Labs: Trees, Hierarchical Clustering, Heatmaps

Thilanka Munasinghe

Data Analytics

ITWS-4600/ITWS-6600/MATP-4450/CSCI-4960

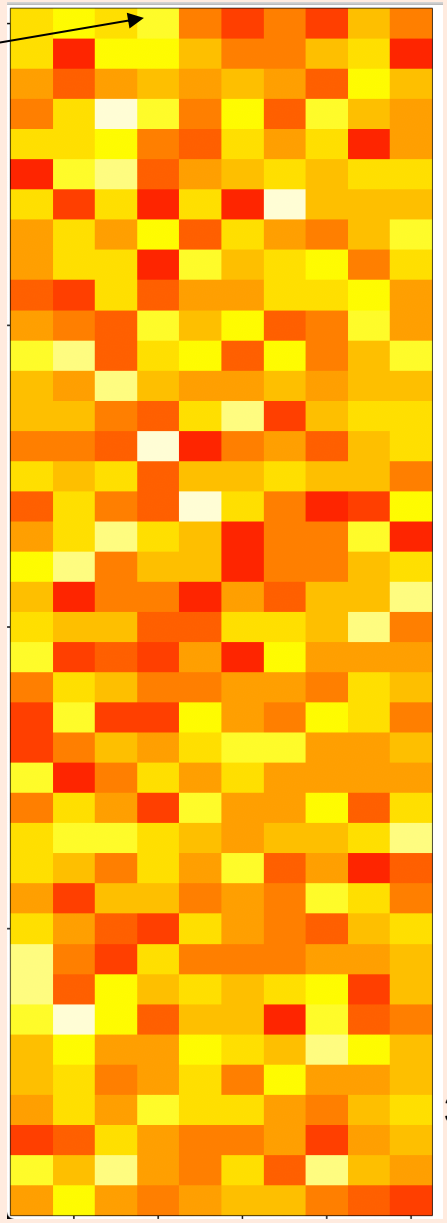
Group 2, Lab-3, Feb 16th 2024

Exercise 1: Heatmap(), image() and hierarchical clustering example

```
# creating a matrix data with random numbers
# and plotting the matrix using the image() function
# you will see there, it does not have a real pattern in the plot.
set.seed(12345)
help(par)
# par can be used to set or query graphical parameters.
# Parameters can be set by specifying them as arguments
# to par in tag = value form, or by passing them as a list of tagged values.
par(mar = rep(0.2,4))
data_Matrix <-matrix(rnorm(400), nrow = 40)
image(1:10, 1:40, t(data_Matrix)[,nrow(data_Matrix):1])
```

Heatmap(), image() and hierarchical clustering example …

There are 40 rows and 10 columns



```
# creating a matrix data with random numbers
# and plotting the matrix using the image() function
# you will see that there it does not have a real pattern in the plot.
set.seed(12345)
par(mar = rep(0.2,4))
data_Matrix <-matrix(rnorm(400), nrow = 40)
image(1:10, 1:40, t(data_Matrix)[,nrow(data_Matrix):1])
```

now we can run a hierarchical cluster analysis on the dataset

we will use the heatmap() function that is available in R

help("heatmap") # read the documentation for
the heatmap() function that is available in
#RStudio

#Read the documentation for rep()

help(rep)

Heatmap(), image() and hierarchical clustering example …

par(mar=rep(0.2,4))

heatmap(data_Matrix)

When we run the heatmap() here, we get the dendrograms printed on the both columns and the rows and still there is no real immerging pattern that is interesting to us,

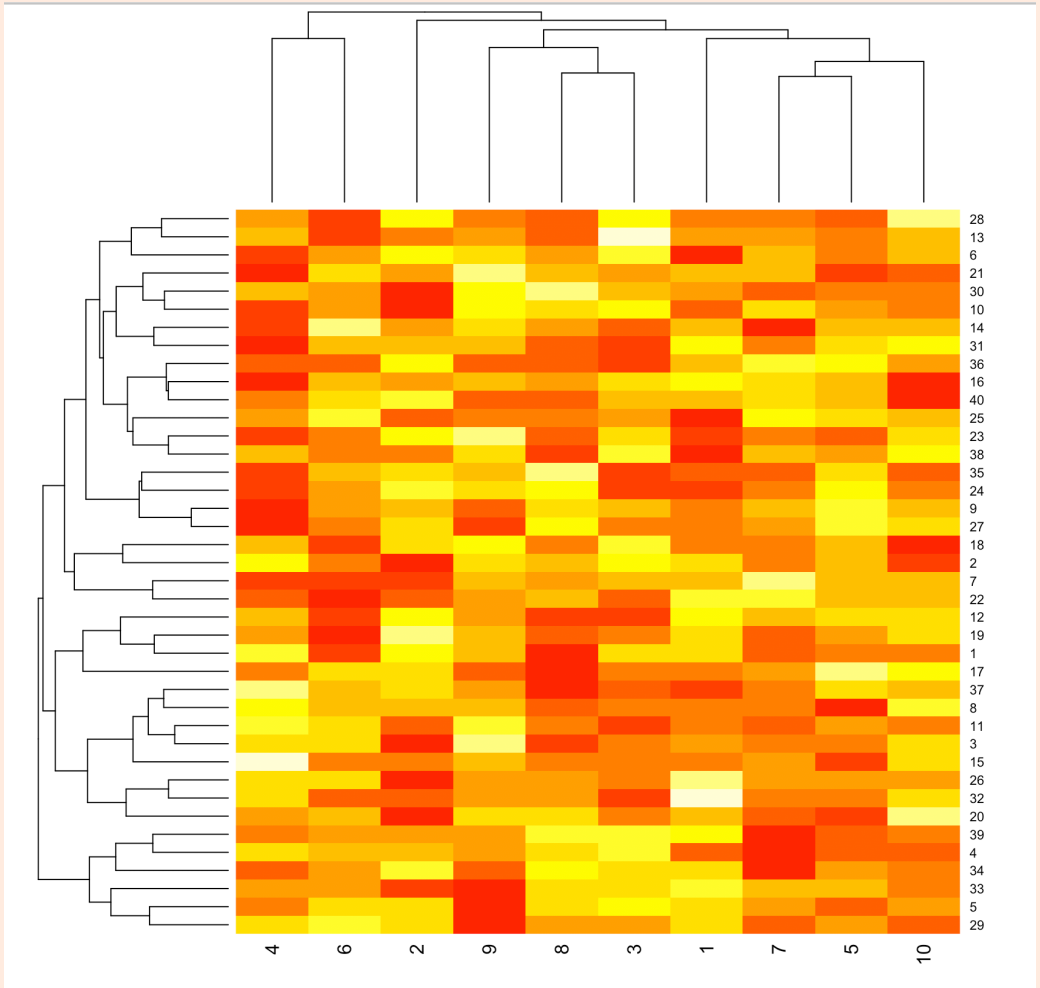
#it is because there is no real interesting pattern underlying in the data we generated.

```
par(mar=rep(0.2,4))
heatmap(data_Matrix)
# When we run the heatmap() here, we get the dendrograms printed on the both coloums,
# and the rows and still there is no real immerging pattern that is interesting to us, it
# is because there is no real interesing pattern underlying in the data we generated.
```

Heatmap(), image() and hierarchical clustering example …

```
par(mar=rep(0.2,4))
heatmap(data_Matrix)
# When we run the heatmap() here, we get the dendrograms printed on the both coloums,
# and the rows and still there is no real immerging pattern that is interesting to us, it
# is because there is no real interesing pattern underlying in the data we generated.
```

Dendrograms printed on columns and rows



Now we will add a pattern to the data by doing a random coin flip.

we will use the rbinom() function along with a for-loop.

help("rbinom") # read the documentation for the rbinom() function that

is available in RStudio

Heatmap(), image() and hierarchical clustering example …

```
set.seed(678910)
for(i in 1:40){
  # flipping a coin and getting the data
  coin_Flip <- rbinom(1, size = 1, prob = 0.5)
  # if the coin is "Heads", add a common pattern to that row,
  if(coin_Flip){
    data_Matrix[i, ] <- data_Matrix[i, ] + rep(c(0,3), each =5)
  }
}
```


```
set.seed(678910) # set seed.
for(i in 1:40){
  # flipping a coin and getting the data
  coin_Flip <- rbinom(1, size = 1, prob = 0.5)
  # if the coin is "Heads", add a common pattern to that row,
  if(coin_Flip){
    data_Matrix[i, ] <- data_Matrix[i, ] + rep(c(0,3), each =5)
  }
}
```

what I did here is, I looped through all the rows and, on a random row, I flipped a coin.

during the coin flip, if is it turn out to be one (true), then, just added a pattern to my data in a way that the five of the columns have a mean of zero and others have mean of three.

```
set.seed(678910) # set seed.
for(i in 1:40){
  # flipping a coin and getting the data
  coin_Flip <- rbinom(1, size = 1, prob = 0.5)
  # if the coin is "Heads", add a common pattern to that row,
  if(coin_Flip){
    data_Matrix[i, ] <- data_Matrix[i, ] + rep(c(0,3), each =5)
  }
}
```

Heatmap(), image() and hierarchical clustering example …

Now we will plot the data

Now we can see that the right hand five columns have more yellow in them,

which means they have a higher value and the left hand five columns that are little bit more in red color which means they have a lower value.

it is because some of the rows have a mean of three in the right hand side, and

some of the rows have mean of zero. Now we have introduced some pattern to it.

Heatmap(), image() and hierarchical clustering example …

Now we will plot the data

Now we can see that the right hand five columns have more yellow in them,

which means they have a higher value and the left hand five columns that are little bit more in red color which means they have a lower value.

it is because some of the rows have a mean of three in the right hand side, and

some of the rows have mean of zero. Now we have introduced some pattern to it.

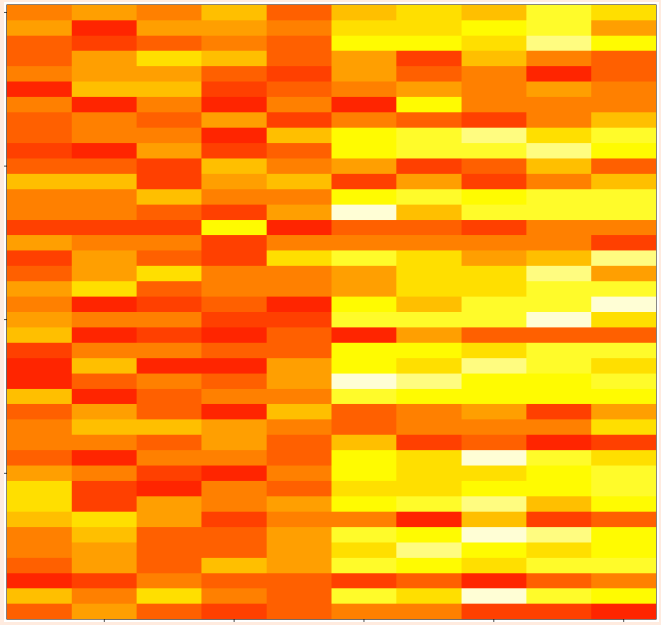
par(mar= rep(0.2, 4))

image(1:10, 1:40, t(data_Matrix)[, nrow(data_Matrix):1])

```
# Now we will plot the data
# Now we can see that the right hand five columns have more yellow in them,
# which means they have a higher value and the left hand five columns that are
# more little bit more red color which means they have a lower value.
# it is becuase some of the rows have a mean of three in the right hand side,and
# some of the rows have mean of zero. Now we have introduce some pattern to it.
par(mar= rep(0.2, 4))
image(1:10, 1:40, t(data_Matrix)[, nrow(data_Matrix):1])
```

Heatmap(), image() and hierarchical clustering example …

```
# Now we will plot the data
# Now we can see that the right hand five columns have more yellow in them,
# which means they have a higher value and the left hand five columns that are
# more little bit more red color which means they have a lower value.
# it is becuase some of the rows have a mean of three in the right hand side,and
# some of the rows have mean of zero. Now we have introduce some pattern to it.
par(mar= rep(0.2, 4))
image(1:10, 1:40, t(data_Matrix)[, nrow(data_Matrix):1])
```



Heatmap(), image() and hierarchical clustering example …

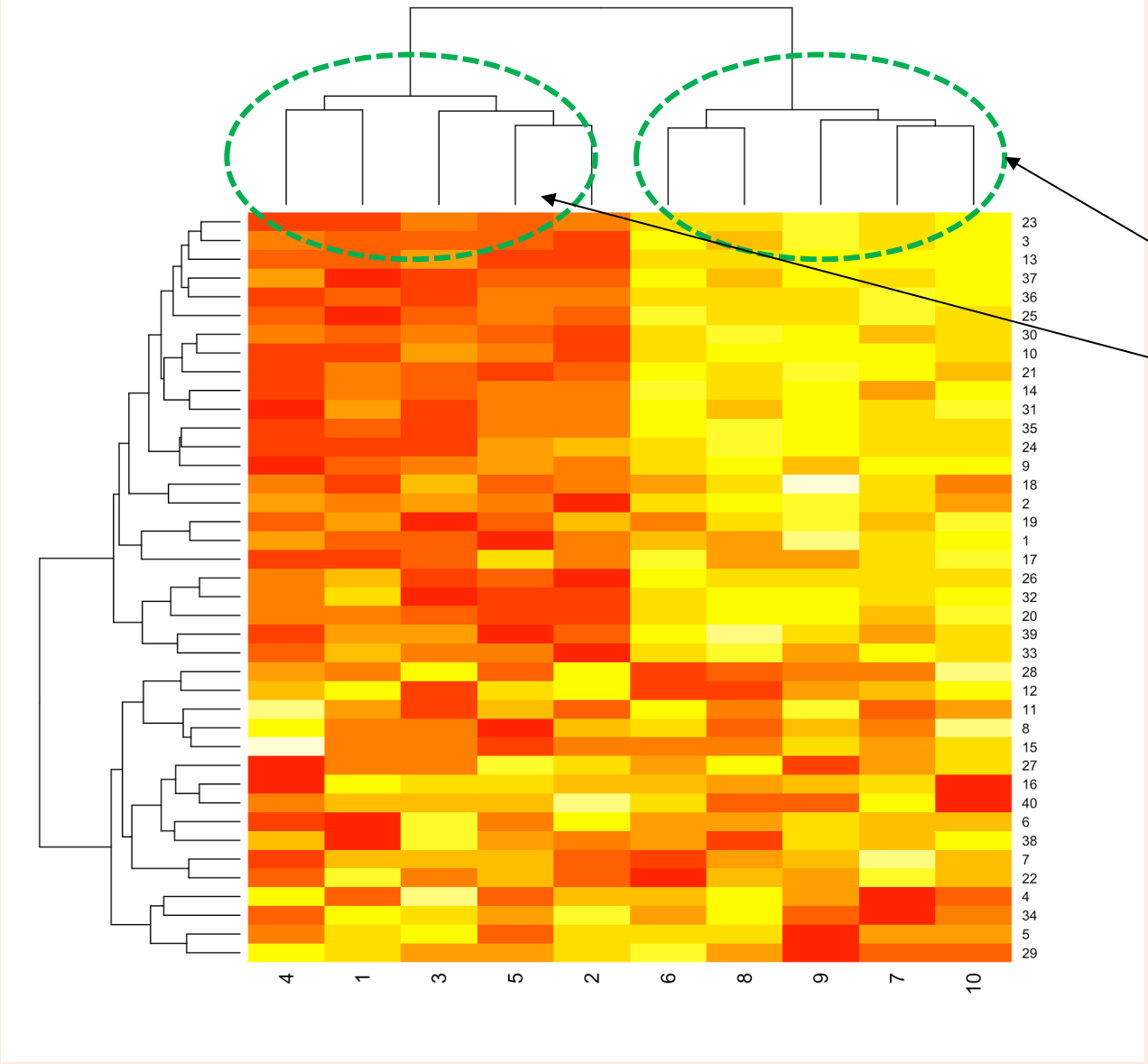
now we will run the heatmap() function on the data, we can see that, two #sets of columns are easily separated.

par(mar=rep(0.2, 4))

heatmap(data_Matrix)

```
# now we will run the heatmap() function on the data, we can see that, two sets of columns are
# easily separated out.
par(mar=rep(0.2, 4))
heatmap(data_Matrix)
```

Heatmap(), image() and hierarchical clustering example …



Separate clusters

Heatmap(), image() and hierarchical clustering example …

The dendrogram is on the top of the of the matrix, (which is on the top of the columns),

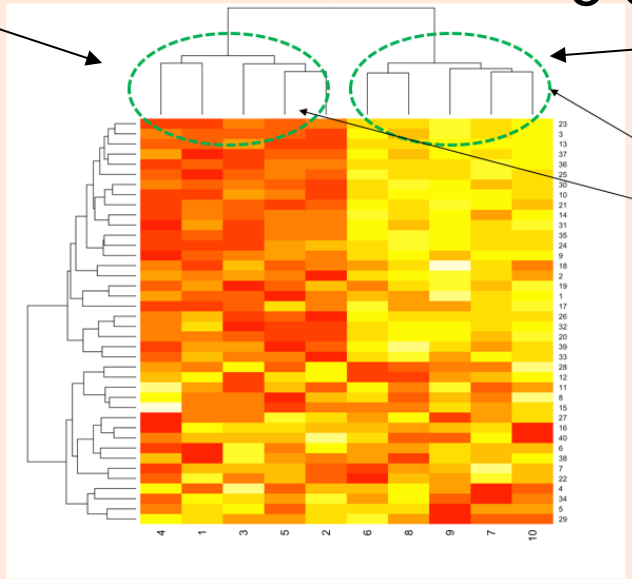
has clearly splits into two separate clusters.

five on the left, and five on the right

on the rows, there is no real pattern that goes along the rows

5 cluster to the Left

5 cluster to the Right



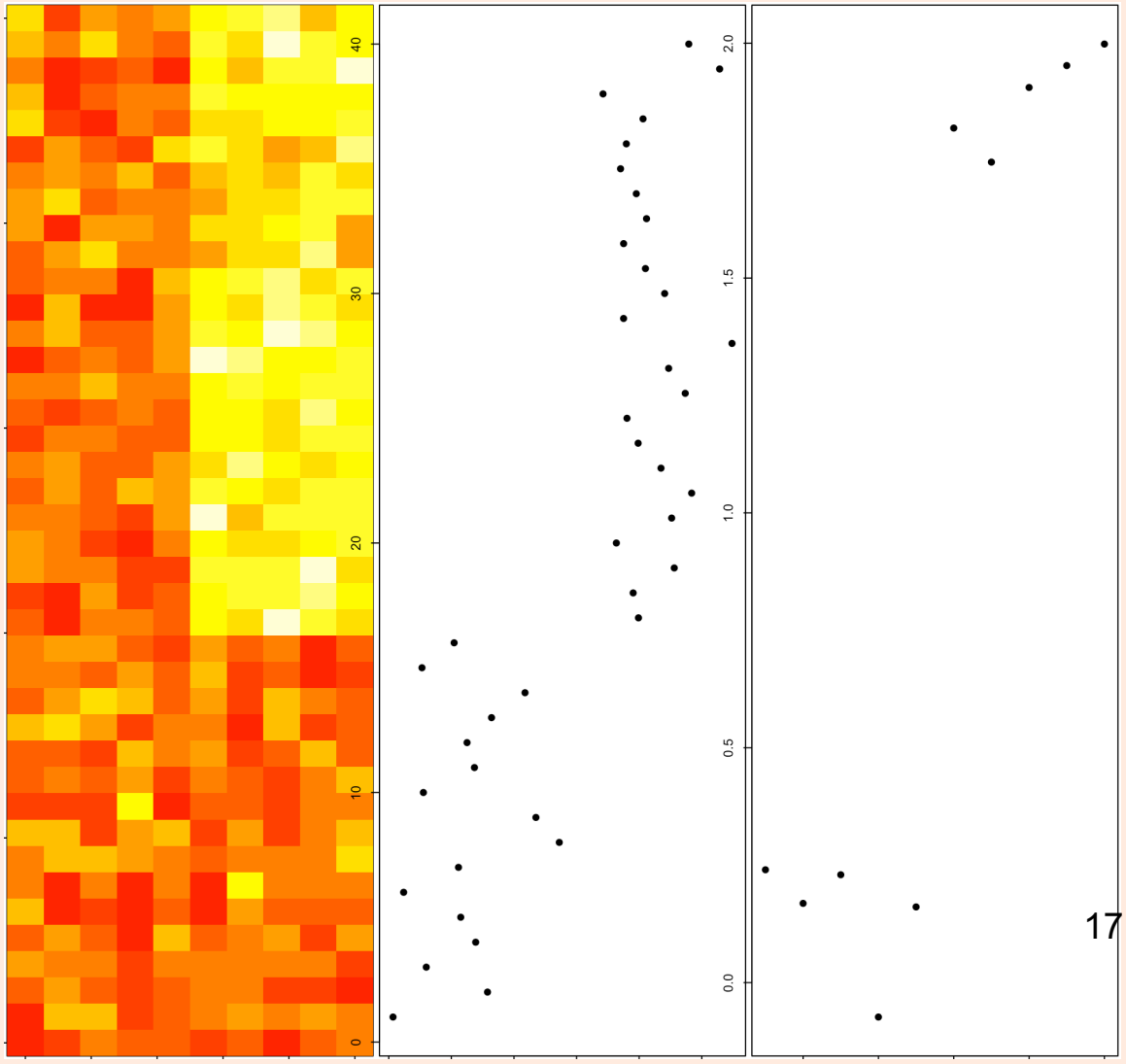
Let's take a closer look at the patters in rows and columns by looking at the marginal

means of the rows and columns.

ten different columns mean and forty different rows means

```
# Let's take a closer look at the patters in rows and colums by looking at the marginal
# means of the rows and colums.
# ten different columns mean and forty different rows means
hh <- hclust(dist(data_Matrix))
data_Matrix_Ordered <- data_Matrix[hh$order,]
par(mfrow = c(1,3))
image(t(data_Matrix_Ordered)[, nrow(data_Matrix_Ordered):1])
plot(rowMeans(data_Matrix_Ordered), 40:1, , xlab = "The Row Mean", ylab = "Row", pch=19)
plot(colMeans(data_Matrix_Ordered), xlab = "Column", ylab = "Column Mean", pch = 19)
```

Pay attention to the number of dots on the middle and right-hand side plot



Interpretation…

left plot has the original data reordered according the the hierarchical cluster analysis

of the rows.

Middle plot has the mean of the each rows.(there are 40 rows and therefore 40 dots representing the mean)

right hand side plot has the means of the each columns (there are 10 columns and therefore 10 dots representing the mean)

Exercise 2: Classification

- Retrieve the abalone.csv dataset
- Predicting the age of abalone from physical measurements.
- The age of abalone is determined by cutting the shell through the cone, staining it, and counting the number of rings through a microscope: a boring and time-consuming task.
- Other measurements, which are easier to obtain, are used to predict the age.
- Perform knn classification to get predictors for Age (Rings). Interpretation not required.

Exercise 3: Clustering

- The Iris dataset (in R use data("iris") to load it)
- The 5th column is the species, and you want to find how many clusters without using that information
- Create a new data frame and remove the fifth column
- Apply kmeans (you choose k) with 1000 iterations
- Use table(iris[,5],<your clustering>) to assess your results

Scripts – work through these

Reminder to finish these code examples See in folder group2/ Lab1

Go over the following scrips,

Lab1_bronx1.R.

Lab1_bronx2.R

Lab1_ctree2.R

Lab1_kknn1.R

Lab1_kknn2.R

Lab1_kknn3.R

Lab1_kmeans1.R

Lab1_nyt.R

Search before you ask! You might need to search your code errors online when you are debugging your code!.

script fragments in R available on the web site:
https://rpi.box.com/s/4rxtho71ko160uprwkubm6rlcwrc42e6

NOTE: you are allowed to work in small groups and discuss during this lab.

Scripts – work through these

Next…

See in folder group2/ Lab3

Go over the following scrips,

Lab3_ctree1.R

Lab3_ctree2.R

Lab3_ctree3.R

…..

And the remaining code snippets in group2/Lab 2 and Lab3

Search before you ask! You might need to search your code errors online when you are debugging your code!

script fragments in R available on the web site:
https://rpi.box.com/s/2xx9ul1fmc6bf5ff8h4jreae69emikmf

NOTE: you are allowed to work in small groups and discuss during this lab.

Trees for the Titanic

data(Titanic)

rpart, ctree, hclust, for:

Survived ~ .

Read the titanic dataset documentation in Rdocumentation:
https://www.rdocumentation.org/packages/titanic/versions/0.1.0