Pattern Recognition Assignment 1

Logistic Regression

2031561 郭超政

1.模型概述

逻辑回归（Logistic Regression）模型的基本形式为：

$$y = \frac{1}{1 + e^{-(\beta^T x)}} \tag{1}$$

在使用逻辑回归模型解决二分类问题时可将分类概率表示为：

$$p(y = 1|x;\beta) = p_1(x;\beta) = \frac{e^{\beta^T x}}{1 + e^{\beta^T x}} \tag{2}$$

$$p(y = 0|x;\beta) = p_0(x;\beta) = \frac{1}{1 + e^{\beta^T x}} \tag{3}$$

为了求解模型的参数，可以通过极大似然法进行估计，则根据(2)(3)得到模型的等价最小化似然函数：

$$L(\beta) = \Sigma_{i=1}^{m}\left(-y_i\beta^T x_i + \ln\left(1 + e^{\beta^T x_i}\right)\right) \tag{4}$$

通过牛顿法来求解似然函数的最优解：

$$\beta^* = \arg\min_{\beta} L(\beta) \tag{5}$$

根据牛顿法以及(4)，可以得到其迭代更新的公式：

$$\beta^{t+1} = \beta^t - \left(\frac{\partial^2 L(\beta)}{\partial\beta\,\partial\beta^T}\right)^{-1}\frac{\partial L(\beta)}{\partial\beta} \tag{6}$$

$$\frac{\partial^2 L(\beta)}{\partial\beta\,\partial\beta^T} = -\Sigma_{i=1}^{m} x_i\left(y_i - p_1(x;\beta)\right) \tag{7}$$

$$\frac{\partial L(\beta)}{\partial\beta} = \Sigma_{i=1}^{m} x_i x_i^T p_1(x;\beta)\left(1 - p_1(x;\beta)\right) \tag{8}$$

2.模型实现

基本函数定义：

根据式(4)定义模型的似然函数：

```
# Likelihood function of logistics regression
def likelihood(data, label, w):
    likelihood = -label * (np.dot(data, w)) +
                            np.log(1 + np.exp(np.dot(data, w)))
    return np.sum(likelihood)
```

根据式(7)(8)，定义相应的梯度以及海森矩阵计算函数：

```
# Gradient of likelihood function
def gradient(data, label, w):
    gradient = np.zeros(w.shape[0])
    for x,y in zip(data,label):
        p1 = np.exp(np.dot(w, x)) / (1 + np.exp(np.dot(w, x)))
        gradient += x * (y - p1)
    return -gradient

# Hessian matrix of likelihood function
def hessian(data, label, w):
    hessian = np.zeros((w.shape[0], w.shape[0]))
    for x,y in zip(data,label):
        p1 = np.exp(np.dot(w, x)) / (1 + np.exp(np.dot(w, x)))
        hessian += np.reshape(x, (x.shape[0], 1)) * x * p1 * (1-p1)
    return hessian
```

牛顿法求解过程实现：

训练阶段模型的输入为训练数据以及对应的数据标签，训练数据 data 的格式为$\mathrm{m} \times \mathrm{n}$的矩阵，包含 m 个数据样本，每个样本包含 n 个属性值，属性值需要均为数值类型；

根据式(1)中的定义，将线性模型$\mathrm{w}^{\mathrm{t}}x + b$简化为$\beta^{\mathrm{T}}x$，需要将数据矩阵拓展为$(x;1)$，则模型输入的数据矩阵尺寸为$\mathrm{m} \times (\mathrm{n}+1)$；

数据标签 label 为对应的 n 维数组，标签对应二分类的结果，使用 0，1 表示；

对应的模型系数 coef 为 n+1 维向量，初始化为零向量；

通过循环迭代更新系数 coef 估计最优解，迭代的过程通过定义最小步长 e 以及最大迭代次数 max_it 来控制；

迭代结束后返回得到的模型系数；

```python
# Solve logistics regression with newton's method
def logistic_regression(data, label, e, max_it=100):
    # Expand data matrix
    data = np.c_[data, np.ones(data.shape[0])]
    # Initialize coeffiencts of model y = 1 / (1 + e^(-wx))
    coef = np.zeros(data.shape[1])
    # Initialize step norm
    d_norm = np.inf
    # Number of iteration
    it_count = 0

    while d_norm > e and it_count < max_it:
        print("Step:", it_count, "Likelihood:", likelihood(data, label, coef))
        d = np.dot(np.linalg.pinv(hessian(data, label, coef)),
                                  gradient(data, label, coef))
        # Update coefficient
        coef = coef - d
        # Calculate step norm
        d_norm = np.linalg.norm(d)
        it_count += 1

    return coef
```

模型评估部分实现：

使用计算的到的模型系数预测未知数据：

```python
# Predict novel data with trained coefficient
def predict(data, w):
    data = np.c_[data, np.ones(data.shape[0])]
    p0 = 1 / (1 + np.exp(np.dot(data, w)))
    p1 = 1 - p0
    res = np.c_[p0,p1]
    res = np.argmax(res, axis=1)
    return res
```

计算预测结果的准确率：

```python
# Calculate the percision of predicted result
def score(predict, ground_truth):
    count = (predict == ground_truth).astype(int).sum()
    return (count/len(predict))
```

3.模型测试

模型的测试选择了 UCI 上的 Breast Cancer 数据集以及 Abalone 数据集：

Breast Cancer 数据集：
数据集描述的是不同的乳腺癌病例分类，病例分为良性以及恶性两个类别，分别表示为 0，1；数据包含了团块厚度等 9 个用于描述病例状况的数值属性，属性均为数值类型，原始数据中还包含一列样本 ID，实验中弃之不用；数据样本数量为 699；

数据处理：
数据中包含少数的缺失值，共 16 个样本存在属性缺失，由于数量较少，将缺失样本直接去除，清理后的数据包含 683 个样本；

数据属性值的数值分布较为平均，且不同属性的数值量纲基本相同，固不作更多的数据预处理；

数据随机打乱后，按照 4:1 的比例划分为训练集及测试集，各包含 546 及 137 个样本；

数据属性值大致分布：

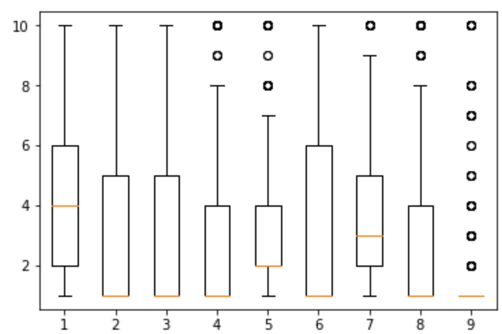

图 1 - Breast Cancer 数据集属性取值箱线图

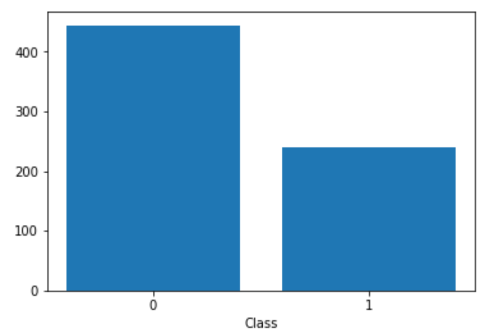

图 2 - Breast Cancer 数据集类别数量分布

模型运行结果：

经过 9 轮迭代后模型收敛，得到的模型系数为：

```
[ 0.55860282,  0.06059213,  0.26964914,  0.39501854,  0.04491559,
  0.41040066,  0.34391083,  0.212355  ,  0.44955754, -9.68085486]
```

测试集中预测结果准确率为：`0.97810`

模型可视化

选择数据集中的前两个属性绘制相应的分类边界：

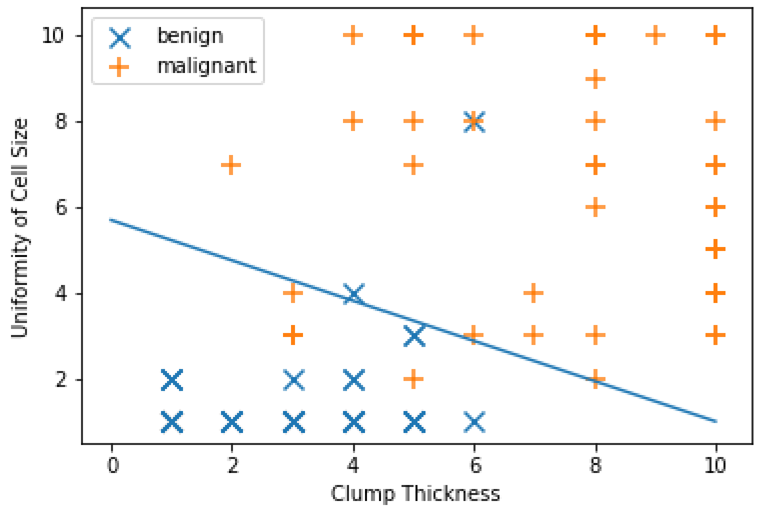

图 3 - Breast Cancer 逻辑回归分类边界

Abalone 数据集：

数据集描述了不同年龄段的鲍鱼个体，通过体重、身长等物理数据来推断鲍鱼的年龄；数据集中对鲍鱼年龄的描述为整数数值，为了将其构建为一个二分类任务，将年龄字段划分为两个类别，≤ 10以及> 10，分别使用 0，1 表示；样本数量共 4177 个，属性数量 8，其中包含一个性别属性为 Nominal 类型，其余属性均为数值类型，实验中仅选取其中 7 个数值类型的属性；

数据处理：

数据无缺失值，且无明显异常数值，同样按照 4:1 的比例划分训练集以及测试集，得到训练集大小 3341，测试集大小 836；

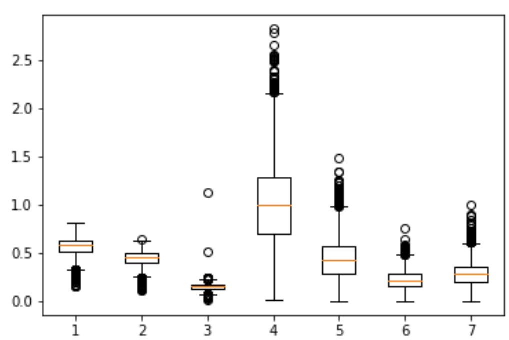

图 4 - Abalone 数据集属性取值箱线图

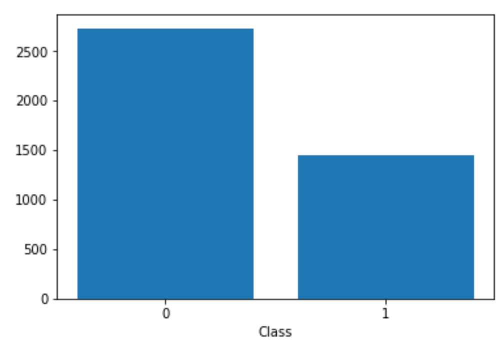

图 5 - Abalone 数据集类别数量分布

模型运行结果：

经过 5 轮迭代后模型收敛，得到的模型系数为：

```
[ -4.37363705,   2.5186944 ,   2.03648743,  18.36825708,
  -18.90044524,  -3.0565954 ,   9.01004543,  -1.91022282]
```

测试集中预测结果准确率为：0.76435

模型可视化：

选择数据集中的两个属性绘制相应的分类边界：

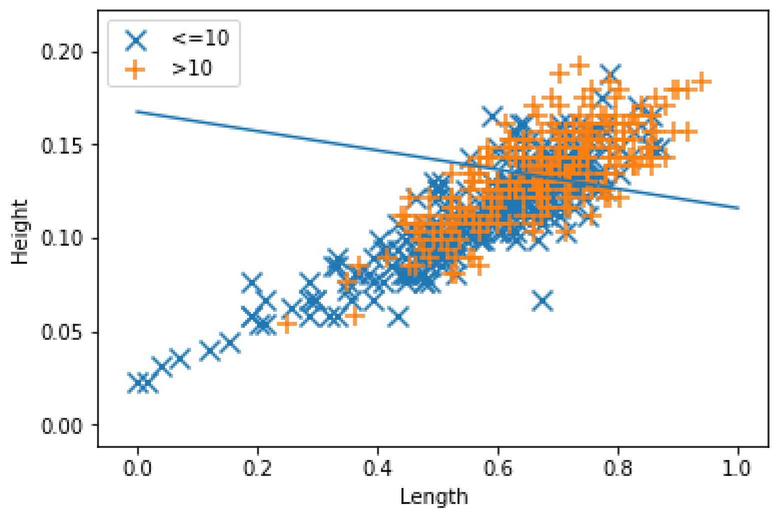

图 6 - Abalone 逻辑回归分类边界

4.结果分析

逻辑回归优势：
逻辑回归的过程简单，计算过程也相当迅速，能够在极短时间内得到一个粗略的模型，且模型的可解释性较强，能够较为清晰的展示不同属性之间的关系；

牛顿法缺陷：
在使用牛顿法求解逻辑回归模型时，因为涉及对 Hessian 的求解，在数据规模较大的时候 Hessian 矩阵的尺寸也会相应变大，增加计算的复杂程度；

线性表达能力较弱：
逻辑回归是广义线性模型，其表达能力较弱，在数据属性间关系较为复杂的状况下无法准确描述属性之间的关系，导致模型的效果较差；