CS269/362 Machine Learning, Spring 2014: Homework X

Zejian Zhan

Instructions Please fill in your name above.

1 Question 1:

(a) (1)

$$\begin{aligned} f &= \begin{pmatrix} n \\ x \end{pmatrix} e^{\log p^x (1-p)^{n-x}} \\ &= \begin{pmatrix} n \\ x \end{pmatrix} e^{\log p^x + \log (1-p)^{n-x}} \\ &= \begin{pmatrix} n \\ x \end{pmatrix} e^{x \log p + (n-x) \log (1-p)} \\ &= \begin{pmatrix} n \\ x \end{pmatrix} e^{x \log \frac{p}{1-p} + n \log (1-p)} \\ &= \begin{pmatrix} n \\ x \end{pmatrix} e^{x\theta - n \log (1+e^{\theta})} \end{aligned} \tag{1}$$

$$Let \theta = \log \frac{p}{1-p}, h(x) = \begin{pmatrix} n \\ x \end{pmatrix}, \eta(\theta) = \theta, T(x) = x, A(\theta) = n \log (1 + e^{\theta}) \tag{2}$$

(2)

$$\begin{aligned} f &= \frac{1}{x!} e^{\log \lambda^x - \lambda} \\ &= \frac{1}{x!} e^{x \log \lambda - \lambda} \end{aligned} \tag{3}$$

$$Let \theta = \log \lambda, h(x) = \frac{1}{x!}, \eta(\theta) = \theta, T(x) = x, A(\theta) = e^{\theta} \tag{4}$$

(3)

$$\begin{aligned} f &= e^{\log \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}} \\ &= e^{\log \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(x-\mu)^2}{2\sigma^2}}} \\ &= e^{\log \frac{1}{\sqrt{2\pi}\sigma} + \frac{-(x-\mu)^2}{2\sigma^2}} \\ &= e^{-\log\sqrt{2\pi}\sigma + \frac{2x\mu - x^2 - \mu^2}{2\sigma^2}} \\ &= e^{-\frac{1}{2}\log 2\pi\sigma^2 - \frac{x^2}{2\sigma^2} + \frac{\mu x - \frac{\mu^2}{2}}{\sigma^2}} \end{aligned} \tag{5}$$

$$Let \theta = \mu, \eta(\theta) = \mu, T(x) = x, A(\theta) = \frac{\theta^2}{2}, h(x) = e^{-\frac{x^2}{2\sigma^2} - \frac{\log 2\pi\sigma^2}{2}} \tag{6}$$

(b) $\log L(\lambda) = \log \prod_i^n \frac{1}{y_i!} e^{y_i\eta(\lambda)-\lambda} = \sum_i^n \log \frac{1}{y_i!} e^{y_i\eta(\lambda)-\lambda} = \sum_i^n \log \frac{1}{y_i!} e^{y_i\eta(\lambda)-\lambda} = \sum_i^n \log \frac{1}{y_i!} + y_i\eta(\lambda) - \lambda$

(c) We call it "linear" because of $h(x) = \frac{1}{1+e^{-w^T x}}$, which is the logistic regression function. And we can use log function to convert it to $w^T x = \log \frac{h(x)}{1-h(x)}$. The right part of the formula is log-odds-ratio and the left part is a linear function of X.

Question 2: $P(Y, X_1, X_2, ..., X_{d-1}) = \sum_{X_d} P(Y, X_1, X_2, ..., X_d) = P(Y) \sum_{X_d} \prod_{i=1}^{d-1} P(X_i|Y)P(X_d|Y) = P(Y) \sum_{X_d} \prod_{i=1}^{d-1} P(X_i|Y)P(X_d|Y) = P(Y) \prod_{i=1}^{d-1} P(X_i|Y)$

Question 3: We'll first look at the core part of NaiveBayes classifier. It takes all the features' likelihoods and get the multiplication of them, which may cause the probability to be 0 if any one of likelihood is 0 regardless of other factors. So in this case, a λ can help solve this problem. But note that a large λ will cause low variance and high bias, while a small one will lead to high variance and low bias. The first situation may be too smooth that it misses significant patterns among the data. And the second one may be fluctuated that it cannot predict good results with testing data.

Question 4: a.Assume that one operation takes constant time. So k operations one iteration means O(k). b.The advantage should be that parameters gets closer to optimal variable after one iteration. The disadvantage is that it increases the time complexity. Say, if the training dataset is so huge that the training process will be very slow. And when k gets close to n, the SGD will become BDG. c. When we try to get optimal variable of vector w of Linear Regression, for instance, StochasticGradient can help to update the w.