

**《机器学习》课程实验报告**

**学 院 软件学院**

**专 业 软件工程**

**组 员**  曹瑞秋

**学 号 201330610031**

**邮 箱 3144148605@qq.com**

**指导教师** 吴庆耀

**提交日期** **2017年 12 月 8 日**

## 【1..实验题目】 线性回归，线性分类与梯度下降

## 【2.实验时间】2017年 12月 2日 ~ 2017年 12月 8日

## 【3. 报告人】 曹瑞秋

## 【4. 实验目的】

* 进一步理解线性回归和梯度下降的原理。
* 在小规模数据集上实践。
* 体会优化和调参的过程。

## 【5. 数据集以及数据分析】

采用LIBSVM Data上提供的Housing和austrilian数据集（以上均采用scaled版本）

* 数据集Housing经分析，比较符合回归问题的特征，采用线性回归模型进行求解。
* 数据集austrilian经分析，比较符合分类问题的特征，采用线性SVM模型求解。

**【6. 实验步骤】**

线性回归和梯度下降

* 读取实验数据，使用sklearn库的[load\_svmlight\_file](http://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_svmlight_file.html" \t "https://www.zybuluo.com/chenyaofo/note/_blank)函数读取数据。
* 将数据集切分为训练集和验证集，本次实验不切分测试集。使用[train\_test\_split](http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html" \t "https://www.zybuluo.com/chenyaofo/note/_blank)函数切分数据集。
* 线性模型参数初始化，可以考虑全零初始化，随机初始化或者正态分布初始化。
* 选择Loss函数及对其求导。
* 求得**所有样本**对Loss函数的梯度G。
* 取梯度G的负方向，记为D。
* 更新模型参数，Wt = Wt-1 + lr \* D。lr为学习率，是人为调整的超参数。
* 在训练集上测试并得到Loss函数值，在验证集上测试并得到Loss函数值。
* 重复步骤5-8若干次，**画出和随迭代次数的变化图**。

线性分类和梯度下降

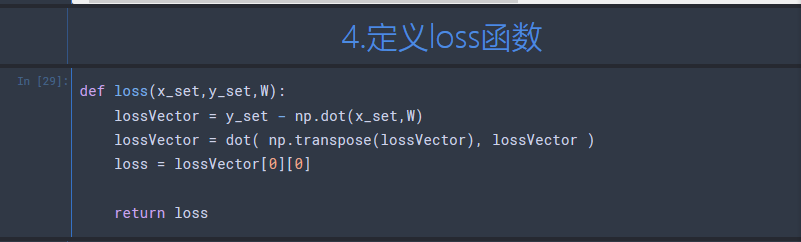
* 读取实验数据，使用sklearn库的[load\_svmlight\_file](http://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_svmlight_file.html" \t "https://www.zybuluo.com/chenyaofo/note/_blank)函数读取数据。
* 将数据集切分为训练集和验证集，本次实验不切分测试集。使用[train\_test\_split](http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html" \t "https://www.zybuluo.com/chenyaofo/note/_blank)函数切分数据集。
* 支持向量机模型参数初始化，可以考虑全零初始化，随机初始化或者正态分布初始化。
* 选择Loss函数及对其求导。
* 求得**所有样本**对Loss函数的梯度G。
* 取梯度G的负方向，记为D。
* 更新模型参数，Wt = Wt-1 + lr \* D。lr为学习率，是人为调整的超参数。
* 在训练集上测试并得到Loss函数值，在验证集上测试并得到Loss函数值。
* 重复步骤5-8若干次，**画出和随迭代次数的变化图**。

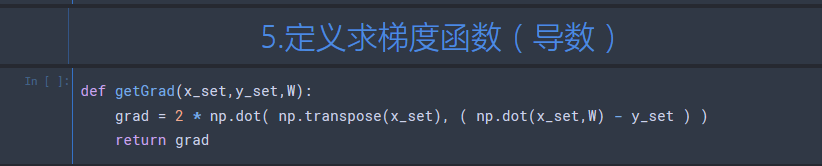
## 【7. 代码内容】

* 线性回归和线性分类，两个实验的非共用部分代码：

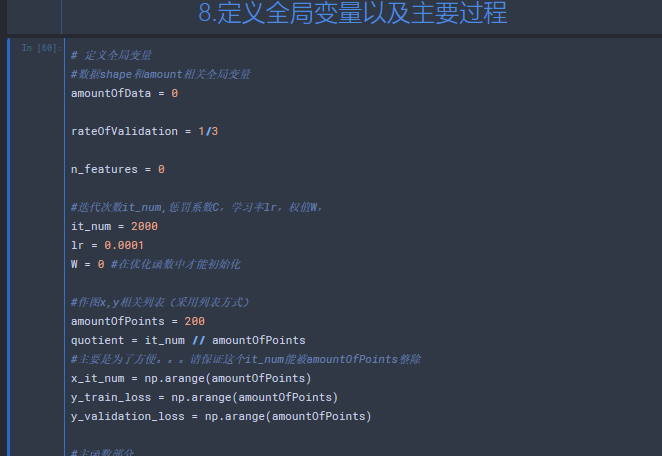
主要是Loss函数，以及其导函数。

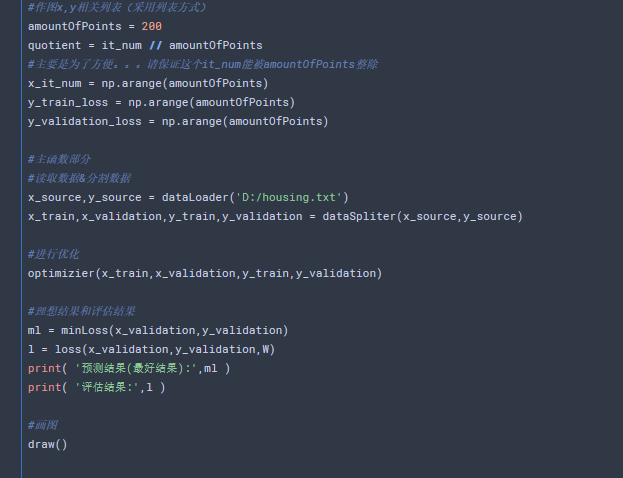
线性回归：

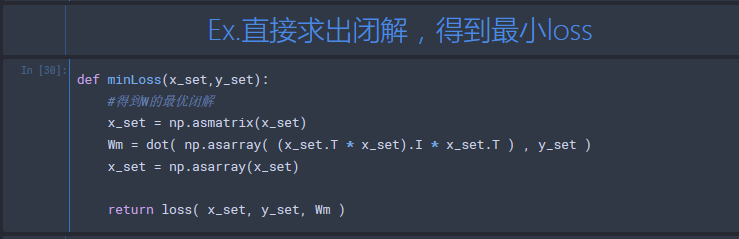






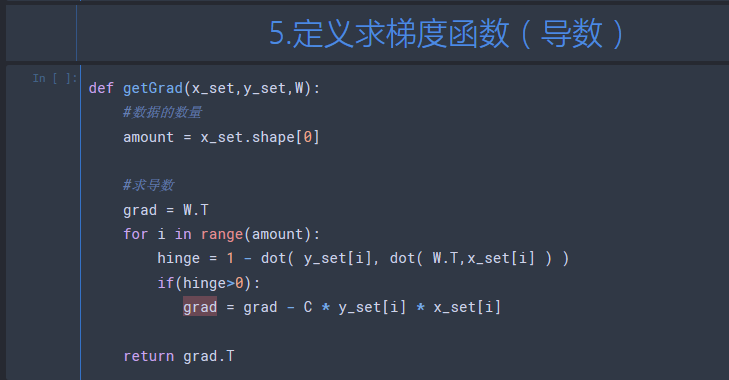






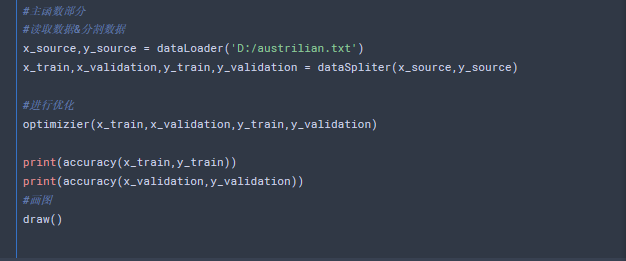
线性分类（线性SVM）：





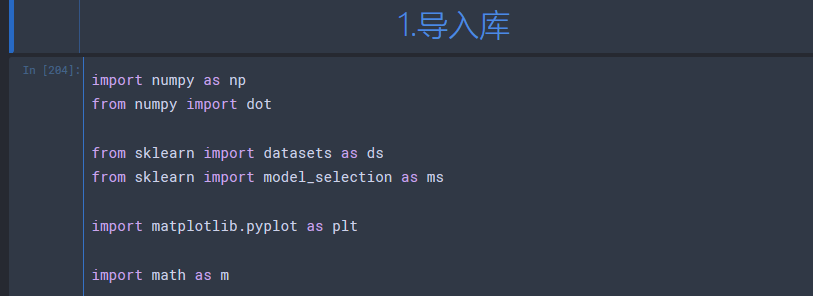




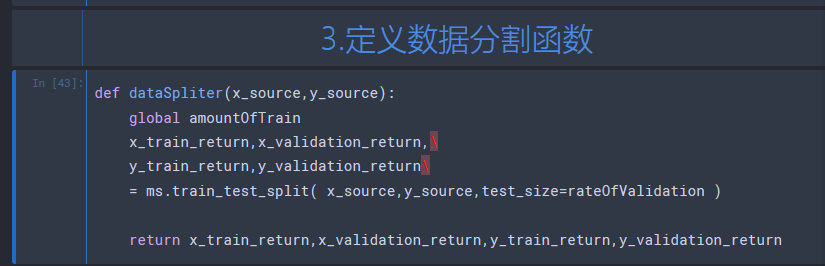


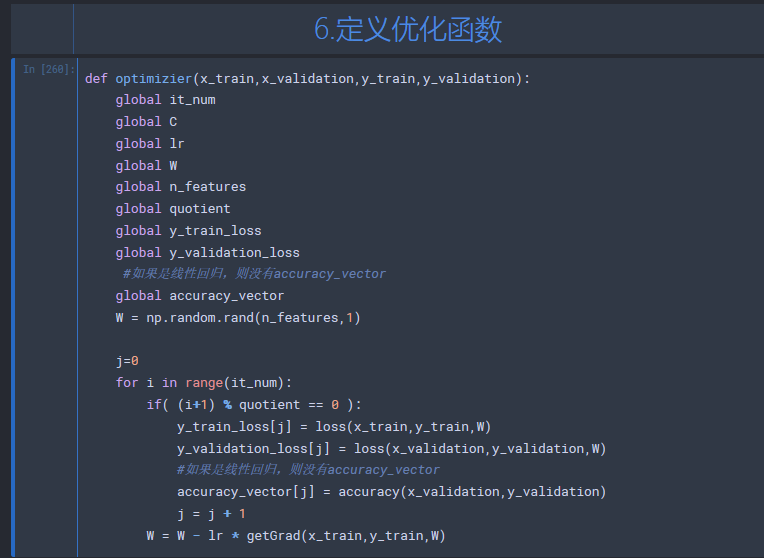


* 两个实验的共用部分代码：









## 8~12依据线性回归，和线性分类分别填写

## ☆线性回归

## 【8. 选择的评估方法】

## 留出法，取数据集的2/3作为训练集，1/3作为验证集

## 【9. 模型参数的初始化方法】

随机初始化

## 【10.选择的loss函数及其导数】

Loss(W) = 1/2 \* (Y-XW)^T(Y\_XW)

Grad(W) = X^T(XW - Y)

## 【11.实验结果和曲线图】

## 超参数选择（η,epoch等）：

## 学习率lr = 0.001

## 迭代次数it\_num = 2000

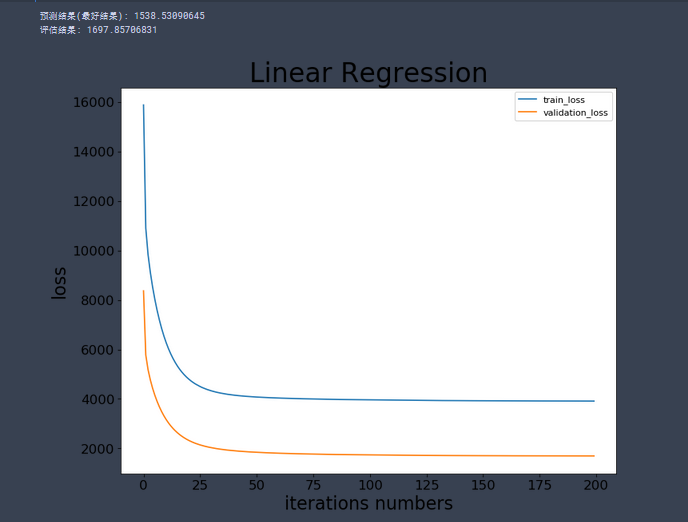
预测结果（最佳结果）：

最佳结果是其闭解，求得为：

评估结果（根据选择的评估方法）：

评估结果为：1697.85706831

## loss曲线图：



## 【12.实验结果分析】

## 1.随着迭代次数的增加，训练集和验证集的loss都逐渐减少到达某个值时，表明对模型进行训练是有效的，能让验证集上的loss也逐渐减少，逐渐接近最优解；

2.而最终评估结果，和预测结果还是有一定的差距，其中部分的原因应该是在训练集产生了一定的过拟合，下一步可以尝试使用正则化等方法来进行改进；

3.另外在调参过程中发现，当学习率过大的时候，会发生在Python浮点数范围内无法收敛的情况，loss不断上升。此处比较疑惑，不确定真的无法收敛，还是只要范围足够大，最后还是能收敛回来的。

## ☆线性分类

## 【8. 选择的评估方法】

## 留出法，取数据集的2/3作为训练集，1/3作为验证集

## 【9. 模型参数的初始化方法】

随机初始化

## 【10.选择的loss函数及其导数】

Loss(W) = ||W||^2/2 + C 

If 1-yiW^TXi>0

Grad(W) = W - CyiXi

if 1-yiW^TXi<=0

Grad(W) = W

## 【11.实验结果和曲线图】

## 超参数选择（η,epoch等）：

## 学习率lr = 0.0001

## 迭代次数it\_num = 3000

惩罚系数 C = 0.1

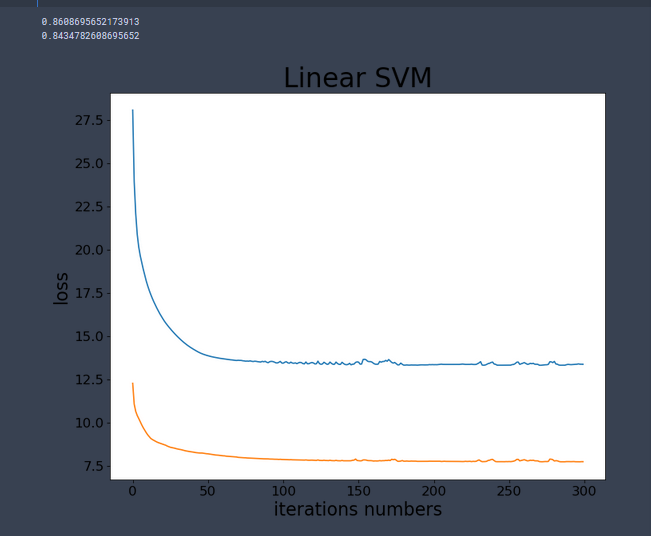
预测结果（最佳结果）：

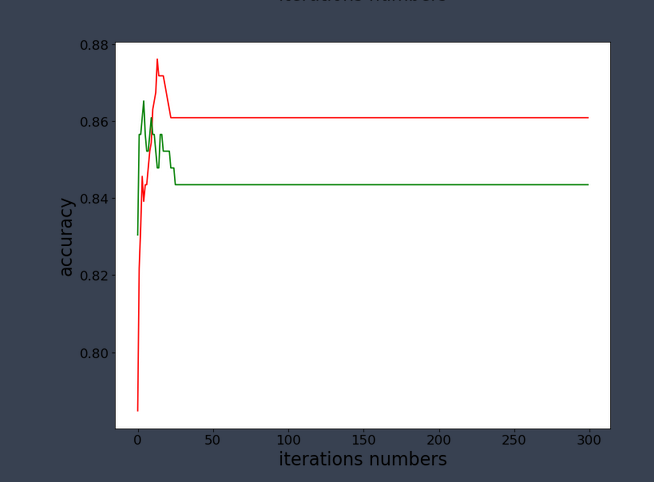
假定为训练集的分类准确率：0.861

评估结果（根据选择的评估方法）：

评估结果为：0.843

## loss曲线图&acurracy曲线图：





## 【12.实验结果分析】

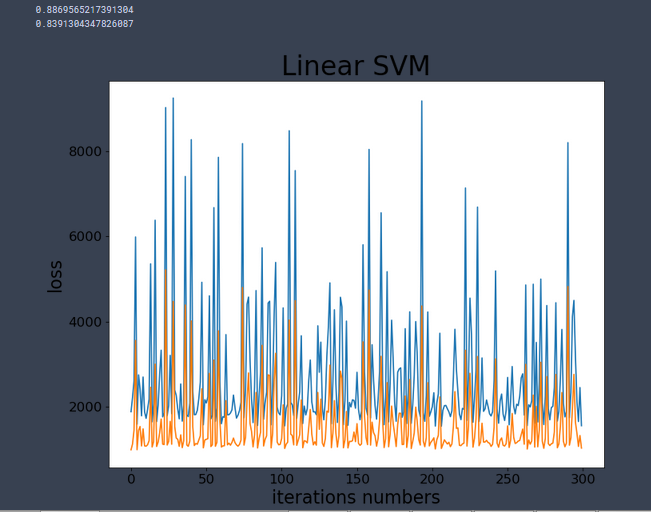
1.和实验1一样，验证集和训练集同趋势的变化，表明了其有效性；

2.准确率的曲线，会发生稍微的下降，应该是hinge loss的原因，

采用软间隔分类，会容忍某些错误的点；

1. 当C过大时，会出现下图状况，究其原因，因为C过大时，优化函数会不遗余力地尽量分类对，难以容忍错误的点，这种情况下，就会出现决策平面的震荡，而无法收敛；
2. 在3中，如果改变学习率，例如缩小10倍，却可以得到收敛的曲线图，

lr和C的关系，仍然不少疑惑，待继续探究。



## 【13.对比线性回归和线性分类的异同点】

线性回归和线性分类的相同点：

1.目标都是拟合一个接近最优的超平面（对于一元线性回归和平面坐标系的线性分类，就是直线）；

2.用梯度下降都能有效地解决；

差异：

1.线性回归的输出是连续的，而线性分类的输出是离散有限的；

2.线性回归是有闭解的，就算用梯度下降，在训练集上，迭代次数足够多，也是可以求出来的；

3.线性回归不会出现，像线性分类这样的震荡现象。

## 【14.实验总结】

## 通过本次实验，虽然只是实现了两个比较简单的机器学习算法，但是真正实现过东西后，学到的东西不再只停留于纸面上后，终于有了一点入门的感觉。

同时，亦发觉，即是只是简单的两个线性的模型，其中也发现了很多暂时还不能理解的问题，自觉自身知识能力之浅薄，必须多加努力。