Logistic Regression

Linear Regression (Review)

- In linear regression we try to predict the value of $y^{(i)}$ for the i'th example $x^{(i)}$ using a linear function

$$y = h_\theta(x) = \theta^\top x$$

Solution is based on the Pseudo-inverse of the data matrix!

Logistic regression (review)

- Two classes: ($y^{(i)} \in \{0,1\}$).

  $P(y=1|x)=h_\theta(x)=1/(1+\exp(-\theta^T x))\equiv\sigma(\theta^T x)$

  $P(y=0|x)= 1- P(y=1|x) = 1- h_\theta(x)$

where the function is the sigmoid function

$$\sigma(z) = \frac{1}{1+\exp(-z)}$$

- For a set of training examples with binary labels $\{(x^{(i)},y^{(i)})\ i=1,\dots,m\}$
- the following cost function can be used to estimate h_θ :

$$J(\theta) = -\sum_i y^{(i)}\log(h_\theta(x^{(i)})) + (1 - y^{(i)})\log(1 - h_\theta(x^{(i)}))).$$

Multi-class Classification Soft Regression

- In the softmax regression setting, we are interested in multi-class classification, and so the label y can take on K different values, rather than only two. Thus, in our training set $\{(x(1),y(1)),\ldots,(x(m),y(m))\}$, we now have that $y(i) \in \{1,2,\ldots,K\}$

$$h_\theta(x) = \begin{bmatrix} P(y=1|x;\theta) \\ P(y=2|x;\theta) \\ \vdots \\ P(y=K|x;\theta) \end{bmatrix} = \frac{1}{\sum_{j=1}^{K} \exp(\theta^{(j)\top} x)} \begin{bmatrix} \exp(\theta^{(1)\top} x) \\ \exp(\theta^{(2)\top} x) \\ \vdots \\ \exp(\theta^{(K)\top} x) \end{bmatrix}$$

We have to learn $\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(K)}$ from the training data

Cost Function

- Maximum likelihood leads to

$$J(\theta) = -\left[\sum_{i=1}^{m}\sum_{k=1}^{K} 1\left\{y^{(i)} = k\right\} \log \frac{\exp(\theta^{(k)\top}x^{(i)})}{\sum_{j=1}^{K}\exp(\theta^{(j)\top}x^{(i)})}\right]$$

where $1\{\cdot\}$ is the ”‘indicator function,”’ so that 1{a true statement}=1, and 1{a false statement}=0

e.g., $1\{y^{(i)}=5\} = 1$, if $y^{(i)}=5$

- Minimization of the cost function J(θ) requires numerical gradient descent method.