사고원인 데이터 분석

INDEX

통계청 데이터에 대한 시각화 -9주차(R스크립트)

```r
#9주차
# 1.데이터 준비하기

# Load
# 원본 데이터 불러오기
ods <- read.csv("통계청사망자데이터셋.csv" , stringsAsFactors = F , header = T)
rs <- read.csv("일반사망요약분류표.txt", sep="\t",stringsAsFactors = F , header = T)

# Column Subset
# 필요한 컬럼만 선택하고, 컬럼이름 부여하기
mycols <- c(3,5,11)
dataset <- ods[ , mycols ]
colnames(dataset) <- c("ymd","age","사망원인코드")

# Missing Value 및 R을 위한 컬럼 Format 설정
summary( dataset )
dataset$ymd <- as.Date( dataset$ymd ); summary( dataset )
dataset <- dataset[ which(dataset$age<150),]; summary( dataset )
dataset$사망원인코드 <- as.character(dataset$사망원인코드); summary( dataset )
# 나이 -> 연령대로 범주화
dataset <- within( dataset, {
   연령대 <- NA
   연령대[age<=9] = "09세이하"
   연령대[age>=10 & age<=19] = "10~19세"
   연령대[age>=20 & age<=29] = "20~29세"
   연령대[age>=30 & age<=39] = "30~39세"
   연령대[age>=40 & age<=49] = "40~49세"
   연령대[age>=50 & age<=59] = "50~59세"
   연령대[age>=60 & age<=69] = "60~69세"
   연령대[age>=70 & age<=79] = "70~79세"
   연령대[age>=80 & age<=89] = "80~89세"
   연령대[age>=90] = "90세이상"
})
dataset <- dataset[-2]
summary(dataset); str(dataset)
```

통계청 데이터에 대한 시각화 -9주차(R스크립트)

```r
# 2. 데이터 탐색

# 기본 집계표 생성
options( digits=5 )
Totals <- nrow( dataset )
TotalYmd <- as.data.frame(table( dataset$ymd ))
colnames(TotalYmd) <- c("ymd","x")
CountReason <- table( dataset$사망원인코드 )
TotalReason <- as.data.frame(CountReason)
CountAge <-  table( dataset$연령대 )
TotalAge <- as.data.frame(CountAge)
TotalReason$per <- 100*TotalReason$Freq/Totals
TotalAge$per <- 100*TotalAge$Freq/Totals

# 시각화
par(mar=c(2,4,2,2))
timeseries <- ts(TotalYmd$x, c(2014,01,01), frequency = 364)
tsdecomp  <- decompose(timeseries)
plot(tsdecomp)
ts_data <- data.frame( TotalYmd$ymd
                       ,tsdecomp$x
                       ,tsdecomp$trend
                       ,tsdecomp$seasonal
                       ,tsdecomp$random )
barplot( CountReason[1:20], main="사망원인", xlab="사망원인", ylab="사망자수" )
barplot( CountReason[21:40], main="사망원인", xlab="사망원인", ylab="사망자수" )
barplot( CountReason[41:60], main="사망원인", xlab="사망원인", ylab="사망자수" )
barplot( CountReason[61:81], main="사망원인", xlab="사망원인", ylab="사망자수" )
barplot( CountAge, main="연령대",xlab="사망원인", ylab="사망자수" )

# 분석 대상 선정
# -> 10대 사망원인=>50대
TopReason <- TotalReason[ order(-TotalReason$Freq),c("Var1","Freq") ]
TopReason <- TopReason[1:50,]
colnames( TopReason ) <- c("사망원인코드","사망자수")
data <- merge(x=dataset,y=TopReason,by='사망원인코드')
data <- data[,c("ymd","사망원인코드","연령대")]

```

통계청 데이터에 대한 시각화 -9주차(R스크립트)

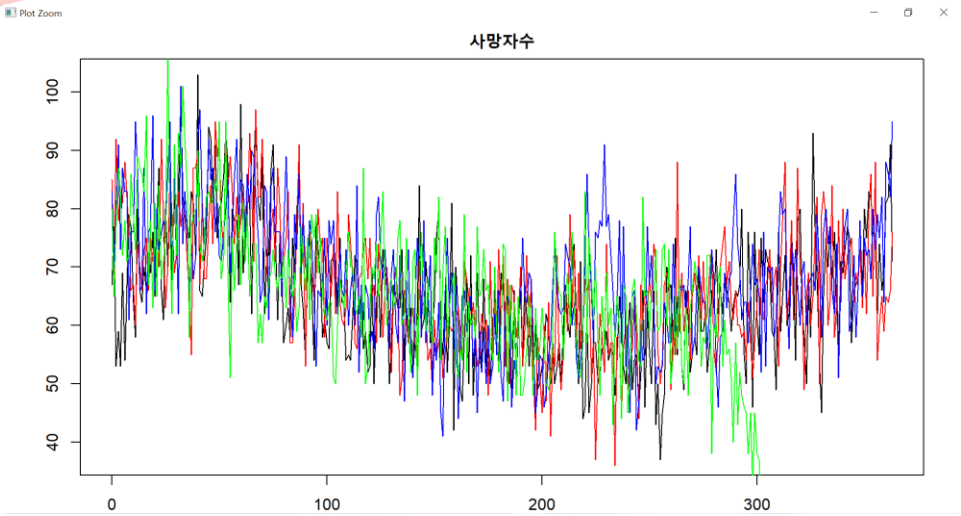
```r
# 시계열 분석을 위한 데이터변환
#install.packages("reshape")
library(reshape)
YmdReason <- cast( data, ymd~사망원인코드 )

# 50대 사망원인만 대상으로 재집계하고 시각화
CountReason <- table( data$사망원인코드 )
TotalReason <- as.data.frame(CountReason)
CountAge <-  table( data$연령대 )
TotalAge <- as.data.frame(CountAge)
TotalReason$per <- 100*TotalReason$Freq/Totals
TotalAge$per <- 100*TotalAge$Freq/Totals
barplot( CountReason, main="사망원인", xlab="사망원인", ylab="사망자수" )
barplot( CountAge, main="연령대",xlab="사망원인", ylab="사망자수" )

# Top1 94번 사망원인으로 시계열 분석 및 시각화
timeseries <- ts(YmdReason$"94", c(2014,01,01), frequency = 364)
tsdecomp  <- decompose(timeseries)
plot(tsdecomp)
ts_data <- data.frame( seq(1:1401)
                       ,TotalYmd$ymd
                       ,tsdecomp$x
                       ,tsdecomp$trend
                       ,tsdecomp$seasonal
                       ,tsdecomp$random )
colnames(ts_data)<-c("순서","년 월 일","X","T","S","R")
ts_data$i<-(ts_data$순서-1)%/%364
ts_data$j<-(ts_data$순서-1)%%364

#x(사망자수)그래프(색상지정)
#ann=F를 사용하여 축 제목이 안나오게 설정함
ts1<-ts_data[ts_data$i==0,]
plot(ts1$j, ts1$X, col='black', type='l',ann=F)
ts2<-ts_data[ts_data$i==1,]
lines(ts2$j,ts2$X,col='red')
ts2<-ts_data[ts_data$i==2,]
lines(ts2$j,ts2$X,col='blue')
ts2<-ts_data[ts_data$i==3,]
lines(ts2$j,ts2$X,col='green')
title("사망자수")
```

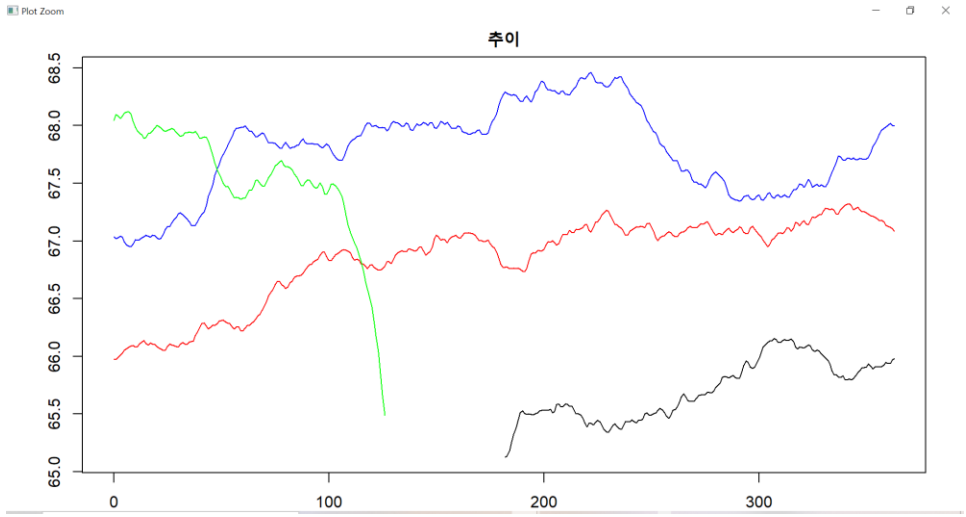
통계청 데이터에 대한 시각화 -9주차(R스크립트)

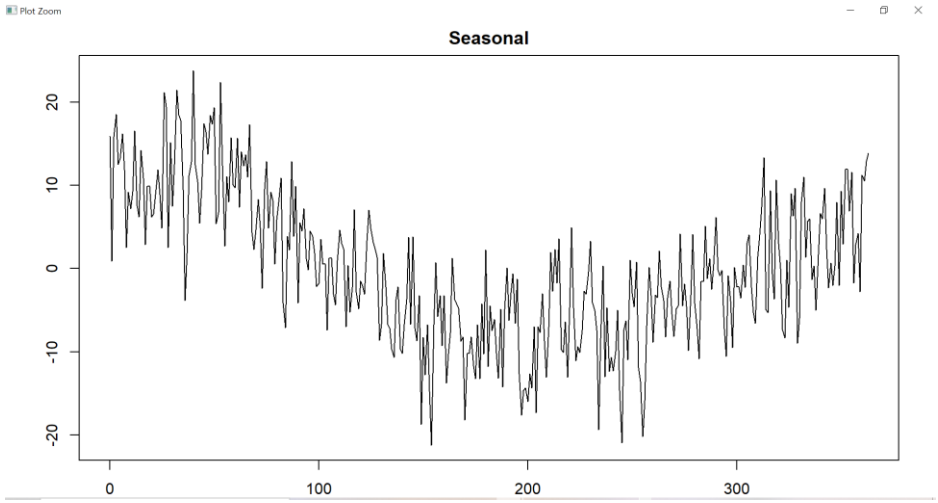
```r
#Trend 그래프(색상지정)
#ann=F를 사용하여 축 제목이 안나오게 설정함
plot(ts_data$j, ts_data$T, type='n',ann=F)
ts2<-ts_data[ ts_data$i==0,]
lines(ts2$j,ts2$T, col='black')
ts2<-ts_data[ ts_data$i==1,]
lines(ts2$j,ts2$T, col='red')
ts2<-ts_data[ ts_data$i==2,]
lines(ts2$j,ts2$T, col='blue')
ts2<-ts_data[ ts_data$i==3,]
lines(ts2$j,ts2$T, col='green')
title("추이")

#Seasonal 그래프(색상지정)
#ann=F를 사용하여 축 제목이 안나오게 설정함
ts1<-ts_data[ts_data$i==0,]
plot(ts1$j, ts1$S, col='black', type='l',ann=F)
title("Seasonal")
```

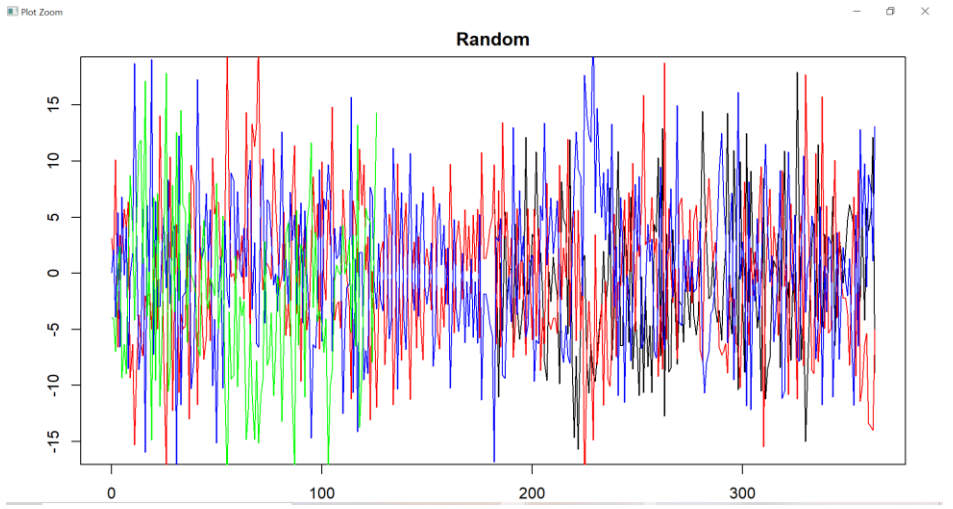
```r
#Random 그래프(색상지정)
#ann=F를 사용하여 축 제목이 안나오게 설정함
ts1<-ts_data[ts_data$i==0,]
plot(ts1$j, ts1$R, col='black', type='l',ann=F)
ts2<-ts_data[ts_data$i==1,]
lines(ts2$j,ts2$R,col='red')
ts2<-ts_data[ts_data$i==2,]
lines(ts2$j,ts2$R,col='blue')
ts2<-ts_data[ts_data$i==3,]
lines(ts2$j,ts2$R,col='green')
title("Random")
```

01 통계청 데이터에 대한 시각화 -9주차(시각화차트)









통계청 데이터에 대한 시각화 -10주차(R스크립트)

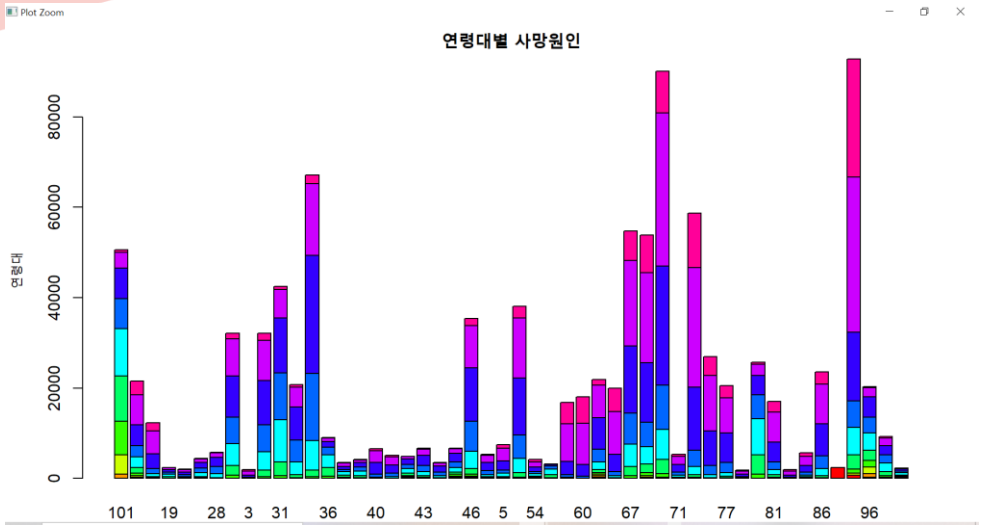
```
#10주차
# 사망원인별 연령대 집계표
AgeTable <- table( data$연령대, data$사망원인코드 )
AgeDF <- data.frame( AgeTable )
colnames( AgeDF ) <- c("연령대","사망원인코드","명")
ReasonAge <- cast( AgeDF, 사망원인코드~연령대,
                   value='명', fun.aggregate=sum )
# 비교 시각화
barplot( AgeTable, main="연령대별 사망원인"
         ,xlab="사망원인", ylab="연령대", col=rainbow(10) )
AgeTableProp<-prop.table(AgeTable,2)
barplot(AgeTableProp,main="연령대별 사망원인"
        ,xlab="사망원인", ylab="연령대",col=rainbow(10))



# 연령대별 사망자수를 비율로 변경
ReasonAge$t1 <- ReasonAge[2]+ReasonAge[3]+ReasonAge[4]+ReasonAge[5]+ReasonAge[6]
ReasonAge$t2 <- ReasonAge[7]+ReasonAge[8]+ReasonAge[9]+ReasonAge[10]+ReasonAge[11]
ReasonAge$total <- ReasonAge[12]+ReasonAge[13]
ReasonAge$age0 <- 100*ReasonAge[2]/ReasonAge[14]
ReasonAge$age1 <- 100*ReasonAge[3]/ReasonAge[14]
ReasonAge$age2 <- 100*ReasonAge[4]/ReasonAge[14]
ReasonAge$age3 <- 100*ReasonAge[5]/ReasonAge[14]
ReasonAge$age4 <- 100*ReasonAge[6]/ReasonAge[14]
ReasonAge$age5 <- 100*ReasonAge[7]/ReasonAge[14]
ReasonAge$age6 <- 100*ReasonAge[8]/ReasonAge[14]
ReasonAge$age7 <- 100*ReasonAge[9]/ReasonAge[14]
ReasonAge$age8 <- 100*ReasonAge[10]/ReasonAge[14]
ReasonAge$age9 <- 100*ReasonAge[11]/ReasonAge[14]

ReasonAge2 <- data.frame( ReasonAge[1],ReasonAge[15:24] )
names(ReasonAge2)[2:11] <- colnames(ReasonAge)[2:11]
colnames(ReasonAge2)
```

```r
# 사망원인별 유사도 계산 및 시각화
rownames(ReasonAge2) <- ReasonAge2[,1]
ReasonAge2 <- ReasonAge2[-1]
ReasonDist <- dist(ReasonAge2, method="euclidean")
two_coord <- cmdscale(ReasonDist)
plot(two_coord, type="n", xlab="x", ylab="y")
text(two_coord, rownames(ReasonAge2) )

# 계층적 군집
library( cluster )
hcl <- hclust( dist(ReasonAge2), method="single")
plot(hcl, hang=-1, xlab="사망원인", ylab="거리")

# 분할적 군집
library( graphics )
kms <- kmeans( ReasonAge2, 4 )
kms
```

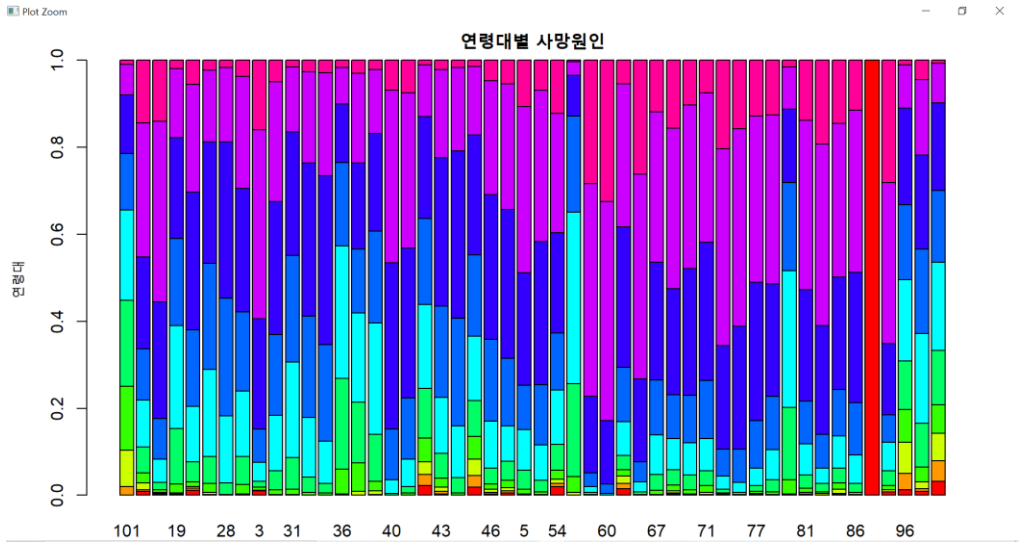
01

통계청 데이터에 대한 시각화 -10주차(시각화차트)

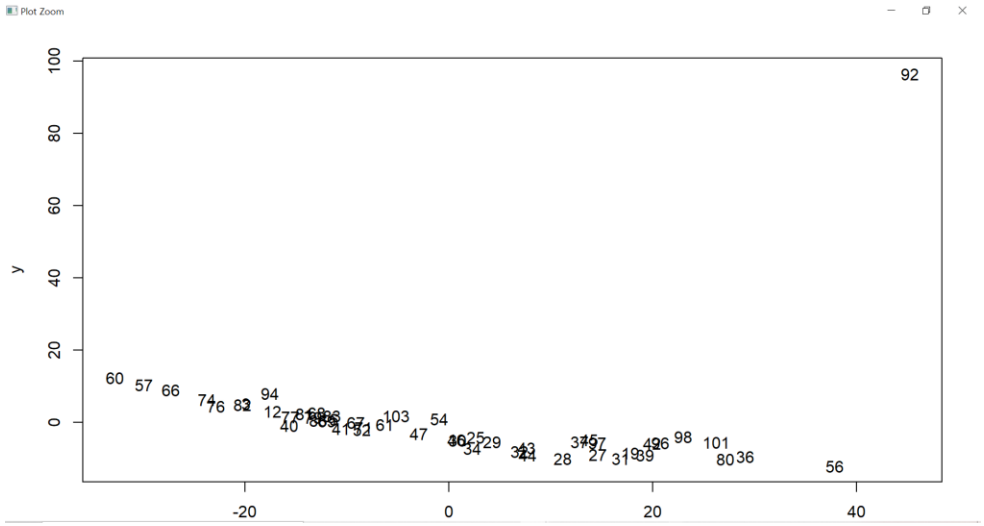
1. 막대그래프(변경전)



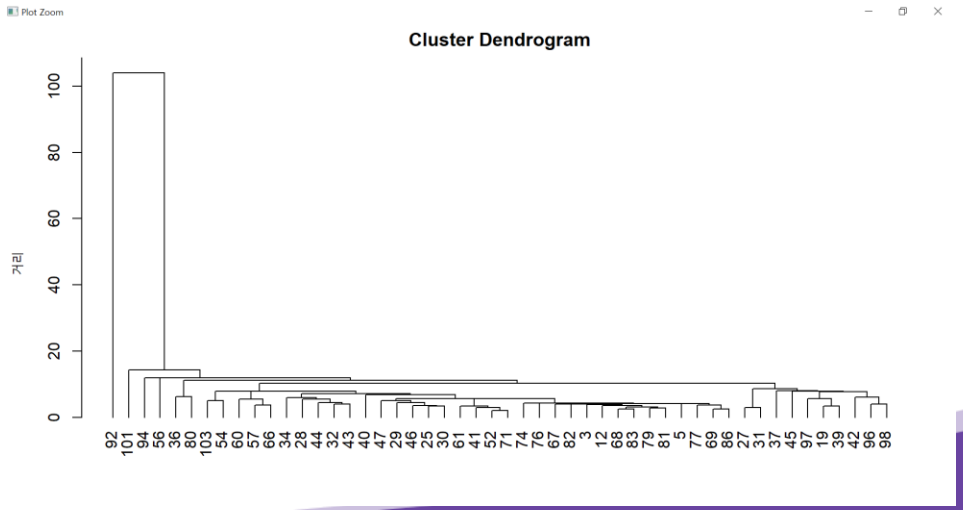
막대그래프(변경후)



2. 차원축소 plot 시각화



3.k-means hclust 군집 시각화

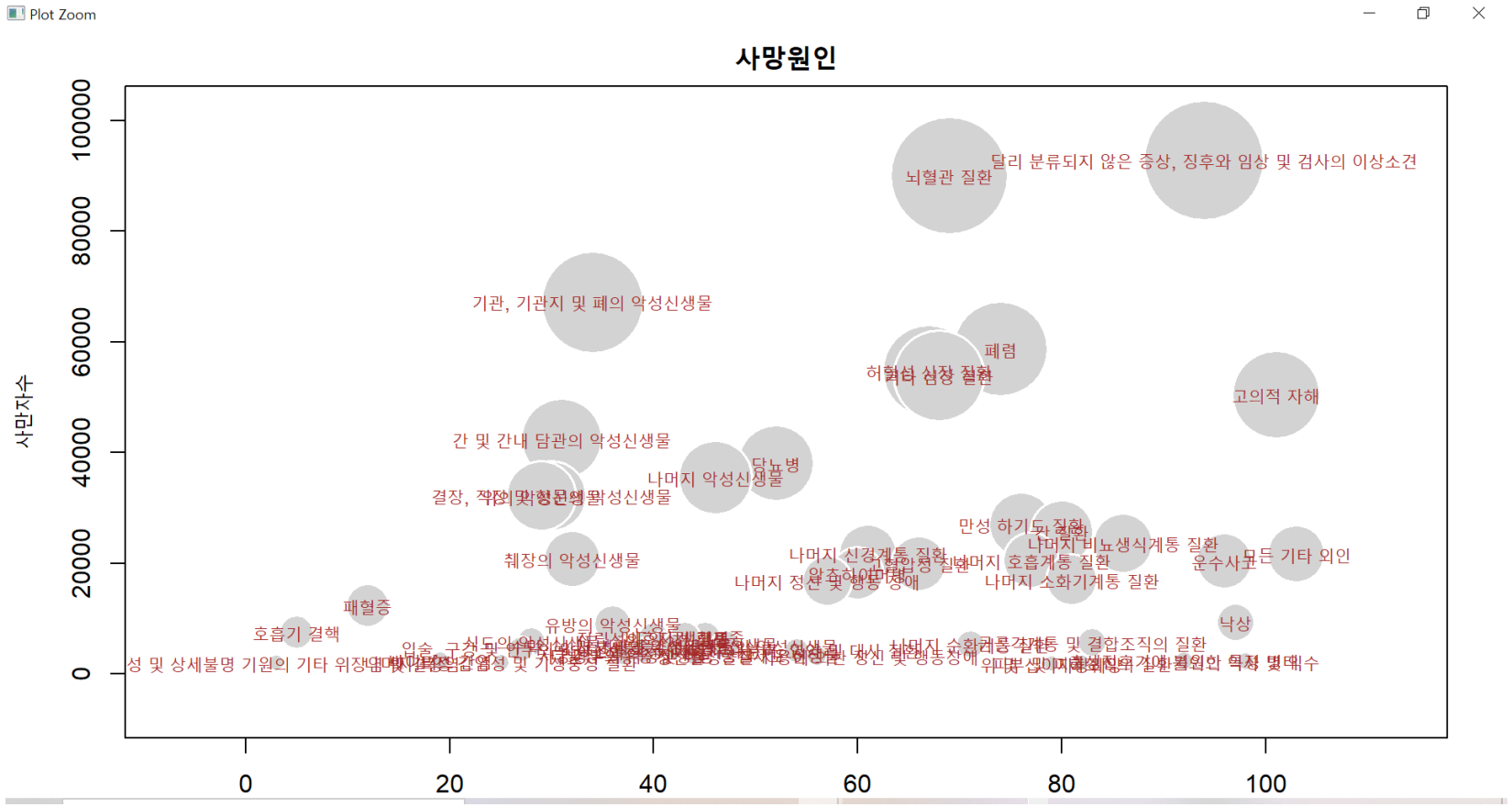


통계청 데이터에 대한 시각화 -11주차(R스크립트)

```r
#11주차
#트리맵을 이용하려고 했으나, 교수님께 보내드린 사진처럼 오류가 떠서 버블차트로 대체하였습니다.
library(MASS)
head(TopReason)
radius<-sqrt(TopReason$사망자수)
symbols(TopReason$사망원인코드,TopReason$사망자수,
        circles=radius, # 각각 써클의 반지름값
        inches=0.4, # 각각 써클의 크기 조절값
        fg="white", # 각각 써클의 테두리 색
        bg="lightgray", # 각각 써클의 바탕색
        lwd=1.5, # 각각 써클의 테두리선 두께
        ylab="사만자수", # y 축 제목 설정
        main="사망원인")
text(TopReason$사망원인코드,TopReason$사망자수, # 문자로 출력할 x,y 위치
     TopReason$사망원인코드, # 문자로 출력할 값
     cex=0.8, # 글자 크기
     col="brown")

#항목분류를 이용하여 사망원인코드를 원인으로 변환시켜서 나오게 작성하였습니다.
rs$X103항목분류 <- as.character(rs$X103항목분류)
reason <- inner_join(x=TopReason, y=rs, by=c("사망원인코드"="X103항목분류"))
head(reason)
radius<-sqrt(reason$사망자수)
symbols(reason$사망원인코드,reason$사망자수,
        circles=radius, # 각각 써클의 반지름값
        inches=0.4, # 각각 써클의 크기 조절값
        fg="white", # 각각 써클의 테두리 색
        bg="lightgray", # 각각 써클의 바탕색
        lwd=1.5, # 각각 써클의 테두리선 두께
        ylab="사만자수", # y 축 제목 설정
        main="사망원인")
text(reason$사망원인코드,reason$사망자수, # 문자로 출력할 x,y 위치
     reason$원인, # 문자로 출력할 값
     cex=0.8, # 글자 크기
     col="brown")
```

01

통계청 데이터에 대한 시각화 -11주차(시각화)



1. 데이터 개수
-3924247개

2. 주요 컬럼에 대한 설명
- dataset: 주요 컬럼으로 소방청 자료에서 필요한 컬럼만 추가하였다.
 연도인 ymd, 발생장소, 사고원인을 가지고 와서 발생장소를 이용하여 사고원인을 분석하고자 3가지로 선정하였다.
-data:발생장소는 도, 광역시 별로 분리 되어있기 때문에 도, 광역시가 아닌 도만 표시하기 위해서 도 이름으로 합쳤고 이 것을 발생도 이름으로 저장하였다.

3. 파일의 크기
-390.6MB

4. 기간 등
-2015년 1월 1일~2019년 12월 31일

5. 일부 데이터에 대한 스크린 샷

1.dataset

	ymd	발생 장소	사고원인
1	2015-01-01	경기도	잠금장치개방
2	2015-01-01	경기도	잠금장치개방
3	2015-01-01	경기도	화재
4	2015-01-01	경기도	승강기
5	2015-01-01	경기도	승강기
6	2015-01-01	경기도	교통
7	2015-01-01	경기도	교통
8	2015-01-01	경기도	잠금장치개방

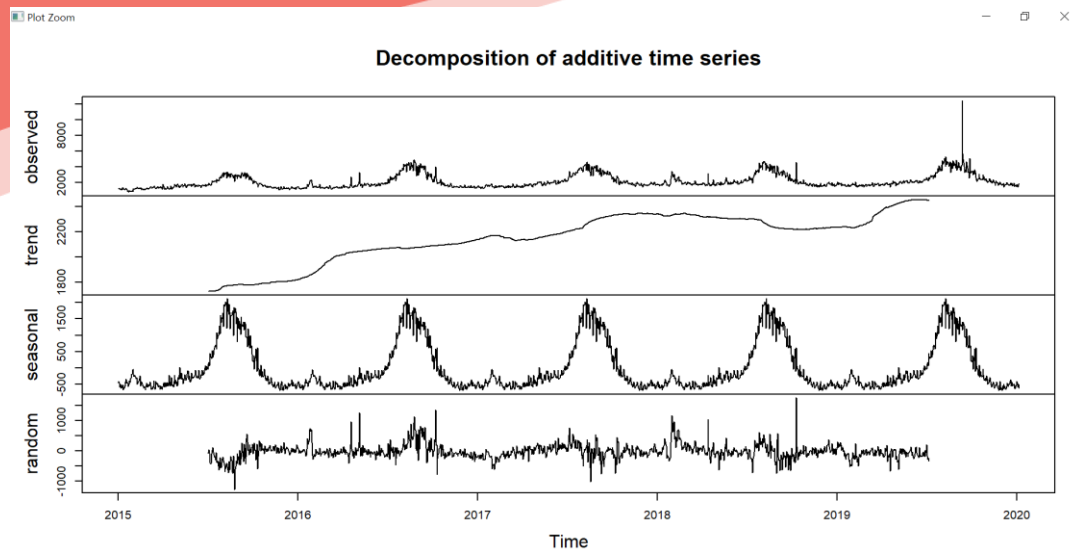
2. data

	ymd	사고 원인	발생 도
6	2015-11-19	교통	경기도
7	2015-10-06	교통	경상남도
8	2016-08-29	교통	경기도
9	2018-01-26	교통	서울
10	2015-05-08	교통	전라북도
11	2018-04-21	교통	경상남도
12	2019-10-01	교통	경상남도

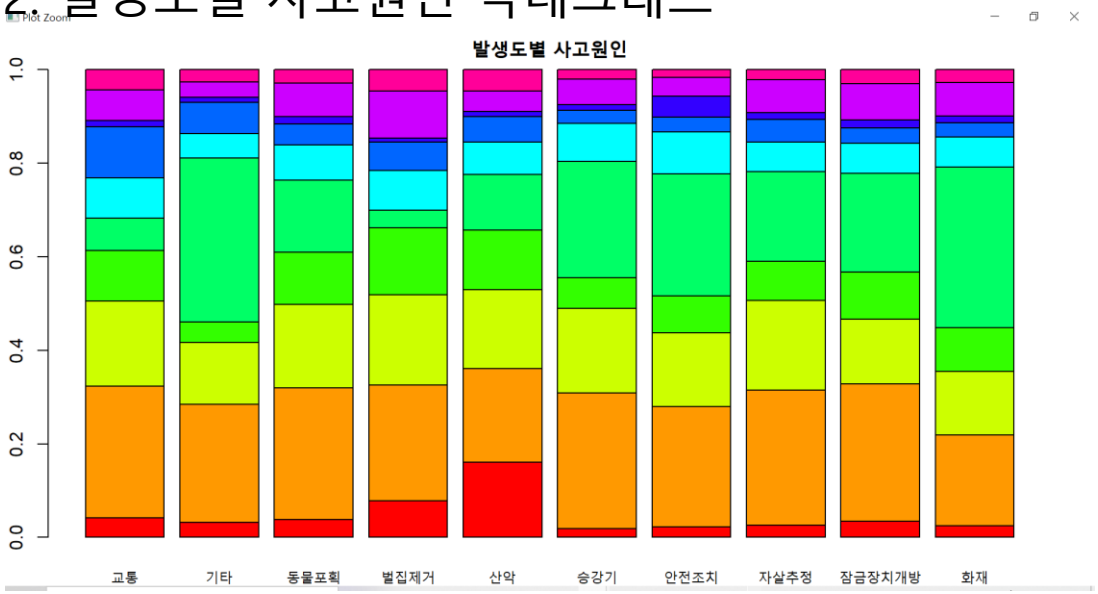
개인 데이터 및 시각화 개요

2.2 시각화 차트(소방청데이터셋)

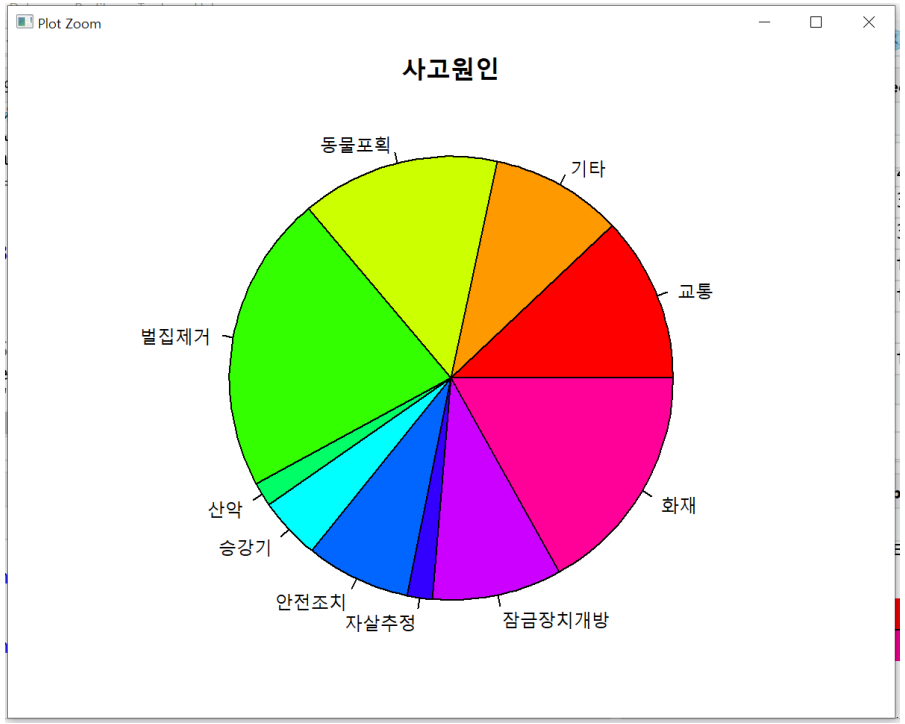
1. 시계열 차트



2. 발생도별 사고원인 막대그래프

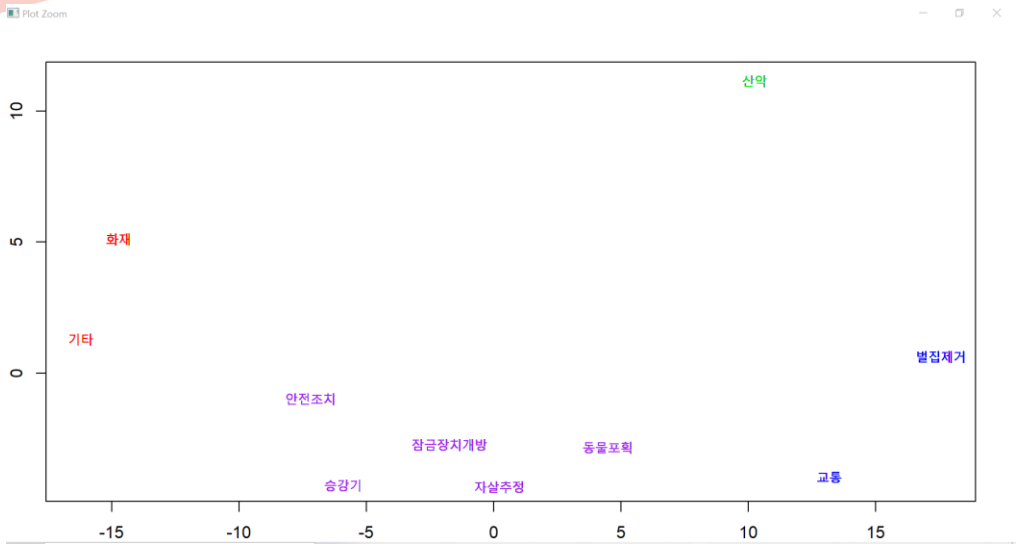


3. 파이차트

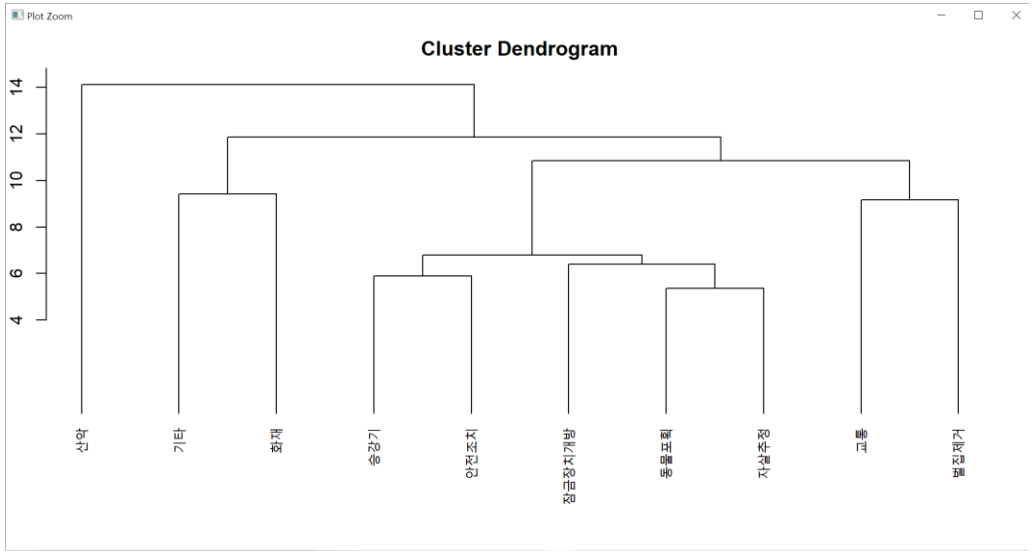


2.2 시각화 차트

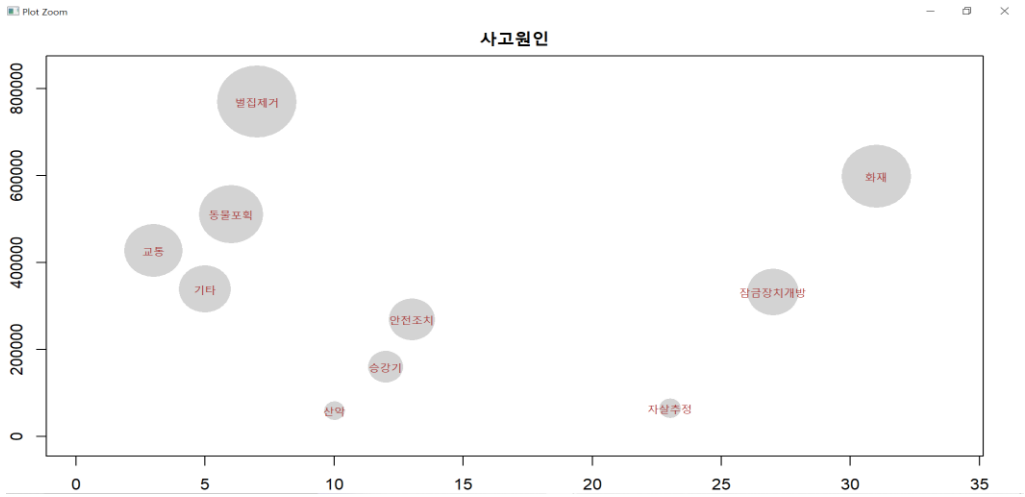
1. 차원 축소 plot 시각화



2. k-means hclust 군집 시각화



3. 버블차트를 이용한 텍스트 시각화



03 개인 데이터 시각화

```r
# 1.데이터 준비하기

# Load
# 원본 데이터 불러오기
ods <- read.csv("소방청.csv" , stringsAsFactors = F , header = T)

# Column Subset
# 필요한 컬럼만 선택하고, 컬럼이름 부여하기
#발생장소를 가지고 사고원인을 알아보고자 3개의 컬럼을 뽑았습니다.
mycols <- c(2,6,9)
dataset <- ods[ , mycols ]
colnames(dataset) <- c("ymd","발생장소","사고원인")

# Missing Value 및 R을 위한 컬럼 Format 설정
summary( dataset )
dataset$ymd <- as.Date( dataset$ymd ); summary( dataset )
dataset$발생장소 <- as.character(dataset$발생장소); summary( dataset )
dataset$사망원인 <- as.character(dataset$사고원인); summary( dataset )

# 여러 광역시들을 해당하는 도의 이름으로 바꾸었습니다.
dataset <- within( dataset, {
  발생도 <- NA
  발생도[발생장소=="서울특별시"] = "서울"
  발생도[발생장소=="경기도"] = "경기도"
  발생도[발생장소=="강원도"] = "강원도"
  발생도[발생장소=="충청북도" ] = "충청북도"
  발생도[발생장소=="충청남도" ]="충청남도"
  발생도[발생장소=="대전광역시"]= "충청남도"
  발생도[발생장소=="세종특별자치도"] ="충청남도"
  발생도[발생장소=="경상북도"]="경상북도"
  발생도[발생장소=="대구광역시"] ="경상북도"
  발생도[발생장소=="경상남도"] = "경상남도"
  발생도[발생장소=="울산광역시"] = "경상남도"
  발생도[발생장소=="부산광역시"] = "경상남도"
  발생도[발생장소=="전라북도"] = "전라북도"
  발생도[발생장소=="전라남도"]= "전라남도"
  발생도[발생장소=="광주광역시"] = "전라남도"
  발생도[발생장소=="제주특별자치도"] = "제주도"

})

dataset <- dataset[-2]
summary(dataset); str(dataset)
# 2. 데이터 탐색

# 기본 집계표 생성
options( digits=5 )
Totals <- nrow( dataset )
TotalYmd <- as.data.frame(table( dataset$ymd ))
colnames(TotalYmd) <- c("ymd","x")
CountReason <- table( dataset$사고원인 )
TotalReason <- as.data.frame(CountReason)
CountAge <-  table( dataset$발생도 )
TotalAge <- as.data.frame(CountAge)
TotalReason$per <- 100*TotalReason$Freq/Totals
TotalAge$per <- 100*TotalAge$Freq/Totals

# 시각화
#plot의 margin크기를 설정하였습니다.
par(mar=c(2,2,2,2))
#그냥 실행하면 지수로 표기되기때문에 숫자표기로 바꾸기 위해서 설정하였습니다.
options(scipen=5)
timeseries <- ts(TotalYmd$x, c(2015,01,01), frequency = 364)
tsdecomp  <- decompose(timeseries)
plot(tsdecomp)
ts_data <- data.frame( TotalYmd$ymd
                       ,tsdecomp$x
                       ,tsdecomp$trend
                       ,tsdecomp$seasonal
                       ,tsdecomp$random )
#사고원인 수가 39개이기 때문에 양이 많아서 두번에 걸쳐서 barplot이 나타나게 설정하였습니다.
barplot( CountReason[1:20], main="사고원인", ylab="사고수" )
barplot( CountReason[21:39], main="사고원인", ylab="사고수" )
barplot( CountAge, main="발생장소",ylab="사고수" )

# 분석 대상 선정
# -> 10대 사고원인
TopReason <- TotalReason[ order(-TotalReason$Freq),c("Var1","Freq") ]
TopReason <- TopReason[1:10,]
colnames( TopReason ) <- c("사고원인","사고수")
data <- merge(x=dataset,y=TopReason,by='사고원인')
data <- data[,c("ymd","사고원인","발생도")]
```

개인 데이터 시각화

```r
# 시계열 분석을 위한 데이터변환
#install.packages("reshape")
library(reshape)
YmdReason <- cast( data, ymd~사고원인 )

# 10대 사고원인만 대상으로 재집계하고 시각화
CountReason <- table( data$사고원인 )
TotalReason <- as.data.frame(CountReason)
CountAge <-  table( data$발생도 )
TotalAge <- as.data.frame(CountAge)
TotalReason$per <- 100*TotalReason$Freq/Totals
TotalAge$per <- 100*TotalAge$Freq/Totals
barplot( CountReason, main="사고원인", xlab="사고원인", ylab="사망자수" )
barplot( CountAge, main="발생장소",xlab="사고원인", ylab="사망자수" )

#사고원인별 파이차트
pie(CountReason,radius=3,main="사고원인"
    ,col=rainbow(10))

# 사고원인별 발생장소 집계표
AgeTable <- table( data$발생도, data$사고원인 )
AgeDF <- data.frame( AgeTable )
colnames( AgeDF ) <- c("발생도","사고원인","명")
ReasonAge <- cast( AgeDF, 사고원인~발생도,
                   value='명', fun.aggregate=sum )

# 비교 시각화(막대그래프)
#발생장소별로 사고원인을 시각화하였습니다.
barplot( AgeTable, main="발생도별 사고원인", col=rainbow(10) )
AgeTableProp<-prop.table(AgeTable,2)
barplot(AgeTableProp,main="발생도별 사고원인",col=rainbow(10))
```

```r
# 발생장소별 사고수를 비율로 변경
#t1에는
ReasonAge$t1 <- ReasonAge[2]+ReasonAge[3]+ReasonAge[4]+ReasonAge[5]+ReasonAge[6]
ReasonAge$t2 <- ReasonAge[7]+ReasonAge[8]+ReasonAge[9]+ReasonAge[10]+ReasonAge[11]
ReasonAge$total <- ReasonAge[12]+ReasonAge[13]
ReasonAge$age0 <- 100*ReasonAge[2]/ReasonAge[14]
ReasonAge$age1 <- 100*ReasonAge[3]/ReasonAge[14]
ReasonAge$age2 <- 100*ReasonAge[4]/ReasonAge[14]
ReasonAge$age3 <- 100*ReasonAge[5]/ReasonAge[14]
ReasonAge$age4 <- 100*ReasonAge[6]/ReasonAge[14]
ReasonAge$age5 <- 100*ReasonAge[7]/ReasonAge[14]
ReasonAge$age6 <- 100*ReasonAge[8]/ReasonAge[14]
ReasonAge$age7 <- 100*ReasonAge[9]/ReasonAge[14]
ReasonAge$age8 <- 100*ReasonAge[10]/ReasonAge[14]
ReasonAge$age9 <- 100*ReasonAge[11]/ReasonAge[14]

ReasonAge2 <- data.frame( ReasonAge[1],ReasonAge[15:24] )
names(ReasonAge2)[2:11] <- colnames(ReasonAge)[2:11]
colnames(ReasonAge2)

# 사고원인별 유사도 계산 및 시각화
rownames(ReasonAge2) <- ReasonAge2[,1]
ReasonAge2 <- ReasonAge2[-1]
ReasonDist <- dist(ReasonAge2, method="euclidean")
two_coord <- cmdscale(ReasonDist)
plot(two_coord, type="n", ylab="y")
#유사도별 텍스트마다 컬러를 설정하였습니다.
text(two_coord, rownames(ReasonAge2),col=c('blue','red','purple','blue','green',
                                           'purple','purple','purple','purple','red'))
```

```
144 #버블차트를 이용한 텍스트시각화
145 library(MASS)
146 head(TopReason)
147 radius<-sqrt(TopReason$사고수)
148 symbols(TopReason$사고원인,TopReason$사고수,
149         circles=radius, # 각각 써클의 반지름값
150         inches=0.4, # 각각 써클의 크기 조절값
151         fg="white", # 각각 써클의 테두리 색
152         bg="lightgray", # 각각 써클의 바탕색
153         lwd=1.5, # 각각 써클의 테두리선 두께
154         ylab="사고수", # y 축 제목 설정
155         main="사고원인")
156 text(TopReason$사고원인,TopReason$사고수, # 문자로 출력할 x,y 위치
157      TopReason$사고원인, # 문자로 출력할 값
158      cex=0.8, # 글자 크기
159      col="brown")
160
161 # 계층적 군집
162 library( cluster )
163 hcl <- hclust( dist(ReasonAge2), method="single")
164 plot(hcl, hang=-1, ylab="거리")
```

1. 서론

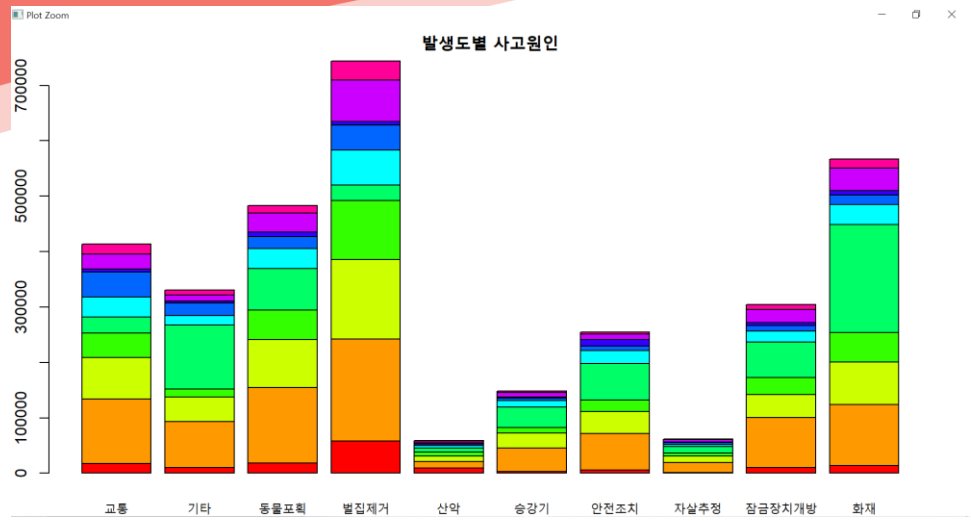
1-1.지역별 사고원인 분석 연구

지역별로 사고원인을 분석하게 되면, 도청 입장에서는 어떤 사고가 제일 자주 발생하는지 파악하여 그 원인에 대하여 전략적 대책을 실행함으로써 주민의 안전에 맞출 수 있다. 현재 계룡시는 교통사고, 화재, 범죄, 생활안전, 자살, 감염병 등을 중심으로 여러 가지 정책을 실행하고 있으며 우수 시로 선정돼 인센티브를 교부 받기도 하였다. (금강일보)

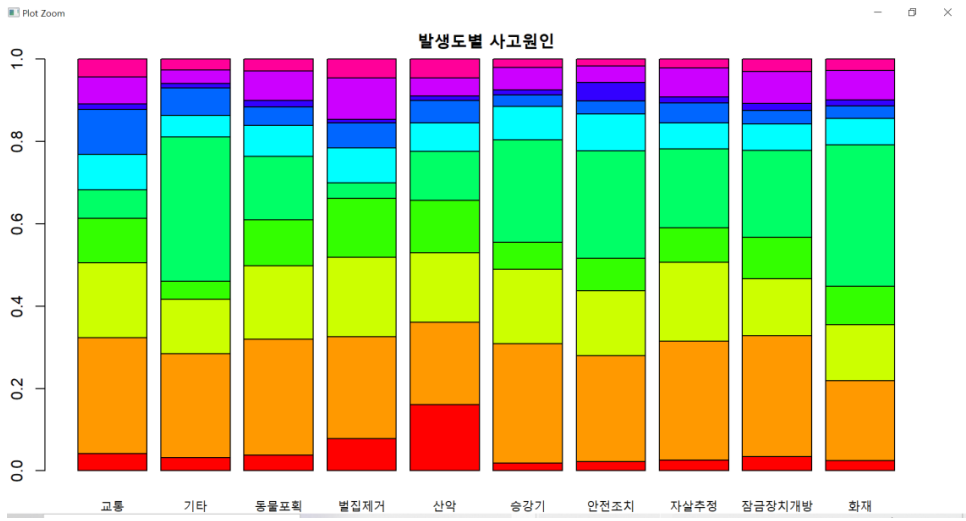
또한, 주민은 어떤 지역이 안전한지를 파악할 수 있어서, 이 점을 고려하여 나중에 살 곳을 정할 수 있다. 그리고 도청은 좀 더 많은 주민을 유치하기 위해서 주만의 안전에 더욱 신경 쓰게 되어서 선순환이 이루어진다.

1-2. 컬럼 선정 기준

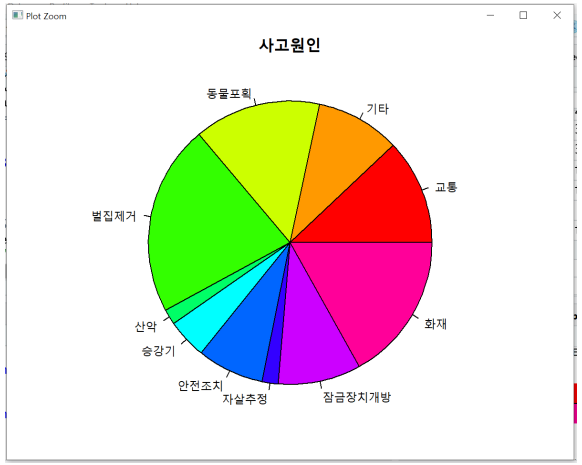
컬럼 선정기준으로 명확성과 전체성을 우선시하여 보았다. 소방청 데이터셋에서는 신고년월일, 신고시간, 출동년월일 ,출동 시간,발생 장소_시, 발생 장소_구, 발생장소_동, 사고원인, 사고원인코드의 컬럼으로 구성되어 있다. 우선 사고원인을 토대로 언제 발생하고 어느 지역에서 발생했는지를 알고 싶었기 때문에 이 중 신고 시간과 출동 시간은 필요하지 않다고 생각하여 제외했다. 그리고 발생장소가 디테일하게 분리되어 있었고, 전체의 지역에서 살펴보고 싶었기 때문에 발생 장소_시 컬럼을 선택하였다. 또한, 사고원인의 경우 전체적으로 어떤 원인이 발생했는지 알고 싶어서 사고원인 컬럼을 선택하였다. 그 외 년월일은 신고년월일을 선택하였다. 그 결과 총 신고년월일, 발생장소_시, 사고원인을 사용하는 컬럼으로 선정하였다.



a. 막대그래프(변경전)



b. 막대그래프(변경후)

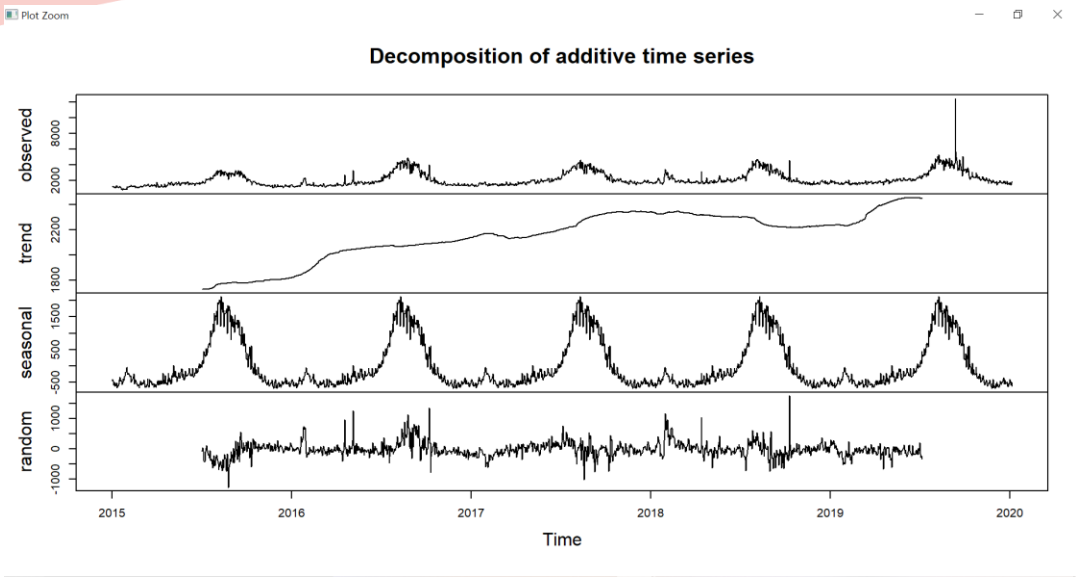


c. 파이차트

2. 시각화 차트 분석 및 해석

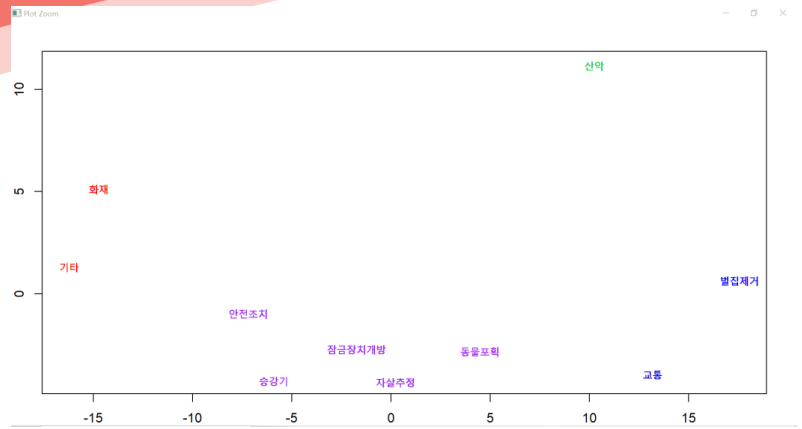
-본 절에서는 사고원인별로 지역별(도별) 빈도를 분석한 후, 사고원인별로 10대 사고원인을 뽑아서 지역별로 분석하였다. a. 막대그래프(변경전)과 c.파이차트를 보면 전반적으로 벌집제거에서 많은 사고가 난다는 것을 알 수 있다. a. 막대그래프(변경전) 와 b. 막대그래프(변경후)는 위에서부터 제주도, 전라남도, 전라북도, 경상남도, 경상북도, 충청남도, 충청북도, 강원도, 경기도, 서울 순 인 것을 확인 할 수 있다.

그 결과 a다른 도에 비해 경기도가 사고가 높게 나타나는 것으로 나왔다. 그리고 전반적으로 경기도가 많은 사고가 발생한다는 점을 알 수 있다. 이 분석에서 경기도는 2020년 기준 인구는 약 1342만 명으로 대한민국 전체에서 가장 인구가 많다는 점에서 불리하게 작용될 수 있으나, 서울 인구수 959만명으로 많은 차이가 안 나지만, 경기도(주황색)과 서울(빨강색)의 차이가 많이 난다는 점을 볼 수 있다. 이에 경기도 도 청 자체 내에서 안전에 더욱 주의에 귀 기울여야할 필요성이 있다.

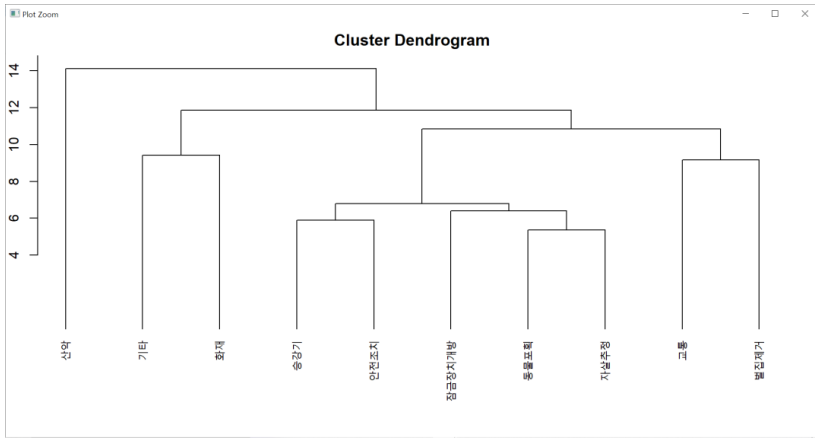


d. 시계열차트

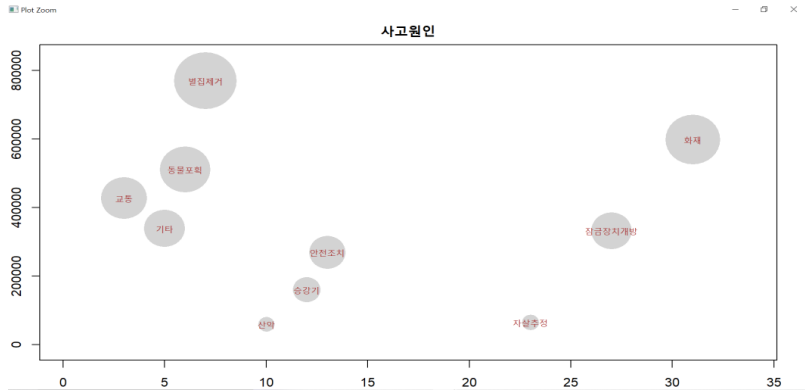
-시계열 차트는 시간 경과에 다른 활동의 그래픽 표시이다. 차트에 표시되는 고점과 저점은 활동이 많고 적음을 나타내는데 seasonal은 1년 주기로 나타나는 계절성이다. Seasonal을 중심으로 살펴보면 거의 모든 사고가 겨울에서 봄으로 이동하는 시기에 많이 나타난다는 점을 알 수 있다. 그 외 시간은 전반적으로 사고 수가 줄어든다. 이 시계열 차트를 보고 겨울과 봄이 다가오는 시기에 더 많은 안전에 유의해야 한다는 사실을 알 수 있다.



e. 차원 축소 plot 시각화



f. k-means hclust 군집 시각화



g. 버블차트를 이용한 텍스트 시각화

-본 절에서는 사고원인을 중점적으로 분석하고자 g.버블 차트를 이용하여 텍스트 시각화를 하였고, 어떤 원인이 서로 유사성이 보이고 군집되어있는지 살펴보고자 e. 차원 축소 plot 시각화와 f. k-means hclust 군집 시각화를 하였다.
이 차트를 보면 앞에서와같이 전반적으로 벌집 제거에서 많은 사고가 난다는 것을 알 수 있다. 그다음으로 많은 빈도수를 보이는 곳은 동물 포획, 화재, 교통, 기타, 잠금장치 등의 문제가 많이 발생한다.또한, e. 차원 축소 plot 시각화와 f. k-means hclust 군집 시각화에서도 유사하고 군집되어 있는 부분이 승강기, 안전장치, 잠금장치개방, 동물포획, 자살 추정이다. 이에 여러 면에서 시설관리에 좀 더 신경을 써야 한다는 점을 알 수 있다.