1 Logistic Regression

a. Definition
For binary dependent variable, with parameter w, the model states

$$P(y = +1|\mathbf{x}, \mathbf{w}) = \frac{e^{H(\mathbf{x})\mathbf{w}}}{1 + e^{H(\mathbf{x})\mathbf{w}}}$$

Then

$$P(y = -1|\mathbf{x}, \mathbf{w}) = 1 - P(y = +1|\mathbf{x}, \mathbf{w}) = \frac{1}{1 + e^{H(\mathbf{x}), \mathbf{w}}}$$

And a typically logistic regression task is to fit $\mathbf{w}$ to the data set ($\mathbf{X}$, $\mathbf{Y}$)

b. Likelihood function and log likelihood function

$$L(\mathbf{w}) = \Pi_{i=1}^{N} P(y_i|x_i, \mathbf{w}) \text{ } (y_i \text{ can be +1 or 0})$$

The solution w maximize the L(w)

$$\begin{aligned} LogL(\mathbf{w}) &= \sum_{i}^{N} (1_{y_i=1} ln(P(y_i = 1|x_i, \mathbf{w})) + 1_{y_i=0} ln(P(y_i = -1|x_i, \mathbf{w}))) \\ &= \sum_{i}^{N} (y_i ln\hat{y}_i + (1 - y_i) ln(1 - \hat{y}_i))) \end{aligned}$$

c. Gradient descent solution
This solution minimize cost function which is the negative of the log likelihood function
Loss(w) = - log Likelihood(w)
Init $w^{(1)} = 0$
While $||gradLogL(w^{(t)})|| > \epsilon$
For j = 0..D
$partial[j] = \sum_{i}^{N} H_{ij}(1_{y_i=+1} CP(y = +1|x_i, \mathbf{w}^{(t)}))$
$w_j^{(t+1)} = w_j^{(}t) + stepsize * partial[j]$
t = t+1;

d. How to choose the step size
1) Picking step size requires a log of trials and error
2) Plot learning curve(cost function vs number of step)
Find the step size that is too small
Find the step size that is too large
Then fine tune the step size in between to find the optimal.

e. Logistic regression with penalty
Use L2 norm as penalty term
Then the new loss function

$$L^{'}(\mathbf{w}) = L(\mathbf{w}) + \sum_{i=j}^{D} \lambda w_j^2$$

For gradient descent, we use

$$\nabla L^{'}(\mathbf{w}) = \nabla L(\mathbf{w}) + 2\lambda \mathbf{w}$$