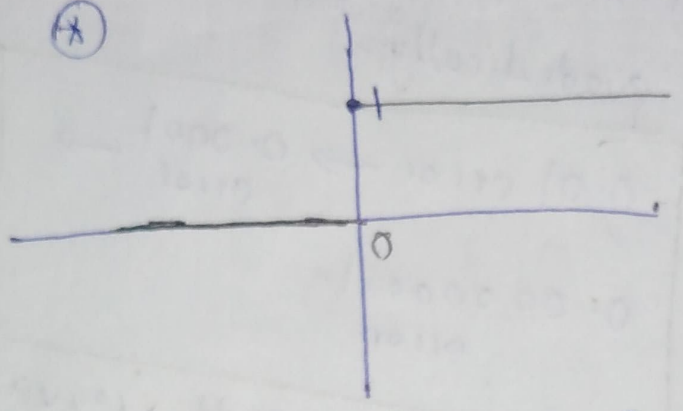
Lecture-4

⊛ Preceptron Algorithm:-

⊛



$$g(z) = \begin{cases} 1 & z \geq 0 \\ 0 & z < 0 \end{cases}$$

→ Hard version of sigmoid.

→ $h_\theta(x) = g(\theta^T x)$.

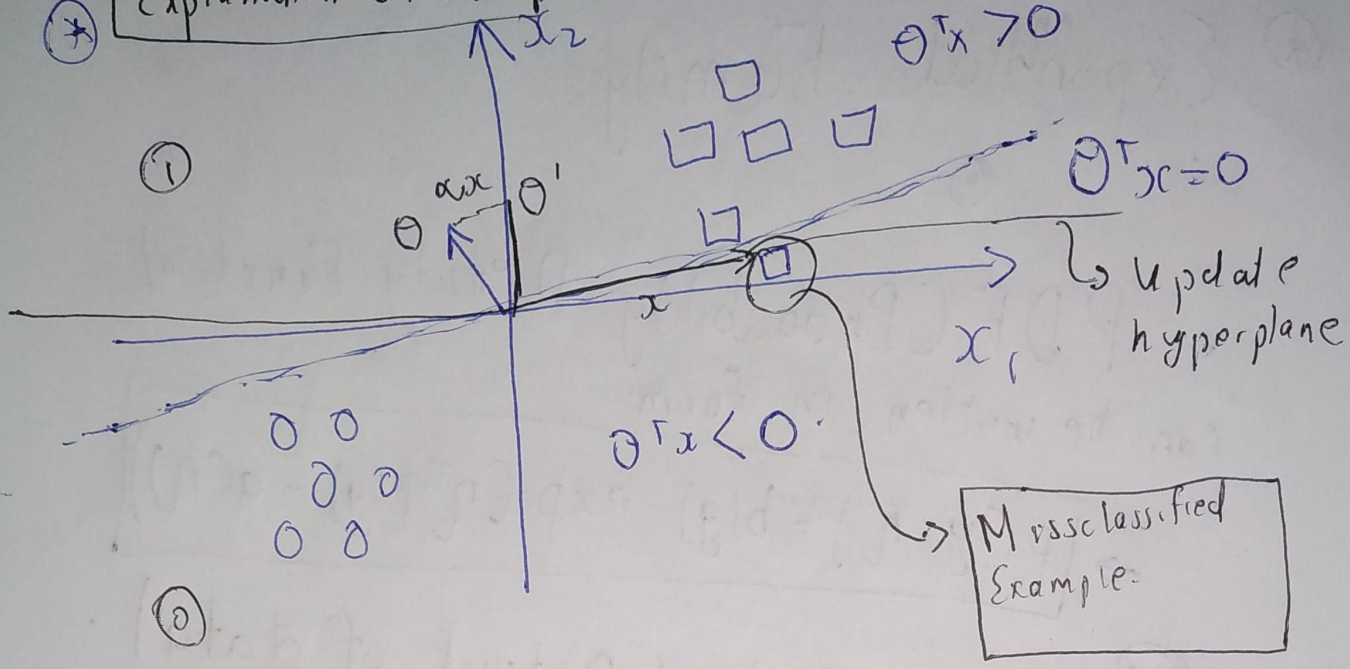
Update Rule for Preceptron algorithm.

⊛ $\boxed{\theta_j := \theta_j + \alpha \left(y^{(i)} - h_\theta(x^{(i)})\right) \cdot x_j^{(i)}}$.

→ Same as logistic Regression.

- $y^{(i)} - h_\theta(x^{(i)})$ would be zero if correctly classify otherwise it would be negative.

(*) Explanation of P.A :-



$\rightsquigarrow$ ① $\theta \approx x \mid y = 1$

$\rightsquigarrow$ ② $\theta \not\approx x \mid y = 0$

→ ① θ and x dot product would be positive if they are similar otherwise it would be negative therefore we want θ' to be close to x as shown in the figure ↓ (This way examples would be classified correctly).

① Remember the update Rule if example is classified correctly then we do nothing otherwise if example is positive[1] and we misclassified it negative[0] then we add to vector θ (See figure)

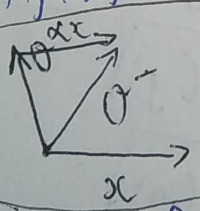
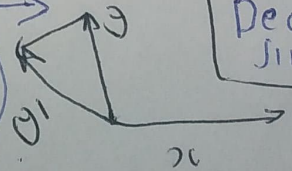


Fig Increasing Similarity

in case of positive misclassification

② Otherwise if example is negative meaning label is zero and we classified it as positive (1) then we subtract vector



Decreasing Similarity

⊛ Exponential Family:

⊙

- PDF (Probability Density Function)

Can be written in form:-

→ $\boxed{p(y;\eta) = b(y) \cdot \exp(\eta^T T(y) - a(\eta))}$.

⊛
- y — Data (output of data).
- η — natural parameter. ← Can be a vector
- $T(y)$ — Sufficient Statistics. ← Can Assume $T(y) = y$
- $b(y)$ — Base measure (function of y)
- $a(\eta)$ — log-partition (function of η). [Scalar]

↳ Reason why it's called log-partition

$p(y;\eta) = b(y) \cdot \boxed{\exp(\eta^T T(y) - a(\eta))}$.

$= \dfrac{\exp(\eta^T \cdot T(y))}{\exp(a(\eta))}$.

* **Partition function** is a technical to indicate normalization constant of probability distribution. s.t whole equation $P(y;\eta)$ integrates to one.

* For an choice of a, b, T as long as expression integrate to one you have ~~exponton~~ ~~expotent~~ ~~expoten~~ exponential family. for different probability distribution

* Algebaric massaging of PDF to bring to this form:- if yes then it is exponsial family.

~ Examples of showing PDFs belonging to exponsial family:-

① Bernoulli Distribution (Binary outcome):-

ϕ = probability of event]

$P(y;\phi) = \phi^y \cdot 1-\phi^{1-y}$ (take this to exponential family form)

$= \log \exp(\log(\phi^y \cdot 1-\phi^{1-y}))$.

$= \exp\left[y \log\left(\frac{\phi}{1-\phi}\right) + \log(1-\phi)\right]$.

$$= \exp\left(y \log \phi + \log(1-\phi) - y \log(1-\phi)\right)$$

$$= \exp\left(y \log \phi - y \log(1-\phi) + \log(1-\phi)\right)$$

$$= \exp\left(y \log \frac{\phi}{1-\phi} + \log(1-\phi)\right)$$

↳ Now comparing to the original equation

- $b(y) = 1$
- $T(y) = y$ (as expected)
- $\eta = \log\left(\frac{\phi}{1-\phi}\right) \Rightarrow \phi = \frac{1}{1+e^{-\eta}}$???
- $a(\eta) = -\log(1-\phi)$

$$\Rightarrow -\log\left(1 - \frac{1}{1+e^{-\eta}}\right)$$

$$= \log(1+e^{\eta})$$

→ This verifies bernoulli is exponential family.

⋆ Example of Gaussian (with fixed variance)

- Assume $\sigma^2 = 1$

$$P(y;\mu) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(y-\mu)^2}{2}\right)$$

$$= \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} \exp\left(\mu y - \frac{1}{2}\mu^2\right)$$

$b(y)$ $\quad \eta \; T(y) \quad a(\eta)$

$$b(y) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{y^2}{2}\right)$$

$$T(y) = y$$

$$\eta = \mu$$

$$a(\eta) = \frac{\mu^2}{2} = \frac{\eta^2}{2}$$

→ Parameter matching for Gaussian.

Example of Exponential family

Real - Gaussian
Binary - Bernoulli
Count - Poisson
R^+ — Gamma, exponential
Distn - Beta, ... Distribution -

⑥ Properties of exponential family:-

- a):- MLE wrt $\eta \Rightarrow$ concave.
  NLL wrt $\eta \Rightarrow$ convex

No Integrals:

b):- $E[y;\eta] = \frac{\partial}{\partial \eta} a(\eta)$.

c):- $Var[y;\eta] = \frac{\partial}{\partial \eta^2} a(\eta)$

⑦ Generalized Linear Model (GLM):-

→ Assumption/Design choice:-

(i):- $y|x;\theta \sim$ Exponential Family(η).

(ii):- $\eta = \theta^T x$ (Design choice)

↳ $\theta \in \mathbb{R}^n, x \in \mathbb{R}^n$

(iii):- Test time:- Output $E[y|x;\theta]$

↳ Mean of the distribution

$\Rightarrow h_\theta(x) = E[y|x;\theta]$.

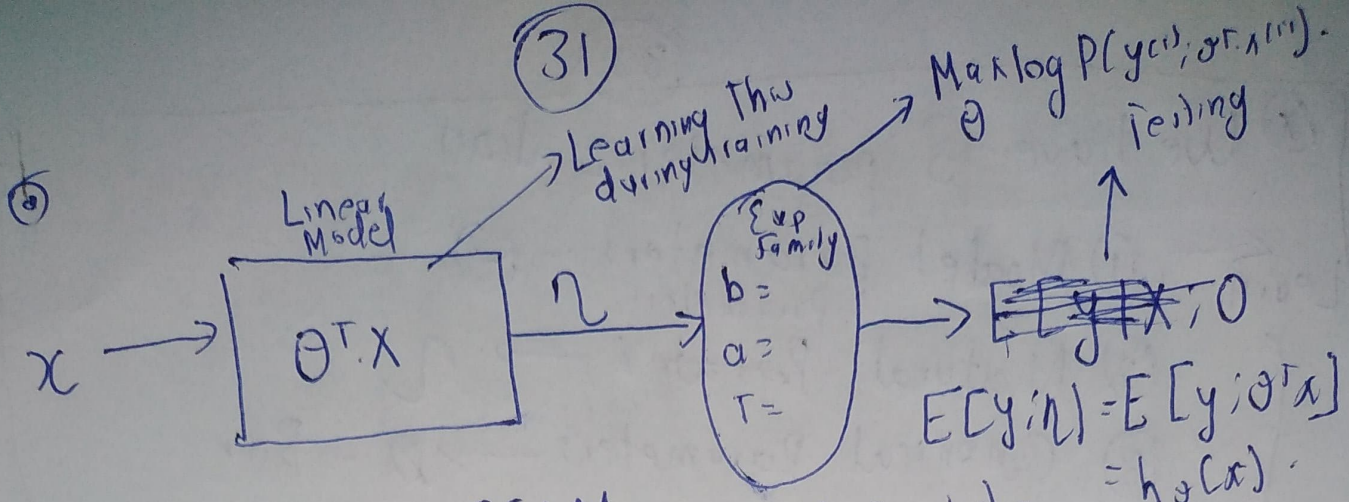


Fig (GLM and Exponential family).
(Overall Summary).

GLM Training:-

⊙ Learning update Rule is Same for all GLMs:-

$$\theta_j := \theta_j + \alpha\left(y^{(i)} - h_\theta(x^{(i)})\right) \cdot x_j^{(i)}$$

⊙ Terminology:

η – natural parameter

$\mu = E[y;\eta] = g(\eta)$ → Canonical Response function.

$\eta = g^{-1}(\mu)$ → Canonical Link function

Also note:

$$g(\eta) = \frac{\partial}{\partial \eta} a(\eta)$$

(*) We have 3 parameterization

Learn ← (1) Model parameters - θ

Parameters

(2) Natural ~~Parameter~~ → η

(3) Canonical Parameters → ϕ - Ber

ϕ σ^2 → Gaussian

λ → Possion

(⊙) → Design choice:-

$\eta = \theta^T \cdot x \rightarrow g(\eta) \rightarrow$

$\leftarrow g^{-1}(\eta)$

Fig

(*) For Logistic Regression:-

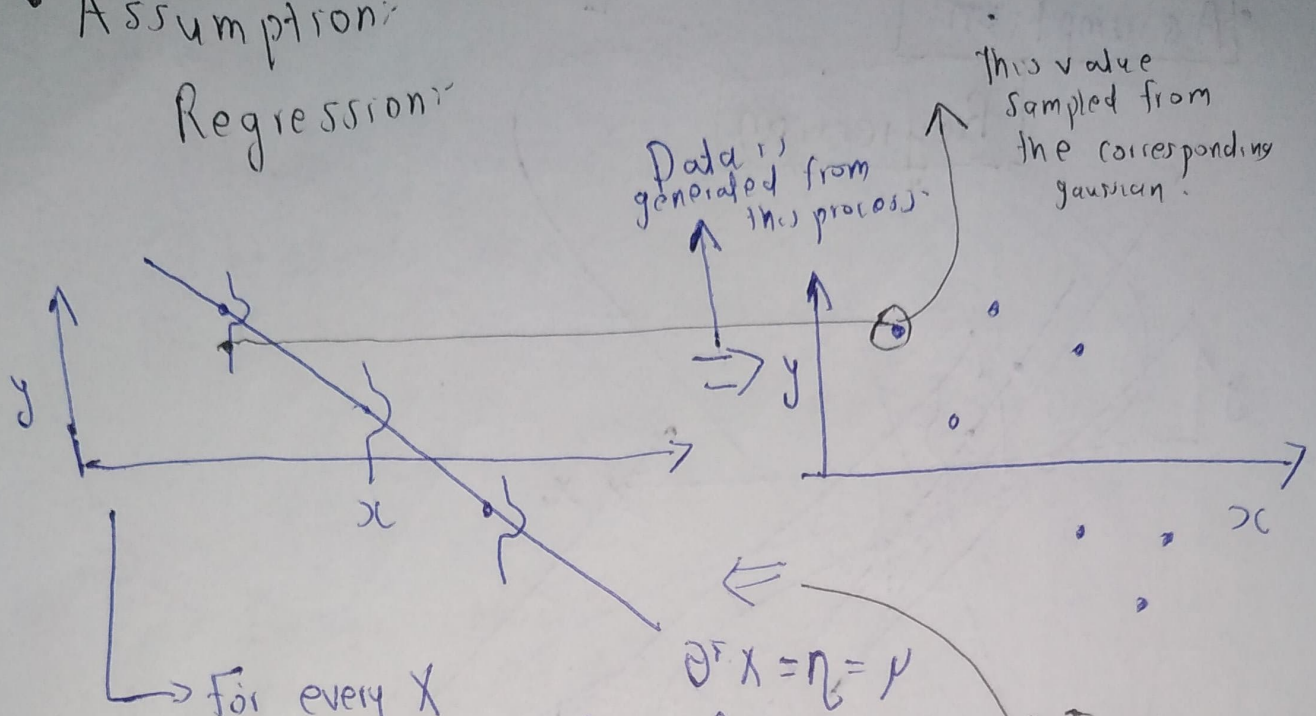
(⊙) $h_\theta(x) = E[y|x;\theta] = \frac{1}{1+e^{-\theta}}$ ϕ

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

(*) GLM is a way to model data → Depending on the task you will choose different models.

- Assumption:

Regression:

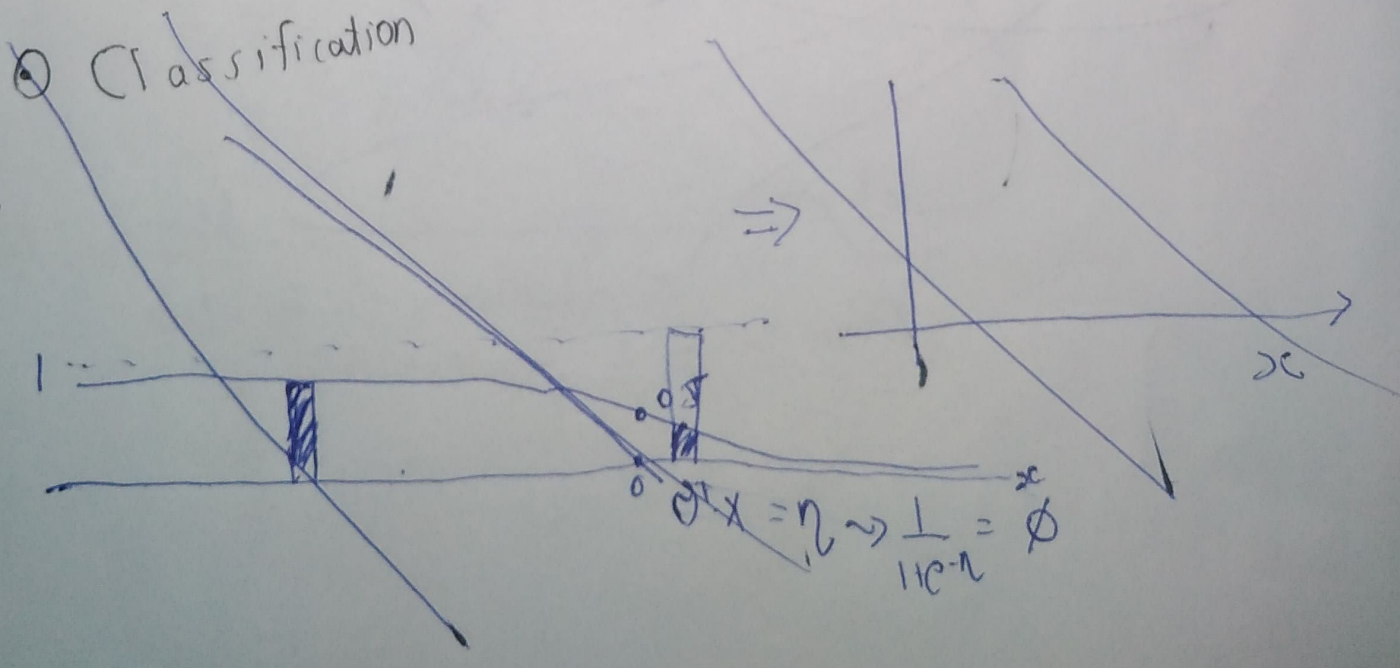


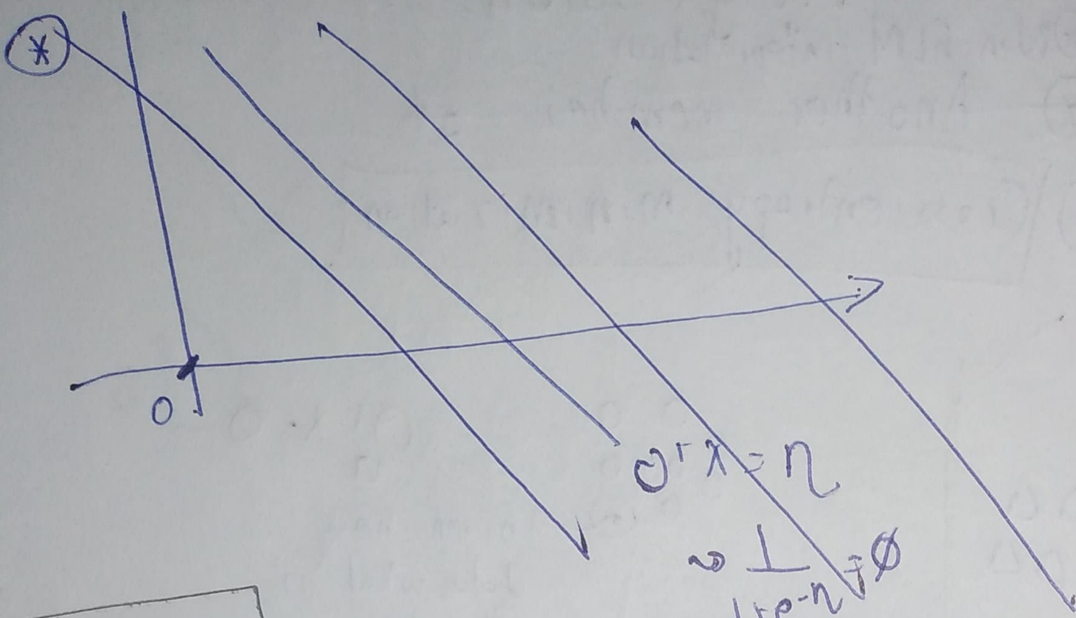
→ for every X there is a distribution (gaussian for y which is parameterized by η

→ Training is backward. by finding θ

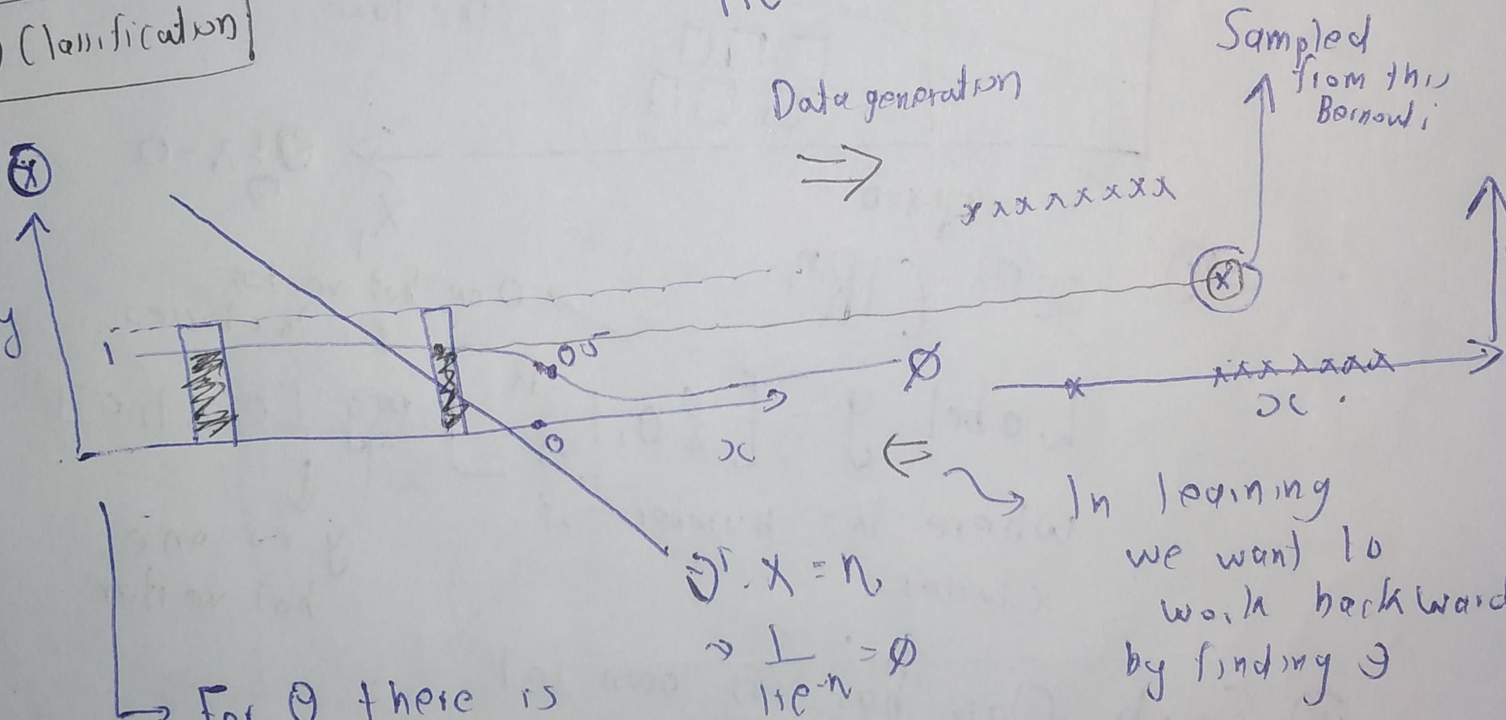
- With GLM we want to work backward from data to pr- modelling the parmaters which generated the data.

- Classification





Classification



$\theta^T . X = \eta$

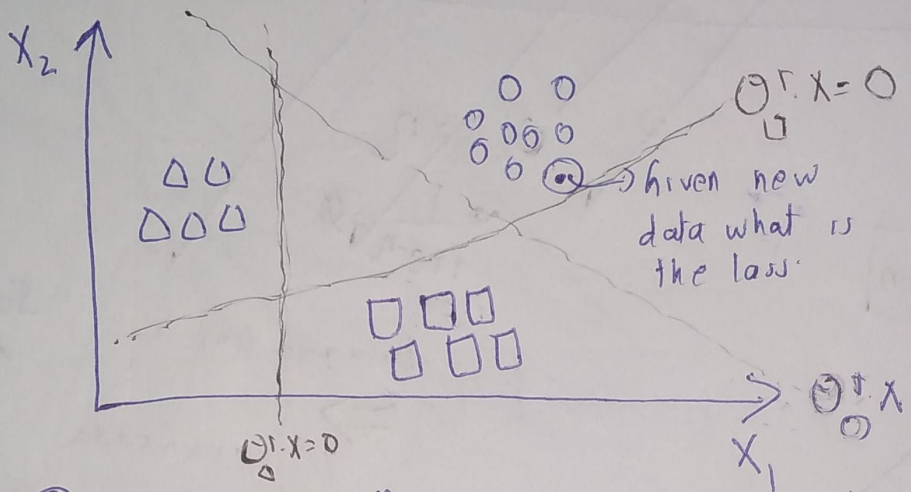
$\to \frac{1}{1+e^{-\eta}} = \phi$

$\Leftarrow$ In learning we want to work backwards by finding θ

For θ there is a ϕ which represent different Bernolui distribution at each X as shown by the bars.

Softmax Regression:

- Non-GLM interpretation
- ~~Another member of~~
- Cross-entropy minimization



- $x^{(i)} \in \mathbb{R}^n$ → one-hot vector for k classes.

  Label $y = [\{0,1\}^k]$ e.g $[0,0,1,0]$

  where k = number of classes

  ↓ y is one hot vector

- Each class has its own set of parameters.

  $\theta_{class} \in \mathbb{R}^n$ k such

  class $\in \{0, 1, \square\}$.

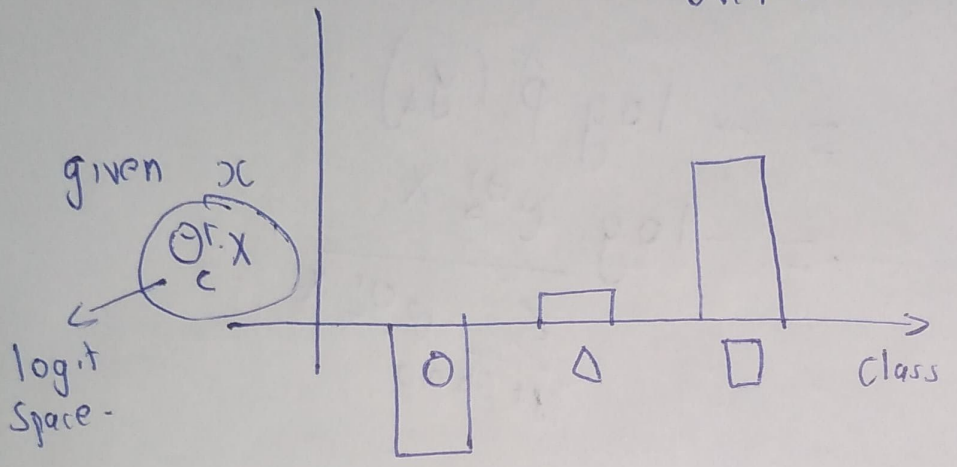
  Can Represent as ~~matrix~~ kxn matrix:

$$\begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_k \end{bmatrix}$$

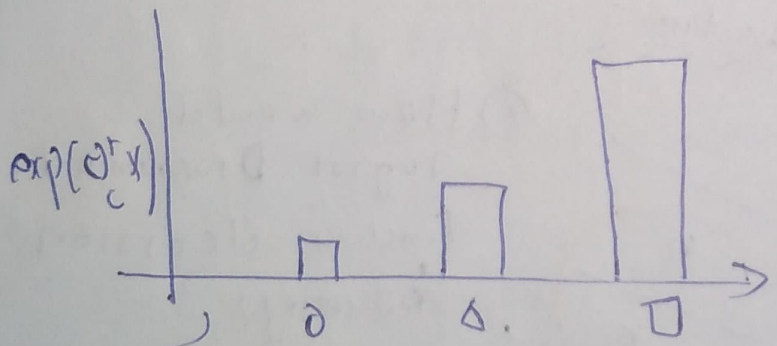
(k ↕, n ↔)

- Generalization of logistic Regression.

* Our goal is to take these parameter and find ~~class of new example~~ probability distribution over the classes.

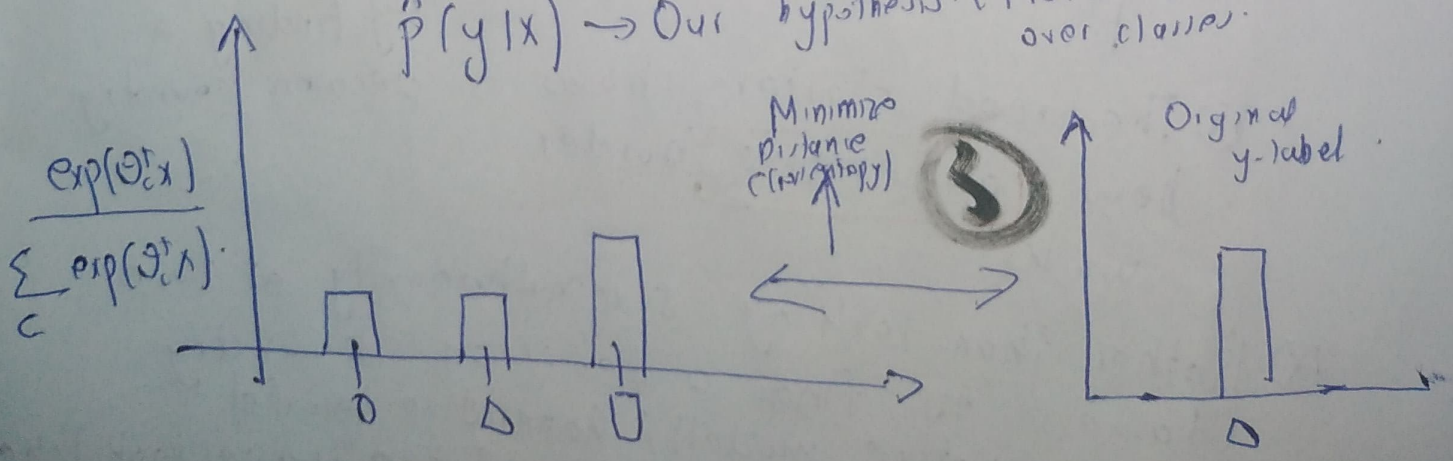


① → exp logits



② → Normalize

$\hat{P}(y|x)$ → Our hypothesis. (Probability over classes.



$$\text{Cross-entropy}(p, \hat{p}) = -\sum_{y \in C} P(y) \log \hat{P}(y)$$

$$= -\log \hat{p}(y_A)$$

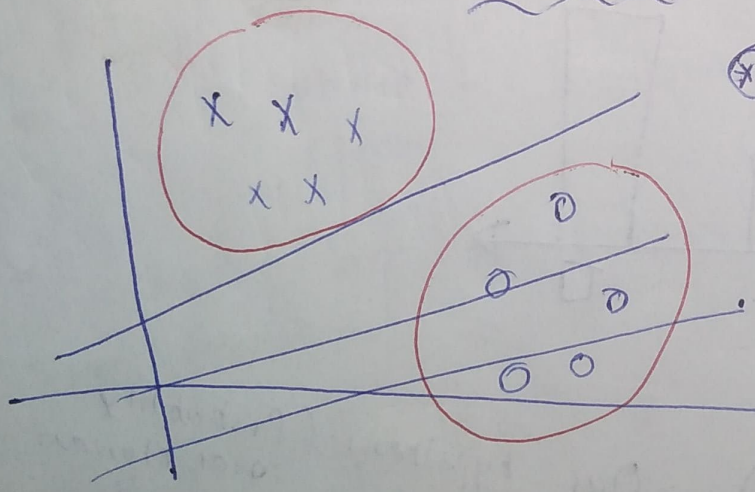
$$= -\log \frac{e^{\theta_A^T \cdot x}}{\sum_{y \in (0,0,0} e^{\theta_C^T \cdot x}}$$

Treat this a loss and you perform gradient descent w.r.t parameters

---

Lecture-5

(*)



(*) How would ~~logistic~~ Discriminative find a decision boundry?

↳ Discriminative model works by finding a decision boundry.

(*) The read cluster shows how generative border works.

(*) Rather than looking simeoullaneowly at two find a decision boundry
lol; generative model