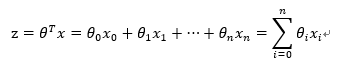
逻辑回归是面对一个回归或者分类问题，建立代价函数，然后通过优化方法迭代求解出最优的模型参数，然后测试验证这个求解的模型的好坏。常规步骤：寻找h函数（即预测函数）、

构造J函数（损失函数）、想办法使得J函数最小并求得回归参数（θ）

Logistic函数（或称为Sigmoid函数），函数形式为： 

对于线性边界的情况，边界形式



构造预测函数为：   
IMG_256

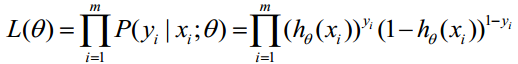
函数h(x)表示结果取1的概率，因此对于输入x分类结果为类别1和类别0的概率分别为：   
  


构造损失函数J（m个样本，每个样本具有n个特征）

概率综合起来写成

IMG_256

取似然函数

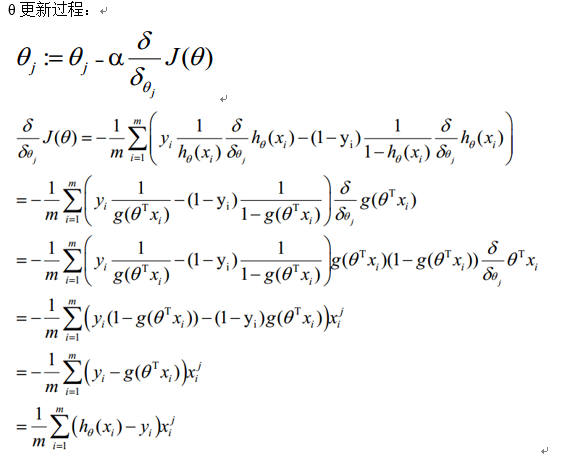


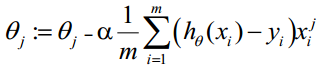
对数似然函数

IMG_256

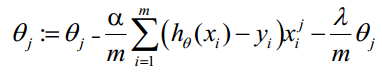
最大似然估计就是求使l(θ)取最大值时的θ，求得的θ就是要求的最佳参数

梯度下降法求解最小值





正则化



α是学习率参数，λ是正则项参数

代码

clear

clc

%二分类 随机生成数据。 200个数据 每个数据2个特征

data=1\*rand(300,2);

label=zeros(300,1);

%label(sqrt(data(:,1).^2+data(:,2).^2)<8)=1;

label((data(:,2)+data(:,1)>1))=1;

%在data上加常数特征项；

data=[data,ones(size(data,1),1)];

%打乱循序

randIndex = randperm(size(data,1));

data\_new=data(randIndex,:);

label\_new=label(randIndex,:);

%80%训练 20%测试

k=0.8\*size(data,1);

X1=data\_new(1:k,:);

Y1=label\_new(1:k,:);

X2=data\_new(k+1:end,:);

Y2=label\_new(k+1:end,:);

[m1,n1] = size(X1);

[m2,n2] = size(X2);

Features=size(data,2); %特征个数

%设定学习率为0.01

delta=1;

lamda=0.2; %正则项系数

theta1=rand(1,Features);

%梯度下降算法求解theta

num = 300; %最大迭代次数

L=[];

while(num)

dt=zeros(1,Features);

loss=0;

for i=1:m1

xx=X1(i,1:Features);

yy=Y1(i,1);

h=1/(1+exp(-(theta1 \* xx')));

dt=dt+(h-yy) \* xx;

loss=loss+ yy\*log(h)+(1-yy)\*log(1-h);

end

loss=-loss/m1;

L=[L,loss];

theta2=theta1 - delta\*dt/m1 - lamda\*theta1/m1;

theta1=theta2;

num = num - 1;

if loss<0.01

break;

end

end

figure

subplot(1,2,1)

plot(L)

title('loss')

subplot(1,2,2)

x=0:0.1:10;

y=(-theta1(1)\*x-theta1(3))/theta1(2);

plot(x,y,'linewidth',2)

hold on

plot(data(label==1,1),data(label==1,2),'ro')

hold on

plot(data(label==0,1),data(label==0,2),'go')

axis([0 1 0 1])

%测试数据

acc=0;

for i=1:m2

xx=X2(i,1:Features)';

yy=Y2(i);

finil=1/(1+exp(-theta2 \* xx));

if finil>0.5 && yy==1

acc=acc+1;

end

if finil<=0.5 && yy==0

acc=acc+1;

end

end

acc/m2

结果

