

**模式识别大作业**

题 目 鸢尾花卉识别系统

学 院 信息科学与工程

专 业 控制科学与工程

组 员 闵佳峰

指导教师 赵海涛

**完成日期： 2018 年 10 月24日**

**模式识别作业报告——鸢尾花卉识别系统**

组员：闵佳峰

**一、IRIS数据集简介**

Iris也称鸢尾花卉数据集,是常用的分类实验数据集，由R.A. Fisher于1936年收集整理的。其中包含3种植物种类，分别是山鸢尾（setosa）变色鸢尾（versicolor）和维吉尼亚鸢尾（virginica），每类50个样本，共150个样本。Iris的每个样本都包含了4个特征：花萼长度，花萼宽度，花瓣长度，花瓣宽度。

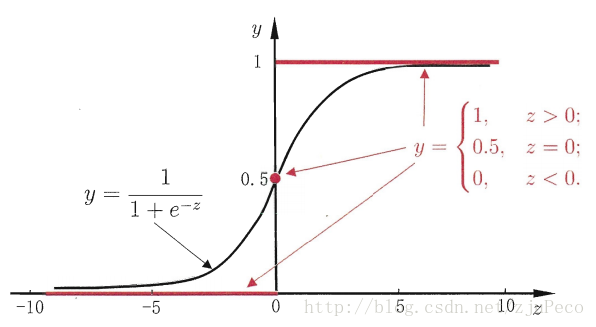
本实验运用逻辑回归算法和极大似然估计求解一个二分类问题，数据源自Iris中的山鸢尾样本集和变色鸢尾样本集，每个样本集的一半数据作为训练集，另一半作为测试集。

**二、逻辑回归算法原理**

逻辑回归与线性回归有一定的相似之处，比如模型的预测值都是,但两者的模型完全不同，线性回归的模型如下所示，其函数不连续。

()

逻辑回归的模型是Sigmoid函数，其函数表达式是，其中，, Sigmoid函数的图像如下图所示。



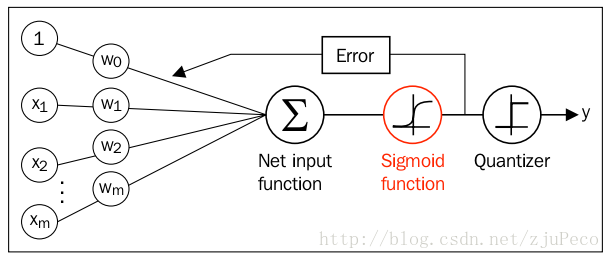
图表 1 Sigmoid函数图像

由于Sigmoid函数的取值范围是，在二分类中通常将作为某一类的后验概率估计值，以类1为例，其后验概率表达式如下所示。

()

()

Sigmoid函数的判别准则是将大于等于0.5的归为类别1，小于0.5的归为类别0，逻辑回归的等效网络示意图如下所示。



图表 2 逻辑回归网络示意图

合并式(2)和式(3)可以得到后验概率的一般形式。

()

然后求解样本的似然函数，记作。

()

为了简化计算，等式(5)两边取对数。

()

由于极大似然估计是求似然函数的最大值，而最优化问题通常是求函数的最小值，因此取代价函数，则：

()

为了计算代价函数的最小值，需要计算代价函数的梯度值，其计算公式如下。

()

()

然后采用梯度下降法迭代和变量，直到满足迭代要求（迭代次数超过限定值或者梯度小于给定值）。

()

()

**三、程序实现**

由于上述算法只能实现两分类，为了实现山鸢尾，变色鸢尾，维吉尼亚鸢尾的三分类，在程序中对三个样本进行两两分类，可以得到各自的判别准则，其中，，分别是第一、第二、第三次分类的预测值。

()

综合上述可以得到三分类的判别准则，通过计算样本在每次分类中的预测值即可判断样本的类别。

()

总体程序如下：

%运用逻辑回归算法对目标进行三分类

clear;

load iris\_dataset %输入iris数据集

data = irisInputs'; %总共150\*4维数据，即特征

setosa\_Train = data(1:25,:); %山鸢尾花训练集

setosa\_Test = data(26:50,:); %山鸢尾花测试集

versicolor\_Train = data(51:75,:); %变色鸢尾花训练集

versicolor\_Test = data(76:100,:); %变色鸢尾花测试集

virginica\_Train = data(101:125,:); %维吉尼亚鸢尾花训练集

virginica\_Test = data(126:150,:); %维吉尼亚鸢尾花测试集

train\_Data{1} = [setosa\_Train;versicolor\_Train]; %山鸢尾和变色鸢尾训练集

train\_Data{2} = [versicolor\_Train;virginica\_Train]; %变色鸢尾和维吉尼亚鸢尾训练集

train\_Data{3} = [virginica\_Train;setosa\_Train]; %维吉尼亚鸢尾和山鸢尾训练集

test\_Data = [setosa\_Test;versicolor\_Test;virginica\_Train]; %测试集数据

train\_Class = [zeros(25,1);ones(25,1)]; %训练集类别

test\_Class = zeros(75,1); %测试集类别

label = [ones(25,1);2\*ones(25,1);3\*ones(25,1)];

theta{1} = [0;0;0;0]; %初始化权重

theta{2} = [0;0;0;0];

theta{3} = [0;0;0;0];

beta = [0;0;0]; %初始化偏置

alpha = 0.003; %学习率取0.003

N = [2;2;2]; %迭代次数

J{1}(1) = 0; %损失函数

J{2}(1) = 0;

J{3}(1) = 0;

for index = 1:3

for i = 1:50

z = train\_Data{index}(i,:) \* theta{index} + beta(index);

phi = Sigmoid(z);

J{index}(2) = J{index}(1) - train\_Class(i) \* log(phi) - (1 - train\_Class(i)) \* log(1 - phi);

end

while abs(J{index}(N(index)) - J{index}(N(index) - 1)) > 0.01 %如果相邻两次损失函数差小于0.01，则停止迭代

N(index) = N(index) + 1;

dtheta = [0;0;0;0];

dbeta = 0;

for i = 1:50

z = train\_Data{index}(i,:) \* theta{index} + beta(index);

phi = Sigmoid(z);

dtheta = dtheta + (train\_Class(i) - phi) \* train\_Data{index}(i,:)'; %计算theta变量的梯度值

dbeta = dbeta + train\_Class(i) - phi; %计算beta变量的梯度值

end

theta{index} = theta{index} + alpha \* dtheta; %更新theta和beta的值

beta(index) = beta(index) + alpha \* dbeta;

for i = 1:50

z = train\_Data{index}(i,:) \* theta{index} + beta(index);

phi = Sigmoid(z);

J{index}(N(index)) = J{index}(N(index) - 1) - train\_Class(i) \* log(phi) - (1 - train\_Class(i)) \* log(1 - phi); %更新代价函数值

end

end

end

figure;

subplot(4,1,1);

plot(1:N(1),J{1}); %绘制训练过程图

xlabel('迭代次数');

ylabel('损失函数');

title('山鸢尾花和变色鸢尾花分类训练');

hold on

subplot(4,1,2);

plot(1:N(2),J{2}); %绘制训练过程图

xlabel('迭代次数');

ylabel('损失函数');

title('变色鸢尾花和维吉尼亚鸢尾花分类训练');

hold on

subplot(4,1,3);

plot(1:N(3),J{3}); %绘制训练过程图

xlabel('迭代次数');

ylabel('损失函数');

title('维吉尼亚鸢尾花和山鸢尾花分类训练');

hold on

%使用测试集检验

for i = 1:75

z1 = test\_Data(i,:) \* theta{1} + beta(1);

z2 = test\_Data(i,:) \* theta{2} + beta(2);

z3 = test\_Data(i,:) \* theta{3} + beta(3);

if z1 < 0 && z3 > 0

test\_Class(i) = 1;

end

if z1 > 0 && z2 < 0

test\_Class(i) = 2;

end

if z2 >0 && z3 < 0

test\_Class(i) = 3;

end

end

subplot(4,1,4);

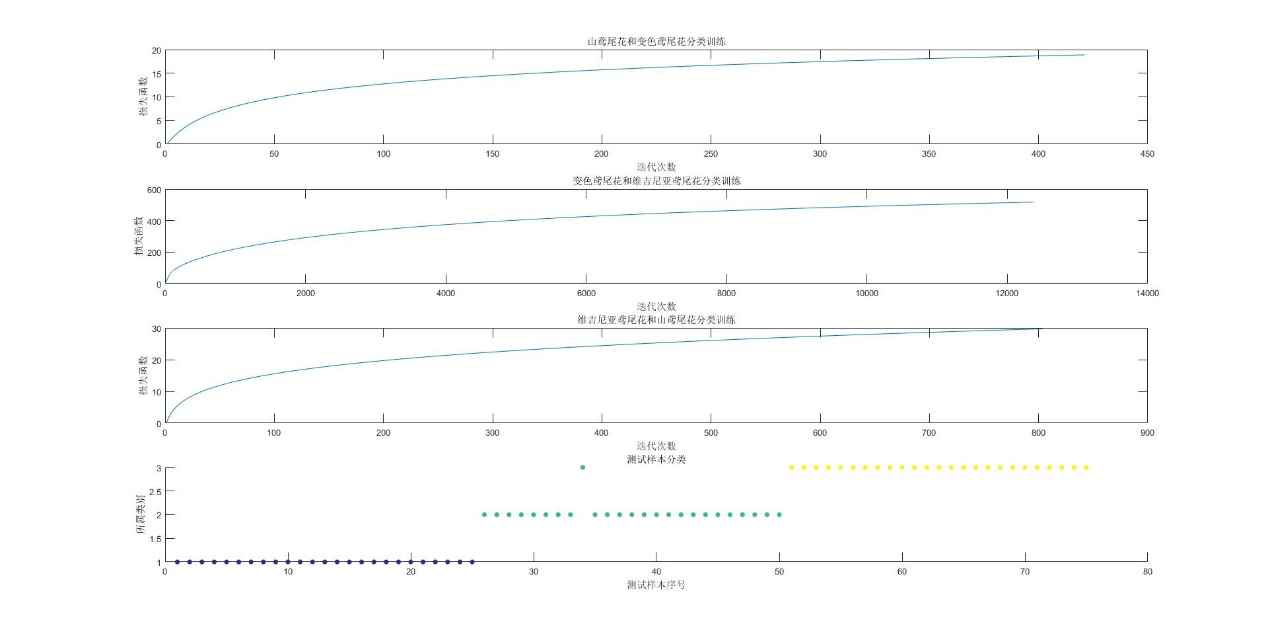
scatter(1:75,test\_Class,30,label,'filled'); %绘制测试集分类结果的散点图

xlabel('测试样本序号');

ylabel('所属类别');

title('测试样本分类');

**四、评定测试**



在测试结果中，第二次训练的时长最长，迭代次数最多。测试样本中有一个样本本来属于变色鸢尾花，却被判断为维吉尼亚鸢尾花，但是总体来说分类效果尚可。