

**《机器学习》课程实验报告**

**学 院 软件学院**

**专 业 软件工程**

**组 员**  **刘志杰**

**学 号 201530612392**

**邮 箱 791638657@qq.com**

**指导教师**  **吴庆耀**

**提交日期** **2017年 12 月 7 日**

## 1. 实验题目: 逻辑回归、线性分类与随机梯度下降

## 2. 实验时间：2017年 12 月 2日

## 3. 报告人:刘志杰

## 4. 实验目的:1. 进一步理解线性回归和梯度下降的原理。

## 2. 在小规模数据集上实践。

## 3. 体会优化和调参的过程。

## 5. 数据集以及数据分析：线性回归使用的是[LIBSVM Data](https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/)中的[Housing](https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/regression.html" \l "housing" \t "_blank)数据，包含506个样本，每个样本有13个属性。请自行下载scaled版本，并将其切分为训练集，验证集。  线性分类使用的是[LIBSVM Data](https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/" \t "_blank)中的[australian](https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html" \l "australian" \t "_blank)数据，包含690个样本，每个样本有14 个属性。请自行下载scaled版本，并将其切分为训练集，验证集。

## 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 "_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)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，过程详见课件ppt。

## 求得所有样本对Loss函数的梯度。

## 取梯度的负方向，记为。

## 更新模型参数，。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 "_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)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，过程详见课件ppt。

## 求得所有样本对Loss函数的梯度。

## 取梯度的负方向，记为。

## 更新模型参数，。为学习率，是人为调整的超参数。

## 选择合适的阈值，将计算结果大于阈值的标记为正类，反之为负类。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:代码内容见提交链接的.ipynb文件中

线性回归梯度下降

def compute\_gradient(w\_current,X\_trainb,y\_train ,learning\_rate):

w\_grad=0

for i in range(0,len(X\_trainb)):

x=X\_trainb[i,:]

y=y\_train[i]

w\_grad+=-y\*x+(x\*x.transpose())\*w\_current

new\_w=w\_current-(learning\_rate\*w\_grad)

return new\_w

线性分类梯度下降

def compute\_gradient(w\_current,b\_current,X\_trainb,y\_train ,learning\_rate):

w\_grad=0

b\_grad=0

gw=0

gb=0

g=0

for i in range(0,len(X\_trainb)):

x=X\_trainb[i]

y=y\_train[i]

w\_current1=np.mat(w\_current)

g=1-y\*((w\_current1\*x.T)[0,0]+b\_current)

x=X\_trainb[i,:]

y=y\_train[i]

if(g>=0):

gw+=-y\*x

gb+=-y

w\_grad=w\_current1+0.1\*gw

b\_grad=0.1\*gb

new\_w=w\_current1-(learning\_rate\*w\_grad)

new\_b=b\_current-(learning\_rate\*b\_grad)

return new\_w,new\_b

## 8. 模型参数的初始化方法:全零初始化

## 9.选择的loss函数及其导数:

## 线性回归采用

Loss函数：L(w) =1/(2\*m)\*

=1/(2\*m)\*

导数：∂L(w)/∂w=-

=+\*w

线性分类采用

Loss函数：L（w）=（||w||^2/2+C\*）/m

导数：∂f(w,b)/w= w + C\*

∂f(w,b)/b= C\*

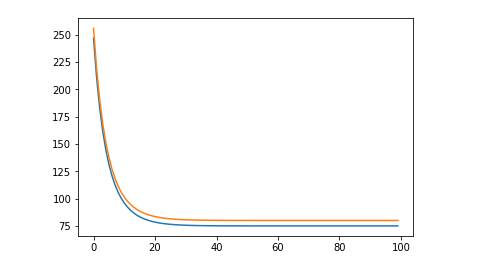
## 10.实验结果和曲线图:（各种梯度下降方式分别填写此项）

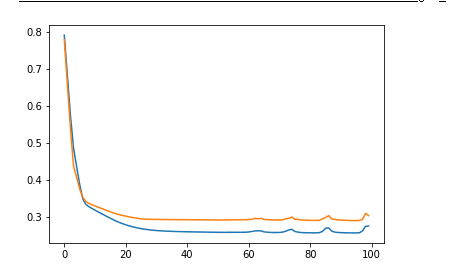
## 超参数选择：线性回归超参数选择学习率选择0.00005，

线性分类超参数选择0.0025

## 预测结果（最佳结果）：线性回归预测最佳结果loss值74.9556

线性分类预测最佳结果loss值 0.2744

**线性回归**loss曲线图：

线性分类曲线图****

## 11.实验结果分析:经过对比，在超参数学习率确定不变的情况下，loss值会根据迭代次数的增加而慢慢降低，最后达到一个比较稳定的值，不会再继续往下下降，其中，最后趋于较为稳定时，训练集的loss会比测试集的loss值稍微低一点，即训练集比测试集更加契合模型一点。

## 12.对比逻辑回归和线性分类的异同点：逻辑是一种广义线性回归，其y=wx+b, 其中w和b是待求参数, 逻辑回归的因变量可以是二分类的，也可以是多分类的，但是二分类的更为常用，所以逻辑回归是一个分类器, 不是真回归，线性分类就是以超平面为决策边界的分类器. 常见的线性分类有逻辑回归与SVM等, 这些个分类算法都不是通过线性回归得到自己的分类超平面的.

## 13.实验总结：这次实验让我手动构建了一个简单的线性回归与线性分类模型，通过调整超参数与迭代次数慢慢地使模型的契合度更高，通过对简单线性回归与分类模型的学习，进一步理解机器学习的基本步骤与调参过程，把课堂上所学的基本理论知识用在实验实践操作上，更加加深了对课堂知识的理解，并从中体会调参的乐趣。