R Handout for February 19, 2019: CSE 487/587 Data-intensive Computing Spring 2018

#Example 1: Feb 19

x<-seq(0, 100, length.out = 100)

y<-rpois(100,10)

model<-lm(y~x)

model #display the parameters of the model

summary(model)

coefs <- coef(model)

plot(x,y)

abline(coefs[1],coefs[2])

# Simple linear regression

x<-c(1,2,4,6,7,9,10,12,30,50)

y<-c(3,5,7,9,10,11,15,35,40,39)

model<-lm(y~x)

model #display the parameters of the model

summary(model)

coefs <- coef(model)

plot(x,y)

abline(coefs[1],coefs[2])

#Example 2 : Kmeans : Feb 19

library(cluster)

library(fpc)

#set.seed(6)

age<-rpois(100,45)

clust<-kmeans(age,centers=4,iter.max=4)

clust

plotcluster(age,clust$cluster)

clust$centers

clust$size

clust

clust$betweenss/clust$totss

#Example 7 : Knn Iris data set again

data(iris3)

iris3

iris

train <- rbind(iris3[1:25,,1], iris3[1:25,,2], iris3[1:25,,3])

test <- rbind(iris3[26:50,,1], iris3[26:50,,2], iris3[26:50,,3])

cl <- factor(c(rep("s",25), rep("c",25), rep("v",25)))

model <-knn(train, test, cl, k = 3, prob=TRUE) # now LEARN how to classify; train (and validate)

attributes(.Last.value)

plot(model)

query1<-c(5.0, 3.2, 4.9, 2.0) #set unknown data

knn(train, query1, cl, k = 3, prob=TRUE) # now classify query1

query2<-c( 5.1, 3.8 , 1.9, 0.4)

knn(train, query2, cl, k = 3, prob=TRUE)

#Example 8 synthetic data

income<-sample(1000:1000000, 1000) # 1000 income values in the range 1000-100000

age<-sample(21:100,1000,replace=T)

gender<-sample(0:1,1000,replace=T)

data3<-data.frame(age,gender,income) # form a data frame that is required for application of k-nn

train<-data3[1:500,] # half the data is training set

test<-data3[500:1000,] #other half is test set

cl<-factor(sample(0:1,500,replace=T)) # randomly select the classes to be either 1 or 0

model<-knn(train, test, cl, k = 3, prob=TRUE)

plot(model)

query<-c(56,1,90000)

classifyMe <-knn(train, query, cl, k = 3, prob=TRUE)

classifyMe

query<-c(56,1,9000)

classifyMe <-knn(train, query, cl, k = 3, prob=TRUE)

classifyMe

#issue 1: data is not normalized for three attributes (ranges 1:0, 1:100, >1000)

#issue 2: used distance as closeness metric

write.csv(data3, "knn.csv", row.names=FALSE)

income<-sample(1:100, 1000, replace=T) # 1000 income values in the range 1000-100000

age<-sample(21:100,1000,replace=T)

#gender<-sample(0:1,1000,replace=T)

data2<-data.frame(age,income) # form a data frame that is required for application of k-nn

train<-data2[1:500,] # half the data is training set

test<-data2[500:1000,] #other half is test set

cl<-factor(sample(0:1,500,replace=T)) # randomly select the classes to be either 1 or 0

model<-knn(train, test, cl, k = 3, prob=TRUE)

plot(model)

query<-c(56,90)

classifyMe <-knn(train, query, cl, k = 3, prob=TRUE)

classifyMe

query<-c(56,9)

classifyMe <-knn(train, query, cl, k = 3, prob=TRUE)

classifyMe

model<-knn(train, test, cl, k = 5, prob=TRUE)

plot(model)

query<-c(56,90)

classifyMe <-knn(train, query, cl, k = 5, prob=TRUE)

classifyMe

query<-c(56,9)

classifyMe <-knn(train, query, cl, k = 5, prob=TRUE)

classifyMe