9181040G0818 黄海浪 机器学习第 5 次作业

softmax 回归

① 归一化（快速收敛（保证能够收敛））:

$$x = \frac{x - min(x)}{max(x) - min(x)}$$

② 假设：

$$p(y = j|x^{(i)}; \theta) = h_\theta^j(x^{(i)}) = \frac{e^{\theta_j^T x^{(i)}}}{\sum_{k=1}^{C} e^{\theta_k^T x^{(i)}}}$$

此题中$\theta_k = [\theta_0; \theta_1; \theta_2]$ 、 $x^{(i)} = [x_0; x_1; x_2]$、$\theta_C = \vec{0}$ “;” 表示列向

③ Cost 函数（类似于概率；每个 theta0 都不进行惩罚）:

$$cost = L(\theta) = -\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{j=1}^{C} 1\{y^{(i)} = j\} log(h_\theta^j(x^{(i)}))\right] + \lambda \sum_{j=1}^{C}\sum_{k=1}^{n} \theta_{jk}^2$$

④ 梯度推导出有（取$\lambda = 0$，则不采取正则化）:

$$\frac{\partial L(\theta)}{\partial \theta_{j0}} = -\frac{1}{m}\sum_{i=1}^{m}[x^{(i)} \times (1\{y^{(i)} = j\} - h_\theta^j(x^{(i)})]$$

$$\frac{\partial L(\theta)}{\partial \theta_{j1}} = -\frac{1}{m}\sum_{i=1}^{m}[x^{(i)} \times (1\{y^{(i)} = j\} - h_\theta^j(x^{(i)})] + \lambda\theta_{j1}$$

$$\frac{\partial L(\theta)}{\partial \theta_{j2}} = -\frac{1}{m}\sum_{i=1}^{m}[x^{(i)} \times (1\{y^{(i)} = j\} - h_\theta^j(x^{(i)})] + \lambda\theta_{j2}$$

⑤ 迭代步骤（$\alpha = 0.01$）：

$$\theta_{j0} = \theta_{j0} - \alpha\frac{\partial L(\theta)}{\partial \theta_{j0}}$$

$$\theta_{j1} = \theta_{j1} - \alpha\frac{\partial L(\theta)}{\partial \theta_{j1}}$$

$$\theta_{j2} = \theta_{j2} - \alpha\frac{\partial L(\theta)}{\partial \theta_{j2}}$$

⑥ 由于归一化后最后求得的θ_i为θ_i'，易得θ_i'并非原始公式的θ_i，需要做变换后才能得到最终求的最后的θ_i，显然此处 y 并不做归一化处理，有：

$$\theta_k^T x^{(i)} = \theta_{k0}' + \theta_{k1}' x_1'^{(i)} + \theta_{k2}' x_2'^{(i)}$$

其中 $x_1'^{(i)} = \frac{x_1^{(i)} - min(x_1)}{max(x_1) - min(x_1)}$ ，$x_2'^{(i)} = \frac{x_2^{(i)} - min(x_2)}{max(x_2) - min(x_2)}$

继而：

$$\theta_{k0} + \theta_{k1} x_1^{(i)} + \theta_{k2} x_2^{(i)} = \theta_{k0}' + \theta_{k1}' x_1'^{(i)} + \theta_{k2}' x_2'^{(i)}$$

$$\theta_{k0} = \theta_{k0}' + \frac{\theta_{k1}' min(x_1)}{max(x_1) - min(x_1)} + \frac{\theta_{k2}' min(x_2)}{max(x_2) - min(x_2)}$$

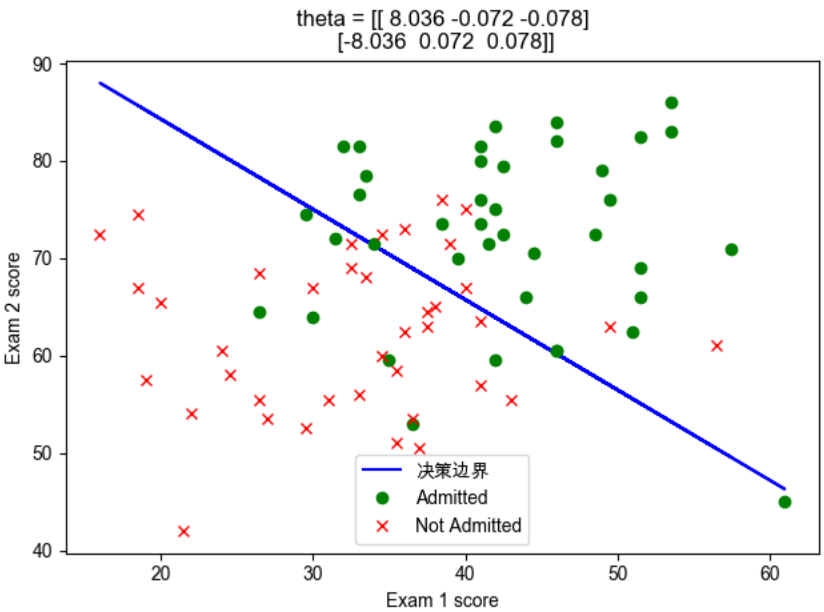
$$\theta_{k1} = \frac{\theta_{k1}'}{max(x_1) - min(x_1)}$$

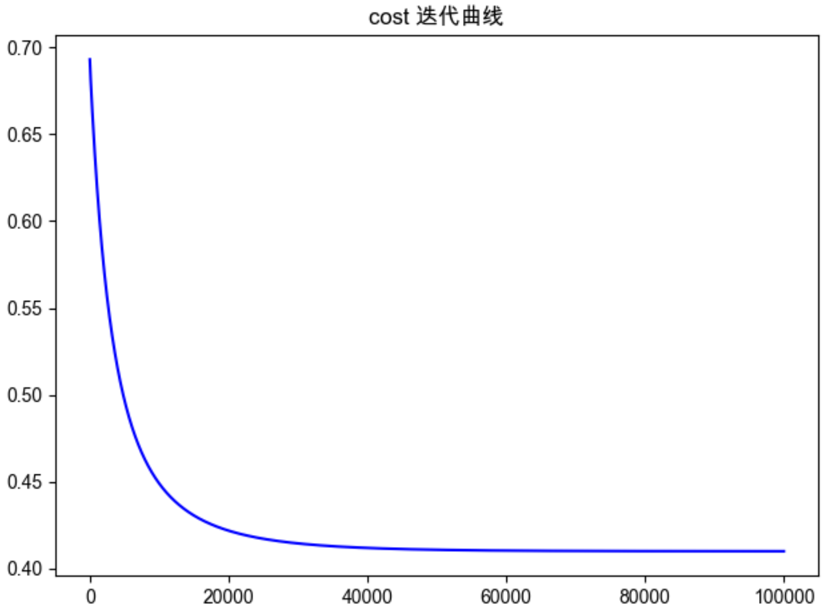
$$\theta_{k2} = \frac{\theta_{k2}'}{max(x_2) - min(x_2)}$$

最后结果：

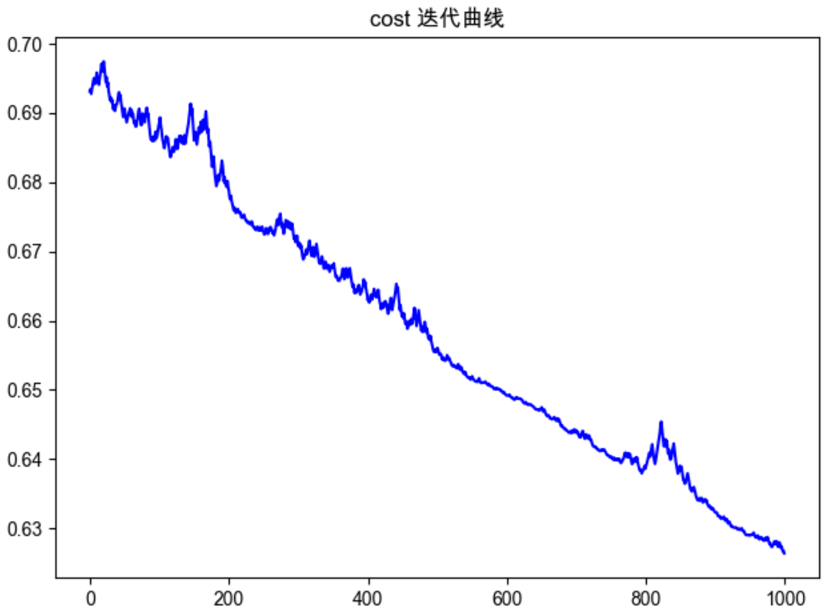
梯度下降：cost 0.40988953387900545

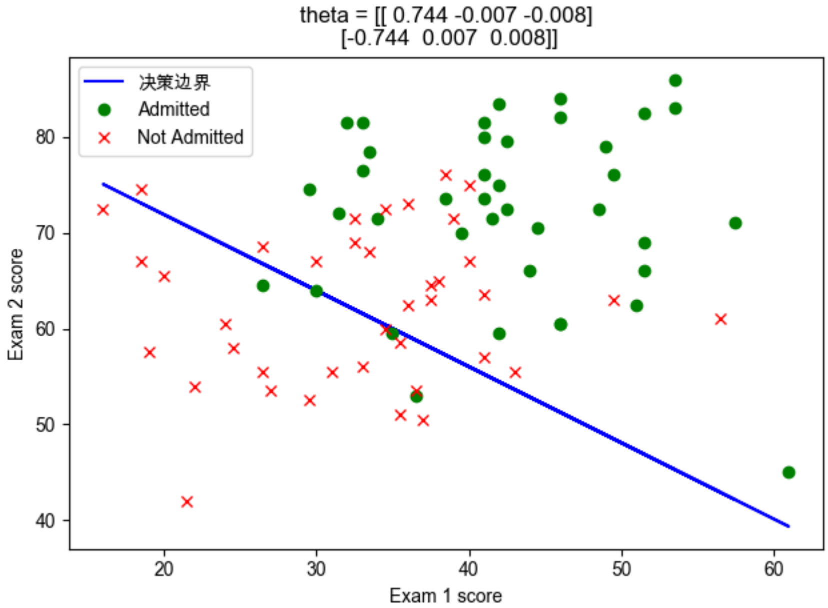
Softmax：cost 0.4098895338790054



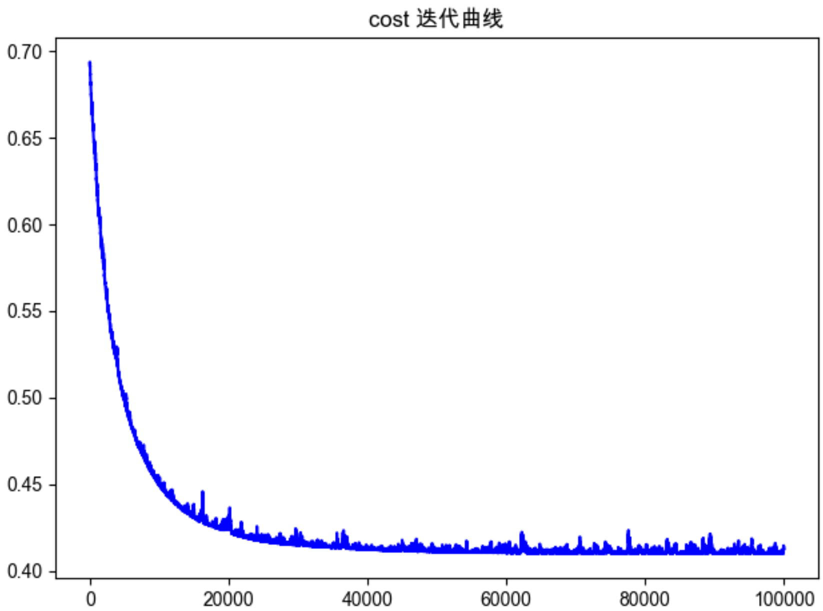


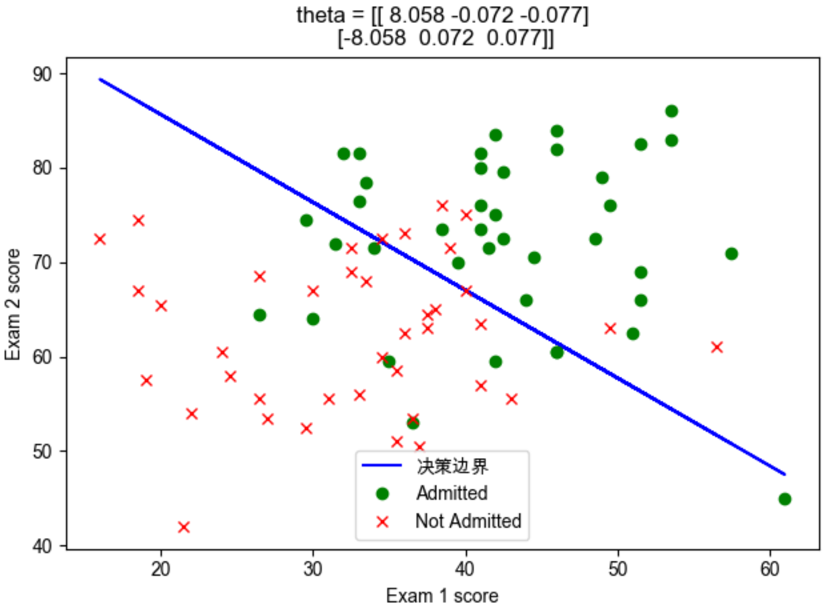
随机 softmax 回归
除了更新时选取一个外其他都一样。
跑 1000 次的结果如下：




theta = [[0.744 -0.007 -0.008]
[-0.744 0.007 0.008]]
决策边界
Admitted
Not Admitted
Exam 2 score
Exam 1 score
80
70
60
50
40
20
30
40
50
60

跑 1e5 次的结果如下：


cost 迭代曲线
0.70
0.65
0.60
0.55
0.50
0.45
0.40
0
20000
40000
60000
80000
100000



theta = [[8.058 -0.072 -0.077]
[-8.058 0.072 0.077]]
90
80
70
60
50
40
Exam 2 score
20
30
40
50
60
Exam 1 score
决策边界
Admitted
Not Admitted

BP 误差逆传播。

①假设：

$$\hat{y}_j = \sigma(\beta_j + \theta_j)$$

$$\beta_j = \sum_{h=1}^{q} w_{hj} b_h$$

$$b_h = \sigma(\alpha_h + \gamma_h)$$

$$\alpha_h = \sum_{i=1}^{d} v_{ih} x_i$$

$\sigma(z)$ 表示 sigmoid 函数。

②Cost 函数（均方差）：

$$E^{(k)} = \frac{1}{2} \sum_{j=1}^{l} (\hat{y}_j^{(k)} + y_j^{(k)})^2$$

③梯度推导出有：

$$\Delta w_{hj} = \eta g_j b_h$$
$$\Delta \theta_j = -\eta g_j$$
$$\Delta v_{ih} = \eta e_h x_i$$
$$\Delta \gamma_h = -\eta e_h$$

其中

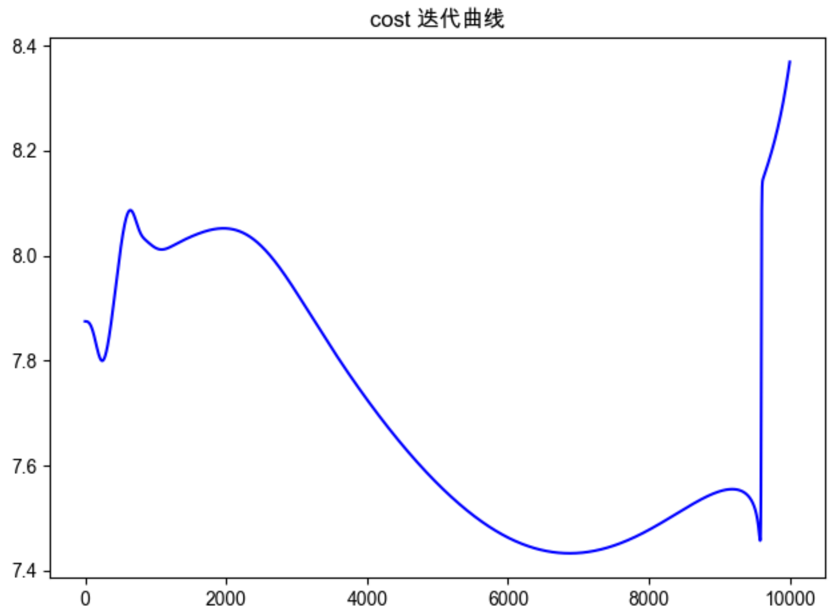
$$g_j = \hat{y}_j^{(k)}(1 - \hat{y}_j^{(k)})(y_j^{(k)} - \hat{y}_j^{(k)})$$

$$e_h = b_h(1 - b_h) \sum_{j=1}^{l} w_{hj} g_j$$

④迭代步骤（$\eta = 0.01$）：

$$w_{hj} = w_{hj} + \Delta w_{hj}$$
$$\theta_j = \theta_j + \Delta \theta_j$$
$$v_{ih} = v_{ih} + \Delta v_{ih}$$
$$\gamma_h = \gamma_h + \Delta \gamma_h$$

不知道为什么，代码我检查了快 10 遍了，还是没跑出来。等下次写作业了重新弄一下吧……附上一个没跑出来的图😭


cost 迭代曲线
8.4
8.2
8.0
7.8
7.6
7.4
0
2000
4000
6000
8000
10000