1 线性回归(Linear Regression)

1.1 当训练集X只有1项时

$$X = \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}_{(n+1)*1} \tag{1}$$

$$\theta = \begin{pmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{pmatrix}_{(n+1)*1} \tag{2}$$

$$y = y \tag{3}$$

$$\begin{aligned} h_\theta(x) &= \theta^T X = X^T \theta \\ &= \begin{pmatrix} 1 & x_1 & x_2 & \dots & x_n \end{pmatrix} \begin{pmatrix} \theta_0 \\ \theta_1 \\ \dots \\ \theta_n \end{pmatrix} \\ &= \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n \end{aligned} \tag{4}$$

1.2 当训练集X有m项时

$$
\begin{aligned}
X &= \begin{pmatrix} x^{(1)} \\ x^{(2)} \\ x^{(3)} \\ \vdots \\ x^{(m)} \end{pmatrix} \\
&= \begin{pmatrix} x_0^{(1)} & x_1^{(1)} & x_2^{(1)} & x_3^{(1)} & \dots & x_n^{(1)} \\ x_0^{(2)} & x_1^{(2)} & x_2^{(2)} & x_3^{(2)} & \dots & x_n^{(2)} \\ x_0^{(3)} & x_1^{(3)} & x_2^{(3)} & x_3^{(3)} & \dots & x_n^{(3)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ x_0^{(m)} & x_1^{(m)} & x_2^{(m)} & x_3^{(m)} & \dots & x_n^{(m)} \end{pmatrix} \\
&= \begin{pmatrix} 1 & x_1^{(1)} & x_2^{(1)} & x_3^{(1)} & \dots & x_n^{(1)} \\ 1 & x_1^{(2)} & x_2^{(2)} & x_3^{(2)} & \dots & x_n^{(2)} \\ 1 & x_1^{(3)} & x_2^{(3)} & x_3^{(3)} & \dots & x_n^{(3)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_1^{(m)} & x_2^{(m)} & x_3^{(m)} & \dots & x_n^{(m)} \end{pmatrix}_{m*(n+1)}
\end{aligned} \tag{5}
$$

$$
\theta = \begin{pmatrix} \theta^{(0)} \\ \theta^{(1)} \\ \theta^{(2)} \\ \theta^{(3)} \\ \vdots \\ \theta^{(n)} \end{pmatrix}_{(n+1)*1} \tag{6}
$$

$$
y = \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ y^{(3)} \\ \vdots \\ y^{(m)} \end{pmatrix}_{m*1} \tag{7}
$$

Cost Function

1. 数值形式:

$$
J(\theta) = \frac{1}{2m} \left[h_\theta(x^{(i)}) - y^{(i)} \right]^2 \tag{8}
$$

2. 矩阵形式:

$$
J(\theta) = \frac{1}{2m} \left[h_\theta(x) - y \right]^T \left[h_\theta(x) - y \right] \tag{9}
$$

梯度下降

1. 数值形式

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m} \left[h_\theta(x^{(i)}) - y^{(i)} \right] x_j^{(i)} \tag{10}$$

迭代方式：

$$\theta_j := \theta_j - \alpha \frac{1}{m} \left[h_\theta(x^{(i)}) - y^{(i)} \right] x_j^{(i)} \tag{11}$$

2. 矩阵形式

$$\nabla J(\theta) = \frac{1}{2m} X^T \left[h_\theta(x) - y \right] \tag{12}$$

迭代方式：

$$\theta := \theta - \alpha \frac{1}{m} X^T \left[h_\theta(x) - y \right] \tag{13}$$

1.3 Feature Normalization

$$x_i = \frac{x_i - \mu}{\sigma} \tag{14}$$

或

$$x_i = \frac{x_i - \mu}{max - min} \tag{15}$$

1.4 公式法求解（Normal Equation）

$$\theta = (X^T X)^{-1} X^T y \tag{16}$$

2 逻辑回归(Logistic Regression)

2.1 当只有2个类别时，使用1个分类器

预测函数:

1. 数值形式

$$h_\theta(x) = \frac{1}{1 + e^{\theta^T x}} \tag{17}$$

2. 矩阵形式

$$h_\theta(X) = \frac{1}{1 + e^{X\theta^T}} \tag{18}$$

Cost Function

1. 数值形式

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m}\left[-y^{(i)} logh_\theta(x^{(i)}) - (1 - y^{(i)}) log(1 - h_\theta(x^{(i)}))\right] \tag{19}$$

2. 矩阵形式

$$J(\theta) = \frac{1}{m}\left[-y^T \log h_\theta(x) - (1 - y^T)\log\left(1 - h_\theta(x)\right]\right. \tag{20}$$

梯度下降

1. 数值形式

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}\left[h_\theta(x^{(i)}) - y^{(i)}\right] x_j^{(i)} \tag{21}$$

迭代方式:

$$\theta_j := \theta_j - \alpha\frac{1}{m}\sum_{i=1}^{m}\left[h_\theta(x^{(i)}) - y^{(i)}\right] x_j^{(i)} \tag{22}$$

2. 梯矩阵形式

$$\nabla J(\theta) = \frac{1}{m}X^T\left[h_\theta(x) - y\right] \tag{23}$$

迭代方式:

$$\theta := \theta - \alpha\frac{1}{m}X^T\left[h_\theta(x) - y\right] \tag{24}$$

2.2 当只有k个类别时，使用k个分类器

2.3 避免过拟合

1. 线性回归

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}[h_\theta(x^{(i)}) - y^{(i)}]^2 + \lambda\frac{1}{2m}\sum_{j=1}^{n}\theta_j^2 \tag{25}$$

$$\theta_j = \theta_j(1 - \alpha\frac{\lambda}{m}) - \frac{\alpha}{m}\sum_{i=1}^{n}\left[h_\theta(x) - y\right]x_j^{(i)} \tag{26}$$

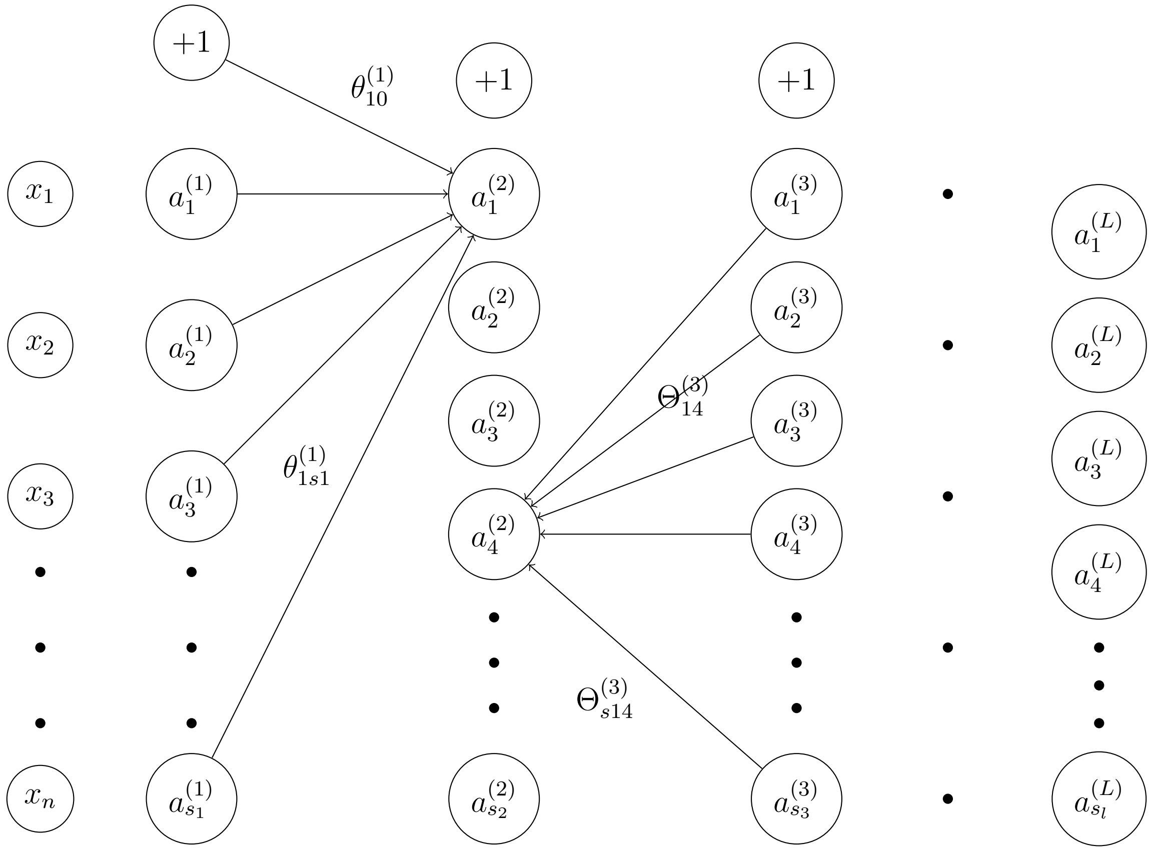
2. 逻辑回归

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m}\left[-y^{(i)}logh_\theta(x^{(i)}) - (1 - y^{(i)})log(1 - h_\theta(x^{(i)}))\right] + \lambda\frac{1}{2m}\sum_{j=1}^{n}\theta_j^2 \tag{27}$$

$$\theta_j = \theta_j - \alpha\left[\frac{1}{m}\sum_{i=1}^{m}[h_\theta(x^{(i)}) - y^{(i)}]x_j^{(i)} + \frac{\lambda}{m}\theta_j\right] \tag{28}$$

3 神经网络图例

3.1 神经网络示意图



3.2 神经网络

3.3 前向算法

3.3.1 各矩阵形状

1. X

$$X = \begin{pmatrix} x^{(1)} \\ x^{(2)} \\ x^{(3)} \\ \vdots \\ x^{(m)} \end{pmatrix} = \begin{pmatrix} x_0^{(1)} & x_1^{(1)} & x_2^{(1)} & x_3^{(1)} & \dots & x_n^{(1)} \\ x_0^{(2)} & x_1^{(2)} & x_2^{(2)} & x_3^{(2)} & \dots & x_n^{(2)} \\ x_0^{(3)} & x_1^{(3)} & x_2^{(3)} & x_3^{(3)} & \dots & x_n^{(3)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ x_0^{(m)} & x_1^{(m)} & x_2^{(m)} & x_3^{(m)} & \dots & x_n^{(m)} \end{pmatrix} = \begin{pmatrix} 1 & x_1^{(1)} & x_2^{(1)} & x_3^{(1)} & \dots & x_n^{(1)} \\ 1 & x_1^{(2)} & x_2^{(2)} & x_3^{(2)} & \dots & x_n^{(2)} \\ 1 & x_1^{(3)} & x_2^{(3)} & x_3^{(3)} & \dots & x_n^{(3)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_1^{(m)} & x_2^{(m)} & x_3^{(m)} & \dots & x_n^{(m)} \end{pmatrix}_{m*(n+1)} \tag{29}$$

2. y 最初始的y

$$y = \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ y^{(3)} \\ \vdots \\ y^{(m)} \end{pmatrix}_{m*1} \tag{30}$$

为进行矩阵运算，要将其转化为如下形式:
PS: 表示为y的值所在的索引位置值为1，其他位置均为0

$$y = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 & 1 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \dots & 0 & 0 \end{pmatrix}_{m*s_L} \tag{31}$$

上式$m * s_L$中的s_L表示共有s_L个分类器
s_L表示的是输出层的unit数

3. $\Theta^{(j)}$

   $\Theta^{(j)}$ 的形状为: $s_{j+1} * (s_j + 1)$

$$\Theta^{(j)} = \begin{pmatrix} \theta^{(j)}_{10} & \theta^{(j)}_{11} & \theta^{(j)}_{12} & \dots & \theta^{(j)}_{1,s_j} \\ \theta^{(j)}_{20} & \theta^{(j)}_{21} & \theta^{(j)}_{22} & \dots & \theta^{(j)}_{2,s_j} \\ \theta^{(j)}_{30} & \theta^{(j)}_{31} & \theta^{(j)}_{32} & \dots & \theta^{(j)}_{3,s_j} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \theta^{(j)}_{s_{j+1},0} & \theta^{(j)}_{s_{j+1},1} & \theta^{(j)}_{s_{j+1},2} & \dots & \theta^{(j)}_{s_{j+1},s_j} \end{pmatrix}_{(s_{j+1},s_j+1)} \tag{32}$$

3.3.2 数值计算方法

1. 单个unit的特殊情况

$$a^{(2)}_1 = g(z^{(2)}_1) = g(a^{(1)}_0\theta^{(1)}_{10} + a^{(1)}_1\theta^{(1)}_{11} + a^{(1)}_2\theta^{(1)}_{12} + \dots + a^{(1)}_{s1}\theta^{(1)}_{1s1}) \tag{33}$$

2. 单个unit的一般情况

$$\begin{aligned} a^{(j)}_i &= g(z^{(j)}_i) \\ &= g(a^{(j-1)}_0\theta^{(j-1)}_{(j-1)0} + a^{(j-1)}_1\theta^{(j-1)}_{(j-1)1} + a^{(j-1)}_2\theta^{(j-1)}_{(j-1)2} + \dots + a^{(j-1)}_{s1}\theta^{(j-1)}_{(j-1)s1}) \\ &= g(\sum_{k=0}^{s_l} a^{(j-1)}_k\theta^{(j-1)}_{(j-1)k}) \end{aligned} \tag{34}$$

3. 单Layer的普通情况
   1. i是否从0开始待确定
   2. 此公式的正确性待确定，似乎没见过计算整Layer的情况

$$\begin{aligned} a^{(j)} &= \sum_{i=0}^{s(j)} a^{(j-1)}_i \\ &= \sum_{i=0}^{s(j)} g(\sum_{k=0}^{s_j} a^{(j-1)}_k\theta^{(j-1)}_{(j-1)k})) \end{aligned} \tag{35}$$

3.3.3 矩阵计算方法

1. 单个unit的特殊情况

$$a_1^{(2)} = g(\Theta^{(1)} a^{(1)}) \tag{36}$$

2. 单个unit的一般情况

$$a_i^{(j)} = g(z_i^{(j)}) = g(\Theta^{(j-1)} a^{(j-1)}) \tag{37}$$