## 线性核第一问的程序

### 数据预处理

导入数据：

data = Import[  
 "G:\\GitHub Local \  
Repository\\Mathematical-Modeling-Campus-Competition\\\  
2020年兰州大学数学建模竞赛赛题 (1)\\Data.xlsx"];

数据预处理

train = data[[1, 2 ;; All]];  
test = data[[2, 2 ;; All]];  
asso=MapThread[Rule,{train[[All,1;;7]],train[[All,8]]}][[2;;All]];  
model=Classify[asso,Method->"SupportVectorMachine"];

## 主成分分析

features=train[[All,1;;7]];  
lables=train[[All,8]];

得到降维后的数据

reduced=(KarhunenLoeveDecomposition[Transpose[features]][[1]]//Transpose)[[All,1;;2]];

求出主向量

{b, m} = KarhunenLoeveDecomposition[Transpose[features],   
 Standardized -> False];

训练集在两个最主成分构成的特征空间中的分布图（有色）：

ListPlot[{point2,point1},PlotRange->{Full,All},AxesLabel->{"第一主成分","第二主成分"},PlotLabel->"训练集在两个最主成分构成的特征空间中的分布",PlotLegends->{0,1}]

计算协方差矩阵，

Covariance[features]

计算协方差矩阵的特征值

eig=Covariance[features]//Eigenvalues;

## 第二问的程序

在某个点(x,y)处以r为窗口宽度，计算某个点群point的程序：

count[x\_,y\_][r\_,point\_]:=(If[(x-r)<#[[1]]<(x+r)&&(y-r)<#[[2]]<(y+r),1,0]&/@point//Total)

对于某个区域已经数出来的两种点的个数，我们可以这样计算明显度函数：

Obvious[x\_,y\_]:=If[  
x!=0||y!=0,Abs@(x-y)/(x+y),0.3  
]//N

计算区域上的明显度数组，

p[r\_,h\_,point\_]:=ParallelTable[{x,y,count[x,y][r,point]},{x,0,2.5,h},{y,0,2,h}];  
list[r\_,h\_]:=MapThread[{#1[[1]],#1[[2]],Obvious[#1[[3]],#2[[3]]]}&,{p[r,h,point1],p[r,h,point2]},2];  
contour=Flatten[list[0.05,0.1],1];

绘制图像：

contour=Flatten[list[0.05,0.1],1];  
  
con=ListContourPlot[contour,PlotLegends->Automatic,Contours->10,PlotLabel->"明显度函数的等高图",FrameLabel->{"第一主特征","第二主特征"}]  
  
lis=ListPlot[{point1,point2},PlotStyle->{Green,Red},AxesLabel->{"第一主成分","第二主成分"},PlotLabel->"训练集在两个最主成分构成的特征空间中的分布",PlotLegends->{0,1}]  
  
show=Show[con,lis,PlotLabel->"点集与明显度函数对比图"]

## 第三问的程序

数据降维

transformedtest = Transpose[m.Transpose[test]][[All, 1 ;; 2]];

作图

compare =   
 ListPlot[{point1, point2, transformedtest},   
 PlotStyle -> {Green, Blue, Red}, PlotLegends -> {1, 0, "测试集"},   
 PlotLabel -> "训练集和测试集对比", AxesLabel -> {"第一主向量", "第二主向量"}]

### 训练线性核

model2 = Classify[asso2,   
 Method -> {"SupportVectorMachine", "KernelType" -> "Linear"}];

作图

linearplot=Show[dens,ListPlot[{point1,point2},PlotStyle->{Yellow,Blue,Red},PlotLegends->{1,0}],PlotLabel->"线性核下的训练结果",FrameLabel->{"第一主向量","第二主向量"}]

linearplot2=Show[dens,plot,PlotLabel->"线性核下的训练结果",FrameLabel->{"第一主向量","第二主向量"}]

### 训练RBF核

modelRBF=Classify[asso2,Method->{"SupportVectorMachine","KernelType"->"RadialBasisFunction"}];

作图

RBFplot=Show[dens2,ListPlot[{point1,point2},PlotStyle->{Yellow,Blue,Red},PlotLegends->{1,0}],PlotLabel->"高斯径向基函数核下的训练结果",FrameLabel->{"第一主向量","第二主向量"}]

RBFplot2=Show[dens2,plot,PlotLabel->"高斯径向基函数核下的训练结果",FrameLabel->{"第一主向量","第二主向量"}]

### 训练多项式核

modelPolinomial=Classify[asso2,Method->{"SupportVectorMachine","KernelType"->"Polynomial"}];

作图

polynomialplot=Show[densPolinomial,ListPlot[{point1,point2},PlotStyle->{Yellow,Blue,Red},PlotLegends->{1,0}],PlotLabel->"多项式核下的训练结果",FrameLabel->{"第一主向量","第二主向量"}]

polynomialplot2=Show[densPolinomial,plot,PlotLabel->"多项式核下的训练结果",FrameLabel->{"第一主向量","第二主向量"}]