1.

Logistic Regression - Cost function

	x_0	x_1	x_2	...	x_d	y
1.	1	x_{11}	x_{12}	...	x_{1d}	y_1
2.	1	x_{21}	x_{22}	...	x_{2d}	y_2
⋮	⋮	⋮	⋮	⋱	⋮	⋮
N	1	x_{N1}	x_{N2}		x_{Nd}	y_N

$y \rightarrow 1/0$

training samples.

$$\hat{y}_B(x) = g(b_0 x_0 + b_1 x_1 + \cdots + x_d b_d)$$

$$= g(B^T \cdot x)$$

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

How to get values for $B = [b_0 \; b_1 \; \ldots b_d]$?

In Linear Reg, we have Error fn as,

$$E(B) = \frac{1}{N} \sum_{i=1}^{N} \underbrace{\frac{1}{2}(\hat{y}_i - y_i)^2}_{cost(B, (x_i, y_i))}$$

$\downarrow$

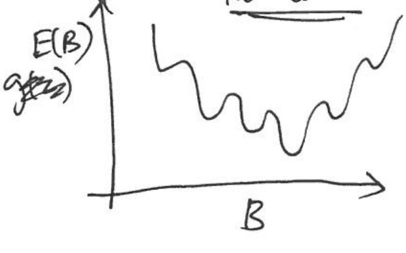
OK for Linear Reg.

But for logistic Reg, when we use the above cost fn, we have, non-convex fn. of B.

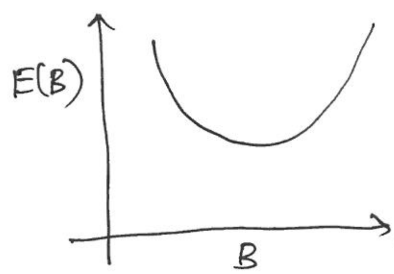
i.e., $\hat{y}_B(x) = g(z) = \left(\frac{1}{1+e^{-B^T \cdot x}}\right)$

$\downarrow$

Non-convex



"convex"

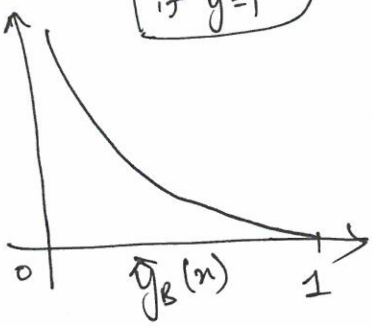


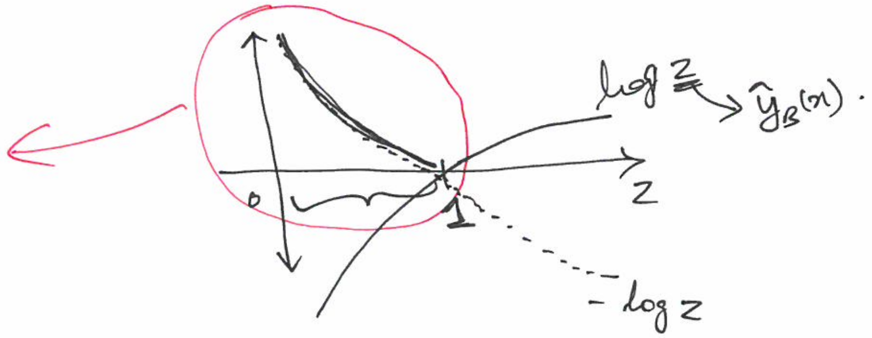
So we need a different Cost/Error fn for logistic Reg.

② **Cost. Fn. for Logistic Reg:**

$$cost(\hat{y}_B(x_i), y_i) = \begin{cases} -\log(\hat{y}_B(x_i)), & \text{if } y = 1. \\ -\log(1 - \hat{y}_B(x_i)), & \text{if } y = 0. \end{cases}$$

$\boxed{\text{if } y=1}$





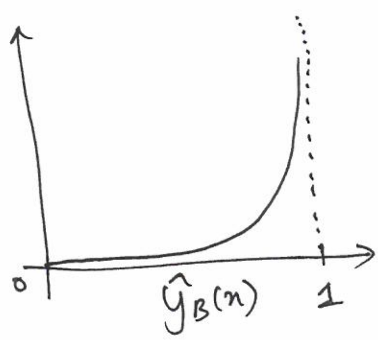
$\Downarrow$

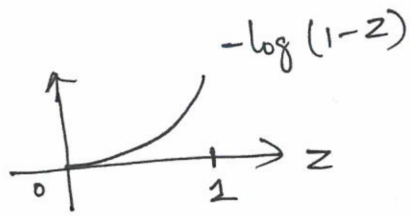
Interesting Properties of Our Cost fn:-

1. Cost $= 0$ if $y = 1$ and $\hat{y}_B(x) = 1$. } if we correctly predict the y, then cost is 0.

   But as $\hat{y}_B(x) \rightarrow 0$
   Cost $\rightarrow \infty$ } if we mispredict the y, then cost is ∞.

$\boxed{\text{if } y=0}$





2. cost $= 0$ if $y = 0$ and $\hat{y}_B(x) = 0$ } if we correctly predict, ~~there~~ y, then cost is 0.

   But as $\hat{y}_B(x) \rightarrow 1$
   cost $\rightarrow \infty$ } if we wrongly predict y, then cost is ∞.

$\boxed{\text{Note:}}$ Our learning algorithms will try to minimize the cost fn. Therefore when cost $\rightarrow \infty$, we will not choose the param B that corresponds cost of ∞.

③

Re-writing Cost fn:

$$E(B) = \frac{1}{N} \sum_{i=1}^{N} \underbrace{Cost\left(\left(\hat{y}_B(x_i), y_i\right)\right)}$$

$$Cost\left(\hat{y}_B(x), y\right) = \begin{cases} -\log\left(\hat{y}_B(x)\right) & \text{if } y=1 \\ -\log\left(1-\hat{y}_B(x)\right) & \text{if } y=0 \end{cases}$$

Output variable $y = 0$ or 1.

$$\therefore \boxed{cost\left(\hat{y}_B(x), y\right) = -y \log\left(\hat{y}_B(x)\right) - (1-y)\log\left(1-\hat{y}_B(x)\right)}$$

if $y=1$) $\Rightarrow$ $cost(\hat{y}_B(x), y) = -(1)\log\left(\hat{y}_B(x)\right) - (0)$

if $y=0$) $\Rightarrow$ $cost\left(\hat{y}_B(x), y\right) = -(0) - (1-0)\log\left(1-\hat{y}_B(x)\right)$

$$\therefore E(B) = \frac{-1}{N}\left[\sum_{i=1}^{N} y_i \log\left(\hat{y}_B(x_i)\right) + (1-y_i)\log\left(1-\hat{y}_B(x_i)\right)\right]$$

To fit parameters B,

$\min_B E(\underline{B}) \longrightarrow$ Get B^* $\longrightarrow$ Use Gradiant Descent.

To make a prediction, given a new x:

Output $\hat{y}_B(x) = \dfrac{1}{\left(1+e^{-B^T \cdot x}\right)}$

$\downarrow$

$P(y=1 \mid x, B)$?

$\downarrow$

$\geq 0.5 \Rightarrow \hat{y}_B(x) = 1$

$< 0.5 \Rightarrow \hat{y}_B(x) = 0.$

④ **G. Descent:**

$$E(B) = \frac{-1}{N}\left[\sum_{i=1}^{N} y_i \log\left(\hat{y}_B(x_i)\right) + (1-y_i)\log\left(1-\hat{y}_B(x_i)\right)\right]$$

We need, $\min\limits_{B} E(B)$:

Repeat {

$$b_j := b_j - \alpha \underbrace{\frac{\partial E(B)}{\partial b_j}} \rightarrow \text{simultaneous update.}$$

(for $j = 1, 2, \ldots d$)

use vectorize implementation. }

$$\boxed{\frac{\partial E(B)}{\partial b_j} = \frac{1}{N}\sum_{i=1}^{N}\left(\hat{y}_B(x_i) - y_i\right) x_{ij}}$$

$B = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_d \end{bmatrix}$; <u>Note</u>: $\frac{\partial E(B)}{\partial b_j}$ look same as linear regression,

but $\boxed{\hat{y}_B(x_i) = B^T.x}$ for $\boxed{\text{linear regression.}}$

for $\boxed{\text{logistic regression,}}$

$$\boxed{\hat{y}_B(x_i) = g(B^T.x)}$$

$$= \frac{1}{(1+e^{-B^T.x})}$$