Clustering for protein data

# 1. Protein 데이터

유럽의 25개 나라에서 9가지 식품군으로부터 단백질 소비량에 대한 1973년도 데이터로 단백질 소비 패턴에 따라 나라를 군집화

<https://github.com/WinVector/zmPDSwR/raw/master/Protein/> 에서 protein.txt를 다운 받을 수 있음

setwd("D:/Dropbox/PDSwR/protein/")  
  
# 데이터 읽기  
protein <- read.table("protein.txt", sep="\t", header=TRUE)  
summary(protein)

## Country RedMeat WhiteMeat Eggs   
## Albania : 1 Min. : 4.400 Min. : 1.400 Min. :0.500   
## Austria : 1 1st Qu.: 7.800 1st Qu.: 4.900 1st Qu.:2.700   
## Belgium : 1 Median : 9.500 Median : 7.800 Median :2.900   
## Bulgaria : 1 Mean : 9.828 Mean : 7.896 Mean :2.936   
## Czechoslovakia: 1 3rd Qu.:10.600 3rd Qu.:10.800 3rd Qu.:3.700   
## Denmark : 1 Max. :18.000 Max. :14.000 Max. :4.700   
## (Other) :19   
## Milk Fish Cereals Starch   
## Min. : 4.90 Min. : 0.200 Min. :18.60 Min. :0.600   
## 1st Qu.:11.10 1st Qu.: 2.100 1st Qu.:24.30 1st Qu.:3.100   
## Median :17.60 Median : 3.400 Median :28.00 Median :4.700   
## Mean :17.11 Mean : 4.284 Mean :32.25 Mean :4.276   
## 3rd Qu.:23.30 3rd Qu.: 5.800 3rd Qu.:40.10 3rd Qu.:5.700   
## Max. :33.70 Max. :14.200 Max. :56.70 Max. :6.500   
##   
## Nuts Fr.Veg   
## Min. :0.700 Min. :1.400   
## 1st Qu.:1.500 1st Qu.:2.900   
## Median :2.400 Median :3.800   
## Mean :3.072 Mean :4.136   
## 3rd Qu.:4.700 3rd Qu.:4.900   
## Max. :7.800 Max. :7.900   
##

# 변수 표준화  
vars.to.use <- colnames(protein)[-1]   
pmatrix <- scale(protein[,vars.to.use])   
pcenter <- attr(pmatrix, "scaled:center")   
pscale <- attr(pmatrix, "scaled:scale")

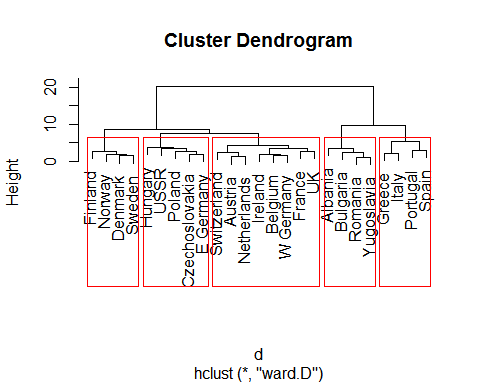
# 2. 계층적 군집화

hclust함수를 이용하여 Ward 방법으로 군집화

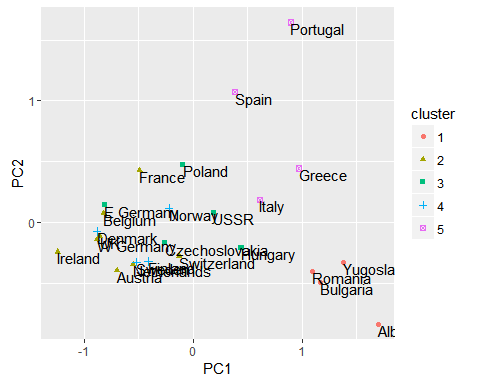
d <- dist(pmatrix, method="euclidean")   
pfit <- hclust(d, method="ward.D")   
plot(pfit, labels=protein$Country)   
  
# 5개의 군집  
rect.hclust(pfit, k=5)  
  
# 군집 결과  
groups <- cutree(pfit, k=5)  
# 각 군집에 속하는 관측값 나열  
print\_clusters <- function(labels, k) { # Note: 1   
 for(i in 1:k) {  
 print(paste("cluster", i))  
 print(protein[labels==i,c("Country","RedMeat","Fish","Fr.Veg")])  
 }  
}  
  
print\_clusters(groups, 5)

## [1] "cluster 1"  
## Country RedMeat Fish Fr.Veg  
## 1 Albania 10.1 0.2 1.7  
## 4 Bulgaria 7.8 1.2 4.2  
## 18 Romania 6.2 1.0 2.8  
## 25 Yugoslavia 4.4 0.6 3.2  
## [1] "cluster 2"  
## Country RedMeat Fish Fr.Veg  
## 2 Austria 8.9 2.1 4.3  
## 3 Belgium 13.5 4.5 4.0  
## 9 France 18.0 5.7 6.5  
## 12 Ireland 13.9 2.2 2.9  
## 14 Netherlands 9.5 2.5 3.7  
## 21 Switzerland 13.1 2.3 4.9  
## 22 UK 17.4 4.3 3.3  
## 24 W Germany 11.4 3.4 3.8  
## [1] "cluster 3"  
## Country RedMeat Fish Fr.Veg  
## 5 Czechoslovakia 9.7 2.0 4.0  
## 7 E Germany 8.4 5.4 3.6  
## 11 Hungary 5.3 0.3 4.2  
## 16 Poland 6.9 3.0 6.6  
## 23 USSR 9.3 3.0 2.9  
## [1] "cluster 4"  
## Country RedMeat Fish Fr.Veg  
## 6 Denmark 10.6 9.9 2.4  
## 8 Finland 9.5 5.8 1.4  
## 15 Norway 9.4 9.7 2.7  
## 20 Sweden 9.9 7.5 2.0  
## [1] "cluster 5"  
## Country RedMeat Fish Fr.Veg  
## 10 Greece 10.2 5.9 6.5  
## 13 Italy 9.0 3.4 6.7  
## 17 Portugal 6.2 14.2 7.9  
## 19 Spain 7.1 7.0 7.2

# 두 주성분을 축으로 데이터를 군집별로 시각  
library(ggplot2)



princ <- prcomp(pmatrix)   
nComp <- 2  
project <- predict(princ, newdata=pmatrix)[,1:nComp]   
project.plus <- cbind(as.data.frame(project),   
 cluster=as.factor(groups),  
 country=protein$Country)  
ggplot(project.plus, aes(x=PC1, y=PC2)) +   
 geom\_point(aes(color=cluster, shape=cluster)) +  
 geom\_text(aes(label=country),  
 hjust=0, vjust=1)



루마니아/유고슬라비아/불가리아/알바니아 군집, 스페인 등 지중해 군집은 다른 군집들과 분리되어 있음

군집분석의 안정성을 알아보기 위해 붓스트랩 표본에 대하여 크기 5인 군집분석 결과 비교. 군집들간의 자카드 측도를 구해 최대값이 0.5보다 작으면 군집이 "분해"되었다고 하고 너무 많이 분해되지 않는 군집은 안정성이 있는 참 군집으로 봄

library(fpc)   
kbest.p<-5   
cboot.hclust <- clusterboot(pmatrix,clustermethod=hclustCBI,   
 method="ward.D", k=kbest.p)

## boot 1   
## boot 2   
## boot 3   
## boot 4   
## boot 5   
## boot 6   
## boot 7   
## boot 8   
## boot 9   
## boot 10   
## boot 11   
## boot 12   
## boot 13   
## boot 14   
## boot 15   
## boot 16   
## boot 17   
## boot 18   
## boot 19   
## boot 20   
## boot 21   
## boot 22   
## boot 23   
## boot 24   
## boot 25   
## boot 26   
## boot 27   
## boot 28   
## boot 29   
## boot 30   
## boot 31   
## boot 32   
## boot 33   
## boot 34   
## boot 35   
## boot 36   
## boot 37   
## boot 38   
## boot 39   
## boot 40   
## boot 41   
## boot 42   
## boot 43   
## boot 44   
## boot 45   
## boot 46   
## boot 47   
## boot 48   
## boot 49   
## boot 50   
## boot 51   
## boot 52   
## boot 53   
## boot 54   
## boot 55   
## boot 56   
## boot 57   
## boot 58   
## boot 59   
## boot 60   
## boot 61   
## boot 62   
## boot 63   
## boot 64   
## boot 65   
## boot 66   
## boot 67   
## boot 68   
## boot 69   
## boot 70   
## boot 71   
## boot 72   
## boot 73   
## boot 74   
## boot 75   
## boot 76   
## boot 77   
## boot 78   
## boot 79   
## boot 80   
## boot 81   
## boot 82   
## boot 83   
## boot 84   
## boot 85   
## boot 86   
## boot 87   
## boot 88   
## boot 89   
## boot 90   
## boot 91   
## boot 92   
## boot 93   
## boot 94   
## boot 95   
## boot 96   
## boot 97   
## boot 98   
## boot 99   
## boot 100

summary(cboot.hclust$result)

## Length Class Mode   
## result 7 hclust list   
## noise 1 -none- logical   
## nc 1 -none- numeric   
## clusterlist 5 -none- list   
## partition 25 -none- numeric   
## clustermethod 1 -none- character  
## nccl 1 -none- numeric

groups<-cboot.hclust$result$partition   
print\_clusters(groups, kbest.p)

## [1] "cluster 1"  
## Country RedMeat Fish Fr.Veg  
## 1 Albania 10.1 0.2 1.7  
## 4 Bulgaria 7.8 1.2 4.2  
## 18 Romania 6.2 1.0 2.8  
## 25 Yugoslavia 4.4 0.6 3.2  
## [1] "cluster 2"  
## Country RedMeat Fish Fr.Veg  
## 2 Austria 8.9 2.1 4.3  
## 3 Belgium 13.5 4.5 4.0  
## 9 France 18.0 5.7 6.5  
## 12 Ireland 13.9 2.2 2.9  
## 14 Netherlands 9.5 2.5 3.7  
## 21 Switzerland 13.1 2.3 4.9  
## 22 UK 17.4 4.3 3.3  
## 24 W Germany 11.4 3.4 3.8  
## [1] "cluster 3"  
## Country RedMeat Fish Fr.Veg  
## 5 Czechoslovakia 9.7 2.0 4.0  
## 7 E Germany 8.4 5.4 3.6  
## 11 Hungary 5.3 0.3 4.2  
## 16 Poland 6.9 3.0 6.6  
## 23 USSR 9.3 3.0 2.9  
## [1] "cluster 4"  
## Country RedMeat Fish Fr.Veg  
## 6 Denmark 10.6 9.9 2.4  
## 8 Finland 9.5 5.8 1.4  
## 15 Norway 9.4 9.7 2.7  
## 20 Sweden 9.9 7.5 2.0  
## [1] "cluster 5"  
## Country RedMeat Fish Fr.Veg  
## 10 Greece 10.2 5.9 6.5  
## 13 Italy 9.0 3.4 6.7  
## 17 Portugal 6.2 14.2 7.9  
## 19 Spain 7.1 7.0 7.2

cboot.hclust$bootmean # 군집 안정성 벡터(최대 자카드의 평균)

## [1] 0.7803333 0.7814921 0.6383135 0.8778929 0.7118333

cboot.hclust$bootbrd # 군집이 분해된 횟수

## [1] 27 13 44 17 41

WSS 계산

# 벡터간의 유클리드 거리 제곱 함수  
sqr\_edist <- function(x, y) {   
 sum((x-y)^2)  
}  
# 한 군집의 WSS 함수  
wss.cluster <- function(clustermat) {   
 c0 <- apply(clustermat, 2, FUN=mean)   
 sum(apply(clustermat, 1, FUN=function(row){sqr\_edist(row,c0)}))   
}  
# 전체 WSS 함수  
wss.total <- function(dmatrix, labels) {   
 wsstot <- 0  
 k <- length(unique(labels))  
 for(i in 1:k)  
 wsstot <- wsstot + wss.cluster(subset(dmatrix, labels==i))   
 wsstot  
}

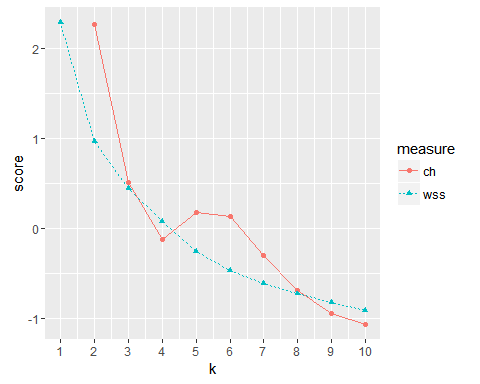
Calinski-Harabasz 지수

* TSS: 전체 제곱합, : 군집수, WSS(k): 군집내 제곱합, BSS(k): 군집간 제곱합
* Calinsik-Harabasz index = [BSS(k)/(k-1)] / [WSS(k)/(n-k)]

# TSS 계산 함수  
totss <- function(dmatrix) {   
 grandmean <- apply(dmatrix, 2, FUN=mean)  
 sum(apply(dmatrix, 1, FUN=function(row){sqr\_edist(row, grandmean)}))  
}  
  
# Calinski-Harabasz 인덱스 계산 함수  
ch\_criterion <- function(dmatrix, kmax, method="kmeans") {   
 if(!(method %in% c("kmeans", "hclust"))) {  
 stop("method must be one of c('kmeans', 'hclust')")  
 }  
 npts <- dim(dmatrix)[1]   
  
 totss <- totss(dmatrix)   
  
 wss <- numeric(kmax)  
 crit <- numeric(kmax)  
 wss[1] <- (npts-1)\*sum(apply(dmatrix, 2, var))   
 for(k in 2:kmax) {   
 if(method=="kmeans") {  
 clustering<-kmeans(dmatrix, k, nstart=10, iter.max=100)  
 wss[k] <- clustering$tot.withinss  
 }else { # hclust   
 d <- dist(dmatrix, method="euclidean")  
 pfit <- hclust(d, method="ward.D")  
 labels <- cutree(pfit, k=k)  
 wss[k] <- wss.total(dmatrix, labels)  
 }  
 }  
 bss <- totss - wss   
 crit.num <- bss/(0:(kmax-1))   
 crit.denom <- wss/(npts - 1:kmax)   
 list(crit = crit.num/crit.denom, wss = wss, totss = totss)   
}

군집수 에 대하여 Ward 방법의 CH 지수와 WSS(k)값 비교

library(reshape2)   
clustcrit <- ch\_criterion(pmatrix, 10, method="hclust") # CH 지수 계산  
critframe <- data.frame(k=1:10, ch=scale(clustcrit$crit),   
 wss=scale(clustcrit$wss))  
critframe <- melt(critframe, id.vars=c("k"),   
 variable.name="measure",  
 value.name="score")  
ggplot(critframe, aes(x=k, y=score, color=measure)) +   
 geom\_point(aes(shape=measure)) + geom\_line(aes(linetype=measure)) +  
 scale\_x\_continuous(breaks=1:10, labels=1:10)



CH 지수는 에서 최대, 에서 국소 최대, WSS는 에서 꺾임

## 연습문제

인 경우에 대하여 결과를 살펴보고 붓스트랩을 이용하여 군집의 안정성을 살펴보시오.

# 3. k-평균 군집

-평균 군집 결과 살펴보기

pclusters <- kmeans(pmatrix, kbest.p, nstart=100, iter.max=100)   
summary(pclusters)

## Length Class Mode   
## cluster 25 -none- numeric  
## centers 45 -none- numeric  
## totss 1 -none- numeric  
## withinss 5 -none- numeric  
## tot.withinss 1 -none- numeric  
## betweenss 1 -none- numeric  
## size 5 -none- numeric  
## iter 1 -none- numeric  
## ifault 1 -none- numeric

pclusters$centers

## RedMeat WhiteMeat Eggs Milk Fish Cereals  
## 1 1.011180399 0.7421332 0.94084150 0.5700581 -0.2671539 -0.6877583  
## 2 -0.508801956 -1.1088009 -0.41248496 -0.8320414 0.9819154 0.1300253  
## 3 -0.570049402 0.5803879 -0.08589708 -0.4604938 -0.4537795 0.3181839  
## 4 -0.807569986 -0.8719354 -1.55330561 -1.0783324 -1.0386379 1.7200335  
## 5 0.006572897 -0.2290150 0.19147892 1.3458748 1.1582546 -0.8722721  
## Starch Nuts Fr.Veg  
## 1 0.2288743 -0.5083895 0.02161979  
## 2 -0.1842010 1.3108846 1.62924487  
## 3 0.7857609 -0.2679180 0.06873983  
## 4 -1.4234267 0.9961313 -0.64360439  
## 5 0.1676780 -0.9553392 -1.11480485

pclusters$size

## [1] 8 4 5 4 4

groups <- pclusters$cluster   
print\_clusters(groups, kbest.p)

## [1] "cluster 1"  
## Country RedMeat Fish Fr.Veg  
## 2 Austria 8.9 2.1 4.3  
## 3 Belgium 13.5 4.5 4.0  
## 9 France 18.0 5.7 6.5  
## 12 Ireland 13.9 2.2 2.9  
## 14 Netherlands 9.5 2.5 3.7  
## 21 Switzerland 13.1 2.3 4.9  
## 22 UK 17.4 4.3 3.3  
## 24 W Germany 11.4 3.4 3.8  
## [1] "cluster 2"  
## Country RedMeat Fish Fr.Veg  
## 10 Greece 10.2 5.9 6.5  
## 13 Italy 9.0 3.4 6.7  
## 17 Portugal 6.2 14.2 7.9  
## 19 Spain 7.1 7.0 7.2  
## [1] "cluster 3"  
## Country RedMeat Fish Fr.Veg  
## 5 Czechoslovakia 9.7 2.0 4.0  
## 7 E Germany 8.4 5.4 3.6  
## 11 Hungary 5.3 0.3 4.2  
## 16 Poland 6.9 3.0 6.6  
## 23 USSR 9.3 3.0 2.9  
## [1] "cluster 4"  
## Country RedMeat Fish Fr.Veg  
## 1 Albania 10.1 0.2 1.7  
## 4 Bulgaria 7.8 1.2 4.2  
## 18 Romania 6.2 1.0 2.8  
## 25 Yugoslavia 4.4 0.6 3.2  
## [1] "cluster 5"  
## Country RedMeat Fish Fr.Veg  
## 6 Denmark 10.6 9.9 2.4  
## 8 Finland 9.5 5.8 1.4  
## 15 Norway 9.4 9.7 2.7  
## 20 Sweden 9.9 7.5 2.0

의 선택

# CH 지수  
clustering.ch <- kmeansruns(pmatrix, krange=1:10, criterion="ch")   
clustering.ch$bestk

## [1] 2

# Average silhousette width 기준  
clustering.asw <- kmeansruns(pmatrix, krange=1:10, criterion="asw")   
clustering.asw$bestk

## [1] 3

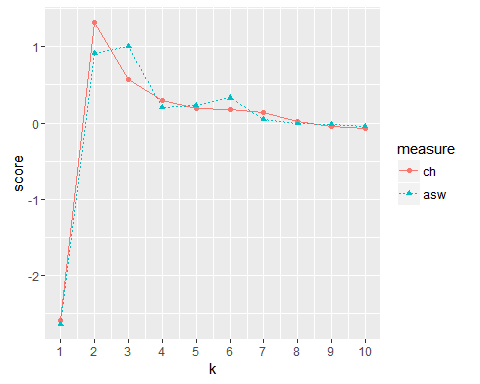
clustering.ch$crit

## [1] 0.000000 14.094814 11.417985 10.418801 10.011797 9.964967 9.861682  
## [8] 9.412089 9.166676 9.075569

clustcrit$crit

## [1] NaN 12.215107 10.359587 9.690891 10.011797 9.964967 9.506978  
## [8] 9.092065 8.822406 8.695065

critframe <- data.frame(k=1:10, ch=scale(clustering.ch$crit),   
 asw=scale(clustering.asw$crit))  
critframe <- melt(critframe, id.vars=c("k"),  
 variable.name="measure",  
 value.name="score")  
# 시각화  
ggplot(critframe, aes(x=k, y=score, color=measure)) +  
 geom\_point(aes(shape=measure)) + geom\_line(aes(linetype=measure)) +  
 scale\_x\_continuous(breaks=1:10, labels=1:10)



summary(clustering.ch)

## Length Class Mode   
## cluster 25 -none- numeric  
## centers 18 -none- numeric  
## totss 1 -none- numeric  
## withinss 2 -none- numeric  
## tot.withinss 1 -none- numeric  
## betweenss 1 -none- numeric  
## size 2 -none- numeric  
## iter 1 -none- numeric  
## ifault 1 -none- numeric  
## crit 10 -none- numeric  
## bestk 1 -none- numeric

기준으로보면 2-3개의 군집이 최적. CH의 경우 에서 hclust와 동일한 값(동일한 군집) => 5 혹은 6이 최적?

에 대한 붓스트랩을 이용한 군집 안정성

kbest.p<-5  
cboot<-clusterboot(pmatrix, clustermethod=kmeansCBI,  
 runs=100,iter.max=100,  
 krange=kbest.p, seed=15555)

## boot 1   
## boot 2   
## boot 3   
## boot 4   
## boot 5   
## boot 6   
## boot 7   
## boot 8   
## boot 9   
## boot 10   
## boot 11   
## boot 12   
## boot 13   
## boot 14   
## boot 15   
## boot 16   
## boot 17   
## boot 18   
## boot 19   
## boot 20   
## boot 21   
## boot 22   
## boot 23   
## boot 24   
## boot 25   
## boot 26   
## boot 27   
## boot 28   
## boot 29   
## boot 30   
## boot 31   
## boot 32   
## boot 33   
## boot 34   
## boot 35   
## boot 36   
## boot 37   
## boot 38   
## boot 39   
## boot 40   
## boot 41   
## boot 42   
## boot 43   
## boot 44   
## boot 45   
## boot 46   
## boot 47   
## boot 48   
## boot 49   
## boot 50   
## boot 51   
## boot 52   
## boot 53   
## boot 54   
## boot 55   
## boot 56   
## boot 57   
## boot 58   
## boot 59   
## boot 60   
## boot 61   
## boot 62   
## boot 63   
## boot 64   
## boot 65   
## boot 66   
## boot 67   
## boot 68   
## boot 69   
## boot 70   
## boot 71   
## boot 72   
## boot 73   
## boot 74   
## boot 75   
## boot 76   
## boot 77   
## boot 78   
## boot 79   
## boot 80   
## boot 81   
## boot 82   
## boot 83   
## boot 84   
## boot 85   
## boot 86   
## boot 87   
## boot 88   
## boot 89   
## boot 90   
## boot 91   
## boot 92   
## boot 93   
## boot 94   
## boot 95   
## boot 96   
## boot 97   
## boot 98   
## boot 99   
## boot 100

groups <- cboot$result$partition  
print\_clusters(cboot$result$partition, kbest.p)

## [1] "cluster 1"  
## Country RedMeat Fish Fr.Veg  
## 1 Albania 10.1 0.2 1.7  
## 4 Bulgaria 7.8 1.2 4.2  
## 18 Romania 6.2 1.0 2.8  
## 25 Yugoslavia 4.4 0.6 3.2  
## [1] "cluster 2"  
## Country RedMeat Fish Fr.Veg  
## 6 Denmark 10.6 9.9 2.4  
## 8 Finland 9.5 5.8 1.4  
## 15 Norway 9.4 9.7 2.7  
## 20 Sweden 9.9 7.5 2.0  
## [1] "cluster 3"  
## Country RedMeat Fish Fr.Veg  
## 5 Czechoslovakia 9.7 2.0 4.0  
## 7 E Germany 8.4 5.4 3.6  
## 11 Hungary 5.3 0.3 4.2  
## 16 Poland 6.9 3.0 6.6  
## 23 USSR 9.3 3.0 2.9  
## [1] "cluster 4"  
## Country RedMeat Fish Fr.Veg  
## 2 Austria 8.9 2.1 4.3  
## 3 Belgium 13.5 4.5 4.0  
## 9 France 18.0 5.7 6.5  
## 12 Ireland 13.9 2.2 2.9  
## 14 Netherlands 9.5 2.5 3.7  
## 21 Switzerland 13.1 2.3 4.9  
## 22 UK 17.4 4.3 3.3  
## 24 W Germany 11.4 3.4 3.8  
## [1] "cluster 5"  
## Country RedMeat Fish Fr.Veg  
## 10 Greece 10.2 5.9 6.5  
## 13 Italy 9.0 3.4 6.7  
## 17 Portugal 6.2 14.2 7.9  
## 19 Spain 7.1 7.0 7.2

cboot$bootmean

## [1] 0.8670000 0.8420714 0.6147024 0.7647341 0.7508333

cboot$bootbrd

## [1] 15 20 49 17 32

## 연습문제

이고 이다. 에 대하여 에서 각각 100개의 난수를 발생시키고 평균 군집을 실시하시오. 에 대하여 에서 하나씩 새로운 데이터를 생성하고 앞에서 찾은 군집의 중심점과의 거리를 계산하여 가장 가까운 군집으로 할당해 보시오.