Name: Janmejay Mohanty **Course:** Applied Machine Learning **Homework Assignment**: 2
CWID: 20009315 **Course Number:** AAI 695

Ans1: Here, it is given that $X = \left(x^{(1)}, x^{(2)}, \dots, x^{(m)}\right)^T$ is the input data and $y = \left(y^{(1)}, y^{(2)}, \dots, y^{(m)}\right)$

Also, $h_w(x) = w_0 + w_1x_1 + w_2x_2 + \cdots + w_nx_n$ and $y^{(j)}$ is the measurement of $h_w(x)$ for the j-th training sample.

Now,

The cost function of the Ridge Regression is $E(w) = \sum_{i=1}^{m}\left(w^T . x^{(i)} - y^{(i)}\right)^2 + \lambda \sum_{i=1}^{m} w_i^2$

$E(w) = (Xw - y)^T(Xw - y) + \lambda w^T w$ [All the matrices are symmetric, so $X^T = X, X^2 = X^T X$]

$E(w) = (Xw)^T(Xw) - (Xw)^T y - y^T(Xw) + y^T y + \lambda w^T w$

$= X^T X w^T w - (Xw)^T y - (Xw)^T y + y^T y + \lambda w^T w$ [$A^T B = B^T A$, A and B are symmetric]

$E(w) = X^T X w^T w - 2(Xw)^T y + y^T y + \lambda w^T w$

To get the closed form equation of Ridge regression differentiate and minimize the cost function,

$$\frac{dE}{dw} = 0$$

$$\frac{dE}{dw} = 2X^T Xw - 2X^T y + 0 + 2\lambda w$$

$$0 = 2X^T Xw - 2X^T y + 2\lambda w$$

$$2X^T Xw + 2\lambda w = 2X^T y$$

$$X^T Xw + \lambda w = X^T y$$

$$(X^T X + \lambda I)w = X^T y$$

Multiplying both sides with $(X^T X + \lambda I)^{-1}$,

$$w = X^T y (X^T X + \lambda I)^{-1}$$

$$\boldsymbol{w = \left(\lambda I + X^T X\right)^{-1} X^T y}$$

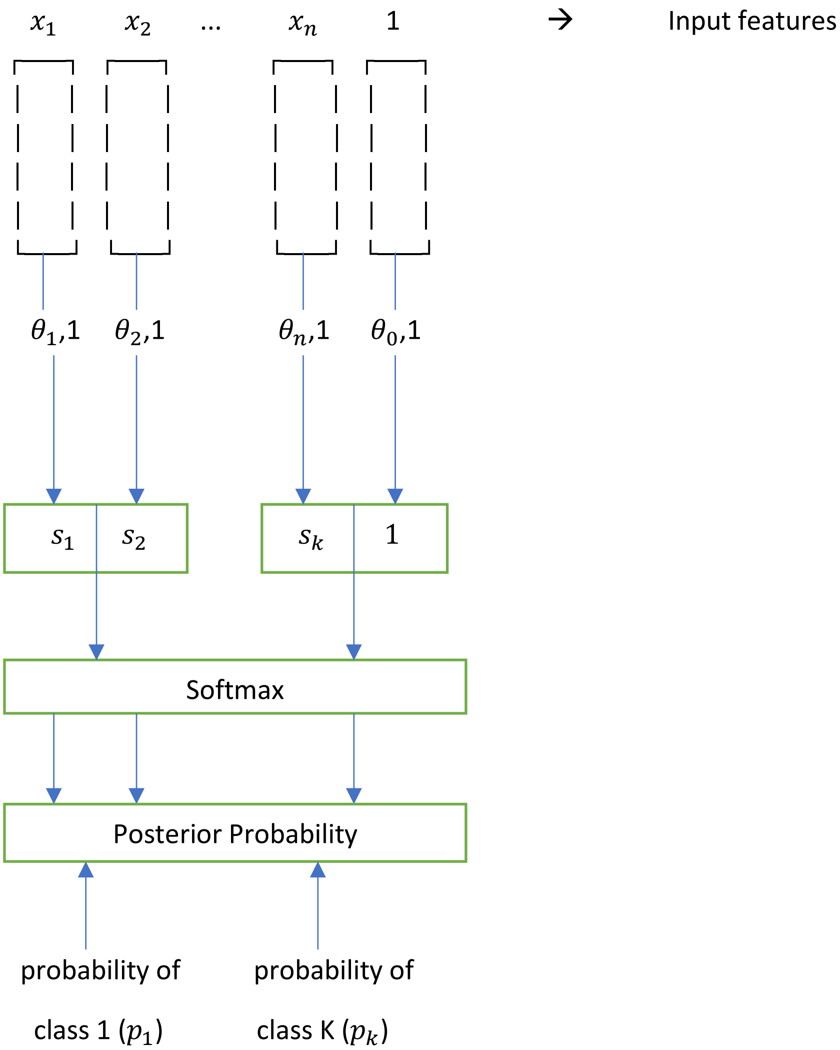
Hence Proved

Here X is matrix of input, y is the measurement of $h_w(x)$ for the j-th training sample and I is the identity matrix.

Ans2:

1. To learn this SoftMax Regression model, we need to estimate $(n + 1)$ parameter and the parameters are $\theta_0, \theta_1, \theta_2, \theta_3, \dots, \theta_n$

Diagram:



2. $J(\theta) = -\frac{1}{m}\sum_{i=k}^{m}\sum_{k=1}^{k} y_k^{(i)} \log\left(\hat{p}_k^{(i)}\right)$

$$\hat{p}(k) = \sigma\big(S(x)\big)_k = \frac{e^{(S_k(x))}}{\sum_{j=1}^{k} e^{(S_j(x))}}$$

Where $S_k(x) = \theta_k^T x$
For minimizing the cost function,

$$J(\theta) = -\frac{1}{m}\sum_{i=k}^{m}\sum_{k=1}^{k} y_k^{(i)} \log\left(\hat{p}_k^{(i)}\right)$$

$$= -\frac{1}{m}\sum_{i=k}^{m}\sum_{k=1}^{k} y_k^{(i)} \log\left(\frac{e^{(S_k(x))}}{\sum_{j=1}^{k} e^{(S_j(x))}}\right)$$

$$= -\frac{1}{m}\sum_{i=k}^{m}\sum_{k=1}^{k} y_k^{(i)} \left\{\log\left(e^{(S_k(x))}\right) - \log\left(\sum_{j=1}^{k} e^{(S_j(x))}\right)\right\}$$

$$= -\frac{1}{m}\sum_{i=k}^{m}\left\{\sum_{k=1}^{k} y_k^{(i)} \log\left(e^{(S_k(x))}\right) - \sum_{k=1}^{k} y_k^{(i)} \log\left(\sum_{j=1}^{k} e^{(S_j(x))}\right)\right\}$$

We know $y_k^{(i)} = 1$ if i^{th} instance belongs to k, otherwise, SoftMax regression gradient for cross-entropy cost function:

$$\nabla_{\theta_k} J(\theta) = \frac{1}{m}\sum_{i=1}^{m}\left(\hat{p}_k^{(i)} - y_k^{(i)}\right)x^{(i)}$$
$$= -\frac{1}{m}\sum_{i=1}^{m} x^{(i)} - \frac{1.e\hat{}(\theta_k^T x^{(i)})x^i}{\sum_{j=1}^{k} e^{\theta_j^T x^{(i)}}}$$
$$= -\frac{1}{m}\sum_{i=1}^{x}\left(1 - \hat{p}_k^{(i)}\right)x^i$$
$$= \frac{1}{m}\sum_{i=1}^{x}\left(\hat{p}_k^{(i)} - y_k^{(i)}\right)x^{(i)}$$
$$= \boldsymbol{\nabla_{\theta_k} J(\theta)}$$

Hence Proved.