使用 Python 实现对数几率回归模型

计算机 1602 陈高欣 1611640212

一、问题描述

下载 Iris 数据集，并读懂数据集的相关说明，了解数据格式。编程实现对数几率回归模型，并对 Iris 数据集进行分类以验证模型的效能：

(1). 将数据集的 50%作为训练集，50%作为测试集，检验模型在测试集上的分类正确率；

(2). 将数据集的 70%作为训练集，30%作为测试集，检验模型在测试集上的分类正确率；

(3). 将数据集的 90%作为训练集，10%作为测试集，检验模型在测试集上的分类正确率。

二、数据集描述

Iris 数据集包含 150 个数据集，每条数据共有 5 个属性，前 4 个属性依次为&Sepal_Length（花萼长度），&Sepal_Width（花萼宽度），&Petal_Length（花瓣长度），Petal_Width（花瓣宽度），单位均为厘米。最后一个属性&Species（种类）共有 3 类，分别为 Iris Setosa（山鸢尾）、Iris Versicolour（杂色鸢尾），以及 Iris Virginica（维吉尼亚鸢尾），每类 50 个数据。

图 2.1、鸢尾花生物结构

	sepal_len	sepal_width	petal_len	petal_width	species
0	4.9	3.0	1.4	0.2	Iris-setosa
1	4.7	3.2	1.3	0.2	Iris-setosa
2	4.6	3.1	1.5	0.2	Iris-setosa
3	5.0	3.6	1.4	0.2	Iris-setosa
4	5.4	3.9	1.7	0.4	Iris-setosa

图 2.2、数据集前 5 行数据

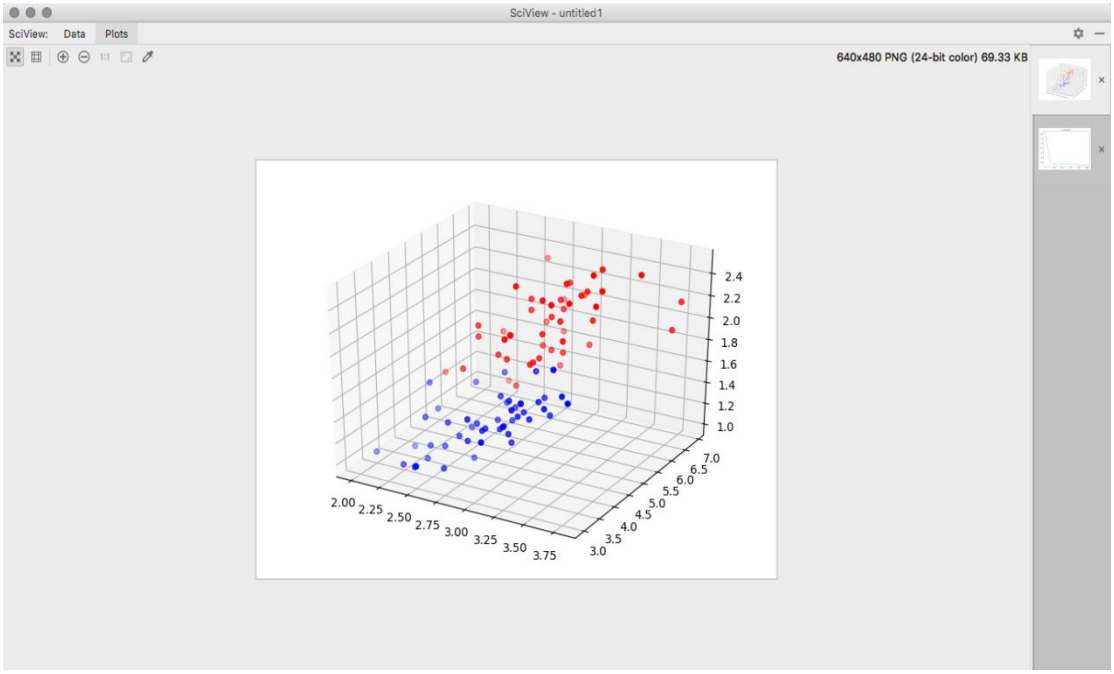

图 2.3、matplotlib 数据分布可视化展示

三、实验结果

3.1、模型函数

已知常见的线性回归模型：$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_n x_n$$

引入了 logit 变换：$$\text{logit}(\text{p}) = In\left(\frac{p}{1-p}\right) = \theta_0 + \theta_1 \text{x}_1 + \theta_2 \text{x}_2 + \ldots + \theta_n \text{x}_n$$

推导变换得出：$$\text{p} = \frac{1}{1 + e^{-(\theta_0 + \theta_1 \text{x}_1 + \theta_2 \text{x}_2 + \ldots + \theta_n \text{x}_n)}}$$

激活函数：$$P(y=1 \mid X, W, b) = \sigma(WX + b) = \frac{1}{1 + e^{-(WX+b)}}$$

损失函数：$$\text{l} = -\frac{1}{m}\sum_{i=1}^{m}(y(i)\log(\sigma(WX+b)) + (1 - y(i))\log(1 - \sigma(WX+b)))$$

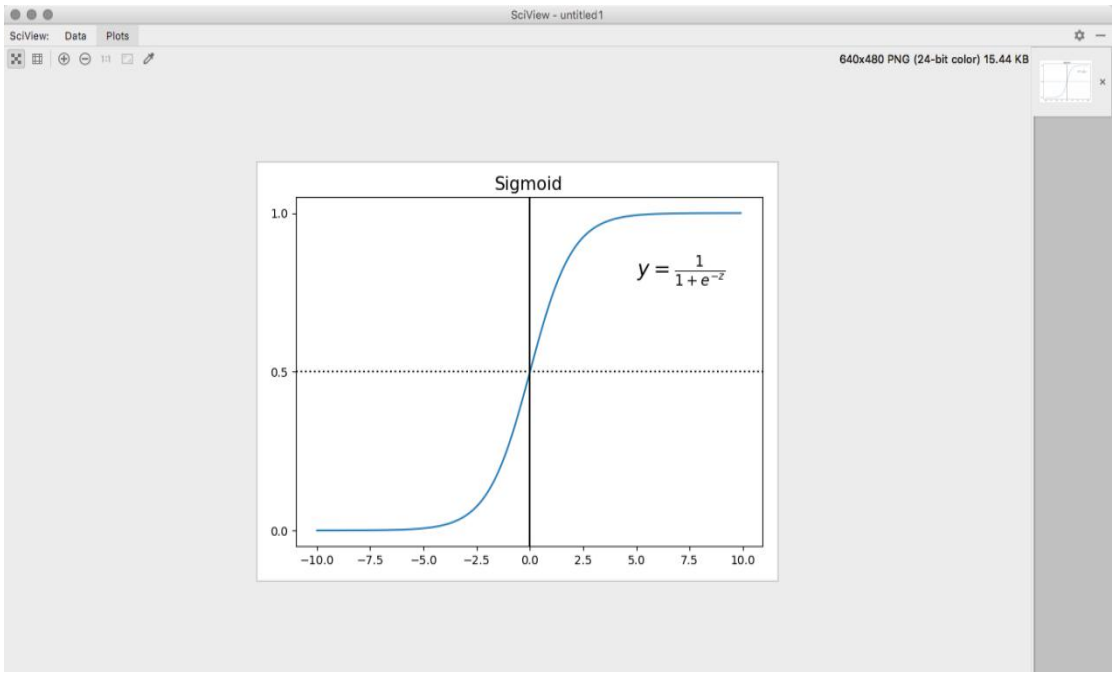

图 3.1、绘制函数曲线

3.2、模型训练与测试

模型训练最大迭代次数为 2000，学习率为 0.001，依次将数据集的 50%作为训练集，50%作为测试集，训练后正确率为 66%；70%作为训练集，30%作为测试集，训练后正确率为 82%；90%作为训练集，10%作为测试集，训练后正确率为 88%。损失函数结果依次如下：

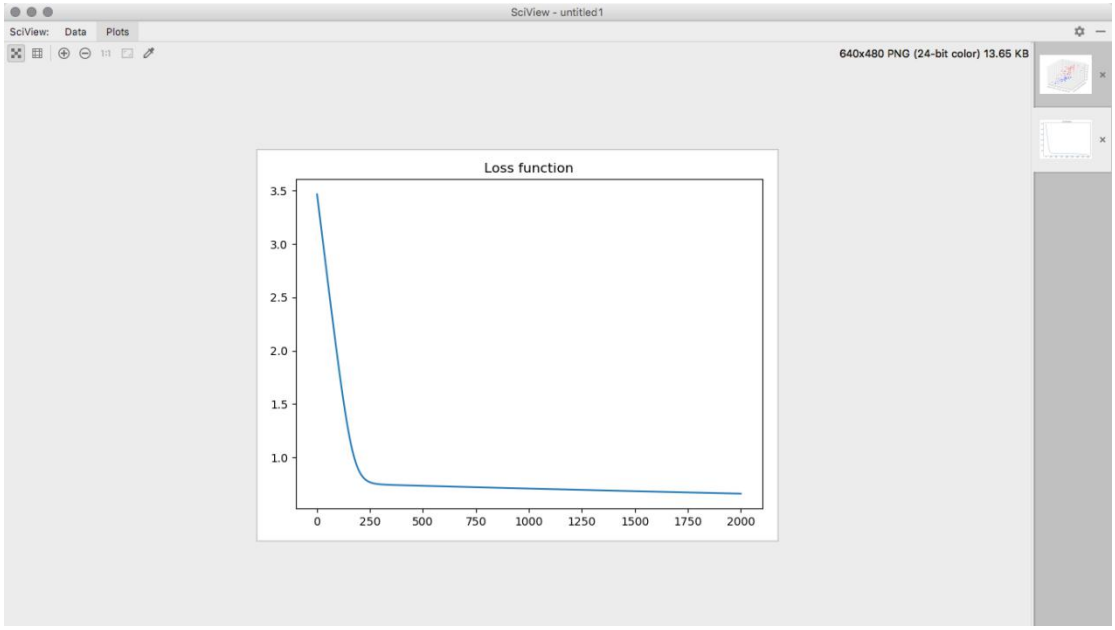

图 3.2、50%训练集时损失函数曲线图

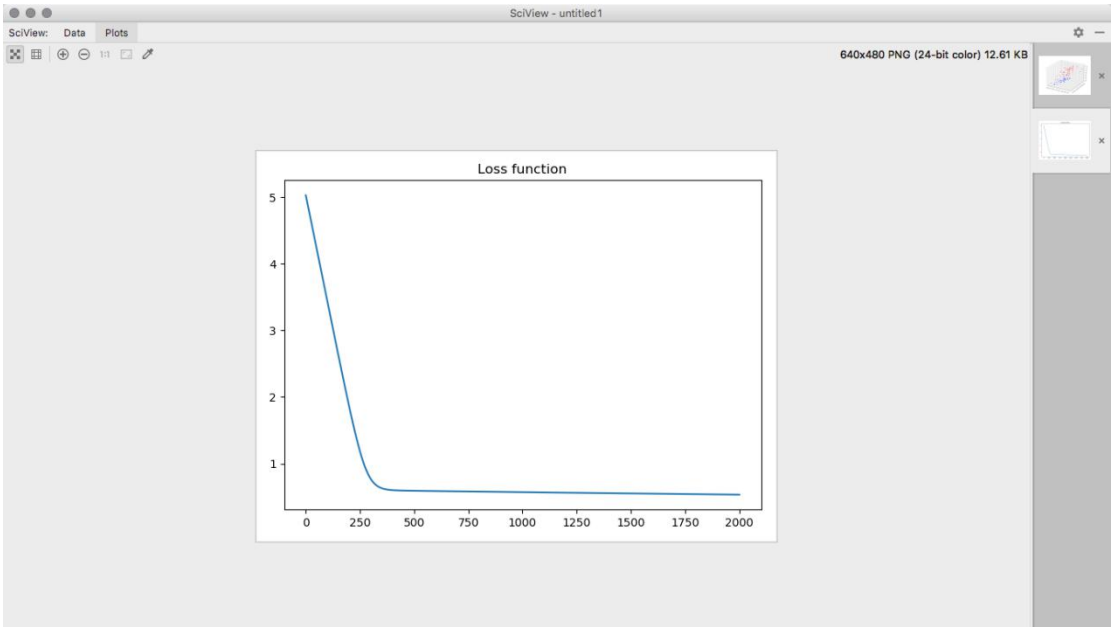

图 3.3、70%训练集时损失函数曲线图

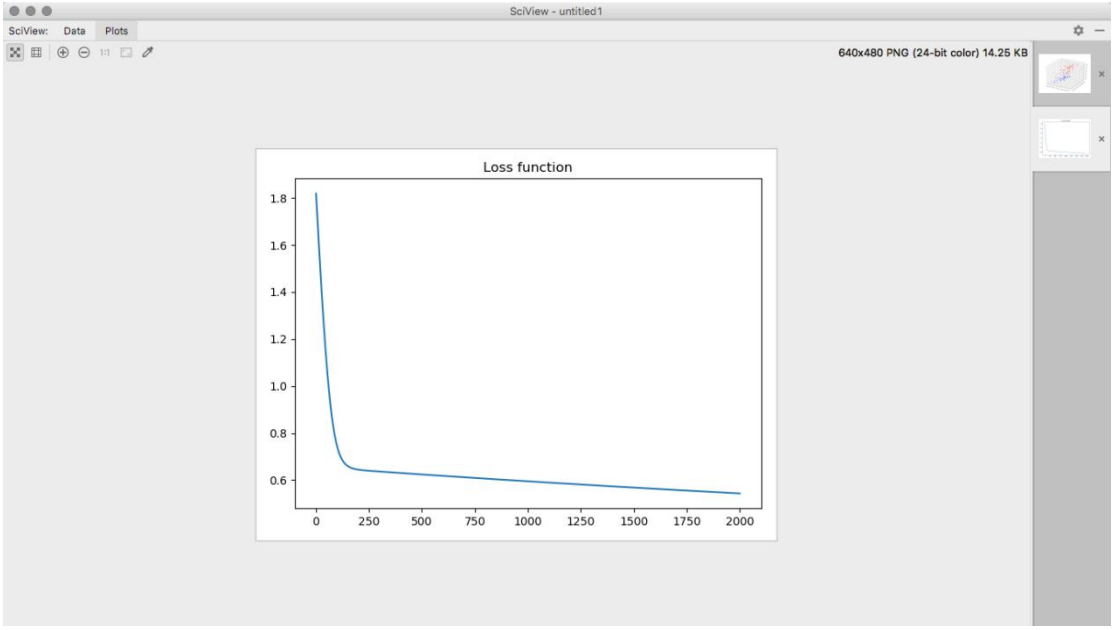

图 3.4、90%训练集时损失函数曲线图

四、实验结果分析

由实验结果可以直观看到，当训练集在所有样本中所占比例越来越大时，其模型训练出来的准确率越来越高，该模型灵敏性较高，具有一定的推广能力。