

Data Science II Homework 5

Camille Okonkwo

Contents

Question 1	3
Background	3
(a) Fit a support vector classifier to the training data. What are the training and test error rates?	5
(b) Fit a support vector machine with a radial kernel to the training data. What are the training and test error rates?	6
Question 2	7
Background	7
(a) Using hierarchical clustering with complete linkage and Euclidean distance, cluster the states. Cut the dendrogram at a height that results in three distinct clusters. Which states belong to which clusters?	8
(b) Hierarchically cluster the states using complete linkage and Euclidean distance, after scaling the variables to have standard deviation one.	10
(e) Does scaling the variables change the clustering results? Why? In your opinion, should the variables be scaled before the inter-observation dissimilarities are computed?	12

```
library(tidymodels)
library(caret)
library(ISLR)
library(kernlab)
library(factoextra)
set.seed(2)
```

Question 1

Background

In this problem, we will apply support vector machines to predict whether a given car gets high or low gas mileage based on the dataset `auto.csv` (used in Homework 3; see Homework 3 for more details of the dataset). The response variable is `mpg_cat`. The predictors are `cylinders`, `displacement`, `horsepower`, `weight`, `acceleration`, `year`, and `origin`. Split the dataset into two parts: training data (70%) and test data (30%).

```
auto = read_csv("data/auto.csv") |>
  drop_na() |>
  mutate(
    mpg_cat = as.factor(mpg_cat),
    mpg_cat = forcats::fct_relevel(mpg_cat, c("low", "high")),
    cylinders = as.factor(cylinders),
    origin = as.factor(origin)
  )

set.seed(2)

# create a random split of 70% training and 30% test data
data_split = initial_split(data = auto, prop = 0.7)

# partitioned datasets
training_data = training(data_split)
testing_data = testing(data_split)

head(training_data)

## # A tibble: 6 x 8
##   cylinders displacement horsepower weight acceleration year origin mpg_cat
##   <fct>          <dbl>         <dbl>  <dbl>          <dbl> <dbl> <fct> <fct>
## 1 4              86           64   1875           16.4   81 1    high
## 2 6             225          100   3651           17.7   76 1    low
## 3 6             231          165   3445           13.4   78 1    low
## 4 5             131          103   2830           15.9   78 2    low
## 5 4              98           65   2380           20.7   81 1    high
## 6 4              97           75   2155           16.4   76 3    high

head(testing_data)

## # A tibble: 6 x 8
##   cylinders displacement horsepower weight acceleration year origin mpg_cat
##   <fct>          <dbl>         <dbl>  <dbl>          <dbl> <dbl> <fct> <fct>
## 1 8             302          140   3449           10.5   70 1    low
## 2 8             390          190   3850            8.5   70 1    low
## 3 4             113           95   2372            15    70 3    high
## 4 6             200           85   2587            16    70 1    low
## 5 4              97           88   2130           14.5   70 3    high
## 6 4             107           90   2430           14.5   70 2    high

# training data
x_1 = model.matrix(mpg_cat ~ ., training_data)[, -1] # matrix of predictors
head(x_1)

##   cylinders4 cylinders5 cylinders6 cylinders8 displacement horsepower weight
```

```
## 1      1      0      0      0      86      64 1875
## 2      0      0      1      0     225     100 3651
## 3      0      0      1      0     231     165 3445
## 4      0      1      0      0     131     103 2830
## 5      1      0      0      0      98      65 2380
## 6      1      0      0      0      97      75 2155
```

```
## acceleration year origin2 origin3
## 1      16.4   81      0      0
## 2      17.7   76      0      0
## 3      13.4   78      0      0
## 4      15.9   78      1      0
## 5      20.7   81      0      0
## 6      16.4   76      0      1
```

```
y_1 = training_data$mpg_cat # vector of response
```

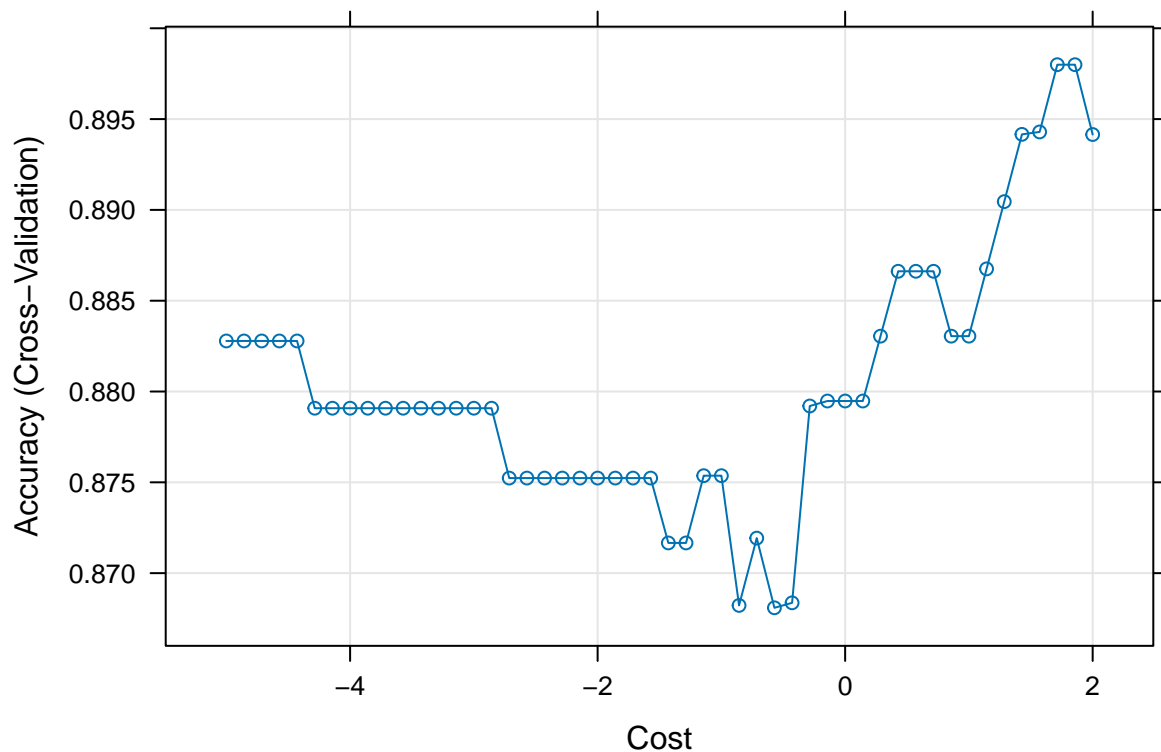
```
# testing data
```

```
x_2 = model.matrix(mpg_cat ~ .,testing_data)[, -1] # matrix of predictors
```

```
y_2 = testing_data$mpg_cat # vector of response
```

(a) Fit a support vector classifier to the training data. What are the training and test error rates?

```
ctrl = trainControl(method = "cv")
# kernlab
set.seed(2)
svml.fit = train(x_1, y_1,
  data = training_data,
  method = "svmLinear",
  tuneGrid = data.frame(C = exp(seq(-5, 2, len = 50))),
  trControl = ctrl)
plot(svml.fit, highlight = TRUE, xTrans = log)
```



what are the training and test error rates?

(b) Fit a support vector machine with a radial kernel to the training data. What are the training and test error rates?

6

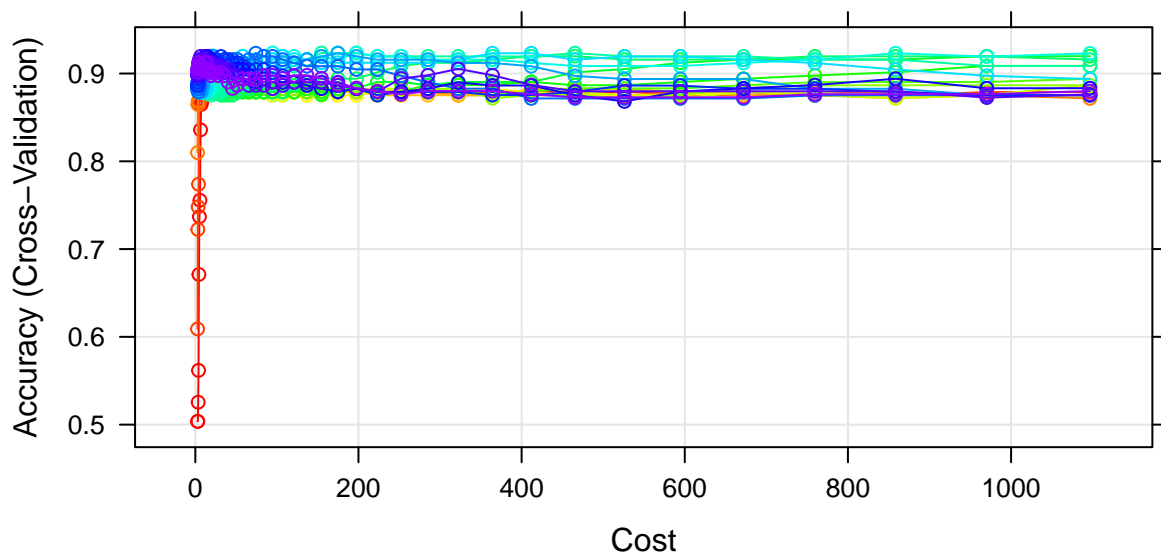
(b) Fit a support vector machine with a radial kernel to the training data. What are the training and test error rates?

```
svmr.grid = expand.grid(C = exp(seq(1, 7, len = 50)),
                        sigma = exp(seq(-10, -2, len = 20))) # how to tune?
# tunes over both cost and sigma
set.seed(2)
svmr.fit = train(x_1, y_1,
                 data = training_data,
                 method = "svmRadialSigma", # what about "svmRadial"? which radial method is best in th
                 tuneGrid = svmr.grid,
                 trControl = ctrl)

myCol = rainbow(25)
myPar = list(superpose.symbol = list(col = myCol),
             superpose.line = list(col = myCol))

plot(svmr.fit, highlight = TRUE, par.settings = myPar)
```

Sigma			
0.000372699966223616	0.00305959206434424		
0.00056783242423576	0.00466148574327131		
0.000865129303016903	0.00710207402743375		
0.00131808026275682	0.0108204676081991		
0.00200818024890684	0.0164856799306543		



Question 2

Background

In this problem, we perform hierarchical clustering on the states using the `USArrests` data in the `ISLR` package. For each of the 50 states in the United States, the dataset contains the number of arrests per 100,000 residents for each of three crimes: **Assault**, **Murder**, and **Rape**. The dataset also contains the percent of the population in each state living in urban areas, `UrbanPop`. The four variables will be used as features for clustering.

```
data(USArrests)
```

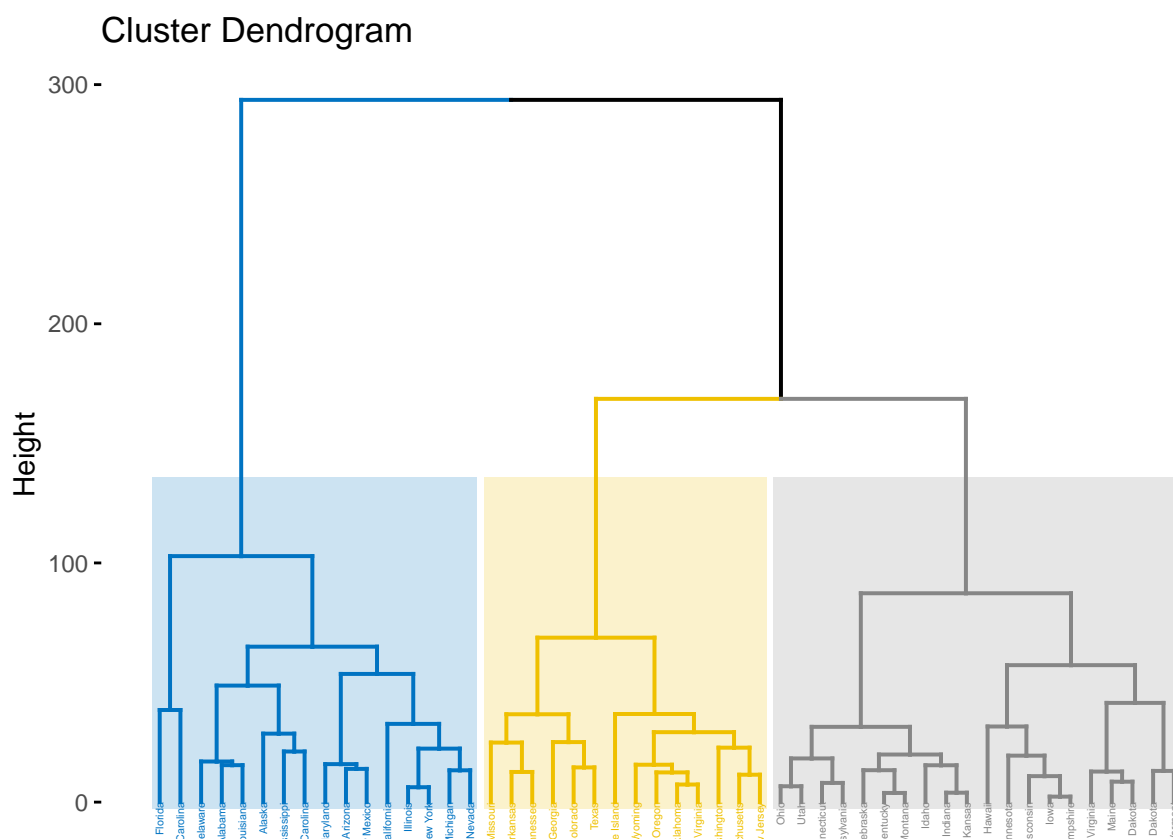
```
us_arrests = na.omit(USArrests)
```

(a) Using hierarchical clustering with complete linkage and Euclidean distance, cluster the states. Cut the dendrogram at a height that results in three distinct clusters. Which states belong to which clusters? 8

(a) Using hierarchical clustering with complete linkage and Euclidean distance, cluster the states. Cut the dendrogram at a height that results in three distinct clusters. Which states belong to which clusters?

```
# complete linkage and euclidean distance
hc.complete = hclust(dist(us_arrests), method = "complete")

# hierarchical clustering
fviz_dend(hc.complete, k = 3,
          cex = 0.3,
          palette = "jco",
          color_labels_by_k = TRUE,
          rect = TRUE, rect_fill = TRUE, rect_border = "jco",
          labels_track_height = 2.5)
```



```
us.complete = cutree(hc.complete, 3)

# 1st cluster states
us_arrests[us.complete == 1,]
```

##		Murder	Assault	UrbanPop	Rape
##	Alabama	13.2	236	58	21.2
##	Alaska	10.0	263	48	44.5
##	Arizona	8.1	294	80	31.0
##	California	9.0	276	91	40.6
##	Delaware	5.9	238	72	15.8
##	Florida	15.4	335	80	31.9

(a) Using hierarchical clustering with complete linkage and Euclidean distance, cluster the states. Cut the dendrogram at a height that results in three distinct clusters. Which states belong to which clusters? 9

```
## Illinois      10.4      249      83 24.0
## Louisiana     15.4      249      66 22.2
## Maryland      11.3      300      67 27.8
## Michigan      12.1      255      74 35.1
## Mississippi   16.1      259      44 17.1
## Nevada        12.2      252      81 46.0
## New Mexico    11.4      285      70 32.1
## New York      11.1      254      86 26.1
## North Carolina 13.0      337      45 16.1
## South Carolina 14.4      279      48 22.5
```

2nd cluster states

```
us_arrests[us.complete == 2,]
```

```
##           Murder Assault UrbanPop Rape
## Arkansas      8.8      190      50 19.5
## Colorado      7.9      204      78 38.7
## Georgia       17.4      211      60 25.8
## Massachusetts  4.4      149      85 16.3
## Missouri       9.0      178      70 28.2
## New Jersey     7.4      159      89 18.8
## Oklahoma       6.6      151      68 20.0
## Oregon         4.9      159      67 29.3
## Rhode Island   3.4      174      87  8.3
## Tennessee     13.2      188      59 26.9
## Texas          12.7      201      80 25.5
## Virginia       8.5      156      63 20.7
## Washington     4.0      145      73 26.2
## Wyoming        6.8      161      60 15.6
```

3rd cluster states

```
us_arrests[us.complete == 3,]
```

```
##           Murder Assault UrbanPop Rape
## Connecticut    3.3      110      77 11.1
## Hawaii          5.3       46      83 20.2
## Idaho           2.6      120      54 14.2
## Indiana         7.2      113      65 21.0
## Iowa            2.2       56      57 11.3
## Kansas          6.0      115      66 18.0
## Kentucky        9.7      109      52 16.3
## Maine           2.1       83      51  7.8
## Minnesota       2.7       72      66 14.9
## Montana         6.0      109      53 16.4
## Nebraska        4.3      102      62 16.5
## New Hampshire   2.1       57      56  9.5
## North Dakota    0.8       45      44  7.3
## Ohio            7.3      120      75 21.4
## Pennsylvania    6.3      106      72 14.9
## South Dakota    3.8       86      45 12.8
## Utah            3.2      120      80 22.9
## Vermont         2.2       48      32 11.2
## West Virginia   5.7       81      39  9.3
## Wisconsin       2.6       53      66 10.8
```

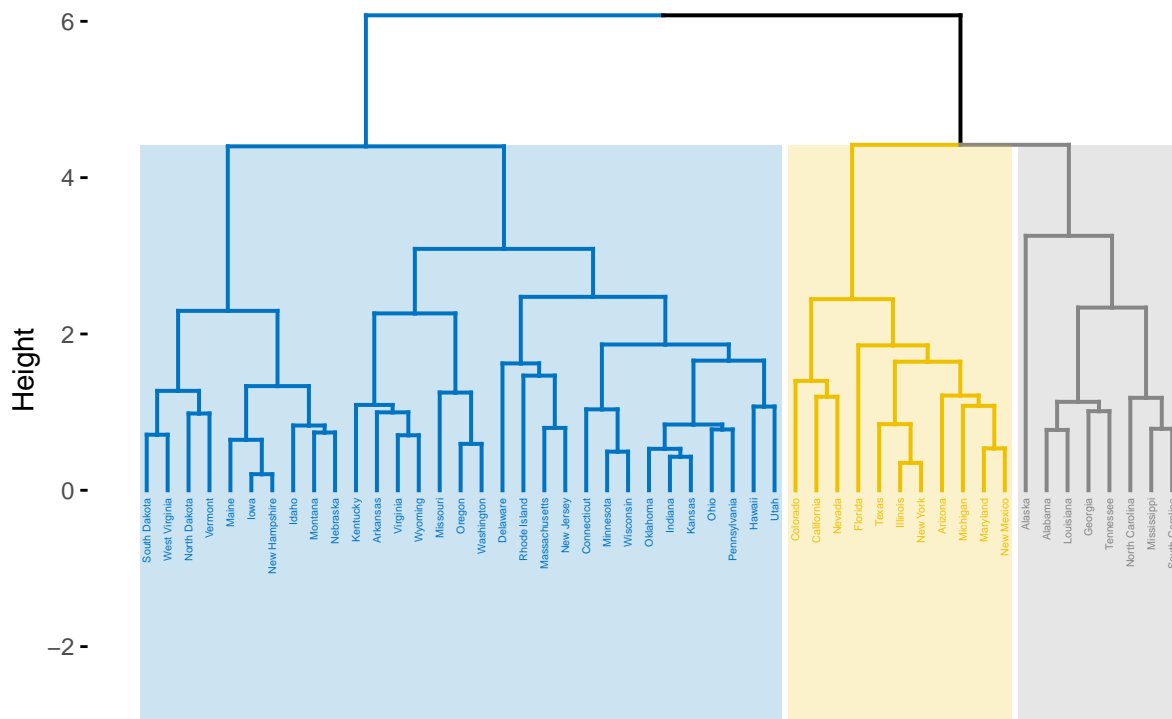
(b) Hierarchically cluster the states using complete linkage and Euclidean distance, after scaling the variables to have standard deviation one.

```
# scale the variables to have standard deviation one
scaled_arrests = scale(us_arrests)

# complete linkage and Euclidean distance
hc_complete = hclust(dist(scaled_arrests), method = "complete")

# hierarchial clustering dendrogram
fviz_dend(hc_complete, k = 3,
  cex = 0.3,
  palette = "jco",
  color_labels_by_k = TRUE,
  rect = TRUE, rect_fill = TRUE, rect_border = "jco",
  labels_track_height = 2.5)
```

Cluster Dendrogram



```
clusters_scaled = cutree(hc_complete, k = 3)
```

```
# 1st cluster states
```

```
us_arrests[clusters_scaled == 1,]
```

##	Murder	Assault	UrbanPop	Rape
## Alabama	13.2	236	58	21.2
## Alaska	10.0	263	48	44.5
## Georgia	17.4	211	60	25.8
## Louisiana	15.4	249	66	22.2
## Mississippi	16.1	259	44	17.1
## North Carolina	13.0	337	45	16.1

(b) Hierarchically cluster the states using complete linkage and Euclidean distance, after scaling the variables to have standard deviation one.

11

```
## South Carolina 14.4 279 48 22.5
## Tennessee 13.2 188 59 26.9
```

```
# 2nd cluster states
```

```
us_arrests[clusters_scaled == 2,]
```

```
##           Murder Assault UrbanPop Rape
## Arizona      8.1    294      80 31.0
## California    9.0    276      91 40.6
## Colorado      7.9    204      78 38.7
## Florida      15.4    335      80 31.9
## Illinois     10.4    249      83 24.0
## Maryland     11.3    300      67 27.8
## Michigan     12.1    255      74 35.1
## Nevada       12.2    252      81 46.0
## New Mexico   11.4    285      70 32.1
## New York     11.1    254      86 26.1
## Texas        12.7    201      80 25.5
```

```
# 3rd cluster states
```

```
us_arrests[clusters_scaled == 3,]
```

```
##           Murder Assault UrbanPop Rape
## Arkansas      8.8    190      50 19.5
## Connecticut    3.3    110      77 11.1
## Delaware       5.9    238      72 15.8
## Hawaii         5.3     46      83 20.2
## Idaho          2.6    120      54 14.2
## Indiana        7.2    113      65 21.0
## Iowa           2.2     56      57 11.3
## Kansas         6.0    115      66 18.0
## Kentucky       9.7    109      52 16.3
## Maine          2.1     83      51  7.8
## Massachusetts  4.4    149      85 16.3
## Minnesota      2.7     72      66 14.9
## Missouri       9.0    178      70 28.2
## Montana        6.0    109      53 16.4
## Nebraska       4.3    102      62 16.5
## New Hampshire  2.1     57      56  9.5
## New Jersey     7.4    159      89 18.8
## North Dakota   0.8     45      44  7.3
## Ohio           7.3    120      75 21.4
## Oklahoma       6.6    151      68 20.0
## Oregon         4.9    159      67 29.3
## Pennsylvania   6.3    106      72 14.9
## Rhode Island   3.4    174      87  8.3
## South Dakota   3.8     86      45 12.8
## Utah           3.2    120      80 22.9
## Vermont        2.2     48      32 11.2
## Virginia       8.5    156      63 20.7
## Washington     4.0    145      73 26.2
## West Virginia  5.7     81      39  9.3
## Wisconsin      2.6     53      66 10.8
## Wyoming        6.8    161      60 15.6
```

(e) Does scaling the variables change the clustering results? Why? In your opinion, should the variables be scaled before the inter-observation dissimilarities are computed?

Scaling the variables before hierarchical clustering does affect the clustering results. Hierarchical clustering and methods like complete linkage and Euclidean distance are sensitive to the scale of variables. When variables are on different scales or have different variances, those with larger variances can dominate the distance computations. This can lead to clusters being formed primarily based on the variables with larger scales or variances, rather than considering the overall patterns across all variables equally. In my opinion, scaling the variables before computing inter-observation dissimilarities can be beneficial. It helps to ensure that all variables contribute equally to the distance computations, preventing dominance by variables with larger scales. This can lead to more balanced and meaningful clusters that capture the underlying patterns in the data more accurately. However, it's important to keep in mind there are certain situations scaling is necessary and scaling is not necessary. For example, if the variables are already on similar scales or if the specific research question or domain knowledge suggests that variable scaling should not be applied, then clustering without scaling might be more appropriate.