logistic regression cost function

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^{(i)}\log(h_\theta(x^{(i)})) + (1-y^{(i)})\log(1-h_\theta(x^{(i)}))$$

neural network

$$J(\Theta) = -\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{k=1}^{K} y_k^{(i)}\log(h_\Theta(x^{(i)}))_k + (1-y_k^{(i)})\log(1-(h_\Theta(x^{(i)}))_k)\right]$$

logistic regression cost function regularization

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^{(i)}\log(h_\theta(x^{(i)})) + (1-y^{(i)})\log(1-h_\theta(x^{(i)})) + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$$

neural network regularization

$$J(\Theta) = -\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{k=1}^{K} y_k^{(i)}\log(h_\Theta(x^{(i)}))_k + (1-y_k^{(i)})\log(1-(h_\Theta(x^{(i)}))_k)\right] + \frac{\lambda}{2m}\sum_{l=1}^{L-1}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}(\Theta_{ji}^{(l)})$$

Feed forward and Back propagation

一些标记:

- L 表示神经网络的总层数
- S_l表示第l层神经网络unit个数，不包括偏差单元`bias unit`
- k表示第几个输出单元
- $\Theta_{i,j}^{(l)}$ 第l层到第$l+1$层的权值矩阵的第i行第j列的分量
- $Z_i^{(j)}$ 第j层第i个神经元的输入值
- $a_i^{(j)}$ 第j层第i个神经元的输出值
- $a^{(j)} = g(Z^{(j)})$

Feed forward computation $h_\theta(x^{(i)})$

```
% computation h(x)
% input layerx
a1 = [ones(m,1) X];
% hidden layer
Z2 = a1*Theta1';
a2 = sigmoid(Z2);
a2 = [ones(size(a2,1),1) a2];
% output layer
Z3 = a2*Theta2';
a3 = sigmoid(Z3);
h = a3;
```

$$J(\Theta) = -\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{k=1}^{K} y_k^{(i)} \log(h_\Theta(x^{(i)}))_k + (1-y_k^{(i)})\log(1-(h_\Theta(x^{(i)}))_k)\right]$$

```
%case 1
J = 0;
Y = zeros(m,num_labels);
for i = 1 : m
    Y(i,y(i)) = 1;
end
J = -1/m * (Y * log(h)' + (1 - Y) * log(1 - h)');
J = trace(J);


%case 2
J = 0;
Y = zeros(m,num_labels);
for i = 1 : m
    Y(i,y(i)) = 1;
end

for i = 1 : m
    J = J + -1*m *(Y(i,:) * log(h(i,:))' + (1 - Y(i,:)* log(1 - h(i,:))');
end
```

Chain Rule

$y = g(x) \quad z = h(y)$

$\Delta x \to \Delta y \to \Delta z \quad \frac{dz}{dx} = \frac{dz}{dy}\frac{dy}{dx}$

$x = g(s) \quad y = h(s) \quad z = k(x, y)$

$\frac{dz}{ds} = \frac{\partial z}{\partial x}\frac{dx}{ds} + \frac{\partial z}{\partial y}\frac{dy}{ds}$

back propagation

我们知道代价函数cost function后，下一步就是按照梯度下降法来计算θ求解cost function的最优解。使用梯度下降法首先要求出梯度，即偏导项$\frac{\partial}{\partial\Theta_{ij}^{(l)}}J(\Theta)$，计算偏导项的过程我们称为back propagation。

根据上面的feed forward computation 我们已经计算得到了 $a^{(1)}$ ，$a^{(2)}$，$a^{(3)}$ ，$Z^{(2)}$，$Z^{(3)}$。

hidden layer to output layer

$$h_\Theta(x) = a^{(L)} = g(z^{(L)})$$

$$z^{(l)} = \Theta^{(l-1)}a^{(l-1)}$$

$$\frac{\partial}{\partial \Theta_{i,j}^{(L-1)}} J(\Theta) = \frac{\partial J(\Theta)}{\partial h_\theta(x)_i} \frac{\partial h_\theta(x)_i}{\partial z_i^{(L)}} \frac{\partial z_i^{(L)}}{\partial \Theta_{i,j}^{(L-1)}} = \frac{\partial J(\Theta)}{\partial a_i^{(L)}} \frac{\partial a_i^{(L)}}{\partial z_i^{(L)}} \frac{\partial z_i^{(L)}}{\partial \Theta_{i,j}^{(L-1)}}$$

$$cost(\Theta) = -y^{(i)} \log(h_\Theta(x^{(i)})) - (1 - y^{(i)}) \log(1 - h_\Theta(x^{(i)}))$$

$$\frac{\partial J(\Theta)}{\partial a_i^{(L)}} = \frac{a_i^{(L)} - y_i}{(1 - a_i^{(L)}) a_i^{(L)}}$$

由下式得

$$\frac{\partial g(z)}{\partial z} = -\left(\frac{1}{1+e^{-z}}\right)^2 \frac{\partial}{\partial z}(1+e^{-z})$$

$$= -\left(\frac{1}{1+e^{-z}}\right)^2 e^{-z} \, (-1)$$

$$= \left(\frac{1}{1+e^{-z}}\right) \left(\frac{1}{1+e^{-z}}\right) (e^{-z})$$

$$= \left(\frac{1}{1+e^{-z}}\right) \left(\frac{e^{-z}}{1+e^{-z}}\right)$$

$$= \left(\frac{1}{1+e^{-z}}\right) \left(\frac{1+e^{-z}}{1+e^{-z}} - \frac{1}{1+e^{-z}}\right)$$

$$= g(z) \, (1 - g(z))$$

$$\frac{\partial a_i^{(L)}}{\partial z_i^{(L)}} = \frac{\partial g(z_i^{(L)})}{\partial z_i^{(L)}} = g(z_i^{(L)})(1 - g(z_i^{(L)})) = a_i^{(L)}(1 - a_i^{(L)})$$

$$\frac{\partial z_i^{(L)}}{\partial \Theta_{i,j}^{(L-1)}} = a_j^{(L-1)}$$

综上

$$\frac{\partial}{\partial\Theta_{i,j}^{(L-1)}}J(\Theta)=\frac{\partial J(\Theta)}{\partial a_i^{(L)}}\frac{\partial a_i^{(L)}}{\partial z_i^{(L)}}\frac{\partial z_i^{(L)}}{\partial\Theta_{i,j}^{(L-1)}}$$

$$=\frac{a_i^{(L)}-y_i}{(1-a_i^{(L)})a_i^{(L)}}a_i^{(L)}(1-a_i^{(L)})a_j^{(L-1)}$$

$$=(a_i^{(L)}-y_i)a_j^{(L-1)}$$

hidden layer / input layer to hidden layer

因为$a^{(1)}=x$，所以可以将 input layer 与 hidden layer同样对待

$$\frac{\partial}{\partial\Theta_{i,j}^{(l-1)}}J(\Theta)=\frac{\partial J(\Theta)}{\partial a_i^{(l)}}\frac{\partial a_i^{(l)}}{\partial z_i^{(l)}}\frac{\partial z_i^{(l)}}{\partial\Theta_{i,j}^{(l-1)}}\ (l=2,3,\ldots,L-1)$$

$$\frac{\partial a_i^{(l)}}{\partial z_i^{(l)}}=\frac{\partial g(z_i^{(l)})}{\partial z_i^{(l)}}=g(z_i^{(l)})(1-g(z_i^{(l)}))=a_i^{(l)}(1-a_i^{(l)})$$

$$\frac{\partial z_i^{(l)}}{\partial\Theta_{i,j}^{(l-1)}}=a_j^{(l-1)}$$

第一部分的偏导比较麻烦，要使用chain rule。

$$\frac{\partial J(\Theta)}{\partial a_i^{(l)}}=\sum_{k=1}^{s_{l+1}}\left[\frac{\partial J(\Theta)}{\partial a_k^{(l+1)}}\frac{\partial a_k^{(l+1)}}{\partial z_k^{(l+1)}}\frac{\partial z_k^{(l+1)}}{\partial a_i^{(l)}}\right]$$

$$\frac{\partial a_k^{(l+1)}}{\partial z_k^{(l+1)}}=a_k^{(l+1)}(1-a_k^{(l+1)})$$

$$\frac{\partial z_k^{(l+1)}}{\partial a_i^{(l)}}=\Theta_{k,i}^{(l)}$$

求得递推式为：

$$
\begin{aligned}
\frac{\partial J(\Theta)}{\partial a_i^{(l)}} &= \sum_{k=1}^{s_{l+1}} \left[\frac{\partial J(\Theta)}{\partial a_k^{(l+1)}} \frac{\partial a_k^{(l+1)}}{\partial z_k^{(l+1)}} \frac{\partial z_k^{(l+1)}}{\partial a_i^{(l)}} \right] \\
&= \sum_{k=1}^{s_{l+1}} \left[\frac{\partial J(\Theta)}{\partial a_k^{(l+1)}} \frac{\partial a_k^{(l+1)}}{\partial z_k^{(l+1)}} \Theta_{k,i}^{(l)} \right] \\
&= \sum_{k=1}^{s_{l+1}} \left[\frac{\partial J(\Theta)}{\partial a_k^{(l+1)}} a_k^{(l+1)} (1 - a_k^{(l+1)}) \Theta_{k,i}^{(l)} \right]
\end{aligned}
$$

定义第l层第i个节点的误差为：

$$
\begin{aligned}
\delta_i^{(l)} &= \frac{\partial}{\partial z_i^{(l)}} J(\Theta) \\
&= \frac{\partial J(\Theta)}{\partial a_i^{(l)}} \frac{\partial a_i^{(l)}}{\partial z_i^{(l)}} \\
&= \frac{\partial J(\Theta)}{\partial a_i^{(l)}} a_i^{(l)} (1 - a_i^{(l)}) \\
&= \sum_{k=1}^{s_{l+1}} \left[\frac{\partial J(\Theta)}{\partial a_k^{(l+1)}} \frac{\partial a_k^{(l+1)}}{\partial z_k^{(l+1)}} \Theta_{k,i}^{(l)} \right] a_i^{(l)} (1 - a_i^{(l)}) \\
&= \sum_{k=1}^{s_{l+1}} \left[\delta_k^{(l+1)} \Theta_{k,i}^{(l)} \right] a_i^{(l)} (1 - a_i^{(l)})
\end{aligned}
$$

$$
\begin{aligned}
\delta_i^{(L)} &= \frac{\partial J(\Theta)}{\partial z_i^{(L)}} \\
&= \frac{\partial J(\Theta)}{\partial a_i^{(L)}} \frac{\partial a_i^{(L)}}{\partial z_i^{(L)}} \\
&= \frac{a_i^{(L)} - y_i}{(1 - a_i^{(L)}) a_i^{(L)}} a_i^{(L)} (1 - a_i^{(L)}) \\
&= a_i^{(L)} - y_i
\end{aligned}
$$

最终代价函数的偏导数为

$$\frac{\partial}{\partial \Theta_{i,j}^{(l-1)}} J(\Theta) = \frac{\partial J(\Theta)}{\partial a_i^{(l)}} \frac{\partial a_i^{(l)}}{\partial z_i^{(l)}} \frac{\partial z_i^{(l)}}{\partial \Theta_{i,j}^{(l-1)}}$$

$$= \frac{\partial J(\Theta)}{\partial z_i^{(l)}} \frac{\partial z_i^{(l)}}{\partial \Theta_{i,j}^{(l-1)}}$$

$$= \delta_i^{(l)} \frac{\partial z_i^{(l)}}{\partial \Theta_{i,j}^{(l-1)}}$$

$$= \delta_i^{(l)} a_j^{(l-1)}$$

总结

- 输出层的误差 $\delta_i^{(L)}$

$$\delta_i^{(L)} = a_i^{(L)} - y_i$$

- 隐层误差 $\delta_i^{(l)}$

$$\delta_i^{(l)} = \sum_{k=1}^{s_{l+1}} \left[\delta_k^{(l+1)} \Theta_{k,i}^{(l)} \right] a_i^{(l)} (1 - a_i^{(l)})$$

- 代价函数偏导项 $\frac{\partial}{\partial \Theta_{i,j}^{(l-1)}} J(\Theta)$

$$\frac{\partial}{\partial \Theta_{i,j}^{(l-1)}} J(\Theta) = \delta_i^{(l)} a_j^{(l-1)}$$

  即

$$\frac{\partial}{\partial \Theta_{i,j}^{(l)}} J(\Theta) = \delta_i^{(l+1)} a_j^{(l)}$$

让我们重新整下back propagation的过程。

首先，我们定义每层的误差

$$\delta^{(l)} = \frac{\partial}{\partial z^{(l)}} J(\Theta)$$

$\delta_j^{(l)}$ 表示第l层第j个节点的误差。为了求出偏导项$\frac{\partial}{\partial \Theta_{ij}^{(l)}} J(\Theta)$，我们首先要求出每一层的$\delta$（不包括第一层，第一层是输入层，不存在误差），对于输出层第四层

$$\delta_i^{(4)} = \frac{\partial}{\partial z_i^{(4)}} J(\Theta)$$

$$= \frac{\partial J(\Theta)}{\partial a_i^{(4)}} \frac{\partial a_i^{(4)}}{\partial z_i^{(4)}}$$

$$= -\frac{\partial}{\partial a_i^{(4)}} \sum_{k=1}^{K} \left[y_k log a_k^{(4)} + (1-y_k) log(1-a_k^{(4)}) \right] g'(z_i^{(4)})$$

$$= -\frac{\partial}{\partial a_i^{(4)}} \left[y_i log a_i^{(4)} + (1-y_i) log(1-a_i^{(4)}) \right] g(z_i^{(4)})(1-g(z_i^{(4)}))$$

$$= \left(\frac{1-y_i}{1-a_i^{(4)}} - \frac{y_i}{a_i^{(4)}} \right) a_i^{(4)}(1-a_i^{(4)})$$

$$= (1-y_i)a_i^{(4)} - y_i(1-a_i^{(4)})$$

$$= a_i^{(4)} - y_i$$

$$\delta_i^{(l)} = \frac{\partial}{\partial z_i^{(l)}} J(\Theta)$$

$$= \sum_{k=1}^{S_{l+1}} \frac{\partial J(\Theta)}{\partial z_k^{(l+1)}} \frac{\partial z_k^{(l+1)}}{\partial a_i^{(l)}} \frac{\partial a_i^{(l)}}{\partial z_i^{(l)}}$$

$$= \sum_{k=1}^{S_{l+1}} \delta_k^{(l+1)} \Theta_{ki}^{(l)} g'(z_i^{(l)})$$

$$= g'(z_i^{(l)}) \sum_{k=1}^{S_{l+1}} \delta_k^{(l+1)} \Theta_{ki}^{(l)}$$

写成向量的形式：

$$\delta^{(l)} = (\Theta^{(l)})^T \delta^{(l+1)} .* g'(z^{(l)})$$

求出所有的δ后，我们可以得到

$$\frac{\partial}{\partial \Theta_{i,j}^{(l)}} J(\Theta) = \delta_i^{(l+1)} a_j^{(l)}$$

```
delta_3 = h - Y;
delta_2 = delta_3 * Theta2 .* a2 .*(1 - a2);
delta_2 = delta_2(:,2:end);

Theta1_grad = delta_2' * a1 / m;
Theta2_grad = delta_3' * a2 / m
```