线性回归与逻辑回归实验报告

任俊彦　520021910546
电子信息与电气工程学院 自动化

目录

1 问题描述

1.1 问题设定

本次作业选择红葡萄酒数据集，设定任务为使用提供的酒品各物理化学成分指标建立线性回归模型及逻辑回归模型预测其酒品质量。

1.2 问题分析

由于给定数据集的酒品质量为 $0-10$ 评分量化，故对于逻辑回归，需要先将其二值化。对于该数据集，二值化有若干种方式，一种方式是将评分不小于最高分 60% 的酒品定义为“合格酒品”，将其它定义为低质量酒品。这种二值方式是否合理，需要观察已知数据的评分分布，该数据集酒品评分分布如下：

评分	0	1	2	3	4	5	6	7	8	9	10
样本数	0	0	0	10	53	681	638	199	18	0	0

对数据集酒品质量评分分布进行分析后，发现小于 6 分的有 744 个酒品样本，大于等于 6 分的有 855 个样本，故这种二值化方式是合理的。实际上，对于此类非二值化样本在较多情况下适合采用 $softmax$ 回归，但出于严格遵守作业任务要求、抵制内卷的考虑，本次作业采用逻辑回归完成任务。实际上，从评分分布上可以发现评分大多集中在 5-7 分间，直接采用 11 分类的 $softmax$ 回归会产生参数冗余和过拟合的问题。

2 数据清洗

在对数据代入模型求解前，需要处理空数据并剔除不合理数据。经分析，红酒数据集中不存在空数据且数据数值区间均合理。

3 算法策略

3.1 矩阵运算

$numpy$ 库中 $array$ 类型可以方便地进行矩阵运算，且其相较一般的循环运算方式有更高的效率，因此设计算法时考虑将线性回归及逻辑回归的各计算式尽可能化成矩阵形式。

对于线性回归，包括一般线性回归及岭回归，其权重均有闭式解，直接使用即可，即

$$w = (X^T X)^{-1} X^T Y \tag{1}$$

$$w = (X^T X + \lambda I)^{-1} X^T Y \tag{2}$$

对于逻辑回归，其似然函数在课件中给出的为求和形式，引入哈达玛积（可以在 $np.array$ 中直接利用运算符 $*$ 实现）并借助 map 函数可以方便地将其化为矩阵乘积形式，对于权重及偏置的梯度亦是如此，详见代码实现。

3.2 五倍交叉验证

评估模型表现使用 K 倍交叉验证 (K-Fold Cross Validation)，其会得到 K 个模型，K 个模型分别在验证集中评估结果，最后的误差 MSE(Mean Squared Error) 取平均即得到交叉验证误差。交叉验证有效利用了有限的数据，并且评估结果能够尽可能接近模型在测试集上的表现，可以做为模型优化的指标使用。本次作业中 K 默认选 5，代码中可调，详见 *main.py*.

3.3 逻辑回归的若干讨论

3.3.1 标准化与归一化

对于具有伸缩不变性的线性回归模型及逻辑回归模型，代入数据前是否进行标准化或归一化不影响最终结果（仅有比例伸缩）。但逻辑回归中涉及到指数运算，观察红酒数据集可以发现酒品样本的物理化学指标中有较大的数，若回归前不进行标准化或归一化，当权重的初值选取的较大时会导致指数溢出，故本次作业中逻辑回归前对各维度数据进行了归一化。

另一方面，相比于直接使用似然函数作为优化目标，使用似然函数除以训练样本总数作为目标函数更为恰当，否则亦容易出现梯度过大导致的指数溢出。

3.3.2 优化算法的选取

由于逻辑回归的优化目标（即其似然函数）为连续可导的凸函数，故其有多种优化方式可以选择，本次作业选取梯度下降法，对于梯度下降法，其每次迭代的步长的选取方式亦有多种，包括固定步长或回溯法等，本次作业选取固定步长法。

3.3.3 初值选取与终止条件

梯度下降的终止条件亦有多种选择，课件中给出的终止条件为似然函数不再上升（上升值小于给定值）为终止条件，另外也可选取梯度近似为 0 做为终止条件（梯度范数小于给定值）。本次作业选取梯度 2 范数小于 0.01 作为终止条件，权重各项及偏置的初值选取均为 0.1。

3.4 类与函数

实际上，实现线性回归及逻辑回归用类来写可以更方便地实现各功能的分块及调度，但本作业中各算法均采用了单个函数的方式来实现，原因是怕和 *sklearn* 过于相似而导致被怀疑抄袭。实际上，*sklearn* 不知道高到哪里去了，并且我其实是考虑到使用单个函数逻辑流畅自然，不用来回跳着看，方便改作业的读者阅读:-)

4 代码实现

4.1 代码结构

线性回归及逻辑回归的函数实现均在 *algorithm.py* 中，*main.py* 写了示例程序，可以直接运行得到本文中所有展示的结果，欢迎尝试。

4.2 函数 API

线性回归及逻辑回归的函数定义如下，其中 lr 为学习率，rk 为正则化系数，$rk = 0$ 时无正则化。data 为训练集，validate 为测试集，返回为一个字典，存储各结果，包括模型参数及表现。

1. def LinearRegression$(data, validate = None, rk = 0.0075)$

2. def LogisticRegression$(data, validate = None, lr = 1, rk = 0, max_iter = 1000)$

关于函数的具体实现，请参见 *algorithm.py*，文件里写有详细的注释，此处不赘述。

5 模型表现

5.1 线性回归与岭回归

5.1.1 正则化与误差

选取不同的正则化系数，将正则化系数设置在 0-0.6 间，进行五倍交叉验证，误差取方根误差的平均值，得到的模型训练集误差与测试集误差如下（蓝线为训练集误差，红线为测试集误差）：

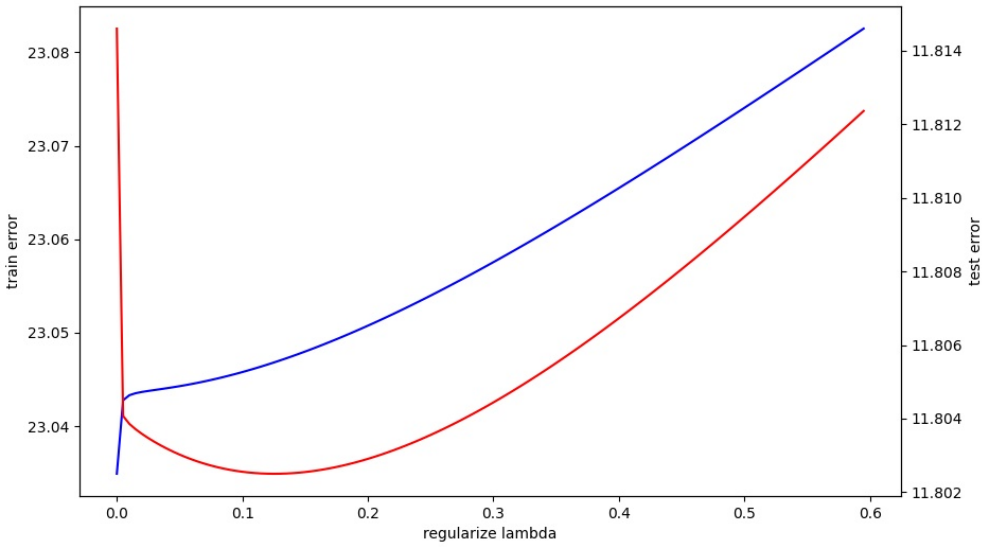

图 1: 误差与正则化系数关系 $(0 < \lambda < 0.6)$

可以发现，当正则化系数较小时，随着 λ 的增加，训练集误差增加，测试集误差减小；但当正则化系数过大时，训练集及测试集误差均增大。将正则化系数的尺度放小，选取最佳的正则化系数为 $\lambda = 0.0075$

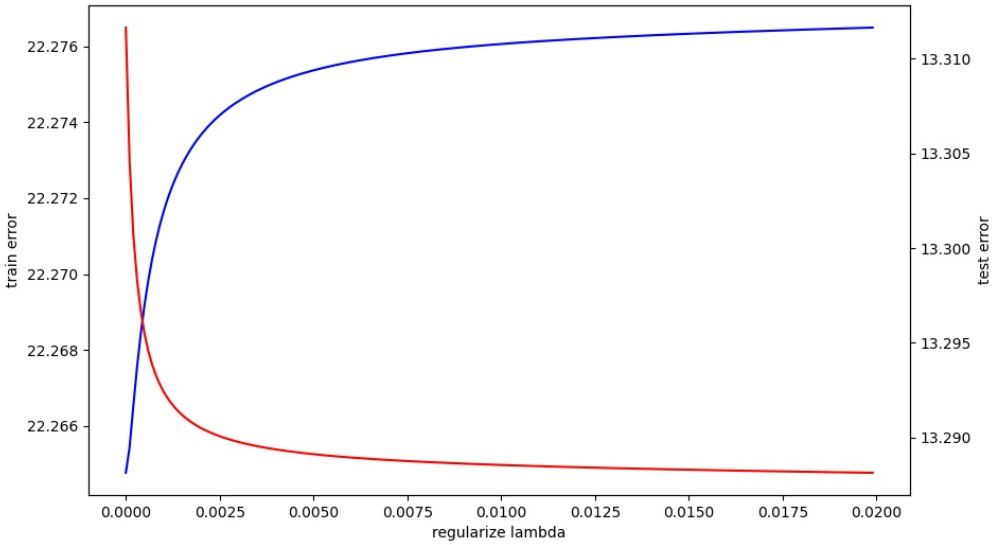

图 2: 误差与正则化系数关系 ($0 < \lambda < 0.02$)

5.1.2 求解结果

对整个数据集按正则化系数 $\lambda = 0.0075$ 进行回归，误差取五倍交叉验证的测试集平均误差，得到最终模型及误差如下表，

方根误差	均方根误差	权重 (含偏置 b)
11.8	0.037	[0.007, -1.097, -1.843, 0.008, -1.898, 0.004, -0.003, 1.054, -0.504, 0.891, 0.294, 3.395]

由结果可知对该数据集使用岭回归模型表现较好，若使用正则化系数为 0 的一般线性回归，其均方根误差亦仅有 0.07，说明该品牌的红酒品质使用线性回归模型进行预测即具有较好的参考价值。

另外，由于未进行归一化，可以观察到特定维度的权重数量级与该维度的数据值的数量级存在一定关联。

5.2 逻辑回归

5.2.1 学习率与迭代次数

学习率的设置会显著影响逻辑回归模型求解的迭代次数。将最大迭代次数设为 1000，正则化系数设置为 0，采用固定步长法，得到学习率与迭代次数间的关系如下图

由图可知，过小或过大的学习率都将导致收敛速度过慢，对于该数据集而言，当学习率设置在 0.5-25 间迭代小于 10 次模型即可完成解算。当学习率设置的较大时，可以设置当迭代次数达到某一阈值后给予一定的学习率衰减。而对于逻辑回归，其优化目标为凸函数，有更多解析判别法来确定学习率衰减的条件，如 sufficient decrease.

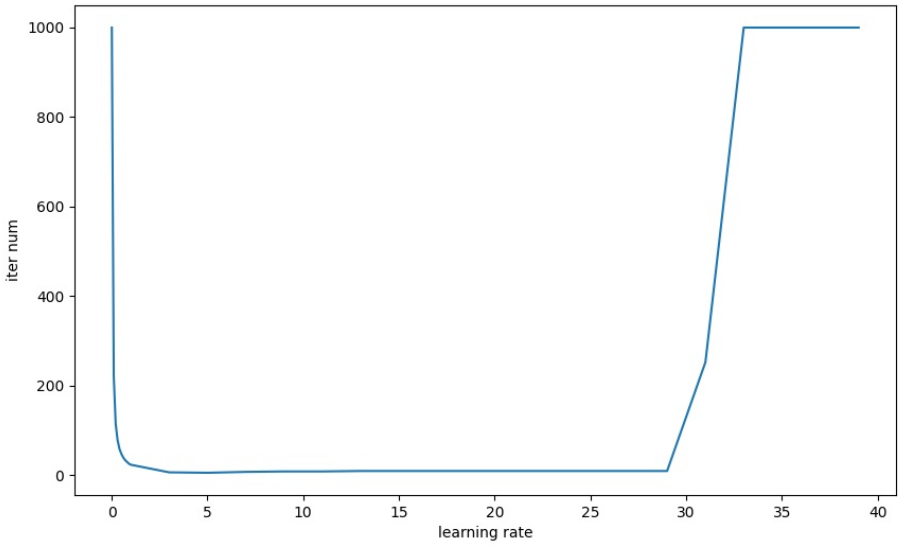

图 3: 学习率与迭代次数

5.2.2 正则化与误差

将逻辑回归的误差定义为

$$error = \sum_i I(\hat{y_i}\ y_i) \tag{3}$$

其中

$$I(x, y) = \begin{cases} 1 & x = y \\ 0 & x \neq y \end{cases} \tag{4}$$

下面探究逻辑回归正则化系数与误差的关系，本次作业采取高斯正则化，即 2 范数正则化。与线性回归相似，首先设置较大的正则化系数区间，选取学习率为 1，得到正则化系数与误差（五倍交叉验证平均误差）关系如图 4.

可以发现，对于该红酒数据集，正则化系数并不影响其最终误差，在测试集上均为 12.6 左右，在训练集上均为 50-51 间。原因应当是在设定任务时合格酒品与低质量酒品的划分标准使得两类酒品在空间上的分布有显著差别，正则化系数的改变引起的权重的改变较小，其二分类结果改变较小，测试集上错误分类的个数保持在 12 左右，且这些错误分类样品应当是模型在该数据分布上的局限性导致的。

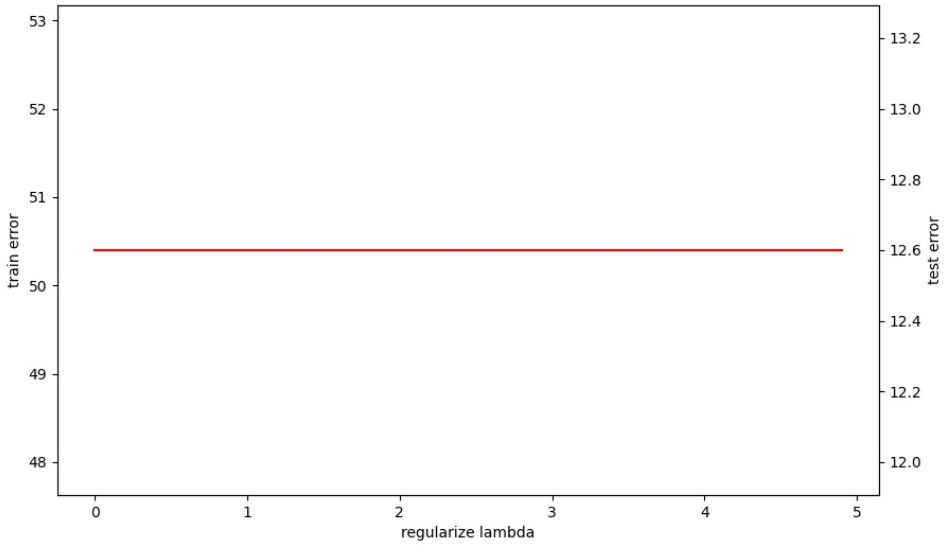

图 4: 逻辑回归误差与正则化系数关系

5.2.3 模型结果

对整个数据集进行无正则化回归，误差取五倍交叉验证的测试集平均误差，得到最终模型及误差如下表，

误差	正确率	权重 w	偏置 b
12.6	96%	[0.565, 0.364, 0.530, 0.234, 0.256 0.427, 0.327, 0.757, 0.675, 0.397, 0.553]	1.484

这可以说明在设定任务时合格酒品与低质量酒品的划分标准十分合理，且在该划分方式下，适合用逻辑回归对酒品进行质量二分类，建立的模型对酒品质量二分类预测具有较高的参考价值。

另外，可以观察到：由于做了归一化，其权重相较于未作归一化的线性回归，在数量级上的差异性显著减小。