L 4

Perceptron algorithm

Similar to sigmoid

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

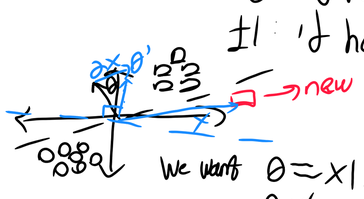
$$h(x) = g(\theta^T x)$$

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

↙

0 if $h(x)$ correct

± 1 if $h(x)$ wrong



We want $\theta \geq x \mid y=1$

$\theta \not\geq x \mid y=0$

perceptron algorithm: add data one by one

disadvantage: ~~x/o~~ never classify as no boundary

Exponential family

$p(y;\eta) = b(y)\, e^{\eta^T T(y) - a(\eta)} = b(y) \frac{e^{\eta^T T(y)}}{e^{a(\eta)}}$

y data η natural parameter } match dimension

$T(y)$ sufficient statistic

$b(y)$ - base measure

$a(\eta)$ log-partition

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

$= 1\, e^{y(\log(\frac{\phi}{1-\phi}))y + \log(1-\phi)}$

η $T(y)$ $a(\eta)$

Gaussian with $\sigma^2 = 1$

$p(y;\mu) = \frac{1}{\sqrt{2\pi}} e^{-(y-\mu)^2/2}$

similar

Why exponential family?

① MLE wrt $\eta \Rightarrow$ concave

NLL is convex

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

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

If η vector $\Rightarrow$ $Var = H(a(\eta))$

GLM

Assumptions

1 $y|x;\theta \sim$ exponential family

- Real – Gaussian
- Binary – Bernoulli
- Count – Poisson
- $\mathbb{R}^+$ – Gamma, exponential
- Distn – Beta, Dirichlet } Bayesian

2 $\eta = \theta^T x$, $\theta \in \mathbb{R}^n$, $x \in \mathbb{R}^n$

3 Test time output $E[y|x;\theta]$

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

GLM: Learning update rule same for exp family

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

$E[y;\eta] = g(\eta)$ → canonical response function

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

$\mu = E[y;\eta) = g(\eta)$

$\Rightarrow g(\eta) = \eta$

Logistic regression

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

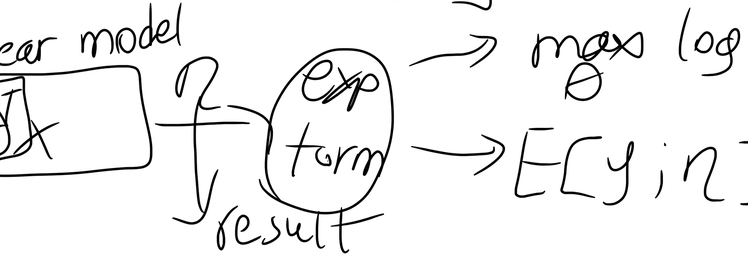
3 – Parameter

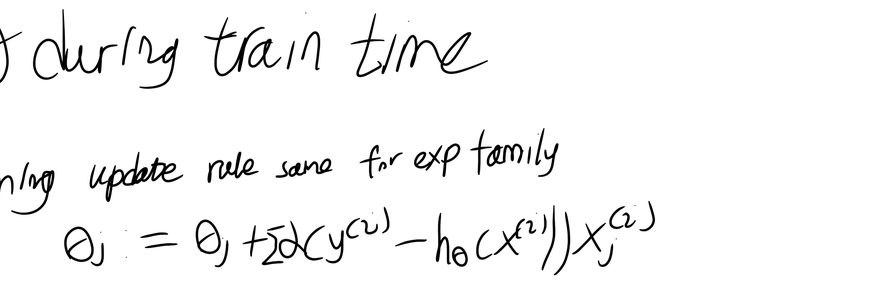
model param	natural param	Canonical param
θ	η	ϕ – Ber
	g →	$\mu\ \phi^2$ – Gauss
	← g^{-1}	λ – Poisson
	y	Link function

Softmax Regression

cross-entropy minimization $K = \#$ classes

Labels $y = [\{1\}^K]$





Cross Entropy $(P, \hat{P}) = -\sum p(y) \log \hat{p}(y)$

minimize — Gradient descend