

**模式识别大作业**

题 目 Logistic回归

学 院 信息科学与工程

专 业 信息与通信工程

组 员 马雪娇

指导教师 赵海涛

**完成日期： 2018 年 10 月24日**

**模式识别作业报告——Logistic回归**

组员：马雪娇

模式识别的课程接近尾声，在赵海涛老师的辛勤指导下，我对模式识别的Logistics回归有了一定的了解，并通过完成本次题目对Logistics回归的公式推导和程序实现进行了更进一步的理解，有了更好的运用能力。实现语言为python。对于题目所给特征数据，进行预处理和预测。

1. Logistic回归简介

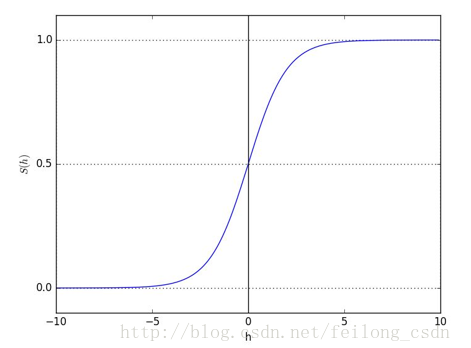
Logistic回归又称logistic回归分析，是一种广义的线性回归分析模型、常用于数据挖掘、经济预测、疾病自动诊断等领域。例如，探讨引发疾病的危险因素，并根据危险因素预测疾病发生的概率等等。

本次实验主要用到两个数据集，是题目所给出的test.csv和train.csv文件。对于test文件中的nan值，经考量，我赋予每一列为-99、-999、0、-1、均值实验，证明均值效果最好。本次实验做的是将点击量进行分析，label列0为未点击，1为点击，我选取了计数特征部分，开始预测。

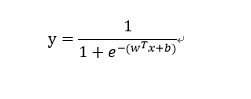
1. 思想技术概述

首先根据课上老师讲解的内容和自己查阅资料，对Logistic回归过程更加了解，简单说就是设定模型wx+b，其中w、b为待求参数，输出标记为y，取{0，1}（本实验未点击为0，点击为1）。使用sigmoid函数如下式：



图像为

嵌套在原来的函数上得到logistic回归的预测函数：



使用梯度下降法求解得：

为了表示方便，设 P（y=1|wx+b）=

则：

P(y=1|x;)=

P(y=0|x;)=

可以写出一个p函数:

P(y|x;)=

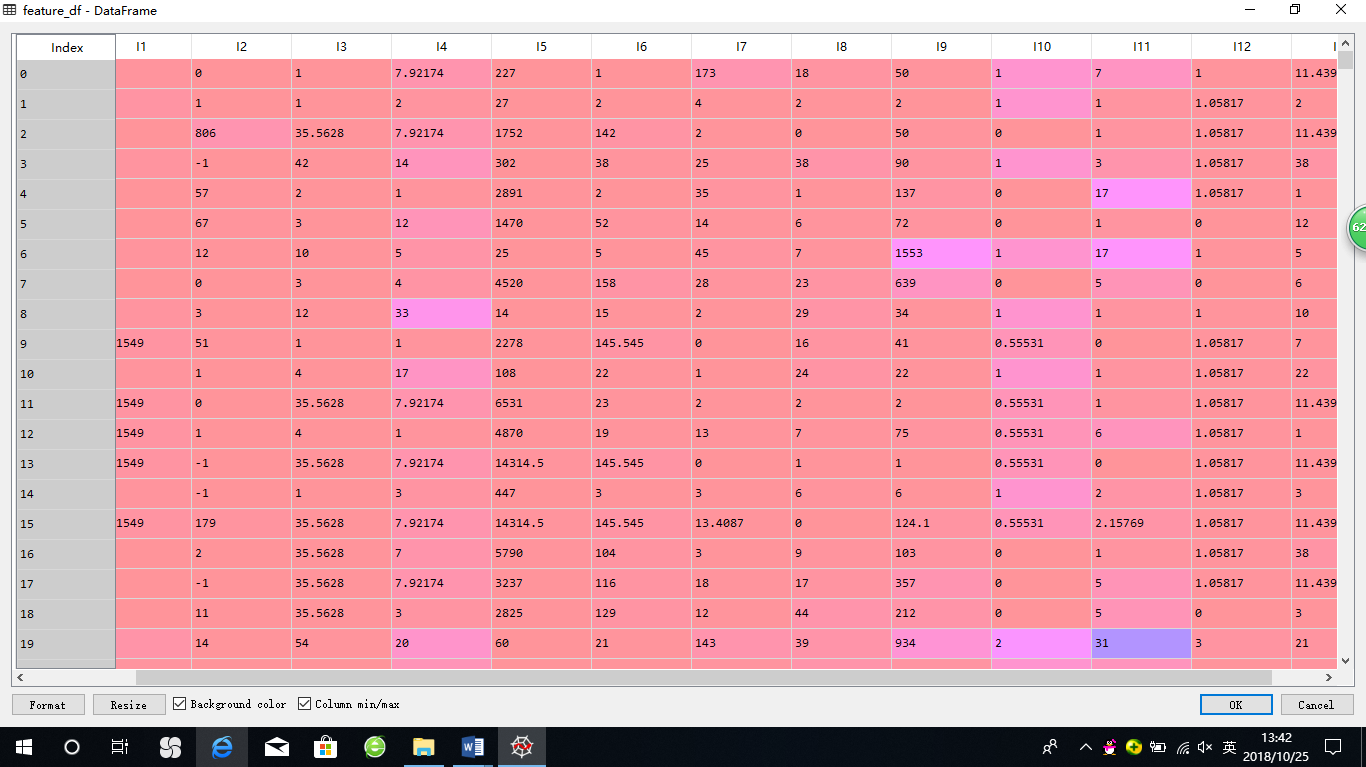
取似然函数得到：

取对数：

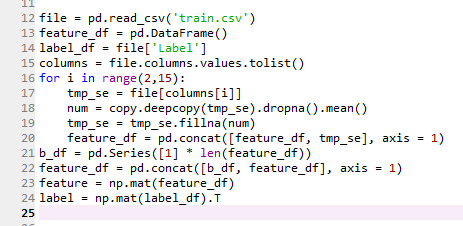
求最大时的值。

1. 数据分析

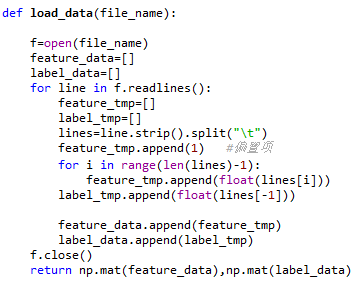
Train文件中用13列数据：

用pandas模块将数据集读取为数据框架（Data Frame），操作如下：

nan部分我取其特征列均值代替（mean（）函数），在预处理之中写入，预处理代码如下：



导入训练数据代码如下：



输入是file\_name,获取字符串训练数据的位置；

输出是future\_data,label\_data,前者为特征，后者为标签。

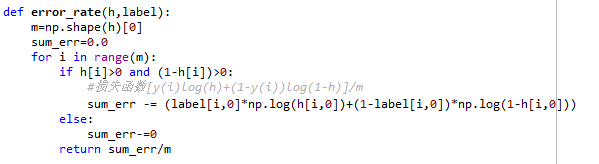
Mat（）将list转化为矩阵形式。

损失函数用J：



和线性回归相似，我们想让数据偏离直线越近越好，也就是损失函数越小越好。

计算执行代码如下：



返回损失函数。

大多数为数学知识，梯度上升优化算法：

def Ir\_train\_bgd(feature,label,maxCycle,alpha):

n=np.shape(feature)[1]

**w=np.mat(np.ones((n,1)))**

**i=0**

**while i<= maxCycle:**

**i+=1**

**h=sig(feature\*w)**

**err=label-h #y(i)-sig(WX(i)+b) 200\*1**

**if i%100 ==0:**

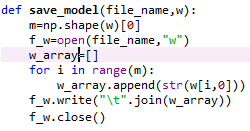
**print("--------iter"+str(i)+",错误率="+str(error\_rate(h,label)))**

**w=w+alpha\*feature.T\*err/np.shape(feature)[0]**

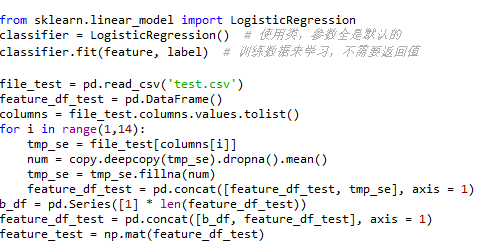
**return w**

其中 maxCycle是最大迭代次数，alpha是学习率，也叫做步长。公式将会迭代进行，每次迭代都将整个数组进行重新计算，并且修正权重。

最后保存模型，记为函数save\_model（）：

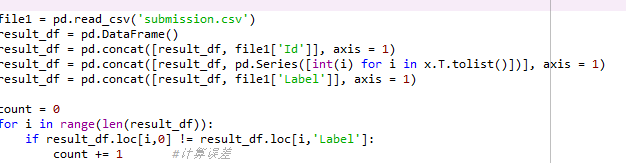


然后用LogisticRegression类预测数据（第一次用要先导入包）：

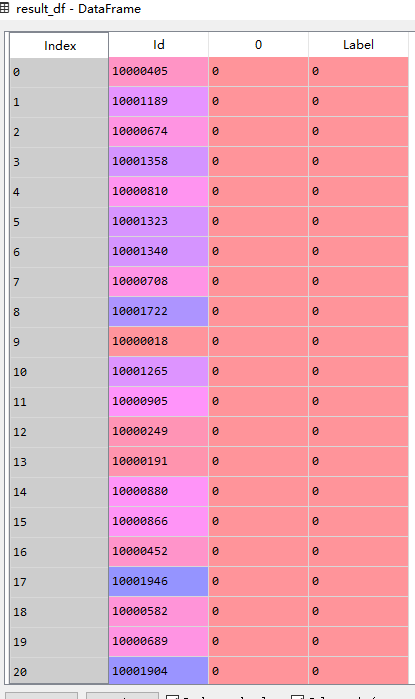


用x = classifier.predict(feature\_test)分类学习，最后返回标记。

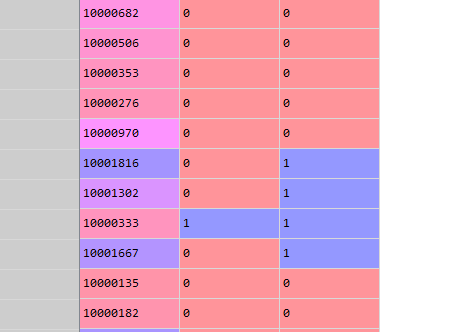
和老师给的标准答案进行对比，并分析误差:



结果图如下：



但有些数据0.1不一致(初步想法是可能由于只做了一部分的特征，如果后面有时间会继续探究)：



由结果可知，误差个数有75个:



四、小组分工

程序设计及编写：马雪娇

程序调试：马雪娇

实验报告：马雪娇

1. 作业总结

由于是第一次接触python并且是用python写出的第一个稍大的程序，在写代码过程中，参考了很多资料，预处理过程中我尝试了将nan赋予几个不同的值（-99，-1，10，9，99），也进行归一化和标准化结果相差不大，都是75-80之间。虽然数据处理不够好，但是觉得自己学到了很多知识，在短短一个月，对python熟悉度大大提高，也对于logistic回归的推导理解的更加透彻。增加了对于一门编程语言的兴趣，感谢老师的传授和指导。

**附：文件说明**

本次附件一共包含有：

1 大作业报告;

2 最终的Python实现程序源码：lr\_homework\_train.py