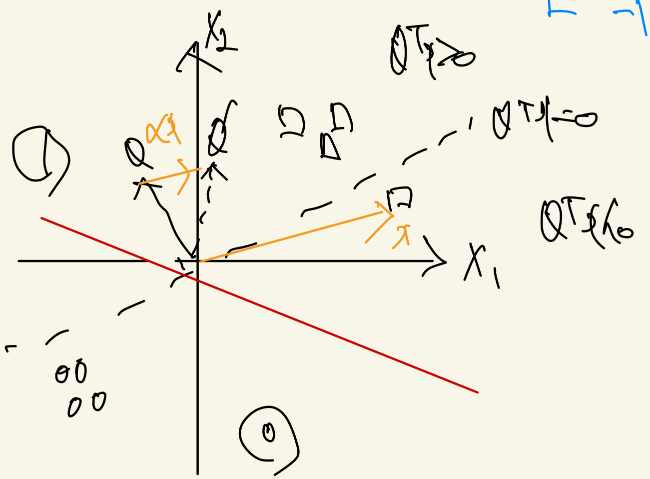
Sigmoid → perceptron

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

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

기본 공식은 $\theta_j := \theta_j + \alpha\left(\boxed{y^{(i)} - h_\theta(x^{(i)})}\right) x_j^{(i)}$ 를 따라 업데이트 해 나간다.

-1, 0, 1



문제 상황이 □가 0 잘못 예측됨. $y^{(i)} - h_\theta(x^{(i)}) = 1$ 이므로

$\theta_j := \theta_j + \alpha x_j^{(i)}$ 로 model update 해 나감.

$$\begin{cases} \theta x\text{가 커 } y=1 \\ \theta x\text{가 커 } y=0 \end{cases}$$

Exponential Family

정의: PDF 가 $\boxed{p(y;\eta) = b(y)\exp(\eta^T T(y) - a(\eta))}$ 로 표현되는 것.

$$= \frac{b(y)\exp(\eta^T T(y))}{e^{a(\eta)}}$$

- y - data
- η - natural parameter
- $T(y)$ - sufficient statistic (y와 같음)
- $b(y)$ - Base measure
- $a(\eta)$ - log-partition

match dimension

scalar.

→ 하지만 같은 η 의 변화시킬 때마다 분포가 달라진다.

이때 우리가 아는 Bernoulli, Gaussian 같은 분포들은 모두 위 수식 같이 PDF가 표현되는 Exponential Family 라는 것이 유의미함.

Ex) Bernoulli

ϕ = probability of event

$$p(y;\phi) = \phi^y (1-\phi)^{1-y}$$

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

$$= \exp\Big(\underbrace{\Big(\log\Big(\frac{\phi}{1-\phi}\Big)\Big)}_{\eta}\ \underbrace{y}_{T(y)} + \underbrace{\log(1-\phi)}_{-a(\eta)}\Big)$$

$b(y) = 1$

$y\log\phi + \log(1-\phi)$

$-y\log(1-\phi)$

$y = y(\log(\phi/(1-\phi)))$

$+\log(1-\phi)$

* Properties

① MLE wrt η ⟶ concave , ∵ A is convex

② $E[y;\eta] = \frac{d}{d\eta} a(\eta)$

③ $Var[y;\eta] = \frac{d^2}{d\eta^2} a(\eta)$

→ 미분으로 평균, 분산 구할 수 있음.

Generalized Linear Model (GLM)

→ 이전까지 우리는 y의 error term이 <u>정규분포를 따른다고 가정함</u>

* 사용하는 경우
  ① y와 x가 non-linear 한 경우
  ② y의 error가 정규분포를 따르지 X

* GLM의 가정

$$x \longrightarrow \underbrace{\boxed{\theta^T \cdot x}}_{\text{Linear Model}} \xrightarrow{\eta} \text{Exp Family } (b = a\ldots,\ a = \ldots,\ T = \ldots) \rightarrow E[y;\eta] = h_\theta(x)$$

(i) $y \mid x;\theta \sim$ Exponential Family (η)

(ii) $\eta = \theta^T \cdot x$ $\theta \in \mathbb{R}^n$, $x \in \mathbb{R}^n$

(iii) Test time 에서 output은 $E[y \mid x;\theta]$

즉 재산분포의 평균이 우리의 가설 h인 지수족에 해당하는
parameter를 찾기위해 θ를 훈련시키는 것..

→ 학습단계는 θ를 훈련시키는 것 $\max_{\theta} \log P(y^{(i)}; \theta^T x^{(i)})$

*GLM의 훈련 과정에서 이점

(i) Learning update Rule은 항상 동일함

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

(ii) Terminology (not Canonical of linked function인가)

η - natural parameter 좋아요

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

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

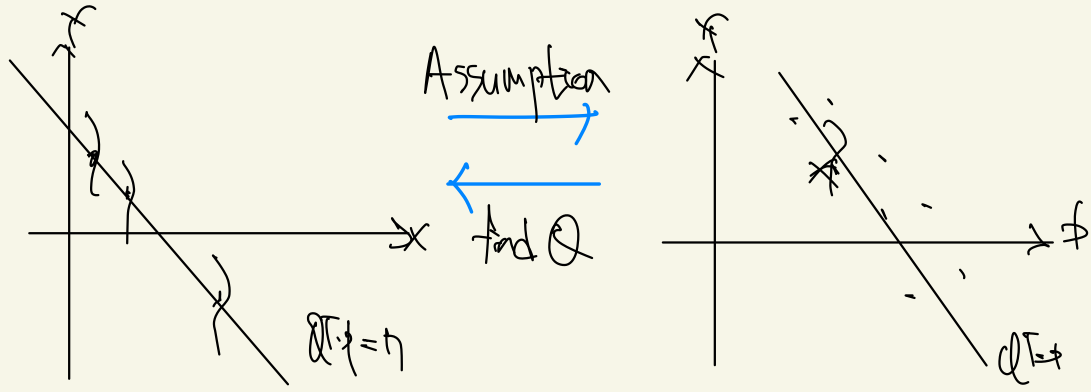
(iii) Parameters.

model param	natural param	Canonical param
θ	η	ϕ - Ber
		μ - Gauss
		λ - Poisson

$$\theta \xrightarrow{\theta^T x} \eta \underset{g^{-1}}{\overset{g}{\rightleftarrows}} \text{canonical param}$$

Assumptions

<Regression>



※ task에 따라 select 하는 Model 달라짐.

- Real → Gaussian
- Binary → Bernoulli
- Count → Poisson ...

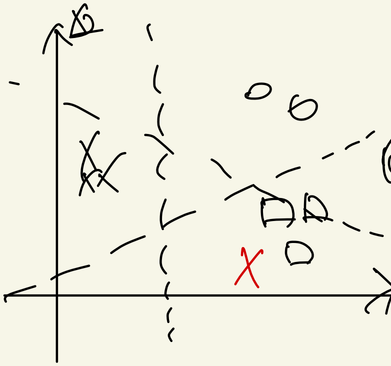
Softmax Regression

: 다분류로 GLM으로 설명가능하지만, 강의에서 보셨이 이런 쪽 방식으로 Softmax가 확률으로 Min cross-entropy 하는 방법.

$K \rightarrow$ #class

$\phi^{(i)} \in \mathbb{R}^n$

$y = [\{0,1\}^K]$ ← one-hot encoding



$\theta_{\circ}^T \phi = 0$

$\theta_{\Delta}^T \phi = 0$

각 클래스가 고유한 파라미터 가짐.

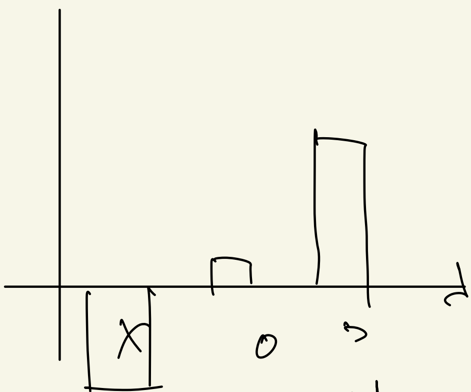
$\theta_{class} \in \mathbb{R}^n$ such

class $\in \{x, \circ, \Delta \ldots\}$

$\theta_x^T \phi = 0$

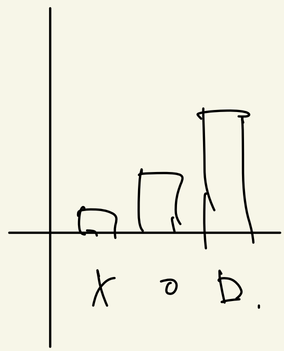
given x

$\theta_c^T \phi$ (logit)



classes.

exp →

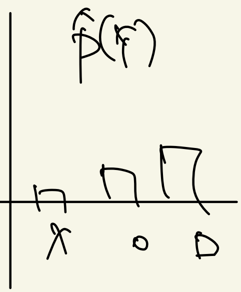


if $x = \Delta$ 라면

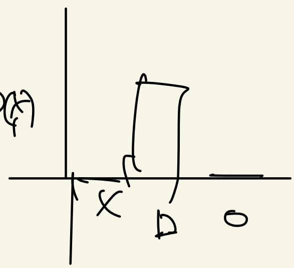
normalize →

$$\frac{e^{\theta_c^T \phi}}{\sum e^{\theta_c^T \phi}}$$

$\hat{p}(x)$



→ $p(x)$



두 분포를 같게 만드는 것이 목적

이 파라미터들에 대해

gradient descent 사용

$$\text{CrossEnt}(p, \hat{p}) = \sum_{x \in \{x, \circ, \Delta\}} p(x) \log \hat{p}(x)$$