
$$\text{Cost}(h_\theta(x), y) = -\log(1 - h_\theta(x)) \quad \text{if } y = 0.$$

$J(\theta)$ vs $h_\theta(x)$ plots when

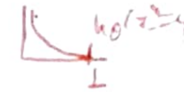
$y = 1$



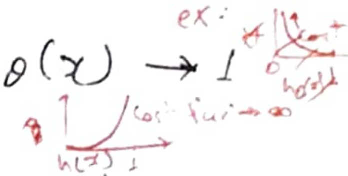

$y = 0$



$Cost(h_\theta(x), y) = 0$ if $h_\theta(x) = y$ (=1 ex:

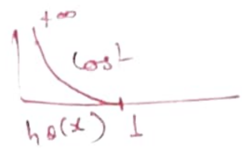


$Cost(h_\theta(x), y) \to \infty$ if $y = 0$ and $h_\theta(x) \to 1$ ex:



$Cost(h_\theta(x), y) \to +\infty$ if $y = 1$ and $h_\theta(x) \to 0$

ex:




If our correct answer 'y' is 0, then the cost funⁿ will be 0 if our hypoth. fun also O/P is 0. If our hypoth. approaches 1, then the cost funⁿ will approach ∞.

If our correct answer 'y' is 1, then the cost funⁿ will be 0 if our hypoth fun also O/P is 1. If the hypoth approaches 0, then the cost funⁿ will approach ∞.

# Writing cost funⁿ in this way guarantees that $J(\theta)$ is convex for Logistic regr.

# (Simplified) Cost function and Grad. Desc.

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} \text{Cost}\left(h_\theta(x^{(i)}), y^{(i)}\right)$$

where, $\text{Cost}(h(x), y) = \begin{cases} -\log(h_\theta(x)) & , \text{ if } y=1 \\ -\log(1-h_\theta(x)) & , \text{ if } y=0 \end{cases}$

$$\boxed{\text{Cost}(h_\theta(x), y) = -y \log(h_\theta(x)) - (1-y)\log(1-h_\theta(x))}$$

So,

if $y = 1$

$$\text{Cost}(h_\theta(x), y) = -1 \cdot \log(h_\theta(x)) - \cancel{(1-1)\log(1-h_\theta(x))}^{\,0}$$

$$= -\log(h_\theta(x))$$

if $y = 0$

$$\text{Cost}(h_\theta(x), y) = -\cancel{0 \cdot \log(h_\theta(x))}^{\,0} - (1-0)\log(1-h_\theta(x))$$

$$= -\log(1-h_\theta(x))$$

$$J(\theta) = -\frac{1}{m}\left[\sum_{i=1}^{m} y^{(i)} \log\left(h_\theta\left(x^{(i)}\right)\right) + \left(1-y^{(i)}\right) \log\left(1-h_\theta\left(x^{(i)}\right)\right)\right]$$

Vectorized implementⁿ :

$$h = g(X\theta)$$

$$J(\theta) = \frac{1}{m} \cdot \left(-y^T \log(h) - (1-y)^T \log(1-h)\right)$$

to fit para.s $\theta$ :

$\min_\theta J(\theta) \rightarrow$ get $\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \end{bmatrix}$

to make a predic' given new $x$ :

Output $h_\theta(x) = \dfrac{1}{1+e^{\theta^T x}}$ $\quad P(y=1 \mid x; \theta)$

prob. that $y=1$, given i/p $x$, parameterized by $\theta$.

Grad. Desc.

general form of grad. desc is

Repeat

{

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$

(simultaneously update all $\theta_j$)

}

# here, $\frac{\partial}{\partial \theta_j} J(\theta) = \frac{1}{m} \sum_{i=1}^{m} \left( h_\theta(x^{(i)}) - y^{(i)} \right) x_j^{(i)}$. So,

$$\boxed{\begin{array}{l} \text{Repeat} \\ \{ \\ \quad \theta_j := \theta_j - \frac{\alpha}{m} \sum_{i=1}^{m} \left( h_\theta(x^{(i)}) - y^{(i)} \right) x_j^{(i)} \\ \} \end{array}}$$

Vectorized Implementⁿ:

$$\cancel{\theta_j := \theta_j -} \quad \boxed{\theta := \theta - \frac{\alpha}{m} X^T \left( g(X\theta) - y \right)} \qquad \theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{bmatrix}$$

# We observe that the grad. desc. algo. for both lin. regr. & log. regr. is identical just that the def of their hypoth. funⁿ has changed $\left( \theta^T x \; / \; \frac{1}{1+e^{-\theta^T x}} \right)$

## Multiclass Classification : one-vs-all.

↓

Classific^n of data for more than 2 categories

i.e., instead of $y = \{0,1\}$, our def. expands to

$$y = \{0, 1, \dots, n\}$$

Since $y = \{0, 1, \dots, n\}$, we divide our problem into $n+1$ (+1 bc index starts at 0) binary classific^n problems; in each one, we predict the probab. that 'y' is a member of one of our classes.

$$y \in \{0, 1, \dots, n\}$$

$$h_\theta^{(0)}(x) = P(y=0 \mid x;\theta)$$

$$h_\theta^{(1)}(x) = P(y=1 \mid x;\theta)$$

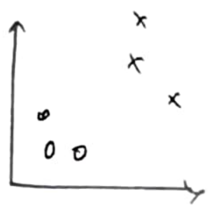
$$\vdots$$

$$h_\theta^{(n)}(x) = P(y=n \mid x;\theta)$$

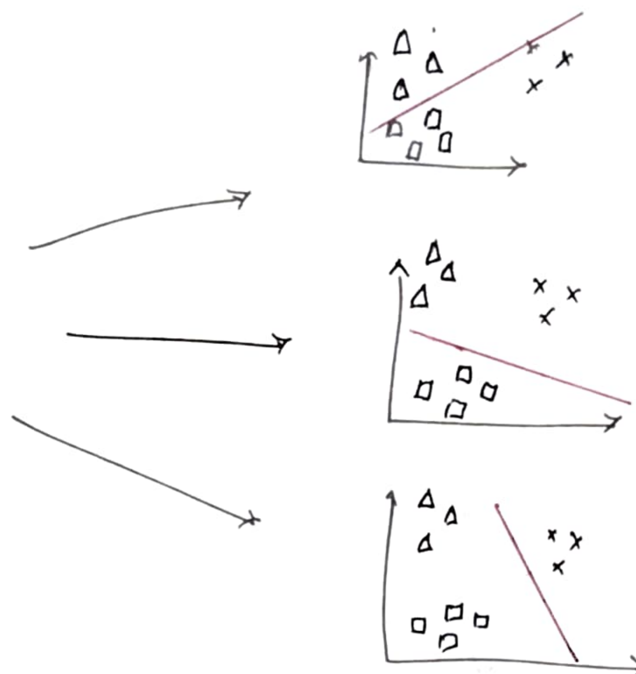
$$\text{prediction} = \max_i (h_\theta^{(i)}(x))$$

We are basically choosing 1 class and lumping all the others into a secondary class.
We do this repeatedly, applying binary log. regr. to each case, & then use the hypoth. that returned the highest val. as our predic^n.

Binary Classificⁿ Multiⁿ- Class Classificⁿ



One - vs - all



Class 1 : Δ

Class 2 : □

Class 3 : ×

$$h_\theta^{(i)}(x) = P(y=i \mid x;\theta)$$
$$(i = 1,2,3)$$

<u>One - vs - all</u>

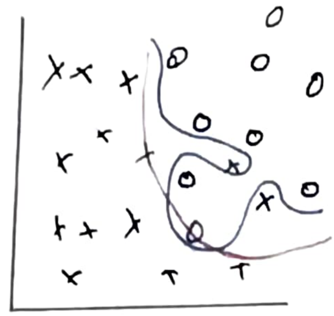
→ Train a log. reg. classifier $h_\theta^{(i)}(x)$ for each class $i$. to predict the prob. that $y=i$

→ On a new I/P $x$, to make a predicⁿ, run $h_\theta^{(i)}(x)$ for all $i = 1,2,3,..$ & pick the class that maximizes $h_\theta(x)$

→ Pick the class $i$ that maximizes $\max_i h_\theta^{(i)}(x)$

# Regularized Logistic Regression

~~optimiz^n objective~~:



$$h_\theta(x) = g(\theta_0 + \theta_1 x + \theta_2 x_1^2 + \theta_3 x_1^2 x_2 + \theta_4 x_1^2 x_2^2 + \theta_5 x_1^2 x_2^3 + \dots)$$

optimiz^n objective:

$$J(\theta) = -\left[\frac{1}{m}\sum_{i=1}^{m} y^{(i)} \log(h_\theta(x^{(i)})) + (1-y^{(i)}) \log(1-h_\theta(x^{(i)}))\right]$$

$$+ \frac{\lambda}{2m}\sum_{j=1}^{n} \theta_j^2$$ ← regularized log. regr.

Grad. Desc.

Repeat {

$$\theta_0 := \theta_0 - \alpha \cdot \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right) x_0^{(i)}$$

$$\theta_j := \theta_j - \underbrace{\alpha\left[\frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right) x_j^{(i)} + \frac{\lambda}{m}\theta_j\right]}_{\frac{\partial}{\partial\theta_j} J(\theta)}$$

$j = 1, 2, \dots, n$

}

$$h_\theta = \frac{1}{1+e^{-\theta^T x}}$$