哈尔滨工业大学计算机科学与技术学院

实验报告

课程名称： 机器学习

课程类型：主修

实验题目： Logistic Regression

学号：1161000309

姓名： 高靖龙

1. 实验目的

理解logistic regress原理，惩罚项，以及在不同的前提下效果。

二、实验要求及实验环境

**实验要求：**

理解逻辑回归模型，掌握逻辑回归模型的参数估计算法。

实现两种损失函数的参数估计（1，无惩罚项；2.加入对参数的惩罚），可以采用梯度下降、共轭梯度或者牛顿法等。

**验证：**

1.可以手工生成两个分别类别数据（可以用高斯分布），验证你的算法。考察类条件分布不满足朴素贝叶斯假设，会得到什么样的结果。

2. 逻辑回归有广泛的用处，例如广告预测。可以到UCI网站上，找一实际数据加以测试。

**实验环境：**

Manjaro linux

三、设计思想（本程序中的用到的主要算法及数据结构）

**1．算法原理**

**原理：**

对01分类问题。

Logistic regress是典型的判别式模型，受naive bayesian的启发我们将P(Y|X)假设为sigmod函数，使用MLE进行估计。估计过程中由于P(X,Y|W)难测，所以使用P(Y|X,W)代替，即使用MCLE最大条件似然估计。

因此实现了对01分类问题的解决。这样求出的分类面是线性的。

推导：

**计算手段：**

**A.使用牛顿法具体求解似然函数**

牛顿法是对最速下降法的改进，是对目标点的二阶导数接近。

**迭代公式：**

**算法的实现**

使用math、numpy库完成矩阵操作。

值得一提的是由于黑塞矩阵值较小，求逆后会很大，容易溢出。所以对某些样本（实测大概8%的几率）会不可用。

具体的解决方法是针对初始化的W进行改进，初始化为全0能有效防止溢出。

其余部分即按照上述算法流程编写。

实验中尝试编写了拟牛顿法，不过有些问题，尚未解决。实验中仅使用牛顿法求解。

1. 使用梯度下降法计算

梯度下降法直观的向梯度反向近似，认为设定学习率。在上次的实验中已经运用，不再赘述原理。其迭代公式隐含在牛顿法算法中。不作为本次实验的重点。

1. 逆牛顿法

逆牛顿法使用迭代方式近似hessian矩阵或其逆矩阵，是对牛顿法的优化。

1. 实验结果与分析

**1、对下列4种样本进行基于统计的对比分析**

1. 满足标准前提假设的样本：各个特征满足相互独立的一维高斯分布、均值取值与特征和y有关、方差仅与特征有关
2. 方差也与y有关
3. 特征不相互独立
4. 不是一维高斯分布，这里测试二项分布和beta分布

**评价标准选择：**

**准确率：TP+TN/TP+TN+FP+FN**

**经测试，accuracy、precision、recall在实验数据量上都能较明显的表现出分类器差异（数据量不大，100以内），因此使用更能表现全局的accuracy作为评价标准。**

**A.以下是对满足假设的样本，准确率在不同惩罚项下的表现：**

可以看到随着惩罚项的增加，性能先上升后下降，虽然W的绝对值确实下降了，但是惩罚项的作用在实验数据对准确率的影响不是很大，改善不明显，猜测是数据生成的不够典型。在下一部分会对这点进行重点测试。

|  |  |  |
| --- | --- | --- |
| punishment | accuracy\_in\_Train | accuracy\_in\_Test |
| 0 | 0.962797 | 0.854762 |
| 1 | 1 | 1 |
| 2 | 1 | 1 |
| 3 | 1 | 1 |
| 4 | 0.97861 | 0.984127 |
| 5 | 0.962567 | 0.952381 |
| 6 | 0.925134 | 0.904762 |
| 7 | 0.925134 | 0.904762 |
| 8 | 0.925134 | 0.904762 |
| 9 | 0.925134 | 0.904762 |

**B．以下是50样本，3特征，100轮次数据生成测试下不同前提的平均训练集准确率和测试集准确率，除了不独立的情况，其他三种基本上都在千分位波动，有参考价值。**

|  |  |  |
| --- | --- | --- |
| **类型** | **accuracy\_in\_Train** | **accuracy\_in\_Test** |
| 正常 | 0.821194 | 0.818560 |
| 方差 | 0.87321 | 0.824737 |
| 不独立 | 0.801071 | 0.804737 |
| 二项分布 | 0.805450 | 0.802851 |
| Beta分布 | 0.774520 | 0.778476 |

我们看到虽然四种类型的有所差异，但是差别都不是很大，基本控制在10%以内。但能发现满足假设时性能会好一点。

这种现象经过多次调整数据生成参数，发现是比较稳定出现的，有参考价值。

**理论解释：**

Logistic回归的前提要求比朴素贝叶斯低的多，实际上对W来说，由于关系是隐含在sigmod函数中的，此发射函数实际上能够学习到X各个特征之间的关联，因此在不满足前提的情况下仍然能够给出相对正确的结果。

但是满足前提使得X特征间关系清晰，能够使得分类器学习到更好的决策面。

**2、针对UCI Heart Disease Data Set 进行数据预处理，并使用logistic regress进行分类，并考察性能。**

**数据简介：**

此数据库包含76个属性，但所有已发布的实验均指使用其中14个属性的子集。特别是，克利夫兰数据库是ML研究人员

迄今为止唯一使用过的数据库 。“目标”字段是指患者中存在心脏病。

**数据观察：**

11个维度，有很多缺失值。Y值进行了排序

**特征工程：**

略过，使用所有特征进行学习，毕竟特征总数不大。

**预处理：**

针对缺失值，我们简单的使用众数进行代替。

**数据拆分：**

X\_train, X\_test, y\_train, y\_test = train\_test\_split(XX, Y,test\_size=0.25, random\_state=0)

然后直接使用logistic regress进行分析

|  |  |  |
| --- | --- | --- |
| **punishment** | **accuracy\_in\_Train** | **accuracy\_in\_Test** |
| 0 | 0.854545 | 0.837838 |
| 0.1 | 0.868182 | 0.837838 |
| 0.2 | 0.868182 | 0.837838 |
| 0.3 | 0.863636 | 0.851351 |
| 0.4 | 0.863636 | 0.851351 |
| 0.5 | 0.863636 | 0.864865 |
| 0.6 | 0.863636 | 0.864865 |
| 0.7 | 0.863636 | 0.864865 |
| 0.8 | 0.863636 | 0.864865 |
| 0.9 | 0.863636 | 0.864865 |
| 1 | 0.863636 | 0.864865 |
| 1.1 | 0.863636 | 0.864865 |
| 1.2 | 0.863636 | 0.851351 |
| 1.3 | 0.863636 | 0.851351 |
| 1.4 | 0.859091 | 0.851351 |
| 1.5 | 0.859091 | 0.851351 |
| 1.6 | 0.854545 | 0.851351 |
| 1.7 | 0.854545 | 0.851351 |
| 1.8 | 0.854545 | 0.851351 |
| 1.9 | 0.854545 | 0.851351 |

再一次我们发现惩罚项能提升logistic regress性能，在次样本中大概提升3%，影响不是很大。同时最优惩罚项大约在0-2范围内。

另一方面我们看到logistic regress在现实数据上表现良好，即使我们没有仔细的处理数据选择合适的特征表达方式。这和part 1中结论相符合：logistic regress性能并不强依赖于前提假设！对X没有特别强的要求，即使是一般的X也能表现良好。

1. 结论

针对朴素贝叶斯要求的前提假设，logistic regress实际上要求并不强烈，即使是一般的数据也能表现良好。但是满足假设的数据上logistic regress也会表现的更好。

惩罚项系数对于logistic regress有限，惩罚项系数取值在10以内。

牛顿法计算迭代次数少，但是迭代速度很慢。不过相对梯度下降来说是优的。

六、参考文献

1. Boyd, S., & Vandenberghe, L. 2011. *Convex optimization*. Cambridge: Cambridge Univ. Pr.
2. Newton's method. 2018. *En.wikipedia.org*. https://en.wikipedia.org/wiki/Newton%27s\_method, October 25, 2018.
3. Overview — NumPy v1.15 Manual. 2018. *Docs.scipy.org*. https://docs.scipy.org/doc/numpy/, October 25, 2018.
4. Search — pandas 0.23.4 documentation. 2018. *Pandas.pydata.org*. http://pandas.pydata.org/pandas-docs/stable/search.html?q=get\_dummies&check\_keywords=yes&area=default, October 25, 2018.
5. UCI Machine Learning Repository: Heart Disease Data Set. 2018. *Archive.ics.uci.edu*. https://archive.ics.uci.edu/ml/datasets/heart+Disease, October 25, 2018.

七、附录：源代码（带注释）