第一个作业-逻辑回归实现分类问题

#应用一个场景（取自网上）这里某位老师想用学生上学期考试的成绩（Last Score）和本学期在学习上花费的时间（Hours Spent）来预期本学期的成绩

#数据是我自己编的不具有代表性，涉及数据量比较少没有太大的数据量用来理解逻辑回归做分类问题的感受

在面对这样一个需求，我们首先想到的是线性回归（用来做预测嘛）所以，我取

X = [“Last Score”, “Hours Spent”]，y = “Score”。

我将前11个样本作为训练集，最后3个样本作为测试集。

这样训练出来之后，得到的预测结果为：[55.33375602 54.29040467 90.76185124]，也就说 id 为 12-14 的三个同学的预测分数为55，54和91。

但是我们与实际情况比较发现：

第一个差别比较大，id 为12的同学，明明考及格了，却被预测为不及格。

原因在于我设计的 id 为4的同学，他只用了20小时在学习上，却考出了很好的成绩。

回想一下线性回归的目标函数，所有训练样本对于目标的贡献是平均的，因此，4号同学这种超常的出现，在数据量本身就小的情况下，有可能影响整个模型。

之后我用逻辑回归如何。用逻辑回归的时候，就是预测这个学生本次能否及格了。（即分类问题）

这里我对数据先做一下转换，把具体分数转变成是否合格，合格标志为1，不合格为0，优秀为2（多分类问题）然后再进行逻辑回归：

[1,0,1]

很明显判断是正确的

LR这里我还额外做了面对的是多分类问题：样本标签的枚举值多于2个，还能用 LR 吗？

答案当然是可以的。我们可以理解为把二分类问题分成多次来做。

假设你一共有 n 个标签（类别），也就是说可能的分类一共有 n 个。那么就构造 n 个 LR 分类模型，第一个模型用来区分 label\_1和 non-label \_1（即所有不属于 label\_1 的都归属到一类），第二个模型用来区分 label\_2 和 non-label \_2……, 第 n 个模型用来区分 label\_n 和 non-label \_n。

测试的时候，每一个输入数据都被这n 个模型同时预测。最后哪个模型得出了 Positive 结果，就是该数据最终的结果。

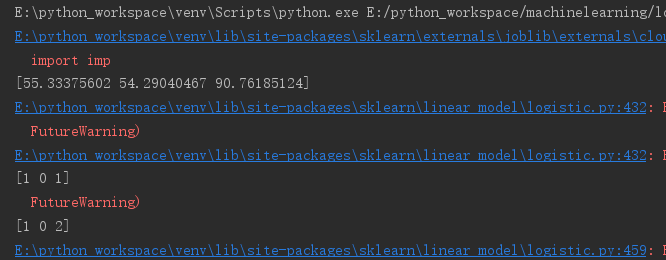
如果有多个模型都得出了 Positive，那也没有关系。因为 LR 是一个回归模型，它直接预测的输出不仅是一个标签，还包括该标签正确的概率。那么对比几个 Positive 结果的概率，选最高的一个就是了。

例如，有一个数据，第一和第二个模型都给出了 Positive 结果，不过 label\_1 模型的预测值是0.95，而 label\_2 的结果是0.78，那么当然是选高的，结果就是 label\_1。

其中的原理来挺麻烦，好在 sklearn 已经为我们处理了多分类问题，我们用 sklearn 来做多分类的时候，只是需要把 y 准备好，其他的，都和做二分类一样就可以了。

这里我是这么设置的：学生的本次成绩是优秀（>=85），及格，还是不及格。我们就在处理 y 的时候给它设置三个值：0 （不及格），1（及格）和2（优秀），然后再做 LR 分类就可以了。

[1 0 2]这里结果还是准确的



图一 第一个homework的结果截图

第二个作业，用distance这样的方法来做分类问题

由于这里笔者这段时间在学CSN231的课程所以这里就写得稍微详细一些，截止我学到的……

L1距离



这个跑出来的时间要很长时间



L2距离





所以我们可以观察到准确率稍微有些不同，是由于：

在这里使用了np.sqrt，但是在实际中可能不用。因为求平方根函数是一个单调函数，它对不同距离的绝对值求平方根虽然改变了数值大小，但依然保持了不同距离大小的顺序。所以用不用它，都能够对像素差异的大小进行正确比较。

L1和L2比较。比较这两个度量方式是挺有意思的。在面对两个向量之间的差异时，L2比L1更加不能容忍这些差异。也就是说，相对于1个巨大的差异，L2距离更倾向于接受多个中等程度的差异。L1和L2都是在p-norm常用的特殊形式。

p-范式，

这里就是像二范式就是欧式距离

先取平方，再开方