机器学习与数据挖掘-HW3

—Linear Regression and Logistic Regression—

19335253 葉珺明

---

1 Ex1: Linear Regression

1.a Gradient Descent

- The Linear Model:

$$\hat{y} = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

- Loss Function:

$$Loss = \sum_{i=1}^{M} (\hat{y}_i - y_i)^2$$

- Want to compute:

$$\hat{\theta}_{MLE} = arg \min_{\theta \in R^p} \sum_{i=1}^{M} (\hat{y}_i - y_i)^2$$

- update θ:

$$\theta = \theta - \alpha * \frac{\partial J(\theta)}{\partial \theta}$$

参数初始 $\theta = [0, 0, 0]$, $iters = 1500000, \ \ \alpha = 0.00015$

所得实验结果:

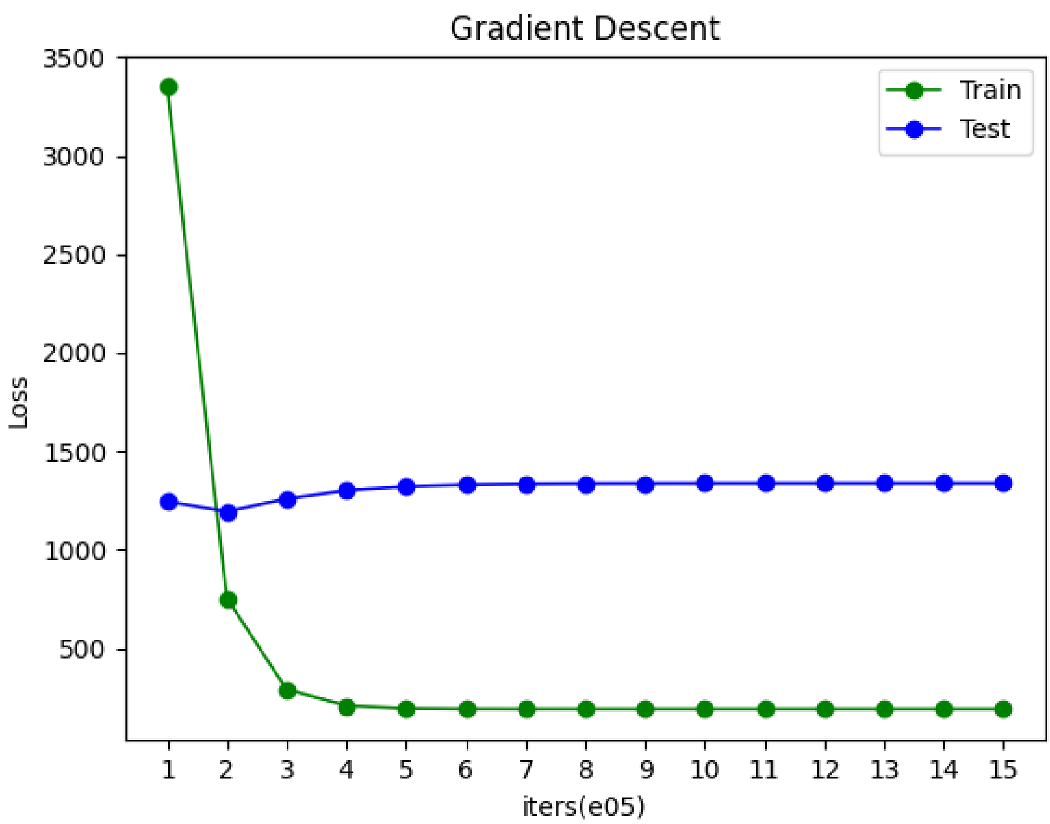
Training Data:

iters(^e05)	1	2	3	4	5	6	7	8
Loss	3350.33	752.955	290.803	208.573	193.941	191.338	190.875	190.792
iters(^e05)	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
Loss	190.778	190.775	190.775	190.774	190.774	190.774	190.774	

Test Data:

iters(^e05)	1	2	3	4	5	6	7	8
Loss	1244.63	1194.55	1258.72	1300.96	1321.48	1330.62	1334.56	1336.24
iters(^e05)	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
Loss	1336.95	1337.25	1337.37	1337.43	1337.45	1337.46	1337.46	

图示：



由图示，

- 训练集的损失函数随着迭代次数的增加而减小，趋于数值190.774
- 测试集的损失函数在迭代次数为200000时达到最小值1194.55
- 迭代次数小于200000时，欠拟合；迭代次数大于200000时，过拟合

当迭代次数为200000时，即测试集损失函数值最小时，θ取值为：$[65.4873, 6.9001, -72.5408]$

即最后Linear Module：

$$\hat{y} = 65.4873 + 6.9001x_1 - 72.5408x_2$$

1.b 改变学习率为0.0002

改变学习率为0.0002后，迭代过程中发现 θ 的输出为nan，即无法收敛。

取学习率为0.000175，

实验结果：

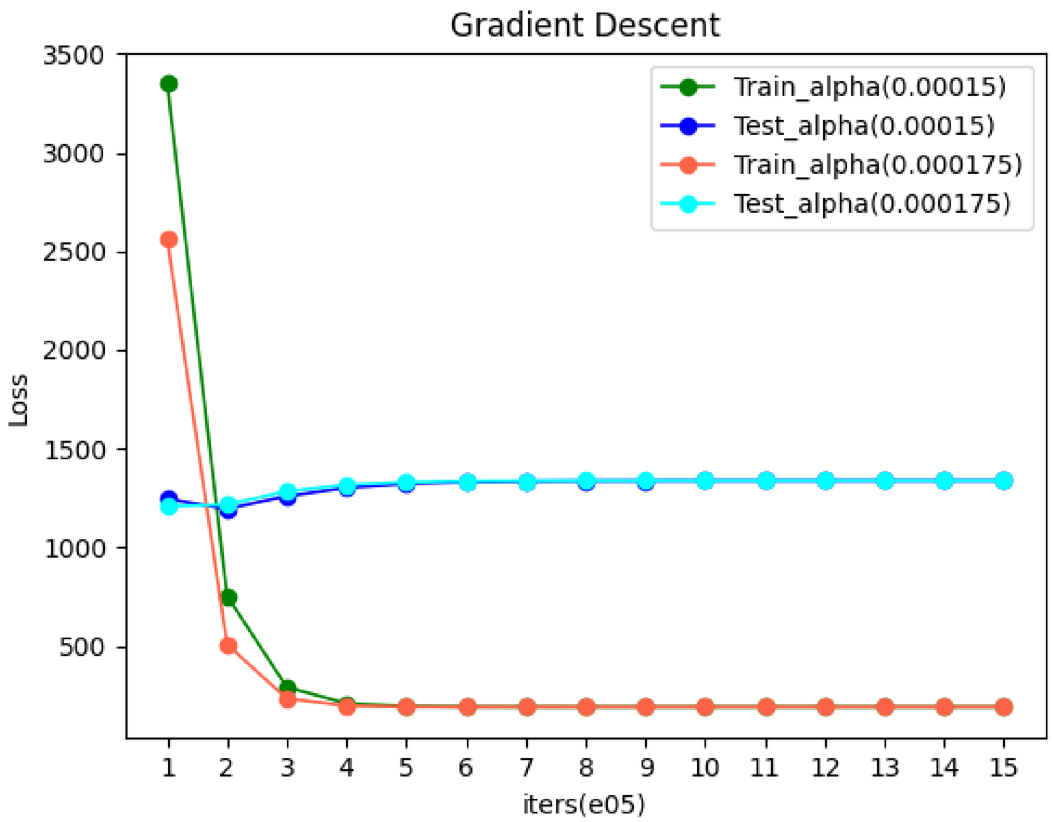
Training Data:

iters(^e05)	1	2	3	4	5	6	7	8
Loss	2560.32	506.970	232.968	196.405	191.526	190.875	190.788	190.776
iters(^e05)	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
Loss	190.775	190.774	190.774	190.774	190.774	190.774	190.774	

Test Data:

iters(^e05)	1	2	3	4	5	6	7	8
Loss	1207.71	1216.09	1283.26	1316.35	1329.57	1334.56	1336.40	1337.08
iters(^e05)	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
Loss	1337.32	1337.41	1337.45	1337.46	1337.46	1337.46	1337.46	

图示:

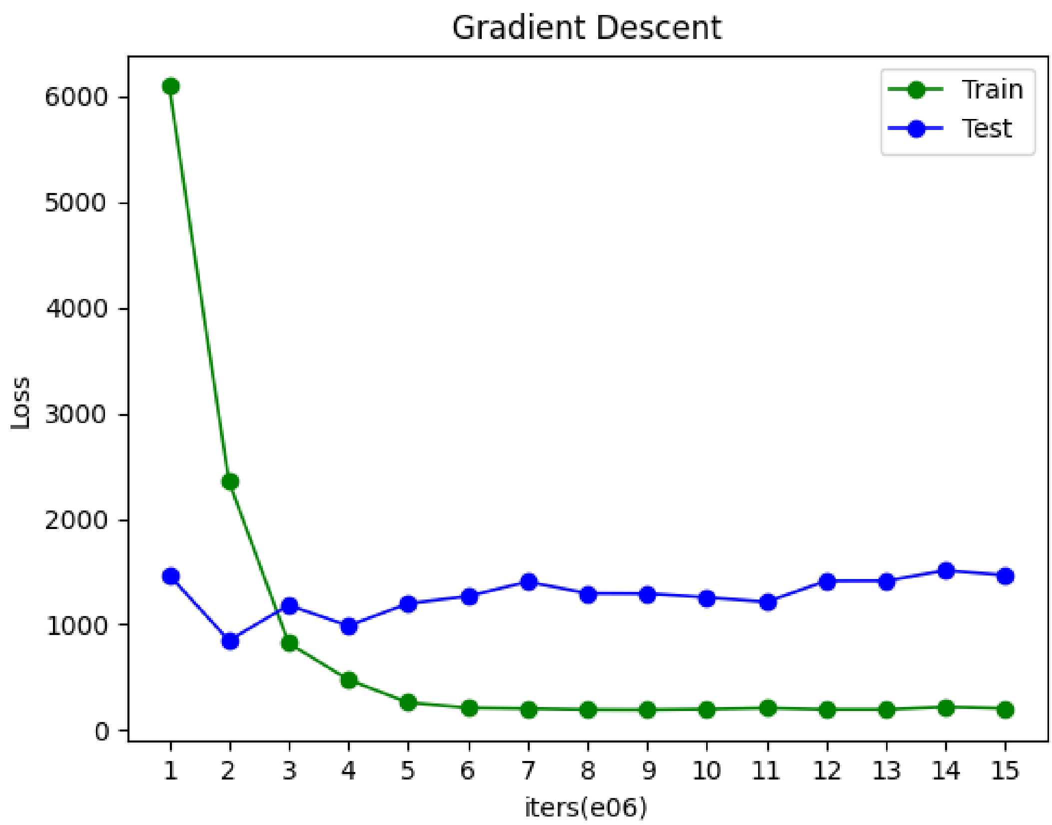


总结,

- 学习率增大,损失函数的下降速度会变快
- 学习率小时,减少震荡,更容易达到局部最优
- 学习率过大时有可能导致不收敛

1.c 随机梯度下降法

设置实验迭代次数为15*e06，每1e*06记录数据，实验结果如下图：



与**1.a**中的实验比较得出，

- 随机梯度下降法的损失函数值初始很大，下降速度比梯度下降更快
- 就本实验而言，随机梯度下降的损失函数值最终结果比梯度下降的最终结果要好
- 从全过程来看，随机梯度在训练集和测试集上的表现都比梯度下降好
- 需要适当调节学习率，如果学习率过大，容易使得损失函数波动过大。

随机梯度下降法在测试集表现最优时的 θ 取值：$[53.5271, 6.9958, -72.6382]$

即最后Linear Module：

$$\hat{y} = 53.5271 + 6.9958x_1 - 72.6382x_2$$

2 Ex2: Logistic Regression

2.a Formula:

$$\mathbf{w} = arg\max_{\mathbf{w}} \sum_l \ln P(y^l|\mathbf{x}^l, \mathbf{w})$$

It can be written as:

$$
\begin{aligned}
l(\mathbf{w}) &= \sum_l y^l \ln P(y^l=1|\mathbf{x}^l,\mathbf{w}) + (1-y^l)\ln P(y^l=0|\mathbf{x}^l,\mathbf{w}) \\
&= \sum_l y^l \ln \frac{P(y^l=1|\mathbf{x}^l,\mathbf{w})}{P(y^l=0|\mathbf{x}^l,\mathbf{w})} + \ln P(y^l=0|\mathbf{x}^l,\mathbf{w}) \\
&= \sum_l y^l(w_0+\sum_{i=1}^n w_i x_i^l) - \ln\left(1+\exp\left(w_0+\sum_{i=1}^n w_i x_i^l\right)\right)
\end{aligned}
$$

2.b Computing

$$
\frac{\partial}{\partial w_0} l(\mathbf{w}) = \sum_l x_i^l (y^l - \hat{P}(y^l=1|\mathbf{x}^l,\mathbf{w}))
$$

2.c

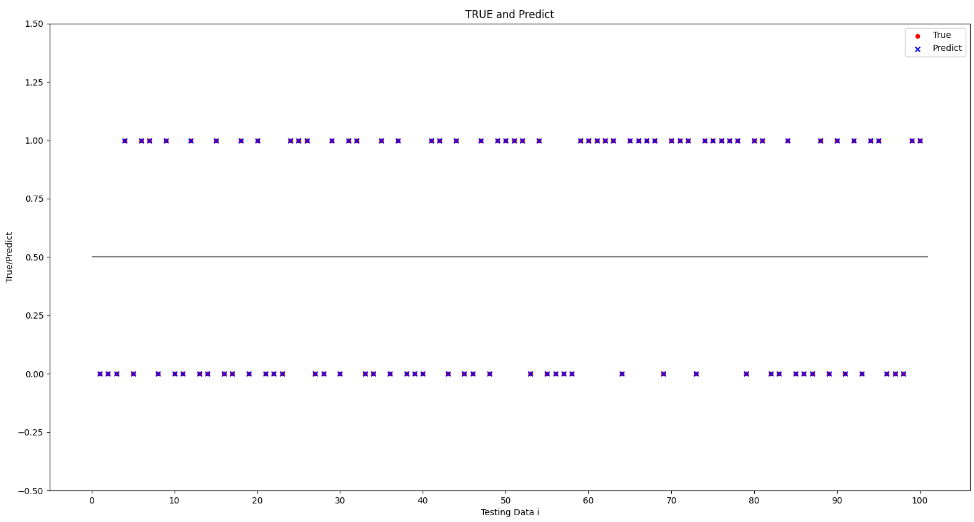
- 阈值为0.5:

$$
\hat{y}=1, if\ P(y^l=1|\mathbf{x}^l)>0.5 \\
\hat{y}=0, if\ P(y^l=1|\mathbf{x}^l)<0.5
$$

- 使用随机梯度下降法优化:

$$
w_i \leftarrow w_i + \eta * x_i^l (y^l - \hat{P}(y^l=1|\mathbf{x}^l,\mathbf{w}))
$$

- 实验结果:



optimal estimated parameters:

$$
\theta_0 = 1.27272928 \\
\theta_1 = -6.78728572 \\
\theta_2 = 9.89956747 \\
\theta_3 = -7.02519041 \\
\theta_4 = 8.80175859 \\
\theta_5 = -4.98863402 \\
\theta_6 = 0.11196229
$$

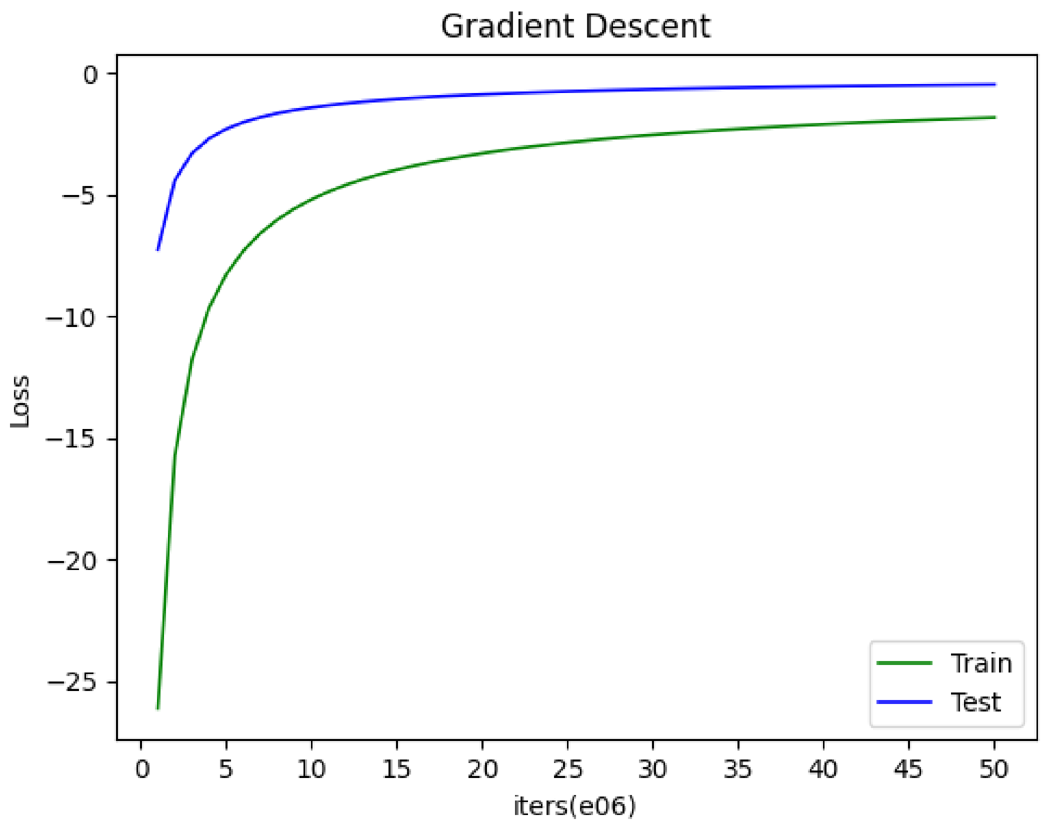
上图是测试集的实验结果，可以看出真实值与预测值一模一样

2.d Misclassified examples in the testing dataset

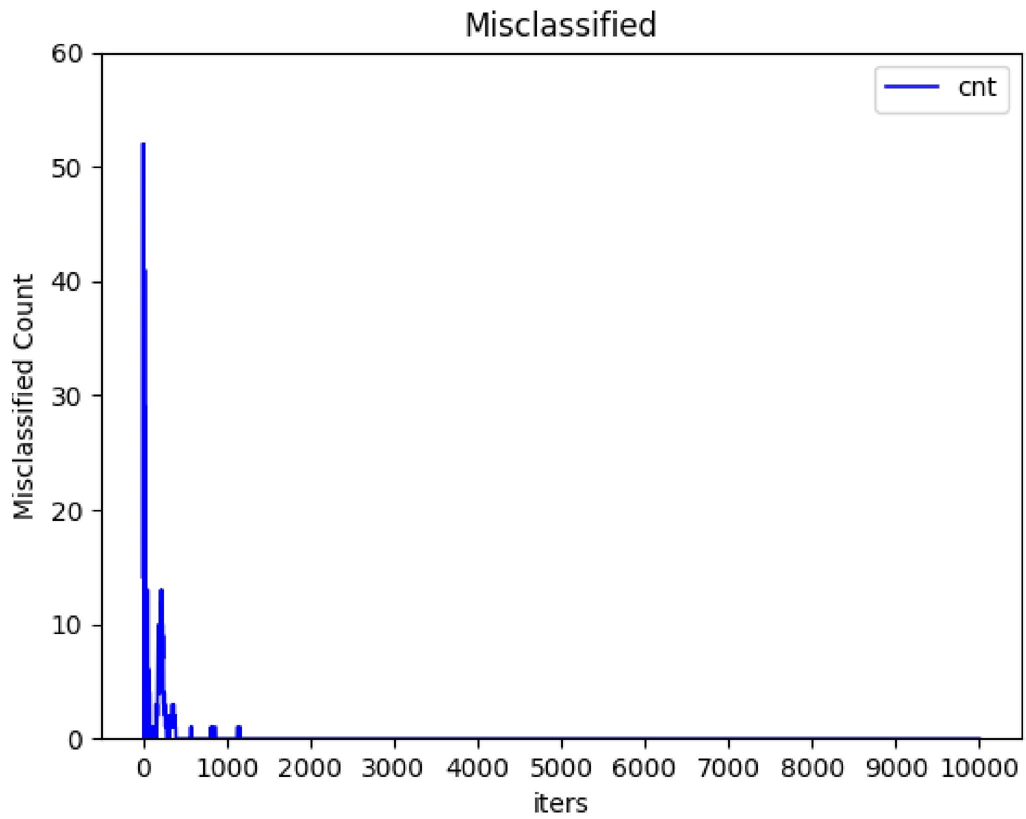
错误预测的个数是0

2.e Plot

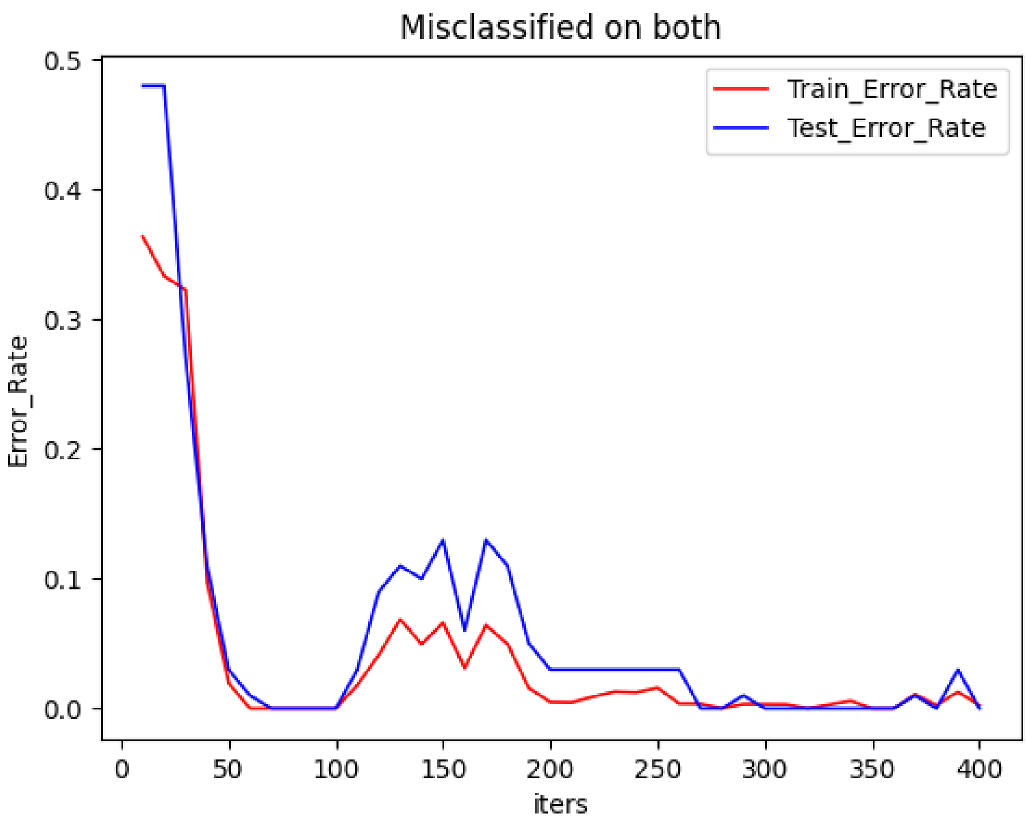
实验结果：



测试集预测错误数与迭代次数的关系：在迭代1200次左右完成该收敛



2.f Misclassified on both with K



由图示,

- 随着训练集的增加，训练集错误率和测试集错误率呈下降趋势

- 在训练集50至100时，错误率到局部最小；而后错误率增大，在训练集270至400错误率又降到最低
- 训练集错误率曲线和测试集错误率曲线分布相似，但训练错率大部分情况下低于测试错误率
- 训练集上表现好的模型，在测试集上不一定表现好

综合，训练样本的质量和训练集大小都会对模型预测影响。