library(signal)

library(tuneR)

C<-function(long,high){

t = seq(0, long, 1/8000)

u =(2^15-1)\*sin(2\*pi\*high\*330\*t)

return(u)

}

D<-function(long,high){

t = seq(0, long, 1/8000)

u =(2^15-1)\*sin(2\*pi\*high\*370\*t)

return(u)

}

E<-function(long,high){

t = seq(0, long, 1/8000)

u = (2^15-1)\*sin(2\*pi\*high\*415\*t)

return(u)

}

F<-function(long,high){

t = seq(0, long, 1/8000)

u =(2^15-1)\*sin(2\*pi\*high\*440\*t)

return(u)

}

G<-function(long,high){

t = seq(0, long, 1/8000)

u =(2^15-1)\*sin(2\*pi\*high\*494\*t)

return(u)

}

A<-function(long,high){

t = seq(0, long, 1/8000)

u =(2^15-1)\*sin(2\*pi\*high\*554\*t)

return(u)

}

B<-function(long,high){

t = seq(0, long, 1/8000)

u =(2^15-1)\*sin(2\*pi\*high\*622\*t)

return(u)

}

#谨以此歌献给世界上最伟大的人！！！

u=c(A(1.5,1),G(0.5,1),E(1,1),G(1,1),C(1,2),A(0.5,1),G(0.5,1),A(2,1),E(1,1),G(0.5,1),A(0.5,1),G(1,1),E(1,1),C(0.5,1),A(0.5,0.5),G(0.5,1),E(0.5,1),D(2,1),D(1.5,1),E(0.5,1),G(1,1),G(0.5,1),A(0.5,1),E(1,1),D(1,1),C(2,1),G(1.5,1),E(0.5,1),D(0.5,1),C(0.5,1),A(0.5,0.5),C(0.5,1),G(3,0.5))

w = Wave(u, samp.rate = 8000, bit=16)

play(w)

###对模拟数据的K-Means聚类###

set.seed(12345)

x<-matrix(rnorm(n=100,mean=0,sd=1),ncol=2,byrow=TRUE)

x[1:25,1]<-x[1:25,1]+3 # 令样本数据包含2个自然小类

x[1:25,2]<-x[1:25,2]-4

par(mfrow=c(2,2))

plot(x,main="样本观测点的分布",xlab="",ylab="")

###将样本聚成2类###

set.seed(12345)

(KMClu1<-kmeans(x=x,centers=2,nstart=1))

plot(x,col=(KMClu1$cluster+1),main="K-Means聚类K=2",xlab="",ylab="",pch=20,cex=1.5)

points(KMClu1$centers,pch=3)

###将样本聚成4类，其中nstart=1###

set.seed(12345)

KMClu2<-kmeans(x=x,centers=4,nstart=1)

plot(x,col=(KMClu2$cluster+1),main="K-Means聚类K=4,nstart=1",xlab="",ylab="",pch=20,cex=1.5)

points(KMClu2$centers,pch=3)

KMClu1$betweenss/(2-1)/KMClu1$tot.withinss/(50-2)

KMClu2$betweenss/(4-1)/KMClu2$tot.withinss/(50-4)

###将样本聚成4类，其中nstart=30###

set.seed(12345)

KMClu2<-kmeans(x=x,centers=4,nstart=30)

plot(x,col=(KMClu2$cluster+1),main="K-Means聚类K=4,nstart=30",xlab="",ylab="",pch=20,cex=1.5)

points(KMClu2$centers,pch=3)

# kmeans对iris进行聚类分析

iris2<-iris[,1:4]

iris.kmeans<-kmeans(iris2,3)

iris.kmeans

#用table函数查看分类结果情况

table(iris$Species,iris.kmeans$cluster)

# 1 2 3

# setosa 50 0 0

# versicolor 0 48 2

# virginica 0 14 36

# K-means clustering with 3 clusters of sizes 50, 62, 38

# Cluster means:

# Sepal.Length Sepal.Width Petal.Length Petal.Width

# 1 5.006000 3.428000 1.462000 0.246000

# 2 5.901613 2.748387 4.393548 1.433871

# 3 6.850000 3.073684 5.742105 2.071053

#

# Clustering vector:

# [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 3 2 2

# [56] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 3 3 3 3 2 3 3 3

# [111] 3 3 3 2 2 3 3 3 3 2 3 2 3 2 3 3 2 2 3 3 3 3 3 2 3 3 3 3 2 3 3 3 2 3 3 3 2 3 3 2

#

# Within cluster sum of squares by cluster:

# [1] 15.15100 39.82097 23.87947

# (between\_SS / total\_SS = 88.4 %)

#

# Available components:

#

# [1] "cluster" "centers" "totss" "withinss" "tot.withinss" "betweenss" "size"

# [8] "iter" "ifault"

#在上述属性中，最常用的就是centers和cluster属性，即中心点和聚类的分类集合

#下边我们将分类以及中心点打印出来

plot(iris2$Sepal.Length,iris2$Sepal.Width,col=iris.kmeans$cluster,pch="\*")

points(iris.kmeans$centers,pch="X",cex=1.5,col=4)

library(factoextra)

# 载入数据

data(iris2)

# 数据进行标准化

df <- scale(USArrests)

# 查看数据的前五行

head(df, n = 5)

#确定最佳聚类数目

fviz\_nbclust(df, kmeans, method = "wss") + geom\_vline(xintercept = 4, linetype = 2)