Homework 3
Introduction to Data Analysis and Mining
Spring 2018
CSCI-B 365

Instructor: Hasan Kurban
Siyi Xian

February 25, 2018

All the work herein is solely mine.

Directions

Please follow the syllabus guidelines in turning in your homework. I am providing the LaTeX of this document too. This homework is due Monday, Feb 26, 2018 10:00p.m. **OBSERVE THE TIME**. Absolutely no homework will be accepted after that time. All the work should be your own. Within a week, AIs can contact students to examine code; students must meet within three days. The session will last no longer than 5 minutes. If the code does not work, the grade for the program may be reduced. Lastly, source code cannot be modified post due date.

k-means Algorithm in Theory

This part is provided to help you implement k-means clustering algorithm.

```
ALGORITHM k-means
INPUT (data $\Delta$, distance $d : \Delta^2 \rightarrow \mathbb{R}_{\geq 0}$, centoid number $k$, threshold $\tau$)
OUTPUT (Set of centoids $\{c_1, c_2, \ldots, c_k\}$)

*** $Dom(\Delta)$ denotes domain of data.

*** Assume centroid is structure $c = (v \in DOM(\Delta), B \subseteq \Delta)$
*** $c.v$ is the centroid value and $c.B$ is the set of nearest points.
*** $c^i$ means centroid at $i^{th}$ iteration.

$i = 0$
*** Initialize Centroids
for $j = 1, k$ do
    $c^i_j.v \leftarrow random(Dom(\Delta))$
    $c^i_j.B \leftarrow \emptyset$
end for

repeat
    $i \leftarrow i + 1$
    *** Assign data point to nearest centroid
    for $\delta \in \Delta$ do
        $c^i_j.B \leftarrow c.B \cup \{\delta\}$, where $\min_{c^i_j}\{d(\delta, c^i_j.v)\}$
    end for
    for $j = 1, k$ do
        *** Get size of centroid
        $n \leftarrow |c_j^i.B|$
        *** Update centroid with average
        $c_j^i.v \leftarrow (1/n) \sum_{\delta \in c_j^i.B} \delta$
        *** Remove data from centroid
        $c_j^i.B \leftarrow \emptyset$
    end for
    *** Calculate scalar product (abuse notation and structure slightly)
    *** See notes
until $((1/k) \sum_{j=1}^{k} ||c_j^{i-1} - c_j^i||) < \tau$
return $(\{c_1^i, c_2^i, \ldots, c_k^i\})$
```

k-means on a tiny data set.

Here are the inputs:

$$\begin{aligned} \Delta &= \{(2,5),(1,5),(22,55),(42,12),(15,16)\} \qquad (1) \\ d((x_1,y_1),(x_2,y_2)) &= [(x_1-x_2)^2+(y_1-y_2)^2)]^{1/2} \qquad (2) \\ k &= 2 \qquad (3) \\ \tau &= 10 \qquad (4) \end{aligned}$$

Observe that $Dom(\Delta) = \mathbb{R}^2$. We now work through k-means. We ignore the uninformative assignments. We remind the reader that T means transpose.

```
$i \leftarrow 0$
*** Randomly assign value to first centroid.
$c_1^0.v \leftarrow random(Dom(\Delta)) = (16, 19)$
*** Randomly assign value to second centroid.
$c_2^0.v \leftarrow random(Dom(\Delta)) = (2, 5)$
$i \leftarrow i + 1$
*** Associate each datum with nearest centroid
$c_1^1.B = \{(22,55),(42,12),(15,16)\}$
$c_2^1.B = \{(2,5),(1,5)\}$
*** Update centroids
$c_1^1.v \leftarrow (26.3, 27.7) = (1/3)((22,55) + (42,12) + (15,16))$
$c_2^1.v \leftarrow (1.5, 5) = (1/2)((2,5) + (1,5))$
*** The convergence condition is split over the next few lines to explicitly show the calculations
$(1/k) \sum_{j=1}^{k} ||c_j^{i-1} - c_j^i|| = (1/2)(||c_1^0 - c_1^1|| + ||c_2^0 - c_2^1||) = (1/2)(||\binom{2}{5} - \binom{1.5}{5}|| + ||\binom{16}{19} - \binom{26.3}{27.7}||)$
$= (1/2)[((\binom{.5}{0})^{\mathsf{T}} \binom{.5}{0})^{(1/2)} + ((\binom{-9.7}{-8.7})^{\mathsf{T}} \binom{-9.7}{-8.7})^{(1/2)})] = (1/2)(\sqrt{.5} + \sqrt{169.7}) \sim (1/2)(13.7) = 6.9$
Since the threshold is met $(6.9 < 10)$, $k$-means stops, returning $\{(26.3, 27.7), (1.5, 5)\}$
```

Problem 1 [10 points]

Answer the following questions for Ionosphere Data Set:

1.1 Briefly describe this data set–what is its purpose? How should it be used? What are the kinds of data it's using?

This radar data was collected by a system in Goose Bay, Labrador. This system consists of a phased array of 16 high-frequency antennas with a total transmitted power on the order of 6.4 kilowatts. See the paper for more details. The targets were free electrons in the ionosphere. "Good" radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" returns are those that do not; their signals pass through the ionosphere.

Received signals were processed using an autocorrelation function whose arguments are the time of a pulse and the pulse number. There were 17 pulse numbers for the Goose Bay system. Instances in this databse are described by 2 attributes per pulse number, corresponding to the complex values returned by the function resulting from the complex electromagnetic signal.

(from description page of the database)

1.2 Using R, show code that answers the following questions:

1.2.1 How many entries are in the data set?

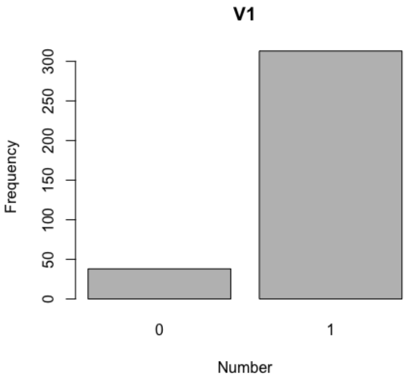
```
> nrow (mydata)
[1] 351
```

1.2.2 How many unknown or missing data are in the data set?

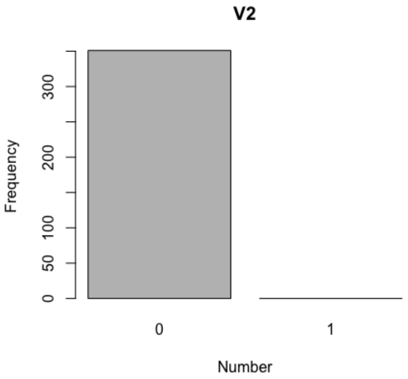
```
> sum (mydata==”?”)
[1] 0
```

1.2.3 Create a bar plot of 1st, 2nd, and 35th variables. Label the plots properly. Discuss the distribution of values *e.g.*, are uniform, skewed, normal. Place images of these bar plots into the document. Show the R code that you used below and discussion below that.

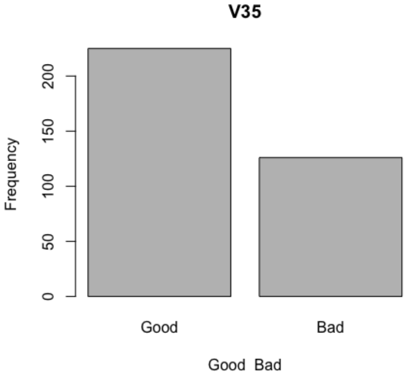
```
v1 <- c(sum(mydata$V1 == ”0”), sum(mydata$V1 == ”1”))
barplot(
     v1,
     xlab = ”Number”,
     ylab = ”Frequency”,
     main = ”V1”,
     names.arg = c(0, 1)
)
```



```
v2 <- c(sum(mydata$V2 == ”0”), sum(mydata$V2 == ”1”))
barplot(
     v2,
     xlab = ”Number”,
     ylab = ”Frequency”,
     main = ”V2”,
     names.arg = c(0, 1)
)
```

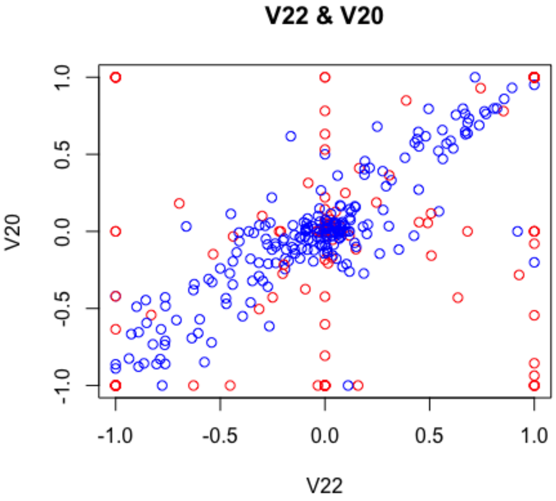


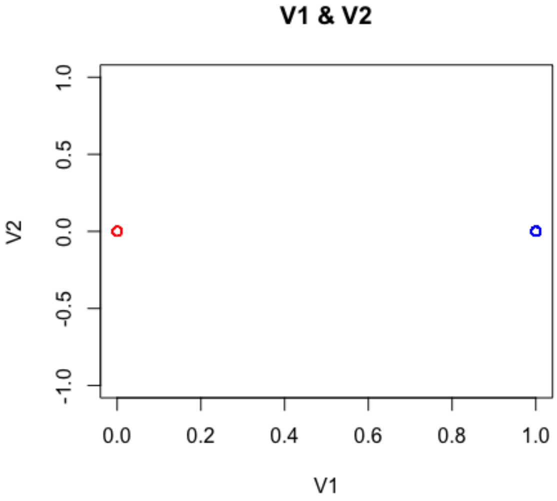
```
v35 <- c(sum(mydata$V35 == "g"), sum(mydata$V35 == "b"))
barplot(
    v35,
    xlab = "Good␣Bad",
    ylab = "Frequency",
    main = "V35",
    names.arg = c("Good", "Bad")
)
```



1.2.4 Make a scatter plots of $[V22, V20]$ and $[V1, V2]$ variables and color the data points with the class variable $[V35]$. Discuss the plots, i.e., do you observe any relationships between variables?

```
chageToColor <- function (x) {
    for (i in 1:nrow(mydata)) {
        if (x[i] == "g") {
            x[i] <- "blue"
        } else {
            x[i] <- "red"
        }
    }
    return (x)
}
v35Color <- chageToColor(mydata$V35)
plot(
    mydata$V22,
    mydata$V20,
    xlab = "V22",
    ylab = "V20",
    main = "V22 & V20",
    col = v35Color
)
plot(
    mydata$V1,
    mydata$V2,
    xlab = "V1",
    ylab = "V2",
    main = "V1 & V2",
    col = v35Color
)
```





Problem 2 [10 points]

The pseudo-code for k-means and a running example of k-means on a small data set are provided above. Answer the following questions:

2.1 Does k-means always converge? Given your answer, a bound on the iterate must be included. How is its value determined?

`$k$-means does not always converge, because for each time with the different initial points, the clusters might be different.`

2.2 What is the run-time of this algorithm?

Problem 3 [20 points]

Implement Lloyd's algorithm for k-means (see algorithm k-means above) in R and call this program C_k. As you present your code explain your protocol for

3.1 initializing centroids

3.2 maintaining k centroids

3.3 deciding ties

3.4 stopping criteria

```r
dis <- function (p1, p2) {
    d <- 0
    for (i in 1:length(p1))
        d <- d + (p1[i] - p2[i]) * (p1[i] - p2[i])
    return (sqrt(d))
}

# stopping criteria
thresHold <- function(c1, c2, t, k, dataSet) {
    ans <- 0
    for (i in 1:k)
        ans <- ans + dis(c1[i, ], c2[i, ])
    ans <- ans / k
    return (ans < t)
}

k_means <- function (dataSet, dis, k, t) {
    # initial central point of each clusters
    len <- ncol(dataSet)
    cv <- matrix(nrow = k, ncol = len)
    cB <- matrix(nrow = k, ncol = nrow(dataSet) + 5)
    c_copy <- cv
    initialC <- sample(1:nrow(dataSet), k, replace = F)
    for (i in 1:k) {
        for (j in 1:len)
            cv[i, j] = dataSet[initialC[i], j]
        cB[i, 1] <- 2
    }

    # maintaining k centroids
    i <- 0
    while (i >= 0) {
        c_copy <- cv

        cB <- matrix(nrow = k, ncol = nrow(dataSet) + 5)
        for (j in 1:k)
            cB[j, 1] <- 2

        lo <- 0
        for (j in 1:nrow(dataSet)) {
            lo <- lo + 1
            minDis <- 0xfffffff
            n <- -1

            for (kk in 1:k) {
                p1 <- as.double(dataSet[j, ])
                p2 <- cv[kk, ]
                if (dis(p1, p2) < minDis) { # deciding ties
                    n <- kk
                    minDis <- dis(p1, p2)
                }
            }
            cB[n, cB[n, 1]] <- lo
            cB[n, 1] <- cB[n, 1] + 1
        }

        for (j in 1:k) {
            co <- c()
            for (kk in 1:len) {
                sum <- 0
                num <- 0
                l <- 2
                while (!is.na(cB[j, l])) {
                    xLoc <- cB[j, l]
                    yLoc <- kk
                    val <- as.double(dataSet[xLoc, yLoc])
                    sum <- sum + val
                    num <- num + 1
                    l <- l + 1
                }
                co <- c(co, sum / num)
            }
            cv[j,] <- co
        }

        # stopping criteria
        if (i > 0 && thresHold(c_copy, cv, t, k, dataSet))
            break
        i <- i + 1
    }

    return (list(v = cv, B = cB))
}

# test case
bigData <- data.frame(mydata[, 1:34])
k_means (bigData, dis, 5, 10)
tinyData <-
    matrix(c(2, 1, 22, 42, 15, 5, 5, 55, 12, 16),
           nrow = 5,
           ncol = 2)
tinyData <- data.frame(tinyData)
k_means (tinyData, dis, 2, 10)
```

Problem 4 [40 points]

In this question, you are asked to run your program, C_k, against Ionosphere Data Set. Upon stopping, you will calculate the quality of the centroids and of the partition. For each centroid c_i, form two counts:

$$\begin{aligned} g_i &\leftarrow \sum_{\delta \in c_i.B} [\delta.C = \text{“g”}], \quad \text{good} \\ b_i &\leftarrow \sum_{\delta \in c_i.B} [\delta.C == \text{“b”}], \quad \text{bad} \end{aligned}$$

where $[x = y]$ returns 1 if True, 0 otherwise. For example, $[2 = 3] + [0 = 0] + [34 = 34] = 2$

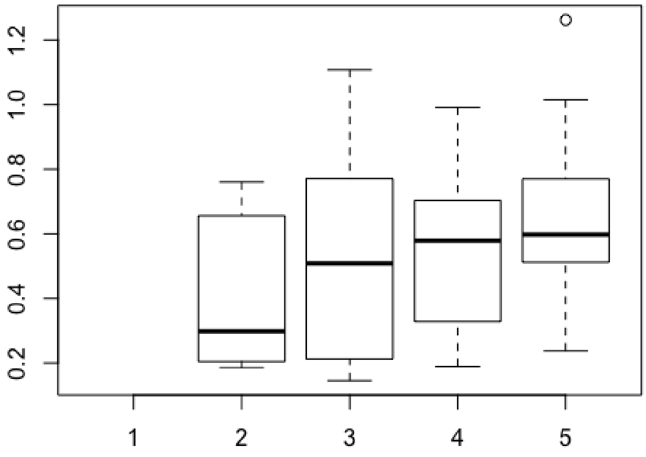
The centroid c_i is classified as good if $g_i > b_i$ and bad otherwise. We can now calculate a simple error rate. Assume c_i is good. Then the error is:

$$error(c_i) = \frac{b_i}{b_i + g_i}$$

We can find the total error rate easily:

$$Error(\{c_1, c_2, \ldots, c_k\}) = \sum_{i=1}^{k} error(c_i)$$

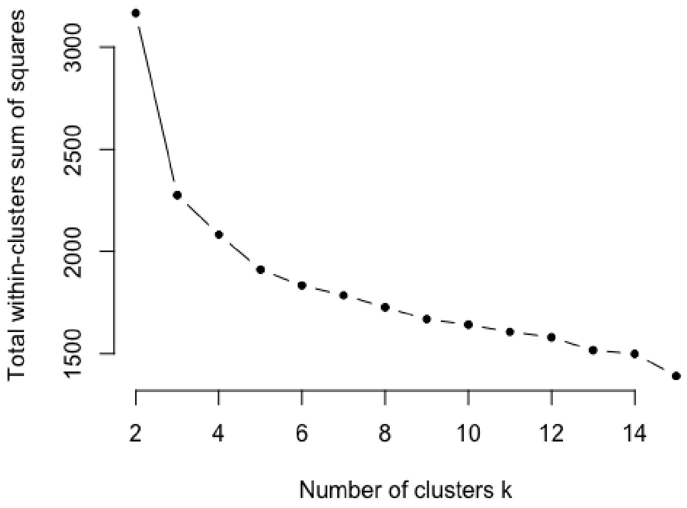
Report the total error rates for $k = 2, \ldots 5$ for 20 runs each, presenting the results that are easily understandable. Plots are generally a good way to convey complex ideas quickly, i.e., box plot. Discuss your results.



For more clusters, there will be larger error rate!

Problem 5 [10 points]

In this question, you are asked to make use of the R package for k-means. Elbow method is one of the techniques to decide the optimal cluster number. Find the optimal cluster number for Ionosphere data set using elbow method (for $2 \leq k \leq 15$). Provide a plot that shows the total SSE for each k. Discuss your results.



I will consider 3 cluster will be better to choose, because through the graph, we can find a obviously turn at 3 clusters.

Problem 6 [10 points]

Let $X \subset \mathbb{R}^n$ ($\mathbb{R}$ is the set of reals) for positive integer $n > 0$. Define a distance $d : X \times X \to \mathbb{R}_{\geq 0}$ as

$$d(x, y) = max\{|x_i - y_i|\}, \forall i \; 1 \leq i \leq n$$

Prove/disprove d is a metric?

```
According to three definition for metric:

(1) d(x, x) = max{|xi - xi|} = 0. And, if x!= y, then there exists some j which x!= y,
    and consequently, |xi - yi| > 0. Therefore, d(x, y) = max{|xi - yi|} > 0
(2) d(x, y) = max{|xi - yi|} = max{|yi - xi|} = d(y, x)
(3) If i = i,...,n. Then,
    |(xi - zi)| = |(xi - yi) + (yi - zj)| <= |xi - yi| + |yi - zi|, by the Triangle Inequality
    Use  max value over i = 1, ..., n of eqution:
    d(x, z) = max{|xi - ji|} <= max{|xi - ji| + |yi - zi|} <= max{|xi - yi|} + max{|yi - zi|} = d(x, y)
    Therefore,  d(x, z) <= d(x, y) + d(y, z)

Therefore, d is a metric.
```

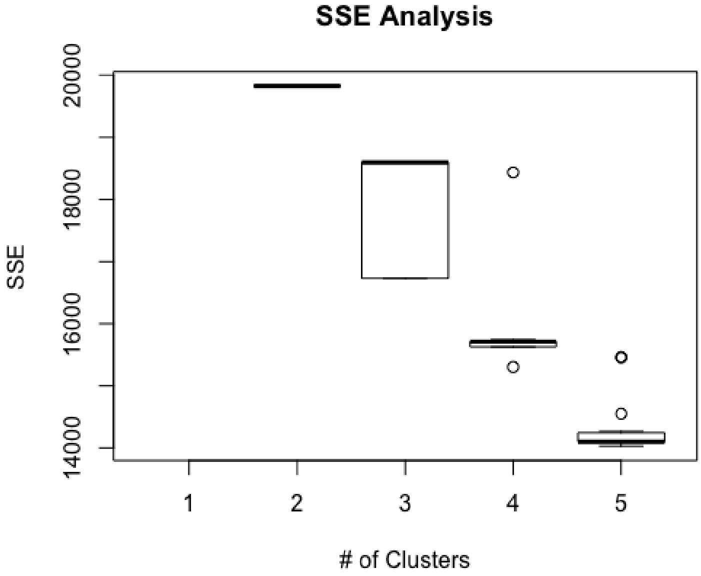
Extra credit [30 points]

This part is optional.

1 The k-means algorithm provided above stops when centroids become stable (Line 34). In theory, k-means converges once SSE is minimized

$$SSE = \sum_{j}^{k} \sum_{x \in c_j.B} ||\mathbf{x} - c_j.v||_2^2$$

In this question, you are asked to use SSE as stopping criterion. Run k-means over Breast Cancer Wisconsin Data Set and report the total SSE in a plot for $k = 2, \ldots, 5$ for 20 runs each. Discuss your results. [15 points].



```
I will consider 3 cluster will be better to choose, because through the graph,
we can find a obviously turn at 3 clusters.
```

2 Traditional k-means initialization is based on choosing values from a uniform distribution. In this question, you are asked to improve k-means through initialization. k-means ++ is an extended k-means clustering algorithm and induces non-uniform distributions over the data that serve as the initial centroids. Read the paper and discuss the idea in a paragraph. Implement this idea to improve your k-means program. [15 points]

What to Turn-in

Submit a .zip file that includes the files below. Name the .zip file as “usename-section number”, i.e., hakurban-B365.

- The *tex and *pdf of the written answers to this document.
- *Rfiles for:
  - implementation of k-means [Problem 4]
  - R package usage [Problem 5]
  - extra credit questions–optional
- A README file that explains how to run your code and other files in the folder