

## 实验报告

成绩:

班级学号 202015116 姓名 梁祺若 实验日期 2022-10-20 仪器编号         

实验名称 实验 6-8 R 的支持向量机，一般聚类 and 特色聚类

### 1. 实验题目

R 的支持向量机，一般聚类 and 特色聚类

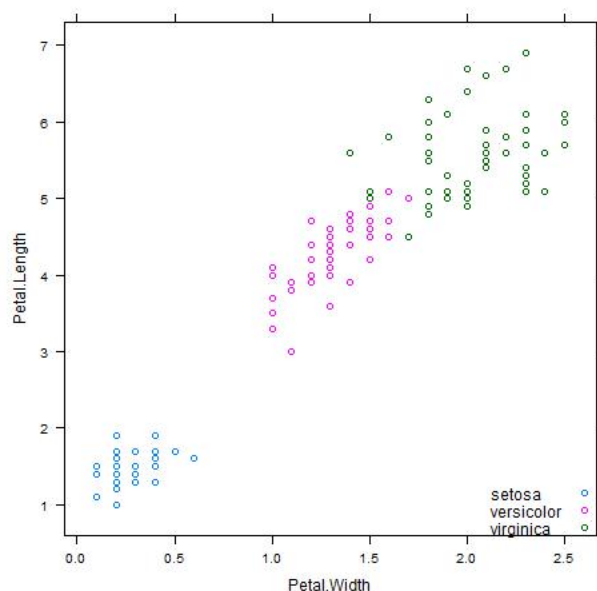
### 2. 实验步骤

本题目由三部分组成：（1）利用 R 软件对 iris 数据做支持向量机；（2）利用 R 软件对 iris 数据做一般聚类；（3）利用 R 软件对\*\*数据做特色聚类。

（1）利用 R 软件对 iris 数据做支持向量机

- 数据来源：<https://cloud.tencent.com/developer/article/1869024>
- 初步通过图像判断数据的分布情况

```
> library(lattice)
> png(file = "C:\\Users\\MarsR\\Desktop\\R语言\\Iris数据集\\iris.png")
> xyplot(Petal.Length ~ Petal.Width, data = iris, groups = Species, auto.key=list(corner=c(1,0)))
> dev.off()
null device
      1
```

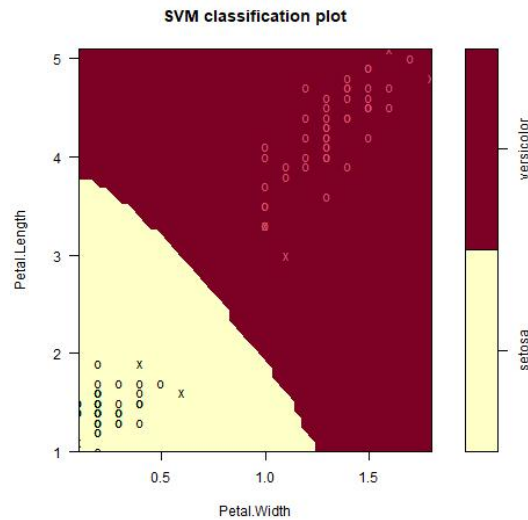


- 由图可以看出 setosa 很容易与 versicolor 和 virginica 分隔开，但后两者分隔不明显，这次试验先选择 setosa 和 versicolor 来进行实验

```
> library(e1071)
> subdata <- iris[iris$Species != 'virginica',] #1两分类方法
> subdata$Species <- factor(subdata$Species) #将类别变为有序因子
> model1 <- svm(Species ~ Petal.Length + Petal.Width, data = subdata) #花瓣长度与宽度做分类 formula:label ~ x + y
```

- SVM 结果可视化

```
> png(file = "C:\\users\\MarsR\\Desktop\\R语言\\Iris数据集\\svm1.png")
> plot(model1, subdata, Petal.Length ~ Petal.Width)
> dev.off()
null device
      1
```



- 使用 `summary` 来查看 SVM 构建情况

```
> summary(model1)

Call:
svm(formula = species ~ Petal.Length + Petal.width, data = subdata)

Parameters:
  SVM-Type:  C-classification
  SVM-Kernel: radial
    cost: 1

Number of Support Vectors: 7

( 3 4 )

Number of Classes: 2

Levels:
setosa versicolor
```

- (2) 利用 R 软件对 iris 数据做一般聚类。

- 数据来源: <https://cloud.tencent.com/developer/article/1869024>
- Kmeans
- Kmeans 对 iris 进行聚类分析

[illegible]

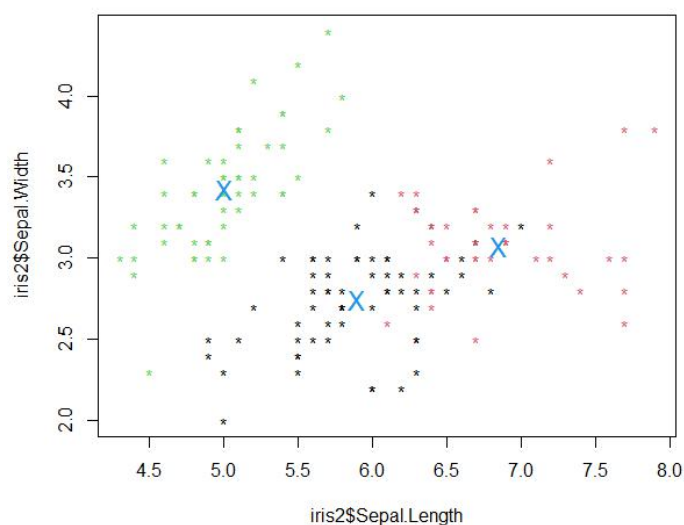
- 用 table 函数查看分类结果情况

```
> #用table函数查看分类结果情况
> table(iris$Species,iris.kmeans$cluster)

      1  2  3
setosa  0  0 50
versicolor 48  2  0
virginica 14 36  0
>
```

- 下边我们将分类以及中心点打印出来

```
> plot(iris2$Sepal.Length,iris2$Sepal.Width,col=iris.kmeans$cluster,pch="x")
>
> points(iris.kmeans$centers,pch="x",cex=1.5,col=4)
>
```



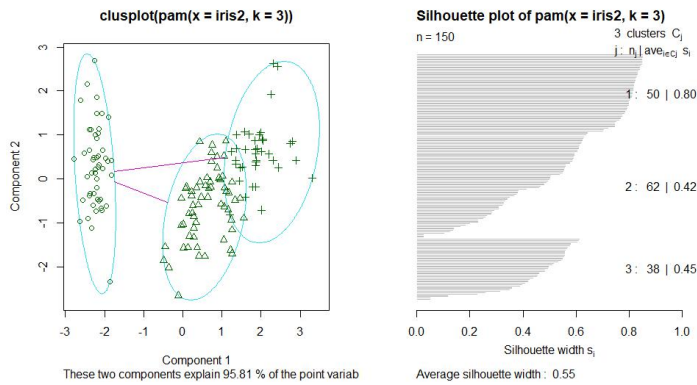
- K-medoids
- 使用 K-medoids 方法中的 pam 算法来进行聚类分析

```
> library(cluster)
>
> iris2.pam<-pam(iris2,3)
> table(iris$Species,iris2.pam$clustering)

      1  2  3
setosa  50  0  0
versicolor  0 48  2
virginica  0 14 36
>
```

- Pam 算法画图

```
> layout(matrix(c(1,2),1,2)) #每页显示两个图
> plot(iris2.pam)
> layout(matrix(1))
> |
```



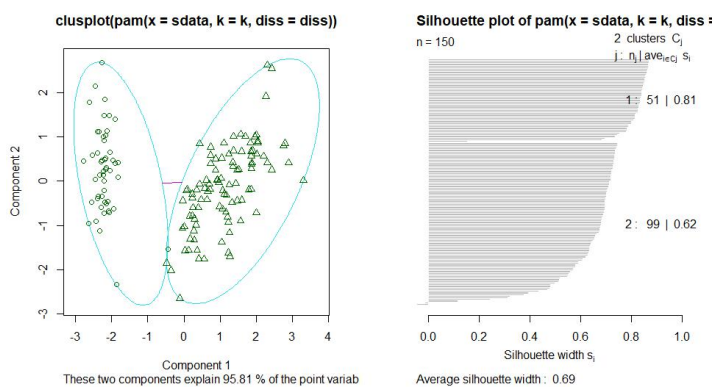
- 使用 K-medoids 方法中的 pamk 算法来进行聚类分析

```
> library(fpc)
> iris2.pamk <- pamk(iris2)
> table(iris2.pamk$pamobject$clustering, iris$species)
```

	setosa	versicolor	virginica
1	50	1	0
2	0	49	50

- pamk 算法画图

```
> layout(matrix(c(1,2),1,2)) #每页显示两个图
> plot(iris2.pamk$pamobject)
> layout(matrix(1))
> |
```



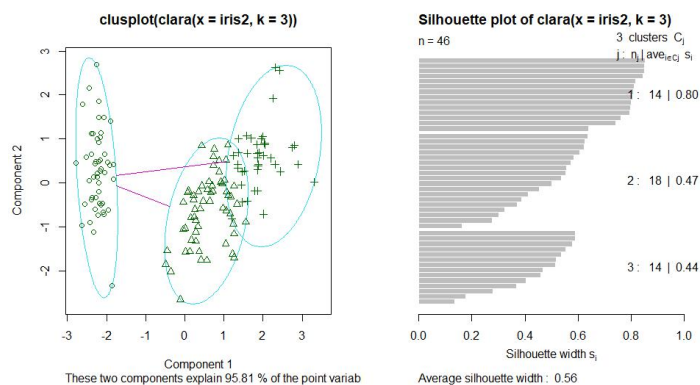
- 使用 K-medoids 方法中的 clara 算法来进行聚类分析

```
> iris2.clara <- clara(iris2,3)
> table(iris$species, iris2.clara$clustering)
```

	1	2	3
setosa	50	0	0
versicolor	0	48	2
virginica	0	13	37

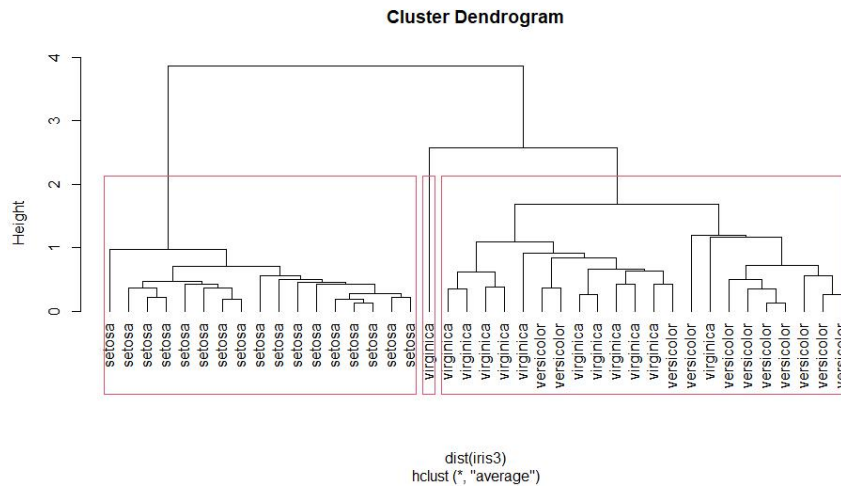
- Clara 算法画图

```
> layout(matrix(c(1,2),1,2)) #每页显示两个图
> plot(iris2.clara)
> layout(matrix(1))
> |
```



- 通过上述分类结果可以看到，pam 和 calra 算法分类结果基本类似，但是 pamk 将三类分为了两类。
- 层次聚类 HCluster

```
> dim(iris)#返回行列数
[1] 150 5
>
> idx<-sample(1:dim(iris)[1],40)
> iris3<-iris[idx,-5]
> iris3
      Sepal.Length Sepal.width Petal.Length Petal.width
37          5.5         3.5         1.3         0.2
8           5.0         3.4         1.5         0.2
107          4.9         2.5         4.5         1.7
94           5.0         2.3         3.3         1.0
15           5.8         4.0         1.2         0.2
4           4.6         3.1         1.5         0.2
142          6.9         3.1         5.1         2.3
33           5.2         4.1         1.5         0.1
102          5.8         2.7         5.1         1.9
135          6.1         2.6         5.6         1.4
20           5.1         3.8         1.5         0.3
125          6.7         3.3         5.7         2.1
72           6.1         2.8         4.0         1.3
100          5.7         2.8         4.1         1.3
114          5.7         2.5         5.0         2.0
70           5.6         2.5         3.9         1.1
87           6.7         3.1         4.7         1.5
26           5.0         3.0         1.6         0.2
139          6.0         3.0         4.8         1.8
25           4.8         3.4         1.9         0.2
64           6.1         2.9         4.7         1.4
12          4.8         3.4         1.6         0.2
22           5.1         3.7         1.5         0.4
112          6.4         2.7         5.3         1.9
82           5.5         2.4         3.7         1.0
47           5.1         3.8         1.6         0.2
140          6.9         3.1         5.4         2.1
10          4.9         3.1         1.5         0.1
124          6.3         2.7         4.9         1.8
105          6.5         3.0         5.8         2.2
36           5.0         3.2         1.2         0.2
44           5.0         3.5         1.6         0.6
56           5.7         2.8         4.5         1.3
97           5.7         2.9         4.2         1.3
49           5.3         3.7         1.5         0.2
28           5.2         3.5         1.5         0.2
78           6.7         3.0         5.0         1.7
63           6.0         2.2         4.0         1.0
17           5.4         3.9         1.3         0.4
106          7.6         3.0         6.6         2.1
> hc<-hclust(dist(iris3),method = "ave") #注意hcluster里边传入的是dist返回值对象
>
> plot(hc,hang=-1,labels=iris$Species[idx]) #这里的hang=-1使得树的节点在下方对齐
> #将树分为3块
> rect.hclust(hc,k=3)
> groups<-cutree(hc,k=3)
>
```



(3) 利用 R 软件对 iris 数据做特色聚类。

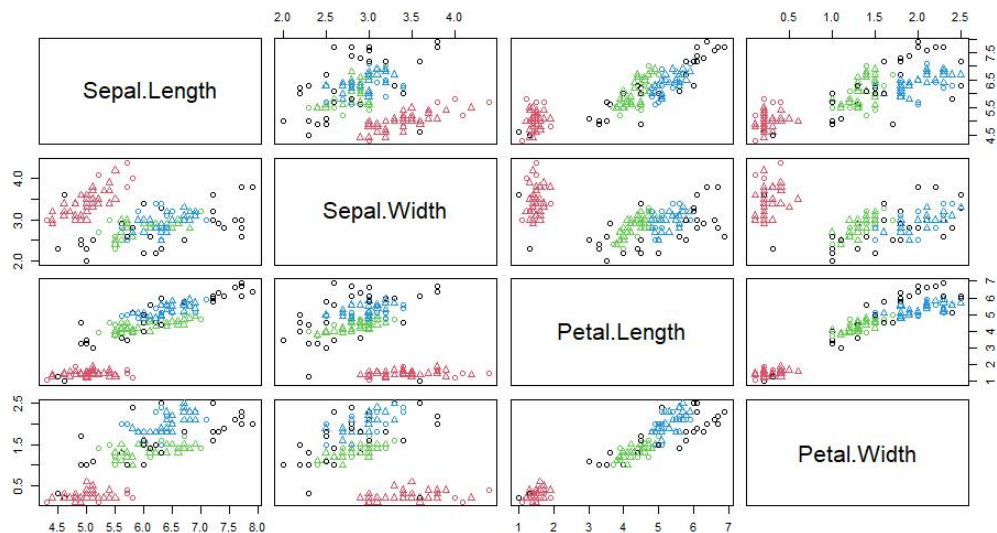
## ● 基于密度的聚类 DBSCAN

```
> library(fpc)
> iris2<-iris[-5]
> ds<-dbscan(iris2,eps=0.42,minpts = 5)
> table(ds$cluster,iris$species)
```

	setosa	versicolor	virginica
0	2	10	17
1	48	0	0
2	0	37	0
3	0	3	33

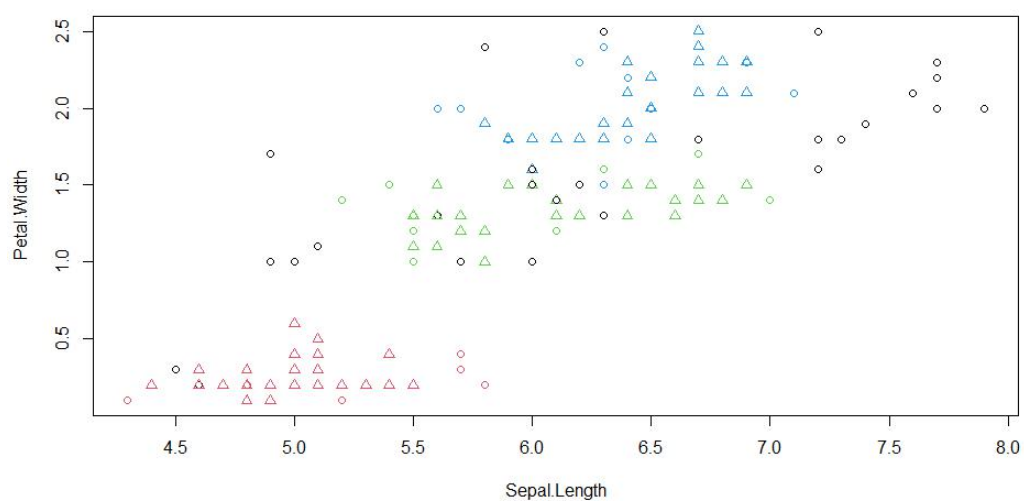
- 打印出 ds 和 irsi2 的聚类散点图

```
> plot(ds,iris2)
```



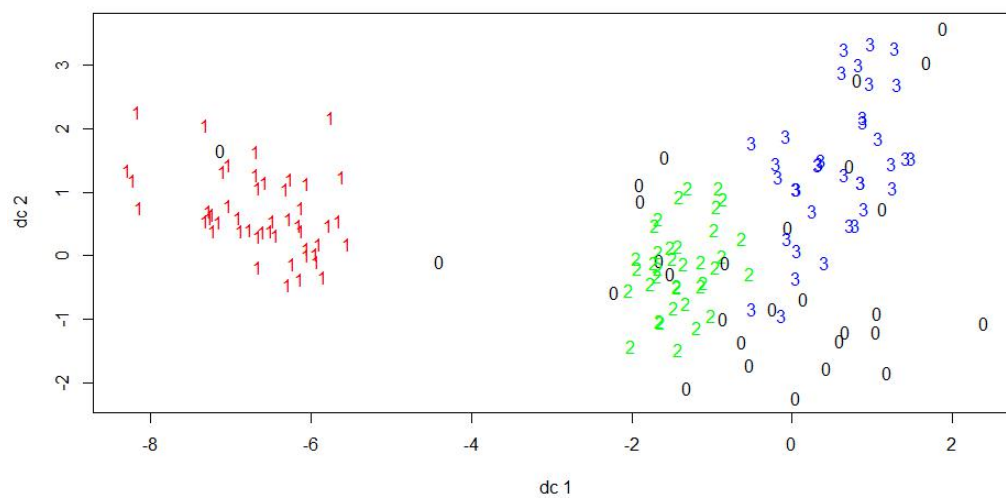
- 打印出 iris 第一列和第四列为坐标轴的聚类结果

```
> plot(ds, iris2[,c(1,4)])
```



- 另一个表示聚类结果的函数，plotcluster

```
> #另一个表示聚类结果的函数，plotcluster  
> plotcluster(iris2,ds$cluster)  
> |
```



### 3. 实验结果

本次实验对 iris 数据集做了支持向量机以及一般聚类 and 特殊聚类分析，特别是对于一般聚类的 K-medoids 方法，使用了 pam、pamk、clara 三种不同的算法。通过上述分类结果可以看到，pam 和 clara 算法分类结果基本类似，但是 pamk 将三类分为了两类。