

## Data Mining Assignment 5

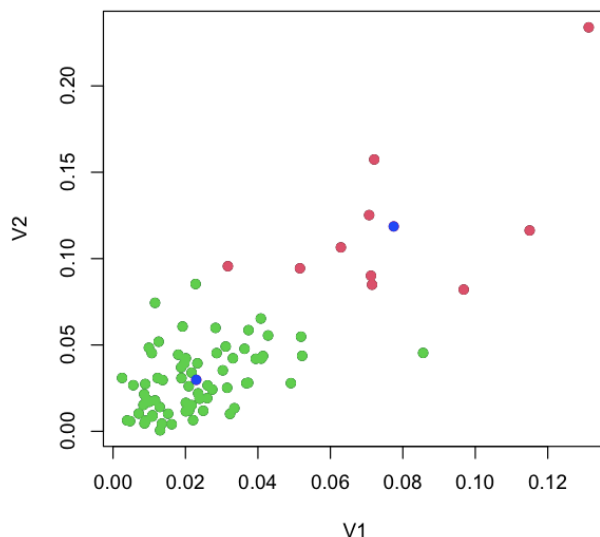
1) Read Chapter 8 (Sections 8.1 and 8.2) and Chapter 2 (Section 2.4).

2) Use `Kmeans()` with all the default values to find the  $k=2$  solution for the first two columns of the sonar test data. Plot these two columns. Also plot the fitted cluster centers using a different color. Finally use the `knn()` function to assign the cluster membership for the points to the nearest cluster center. Color the points according to their cluster membership. Show your R commands for doing so.

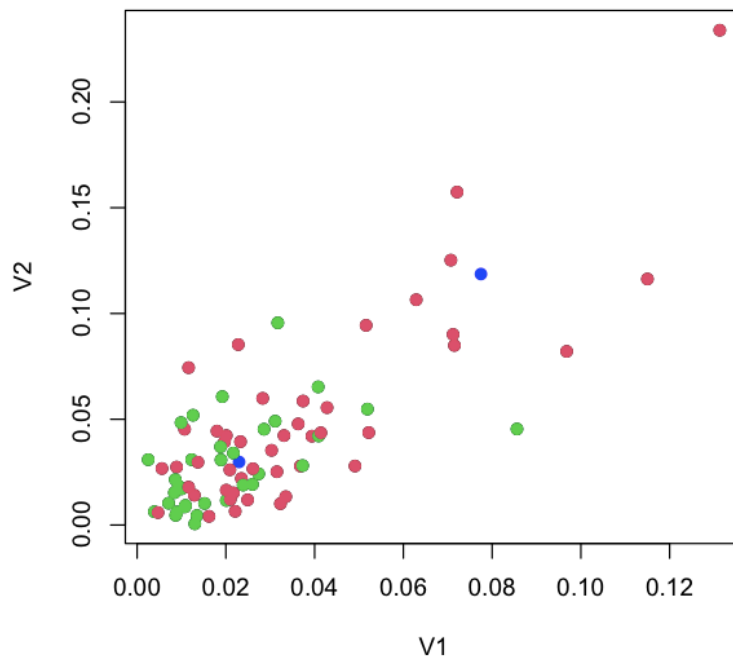
```
> test <- read.csv("sonar_test.csv", header=F)
> sonar <- test[,1:2]
> results <- kmeans(sonar, 2)
> plot(sonar, pch=19, xlab="V1", ylab="V2")
> points(results$centers, pch=19, col="red")

> library(class)
> model <- knn(results$centers, sonar, as.factor(c(-1,1)))
> points(sonar, col=1+1*as.numeric(model), pch=19)
```

3) Graphically compare the cluster memberships from the previous problem to the actual labels in the test data. Also compute the misclassification error that would result if you used your clustering rule to classify the data. Show your R commands for doing so.



```
> model1 <- knn(sonar, sonar, y_test)
```

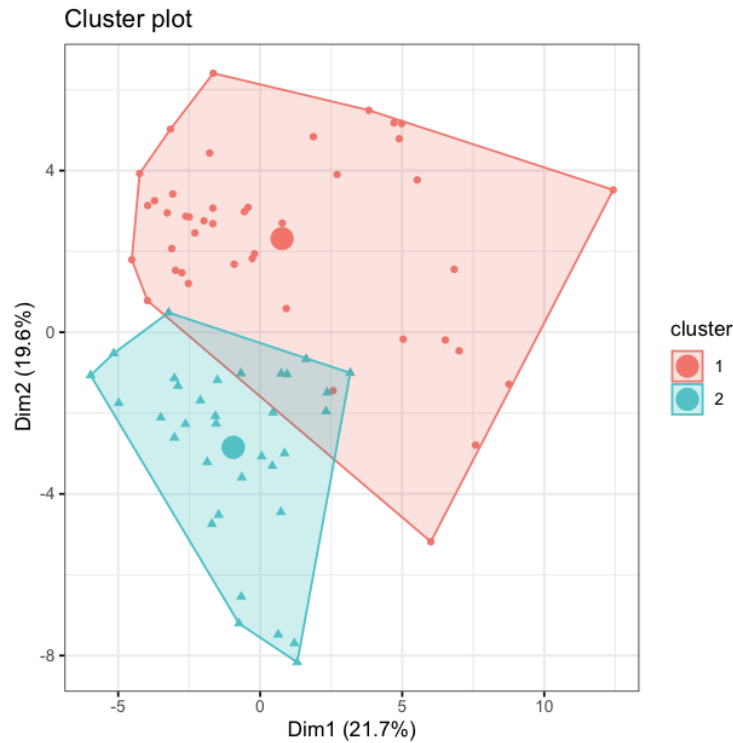


```
> 1 - sum(y_test == model) / length(y_test)
[1] 0.474359
```

4) Repeat the previous problem using all 60 columns. Show your R commands for doing so.

```
> x_test <- test[,1:60]
> fit <- kmeans(x_test, 2)

> library(ggpubr)
> fviz_cluster(fit, data=x_test,
  geom="point", ellipse.type="convex", ggtheme=theme_bw()) +
  stat_mean(aes(color=cluster), size=5)
```



```
> fit_knn <- knn(fit$centers, x_test, as.factor(c(-1,1)))
> 1 - sum(y_test == fit_knn) / length(y_test)
[1] 0.5641026
```

5) Consider the one dimensional data set given  $x \leftarrow c(1, 2, 2.5, 3, 3.5, 4, 4.5, 5, 7, 8, 8.5, 9, 9.5, 10)$ . Starting with initial cluster center values of 1 and 2 carry out algorithm 10 until convergence by hand for  $k=2$  clusters. Show all your work for each step and be sure to say specifically which points are in each cluster at each step.

Given:

$$X = \{1, 2, 2.5, 3, 3.5, 4, 4.5, 5, 7, 8, 8.5, 9, 9.5, 10\}$$

$k = 2$  clusters.

$k$  initial centers = 1, 2  $\{c_1 = 1, c_2 = 2\}$

assign points  
closest to the  
centers

1. cluster 1 points =  $\{1\}$   
cluster 2 points =  $\{2, 2.5, 3, 3.5, 4, 4.5, 5, 7, 8, 8.5, 9, 9.5, 10\}$

2. Recompute the centers:

$$c_1 = 1/1 = 1 //$$

$$c_2 = 76.5/13 = 5.88 //$$

(Repeat step 1 & 2)

3. cluster 1 points =  $\{1, 2, 2.5, 3, 3.5\}$   
cluster 2 points =  $\{4, 4.5, 5, 7, 8, 8.5, 9, 9.5, 10\}$

4.  $c_1 = 12/5 = 2.4 //$   
 $c_2 = 65.5/9 = 7.27 //$

5. cluster 1 points =  $\{1, 2, 2.5, 3, 3.5, 4, 4.5, 5\}$   
cluster 2 points =  $\{7, 8, 8.5, 9, 9.5, 10\}$

6.  $c_1 = 25.5/8 = 3.19 //$

$$c_2 = 52/6 = 8.67 //$$

7. cluster 1 points =  $\{1, 2, 2.5, 3, 3.5, 4, 4.5, 5\}$   
cluster 2 points =  $\{7, 8, 8.5, 9, 9.5, 10\}$

8.  $c_1 = 25.5/8 = 3.19 //$

$$c_2 = 52/6 = 8.67 //$$

6) Repeat the previous problem by writing a loop and verify that the final answer is the same and show your R commands for doing so.

```
> x <- c(1,2,2.5,3,3.5,4,4.5,5,7,8,8.5,9,9.5,10)
> c1 <- 1
> c2 <- 2
> cluster1_points <- c()
> cluster2_points <- c()
> iter <- 1
> while (iter <= 10) {
+ for (val in x) {
+ if (dist(rbind(val, c1)) < dist(rbind(val, c2))) {
+ cluster1_points <- c(cluster1_points, val) }
+ else cluster2_points <- c(cluster2_points, val)
+ }
+ temp1 <- c1
+ temp2 <- c2
+ c1 <- mean(cluster1_points)
+ c2 <- mean(cluster2_points)
+ if (temp1 == c1 && temp2 == c2) break
+ else {
+ cluster1_points <- c()
+ cluster2_points <- c()
+ iter <- iter + 1
+ }
+ }
>
> cluster1_points
[1] 1.0 2.0 2.5 3.0 3.5 4.0 4.5 5.0
> cluster2_points
[1] 7.0 8.0 8.5 9.0 9.5 10.0

> c1
[1] 3.1875
> c2
[1] 8.666667
```

7) Verify that the kmeans function gives the same solution for the previous problem when you use all of the default values and show your R commands for doing so.

```

> results <- kmeans(x, 2)
> results
K-means clustering with 2 clusters of sizes 8, 6

Cluster means:
      [,1]
1 3.187500
2 8.666667

Clustering vector:
[1] 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2

Within cluster sum of squares by cluster:
[1] 12.468750  5.833333
(between_SS / total_SS =  84.9 %)

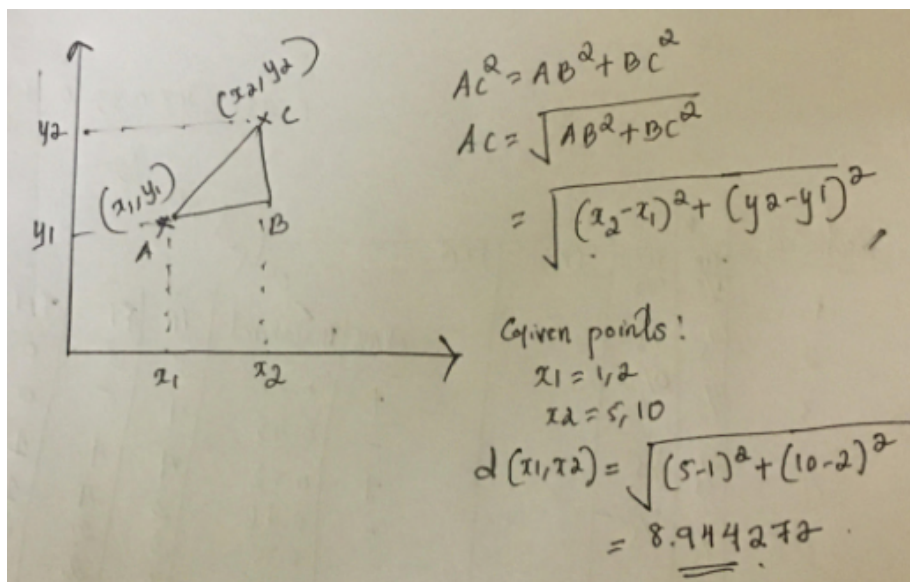
Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"

```

8) Consider the points  $x_1 \leftarrow -c(1, 2)$  and  $x_2 \leftarrow -c(5, 10)$ .

a) Compute the (Euclidean) distance by hand. Show your work and include a picture of the triangle for the Pythagorean Theorem.

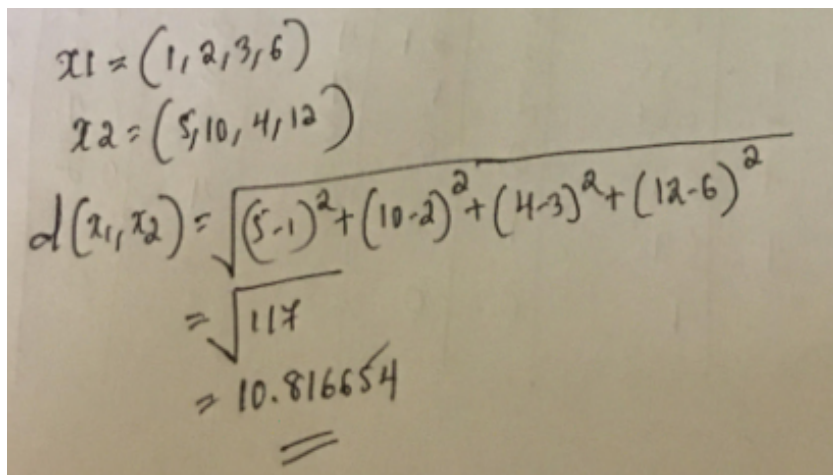


b) Verify that the dist function in R gives the same value as you got in part a. Show your R commands for doing so.

```
> x1 <- c(1,2)
> x2 <- c(5,10)
> dist(rbind(x1,x2))
      x1
x2 8.944272
```

9) Consider the points  $x1 <- c(1,2,3,6)$  and  $x2 <- c(5,10,4,12)$ .

a) Compute the (Euclidean) distance by hand. Show your work.



Handwritten calculation of the Euclidean distance between two points  $x_1 = (1, 2, 3, 6)$  and  $x_2 = (5, 10, 4, 12)$ . The formula used is  $d(x_1, x_2) = \sqrt{(5-1)^2 + (10-2)^2 + (4-3)^2 + (12-6)^2}$ . The calculation simplifies to  $\sqrt{118}$ , which is approximately 10.816654.

b) Verify that the dist function in R gives the same value as you got in part a. Show your R commands for doing so.

```
> x1 <- c(1,2,3,6)
> x2 <- c(5,10,4,12)
> dist(rbind(x1,x2))
      x1
x2 10.81665
```

10) Read Chapter 10.

11) Use a z score cut off of 3 to identify any outliers using the grades for the first midterm at [www.stats202.com/spring2008exams.csv](http://www.stats202.com/spring2008exams.csv). Are there any outliers according to the  $z=\pm 3$  rule? What is the value of the largest z score and what is the value of the smallest (most negative) z score? Show your R commands.

```
> zscores <- (spring2008exams$Midterm.1 - mean(spring2008exams$Midterm.1)) /  
sd(spring2008exams$Midterm.1)  
> spring2008exams <- cbind(spring2008exams, zscores=zscores)  
  
> zscores > 3 | zscores < -3  
[1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
[13] FALSE FALSE FALSE FALSE FALSE  
.  
  
> max(zscores)  
[1] 1.84958  
> min(zscores)  
[1] -2.283753  
.
```

12) Use a z score cut off of 3 to identify any outliers using the grades for the second midterm at [www.stats202.com/spring2008exams.csv](http://www.stats202.com/spring2008exams.csv). Are there any outliers according to the  $z=\pm 3$  rule? What is the value of the largest z score and what is the value of the smallest (most negative) z score? Show your R commands.

```
.  
> zscores_1 <- (spring2008exams$Midterm.2 - mean(spring2008exams$Midterm.2)) /  
sd(spring2008exams$Midterm.2)  
> zscores_1 > 3 | zscores_1 < -3  
[1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
[13] FALSE FALSE FALSE FALSE FALSE  
> max(zscores_1)  
[1] 1.299726  
> min(zscores_1)  
[1] -2.396223
```



13) Compute the count of each ip address (1<sup>st</sup> column) in the data stats202log.txt, then use a z score cut off of 3 to identify any outliers for these counts using Excel for the user agent column of the data at [www.stats202.com/stats202log.txt](http://www.stats202.com/stats202log.txt). (The user agent column is the second to last column and the value for it in the first row is "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; .NET CLR 1.1.4322)"). What user agents are identified as outliers using the  $z = \pm 3$  rule on the counts of the user agents? What are the z scores for these outliers? (You do not need to show any work for this problem because you are using Excel.)

Following user agents are identified as outliers using the  $z = \pm 3$  rule on the counts of the user agents, and its corresponding z scores are given as well.

User Agents	Z scores
Mozilla/5.0 (compatible; Google Desktop)	5.94089886
Mozilla/5.0 (Macintosh; U; Intel Mac OS X; en-US; rv:1.8.0.7) Gecko/20060909 Firefox/1.5.0.7	6.76565756
Mozilla/5.0 (Macintosh; U; Intel Mac OS X; en-US; rv:1.8.1.4) Gecko/20070515 Firefox/2.0.0.4	11.4392902
Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.0.12) Gecko/20070508 Firefox/1.5.0.12	5.52851951
Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.1.1) Gecko/20061204 Firefox/2.0.0.1	4.29138146
Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.1.4) Gecko/20070515 Firefox/2.0.0.4	38.1064882
Mozilla/5.0 (X11; U; Linux i686 (x86_64); en-US; rv:1.8.0.11) Gecko/20070327 Ubuntu/dapper-security Firefox/1.5.0.11	21.4738544
Mozilla/5.0 (X11; U; Linux i686 (x86_64); en-US; rv:1.8.1.3) Gecko/20070309 Firefox/2.0.0.3	79.8942624
Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.7.8) Gecko/20050519 Red Hat/1.7.8-0.90.1gg1	3.60408254
Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.8.0.6) Gecko/20060728 Firefox/1.5.0.6	4.42884124
Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.8.1.3) Gecko/20070309 Firefox/2.0.0.3	11.8516696
Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.8.1.4) Gecko/20070515 Firefox/2.0.0.4	8.55263475

14) Identify any outliers more than 1.5 IQR's above the 3<sup>rd</sup> quartile or below the 1<sup>st</sup> quartile. Verify that these are the same outliers found by the boxplot function using the grades for the second midterm at [www.stats202.com/spring2008exams.csv](http://www.stats202.com/spring2008exams.csv). Show your R commands and include the boxplot. Are

any of the grades for the second midterm outliers by this rule? If so, which ones?

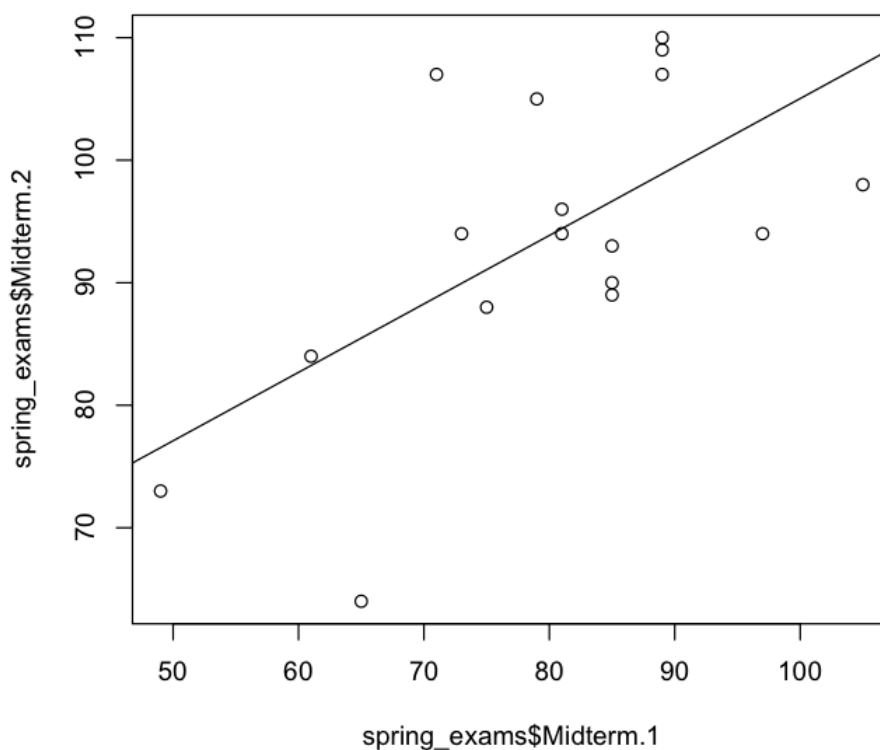
```
> spring_exams <- read.csv("spring2008exams.csv")
> lower_threshold <- quantile(spring_exams$Midterm.2)[2] - (1.5 *
IQR(spring_exams$Midterm.2))
> lower_threshold
25%
65
> upper_threshold <- quantile(spring_exams$Midterm.2)[4] +
(1.5*IQR(spring_exams$Midterm.2))
> upper_threshold
75%
129
> boxplot(spring_exams$Midterm.2)
```

Only one outlier i.e 64 is present in the second midterm grades as per this given rule.



15) Use functions to fit a least squares regression model which predicts the exam 2 score as a function of the exam 1 score for the data spring2008exams.csv. Plot the fitted line and determine for which points the fitted exam 2 values are the furthest from the actual values using the model residuals using the midterm grades at [www.stats202.com/spring2008exams.csv](http://www.stats202.com/spring2008exams.csv). Be sure to include the plot. Which student # had the largest POSITIVE residual? Show your R commands.

```
> plot(spring_exams$Midterm.1, spring_exams$Midterm.2)
> 
> model1 <- lm(spring_exams$Midterm.2 ~ spring_exams$Midterm.1)
> abline(model1)
```



```
> summary(model1)
```

```
Call:
```

```
lm(formula = spring_exams$Midterm.2 ~ spring_exams$Midterm.1)
```

```
Residuals:
```

Min	1Q	Median	3Q	Max
-21.4761	-6.6498	-0.4151	8.1154	18.1718

```
Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t )	
(Intercept)	49.1616	15.2538	3.223	0.00569	**
spring_exams\$Midterm.1	0.5587	0.1883	2.967	0.00959	**

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 10.2 on 15 degrees of freedom
```

```
Multiple R-squared:  0.3699, Adjusted R-squared:  0.3279
```

```
F-statistic: 8.804 on 1 and 15 DF, p-value: 0.009591
```

```
> model1$residuals
```

1	2	3	4	5	6
1.5849223	4.0543983	11.1154462	-9.8235058	18.1717673	8.1154462
7	8	9	10	11	12
-9.3540298	-6.6498157	11.7022913	0.7586124	-21.4761256	-3.6498157
13	14	15	16	17	
-0.4150777	-3.5371735	-3.0629707	-7.6498157	10.1154462	

Points 71, and 65 are the furthest from the actual values as per the residual model.

Student #5 has the largest positive residual.