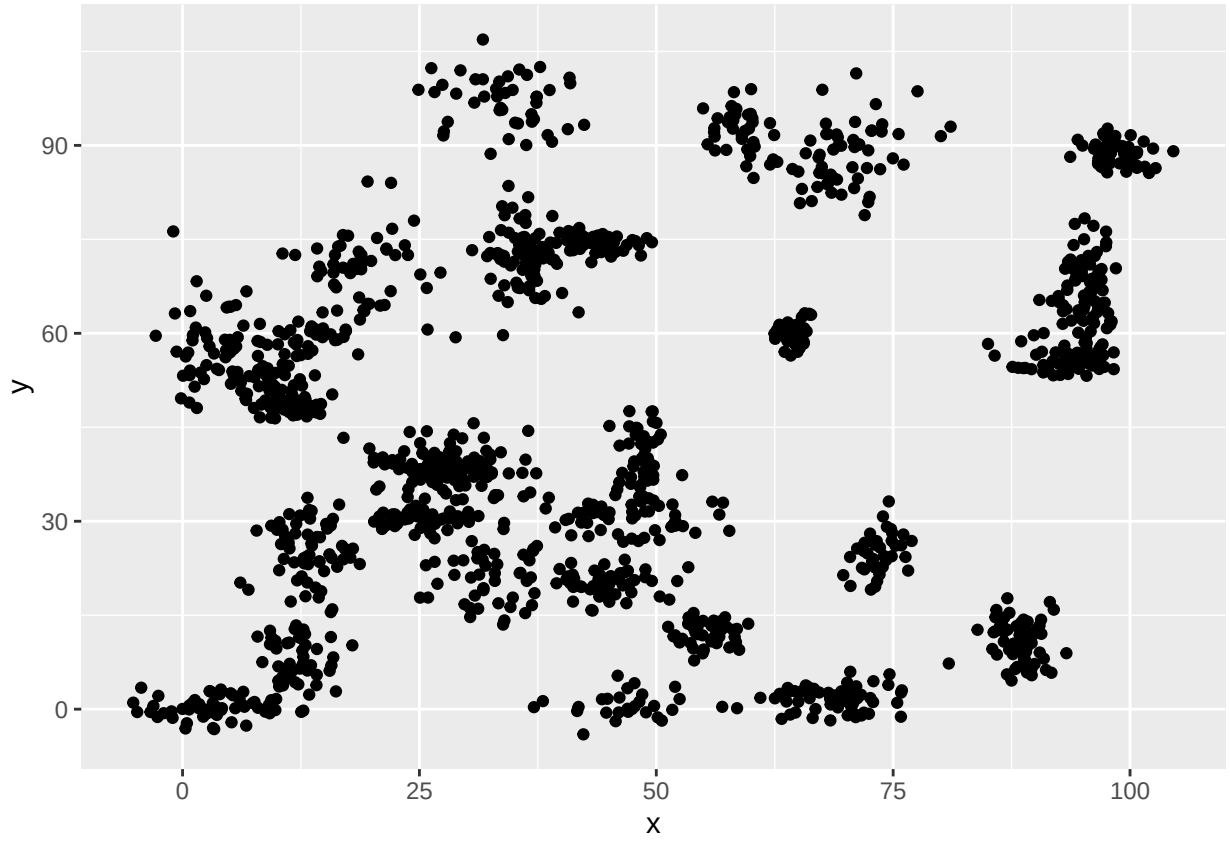
Assignment 11.2.1

Kiran Komati

2021-06-05

```
setwd("C:/Users/kiran/dsc520/data")
library(ggplot2)
library(class)
library(ggplot2)

bcd <- read.csv('binary-classifier-data.csv')
tcd <- read.csv('trinary-classifier-data.csv')
ggplot(bcd,aes(x=x,y=y)) + geom_point()
```



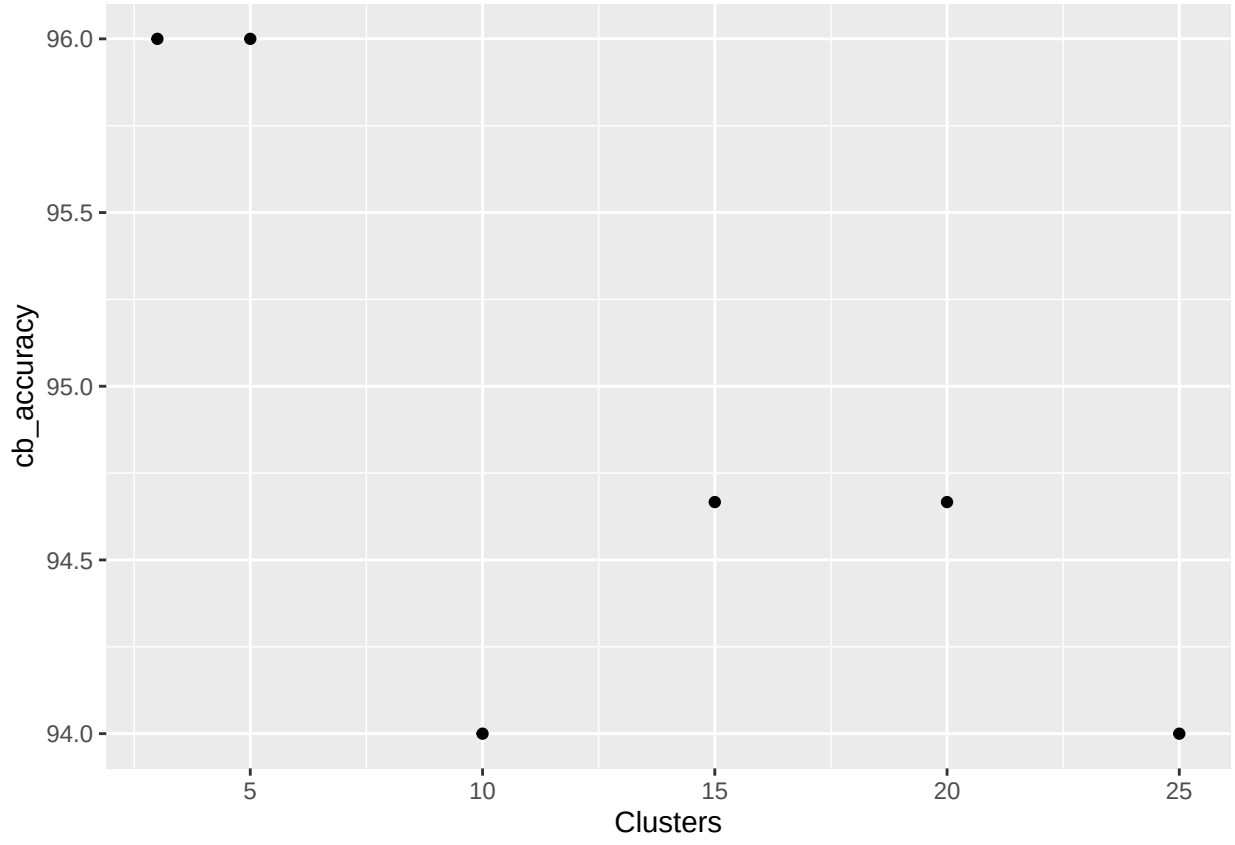

```
random <- sample(1:nrow(bcd), 0.9 * nrow(bcd))
normalize <- function(x) {
    return ((x - min(x)) / (max(x) - min(x))) }
```

```
bcd_norm <- as.data.frame(lapply(bcd[2:3], normalize))
summary(bcd_norm)
```

```
##        x                y
##  Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.2275   1st Qu.:0.2274
##  Median :0.4278   Median :0.4386
##  Mean   :0.4580   Mean   :0.4421
##  3rd Qu.:0.6522   3rd Qu.:0.6556
##  Max.   :1.0000   Max.   :1.0000
```

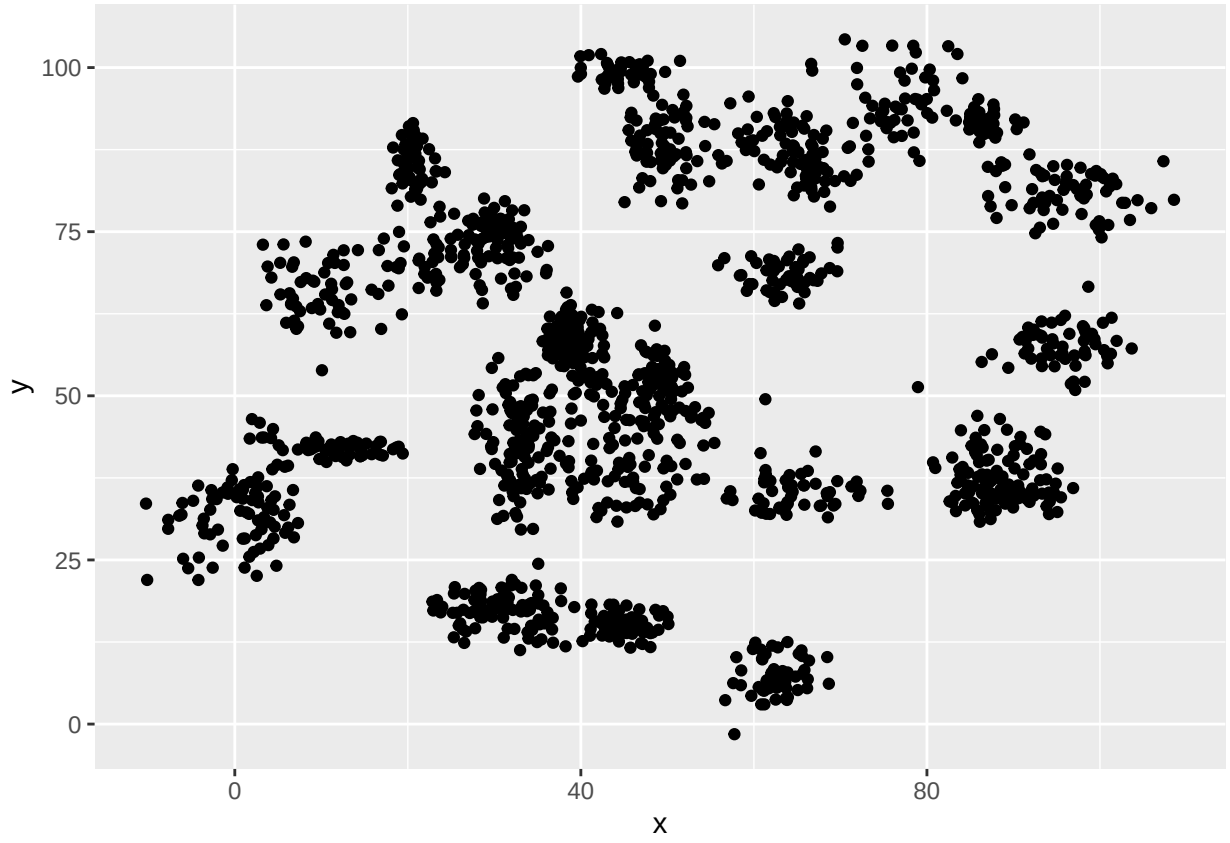
```
##extract training set
bcd_train <- bcd_norm[random,]

##extract testing set
bcd_test <- bcd_norm[-random,]

##extract 1st column of train dataset because it will be used as 'cl' argument in knn function.

bcd_target_category <- bcd[random,1]
##extract 1st column of test dataset to measure the accuracy
bcd_test_category <- bcd[-random,1]

##run knn function with no. of clusters 3
pr_3 <- knn(bcd_train,bcd_test,cl=bcd_target_category,k=3)

##create confusion matrix
tab_3 <- table(pr_3,bcd_test_category)

##this function divides the correct predictions by total number of predictions that tell us how accurat
accuracy <- function(x){sum(diag(x)/(sum(rowSums(x)))) * 100}

a3_bcd<-accuracy(tab_3)

##run knn function with no. of clusters 5
pr_5 <- knn(bcd_train,bcd_test,cl=bcd_target_category,k=5)

##create confusion matrix
tab_5 <- table(pr_5,bcd_test_category)

a5_bcd<-accuracy(tab_5)

##run knn function with no. of clusters 10
pr_10 <- knn(bcd_train,bcd_test,cl=bcd_target_category,k=10)

##create confusion matrix
tab_10 <- table(pr_10,bcd_test_category)

a10_bcd<-accuracy(tab_10)

##run knn function with no. of clusters 15
pr_15 <- knn(bcd_train,bcd_test,cl=bcd_target_category,k=15)
```

```r
##create confusion matrix
tab_15 <- table(pr_15,bcd_test_category)

a15_bcd<-accuracy(tab_15)

##run knn function with no. of clusters 20
pr_20 <- knn(bcd_train,bcd_test,cl=bcd_target_category,k=20)

##create confusion matrix
tab_20 <- table(pr_20,bcd_test_category)

a20_bcd<-accuracy(tab_20)

##run knn function with no. of clusters 25
pr_25 <- knn(bcd_train,bcd_test,cl=bcd_target_category,k=25)

##create confusion matrix
tab_25 <- table(pr_25,bcd_test_category)

a25_bcd<-accuracy(tab_25)

Clusters <- c(3,5,10,15,20,25)
cb_accuracy <- c(a3_bcd,a5_bcd,a10_bcd,a15_bcd,a20_bcd,a25_bcd)
ggplot(data.frame(Clusters,cb_accuracy),aes(x=Clusters,y=cb_accuracy)) + geom_point()
```



Calculating for trinary classifier data

```
tcd <- read.csv('trinary-classifier-data.csv')
ggplot(tcd,aes(x=x,y=y)) + geom_point()
```

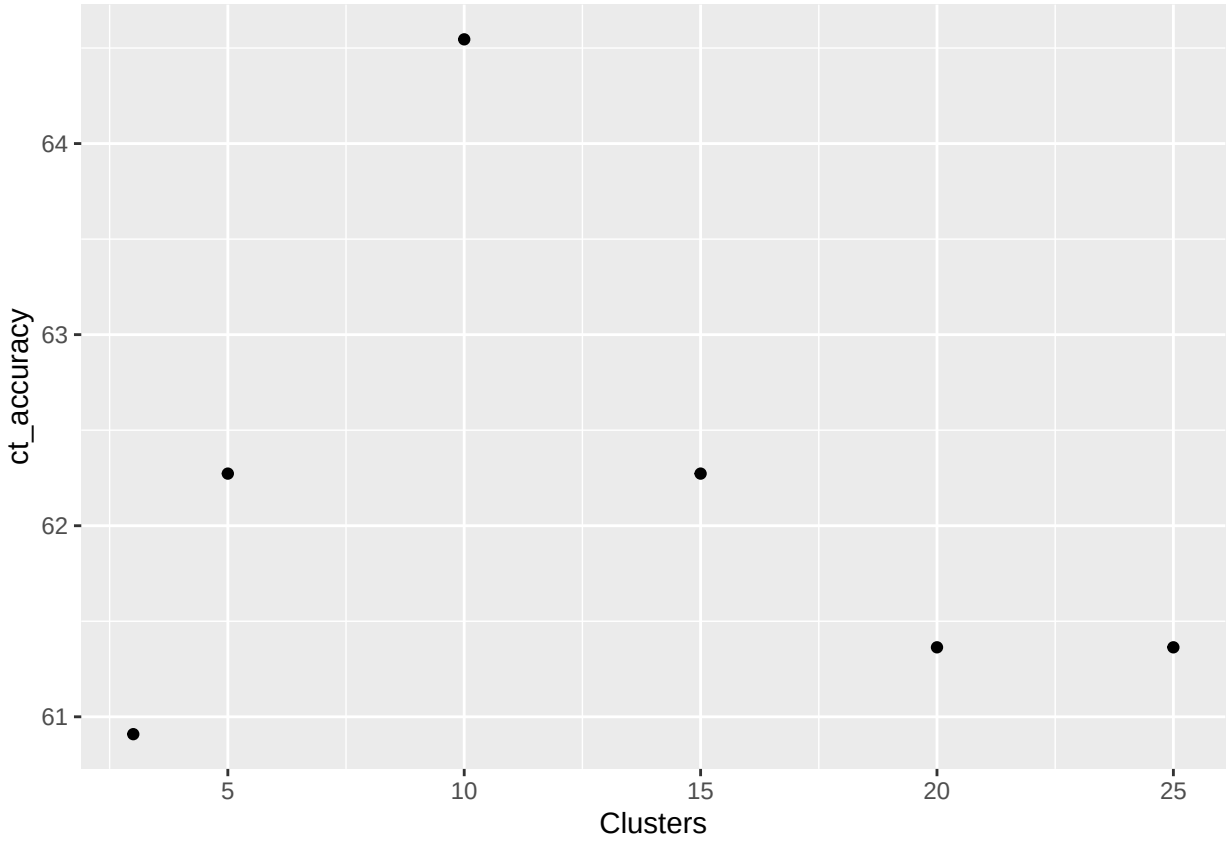



```
tcd_norm <- as.data.frame(lapply(tcd[2:3], normalize))

summary(tcd_norm)
```

```
##        x                y         
##  Min.   :0.0000   Min.   :0.0000  
##  1st Qu.:0.3485   1st Qu.:0.3538  
##  Median :0.4701   Median :0.5349  
##  Mean   :0.4976   Mean   :0.5369  
##  3rd Qu.:0.6441   3rd Qu.:0.7459  
##  Max.   :1.0000   Max.   :1.0000
```

```
##extract training set
tcd_train <- tcd_norm[random,]

##extract testing set
tcd_test <- tcd_norm[-random,]
```

```
##extract 1st column of train dataset because it will be used as 'cl' argument in knn function.
tcd_target_category <- tcd[random,1]

##extract 1st column of test dataset to measure the accuracy
tcd_test_category <- tcd[-random,1]

##run knn function with no. of clusters 3
tpr_3 <- knn(tcd_train,tcd_test,cl=tcd_target_category,k=3)

##create confusion matrix
tab_3 <- table(tpr_3,tcd_test_category)
a3_tcd<-accuracy(tab_3)

##run knn function with no. of clusters 5
tpr_5 <- knn(tcd_train,tcd_test,cl=tcd_target_category,k=5)

##create confusion matrix
tab_5 <- table(tpr_5,tcd_test_category)
a5_tcd<-accuracy(tab_5)

##run knn function with no. of clusters 10
tpr_10 <- knn(tcd_train,tcd_test,cl=tcd_target_category,k=10)

##create confusion matrix
tab_10 <- table(tpr_10,tcd_test_category)
a10_tcd<-accuracy(tab_10)

##run knn function with no. of clusters 15
tpr_15 <- knn(tcd_train,tcd_test,cl=tcd_target_category,k=15)

##create confusion matrix
tab_15 <- table(tpr_15,tcd_test_category)
a15_tcd<-accuracy(tab_15)

##run knn function with no. of clusters 20
tpr_20 <- knn(tcd_train,tcd_test,cl=tcd_target_category,k=20)

##create confusion matrix
tab_20 <- table(tpr_20,tcd_test_category)
a20_tcd<-accuracy(tab_20)

##run knn function with no. of clusters 25
tpr_25 <- knn(tcd_train,tcd_test,cl=tcd_target_category,k=25)

##create confusion matrix
tab_25 <- table(tpr_25,tcd_test_category)
a25_tcd<-accuracy(tab_25)

#plotting clusters against accuracy
ct_accuracy <- c(a3_tcd,a5_tcd,a10_tcd,a15_tcd,a20_tcd,a25_tcd)
ggplot(data.frame(Clusters,ct_accuracy),aes(x=Clusters,y=ct_accuracy)) + geom_point()
```



Accuracy from last week for binary classifier was 58.3. The accuracy is maximum at k=20 with 61.3. Clustering puts subjects into groups. Ideally, the composition of those groups illuminates something about the nature of the sample and the population. we could then use those clusters as an independent variable in a logistic regression.