Homework 3: Linear Regression and Logistic Regression

> Student ID:
>
> Student Name:
>
> Lectured by: Shangsong Liang
>
> Course: Information Retrieval Course, Sun Yat-sen University

Exercise One: Linear Regression

线性回归模型为：

$$h(x) = \theta_0 + \theta_1 x_1 + \theta_3 x_2$$

梯度下降迭代，单次迭代(a为学习率)：

$$\theta_0 = \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)} - y^{(i)}) \cdot x_0^{(i)})$$

$$\theta_1 = \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)} - y^{(i)}) \cdot x_1^{(i)})$$

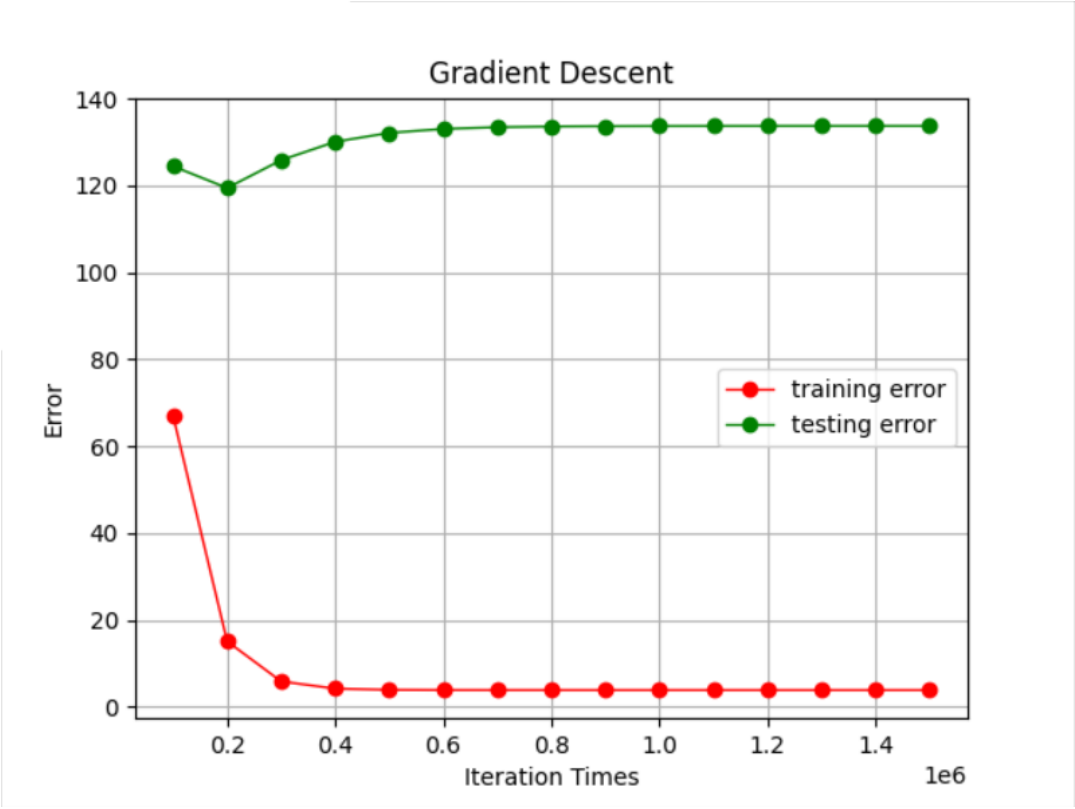
$$\theta_2 = \theta_2 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)} - y^{(i)}) \cdot x_2^{(i)})$$

计算误差：

$$E_\theta = \frac{1}{m} \sum (h_\theta(x^{(i)}) - y^{(i)})^2$$

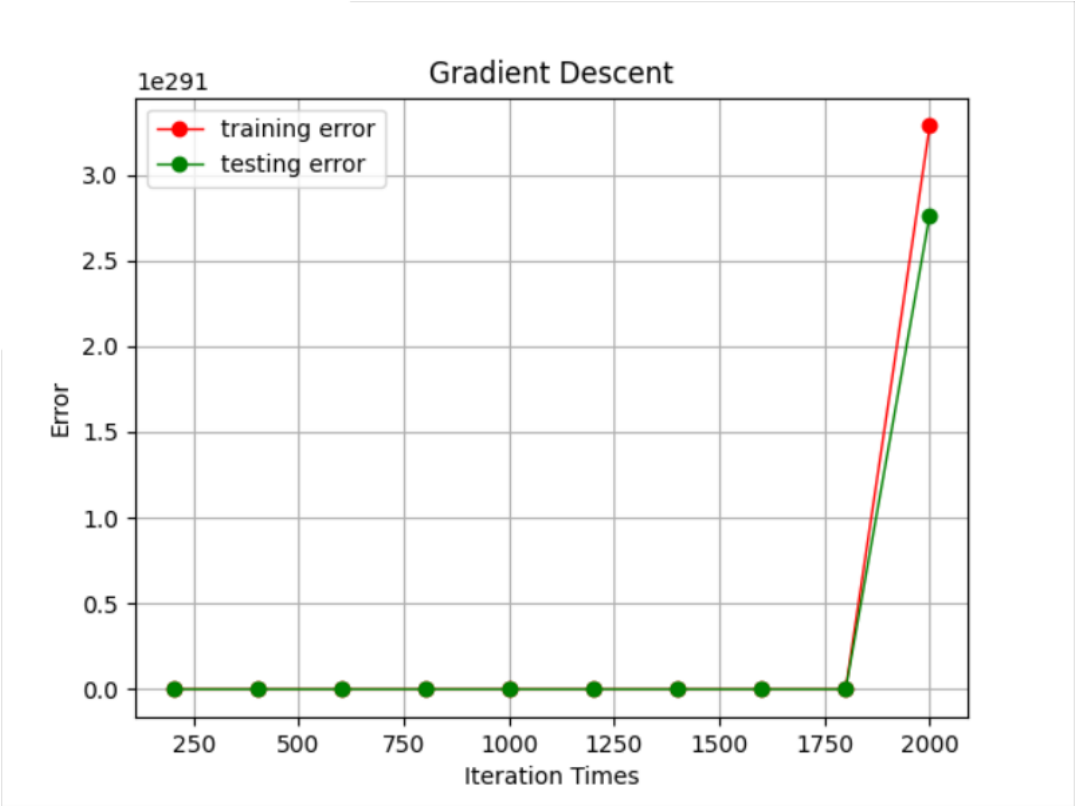
(a)

结果如图：



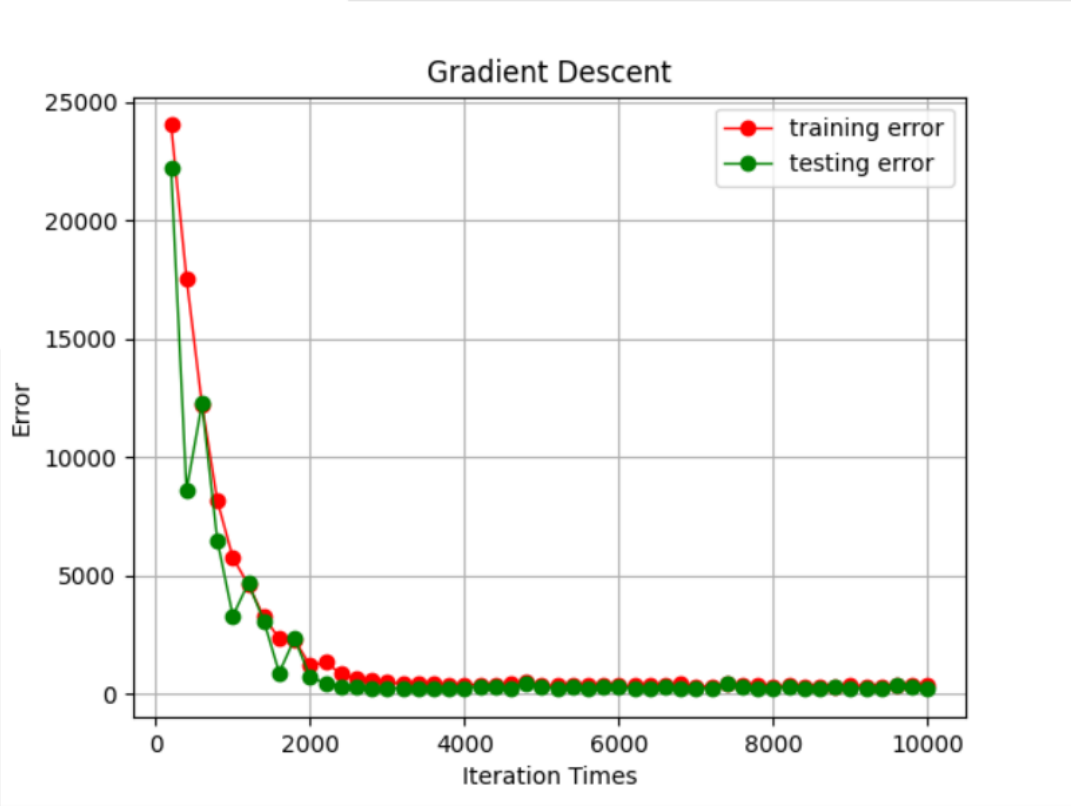
可以发现，训练到2万次之后出现了过拟合的问题，

(b)



可以发现，训练一段时间后误差变得极大，难以收敛，因此需要适当降低学习率，从而加快模型收敛。

(c)



可以发现，对于每次迭代，用随机选择 k 个数据取代原本的使用全部数据，降低了训练时间。但是由于训练数据的随机性，会出现在测试集的误差波动大的情况，但最终区域收敛。

Exercise Two: Logistic Regression

(a)

Sigmoid 函数作为逻辑函数：

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

逻辑回归模型：

$$P(y=1|x;\theta) = g(\theta^T x) = \frac{1}{1+e^{-\theta^T x}}, \qquad \theta = [w_0 \quad w_1 \quad w_2 \quad w_3 \quad w_4 \quad w_5 \quad w_6]$$

因为 y 符合二项分布，对应条件的对数似然函数为：

$$LCL = \sum \log p^{(i)} + \sum \log\left(1-p^{(i)}\right)$$

(b)

对上述函数求偏导：

$$\text{对}w_0\text{求偏导} \qquad \frac{\partial}{\partial w_0} LCL = \sum (y^{(i)} - p^{(i)})$$

$$\text{对}w_j\text{求偏导} \qquad \frac{\partial}{\partial w_j} LCL = \sum (y^{(i)} - p^{(i)}) x_j^{(i)}, \qquad j \in [1,6],\ j\text{是整数}$$

梯度下降：

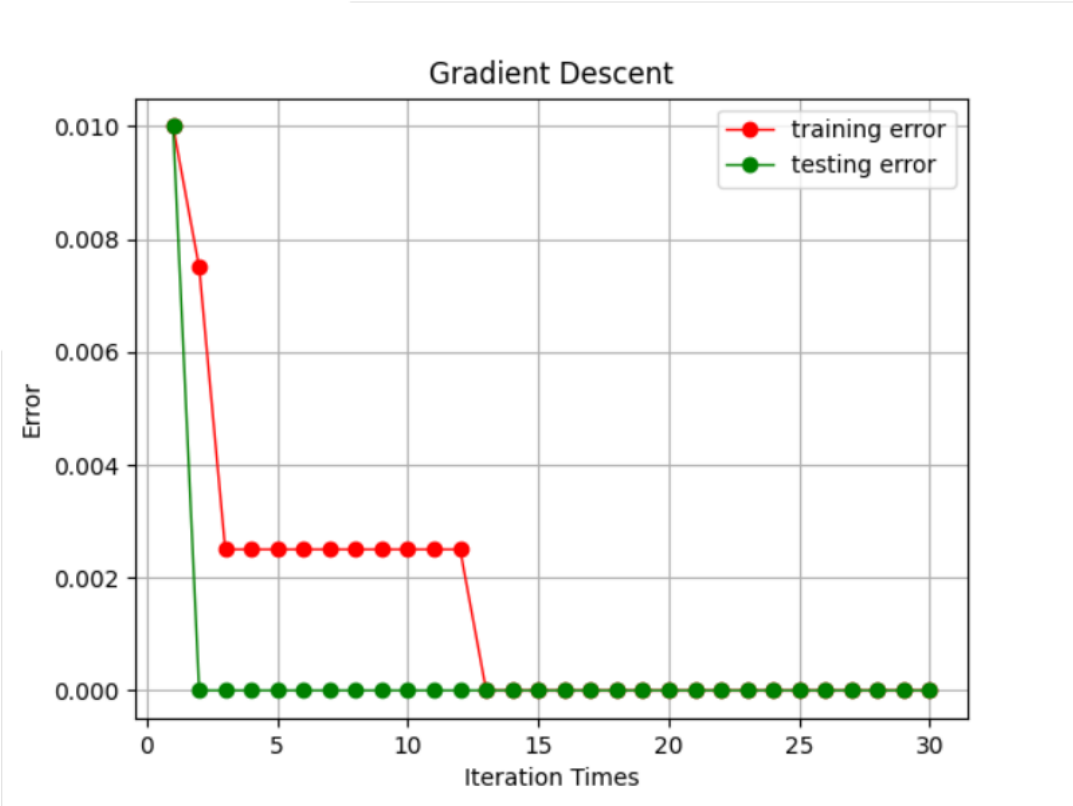
$$w_0 := w_0 + \alpha \frac{\partial}{\partial w_0} LCL = w_0 + \alpha \sum (y^{(i)} - p^{(i)})$$

$$w_j := w_j + \alpha \frac{\partial}{\partial w_j} LCL = w_j + \alpha \sum (y^{(i)} - p^{(i)}) x_j^{(i)}, \qquad j \in [1, 6],\ j\text{是整数}$$

(c)

在exercise one代码的基础上进行修改，按照模型公式进行设计，并采用训练集进行正常梯度下降训练。

训练结果如下：



最终收敛时各个参数值：[0.5662 -2.6123   3.5120 -2.6362   3.4185 -2.1779   0.0474]

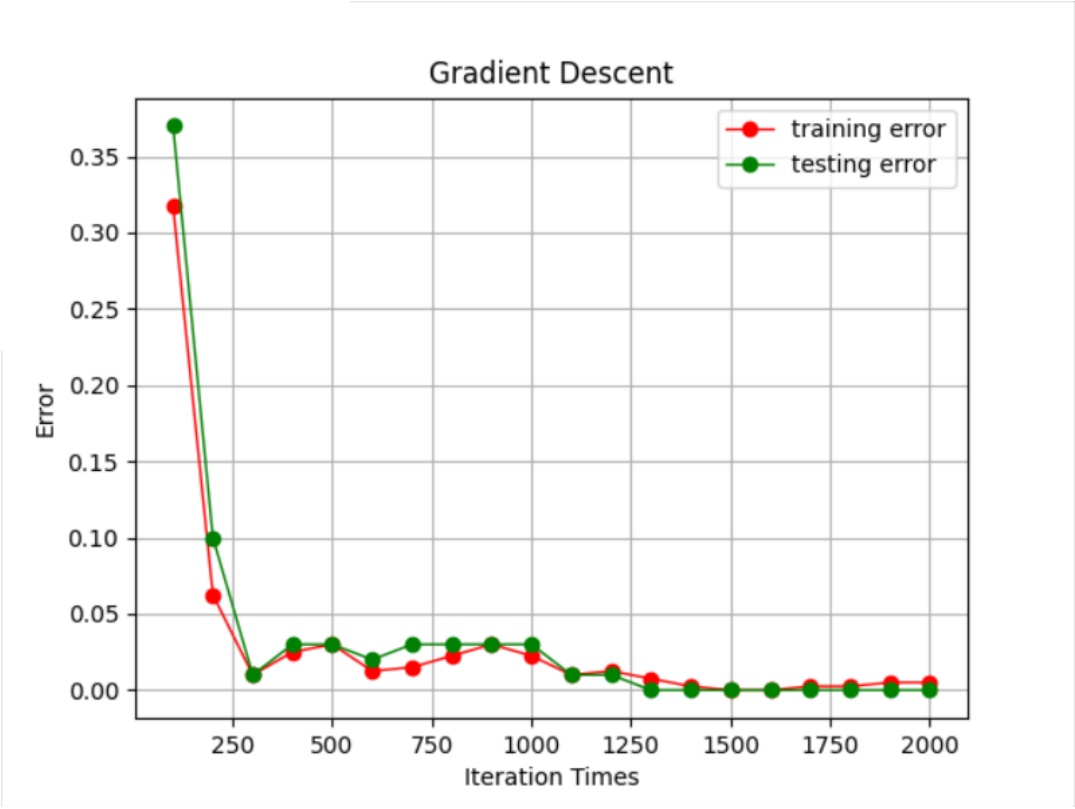
在测试集上预测的准确率达到100%

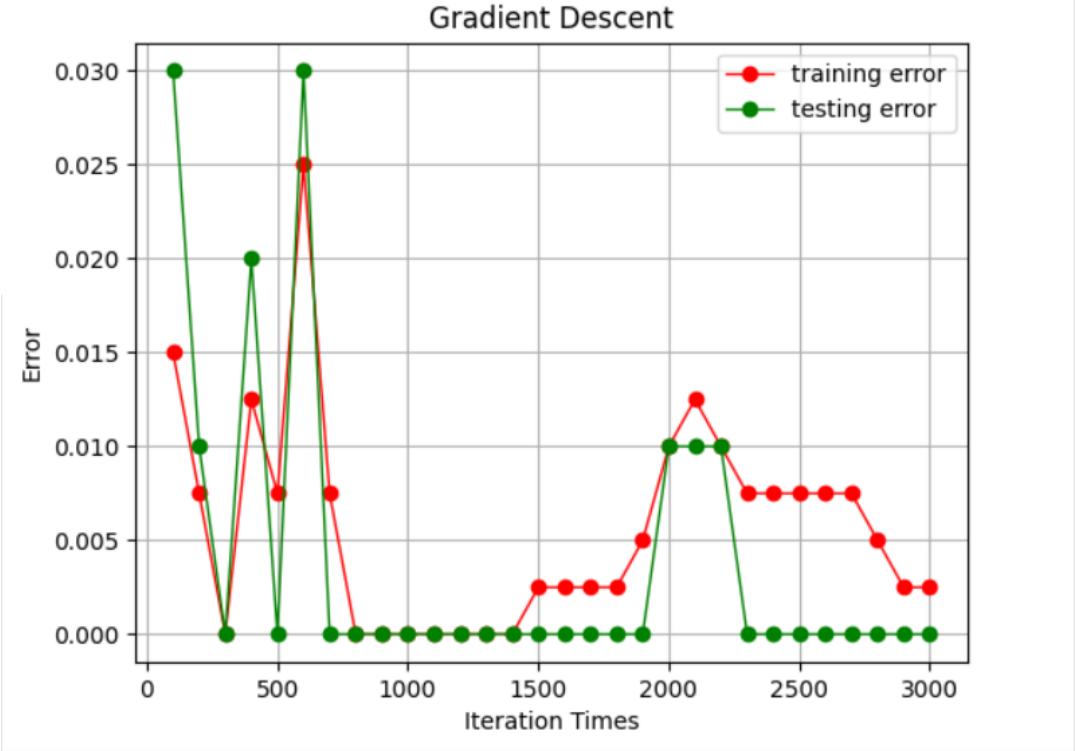
(d)

最终收敛时，没有missclassified examples。

(e)

部分训练结果：


Gradient Descent
training error
testing error
Error
Iteration Times
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
250
500
750
1000
1250
1500
1750
2000


Gradient Descent
training error
testing error
Error
Iteration Times
0.030
0.025
0.020
0.015
0.010
0.005
0.000
0
500
1000
1500
2000
2500
3000

iteration_times	theta_0	theta_1	theta_2	theta_3	theta_4	theta_5	theta_6	train_error_rate	test_error_rate
100	0.0059	-0.0123	0.0215	-0.0124	0.0238	-0.0126	0.0008	0.015000000000000013	0.030000000000000027
200	0.0114	-0.0278	0.0416	-0.0290	0.0458	-0.0239	0.0034	0.007499999999999951	0.010000000000000009
300	0.0110	-0.0446	0.0600	-0.0463	0.0651	-0.0424	0.0015	0.0	0.0
400	0.0055	-0.0643	0.0775	-0.0671	0.0791	-0.0610	-0.0038	0.012499999999999956	0.020000000000000018
500	0.0118	-0.0785	0.0975	-0.0826	0.1022	-0.0746	-0.0040	0.007499999999999951	0.0
600	0.0032	-0.0975	0.1118	-0.1008	0.1145	-0.0934	-0.0077	0.025000000000000022	0.030000000000000027
700	0.0127	-0.1091	0.1341	-0.1120	0.1382	-0.0983	-0.0041	0.007499999999999951	0.0
800	0.0328	-0.1150	0.1612	-0.1201	0.1666	-0.0998	0.0000	0.0	0.0
900	0.0323	-0.1330	0.1796	-0.1361	0.1826	-0.1156	-0.0016	0.0	0.0
1000	0.0357	-0.1471	0.1999	-0.1503	0.1999	-0.1295	-0.0032	0.0	0.0
1100	0.0408	-0.1589	0.2191	-0.1627	0.2191	-0.1419	-0.0017	0.0	0.0
1200	0.0455	-0.1702	0.2377	-0.1733	0.2388	-0.1529	-0.0026	0.0	0.0
1300	0.0508	-0.1834	0.2542	-0.1856	0.2568	-0.1614	0.0019	0.0	0.0
1400	0.0489	-0.1997	0.2683	-0.2018	0.2703	-0.1754	-0.0003	0.0	0.0
1500	0.0438	-0.2167	0.2818	-0.2180	0.2824	-0.1914	-0.0026	0.0024999999999999467	0.0
1600	0.0442	-0.2330	0.2968	-0.2340	0.2975	-0.2076	-0.0032	0.0024999999999999467	0.0
1700	0.0430	-0.2464	0.3118	-0.2483	0.3135	-0.2244	-0.0037	0.0024999999999999467	0.0
1800	0.0454	-0.2591	0.3284	-0.2621	0.3314	-0.2352	-0.0061	0.0024999999999999467	0.0
1900	0.0419	-0.2734	0.3429	-0.2786	0.3444	-0.2507	-0.0117	0.005000000000000044	0.0
2000	0.0357	-0.2906	0.3567	-0.2953	0.3553	-0.2662	-0.0177	0.010000000000000009	0.010000000000000009
2100	0.0333	-0.3059	0.3711	-0.3107	0.3676	-0.2817	-0.0193	0.012499999999999956	0.010000000000000009
2200	0.0410	-0.3152	0.3899	-0.3210	0.3877	-0.2909	-0.0217	0.010000000000000009	0.010000000000000009
2300	0.0481	-0.3244	0.4084	-0.3316	0.4063	-0.2984	-0.0194	0.007499999999999951	0.0
2400	0.0502	-0.3388	0.4233	-0.3434	0.4233	-0.3097	-0.0203	0.007499999999999951	0.0
2500	0.0513	-0.3517	0.4383	-0.3560	0.4386	-0.3199	-0.0220	0.007499999999999951	0.0
2600	0.0513	-0.3633	0.4531	-0.3712	0.4542	-0.3305	-0.0240	0.007499999999999951	0.0
2700	0.0538	-0.3744	0.4674	-0.3851	0.4703	-0.3398	-0.0228	0.007499999999999951	0.0
2800	0.0585	-0.3859	0.4839	-0.3975	0.4862	-0.3494	-0.0211	0.005000000000000044	0.0
2900	0.0629	-0.3944	0.4997	-0.4067	0.5031	-0.3581	-0.0171	0.0024999999999999467	0.0
3000	0.0656	-0.4069	0.5133	-0.4170	0.5181	-0.3676	-0.0182	0.0024999999999999467	0.0

随机多次进行训练，平均迭代1300次后收敛。

(f)

可以发现，随着训练样本数量的增加，误差逐渐降低，并逐渐趋于收敛。
但是由于采样随机采样，具有随机性，因此训练结果的误差会出现一些波动。