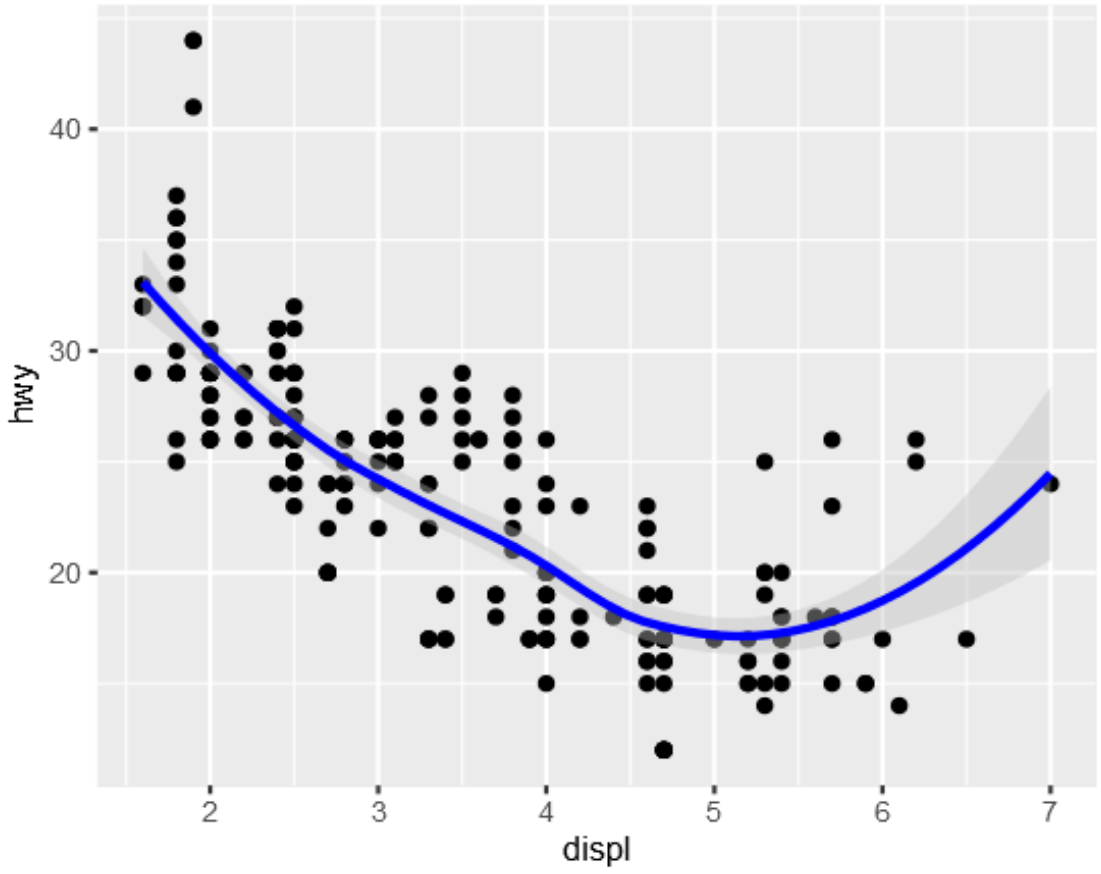
2020년 1학기 통계 그래픽스 중간고사

192STG11_우나영

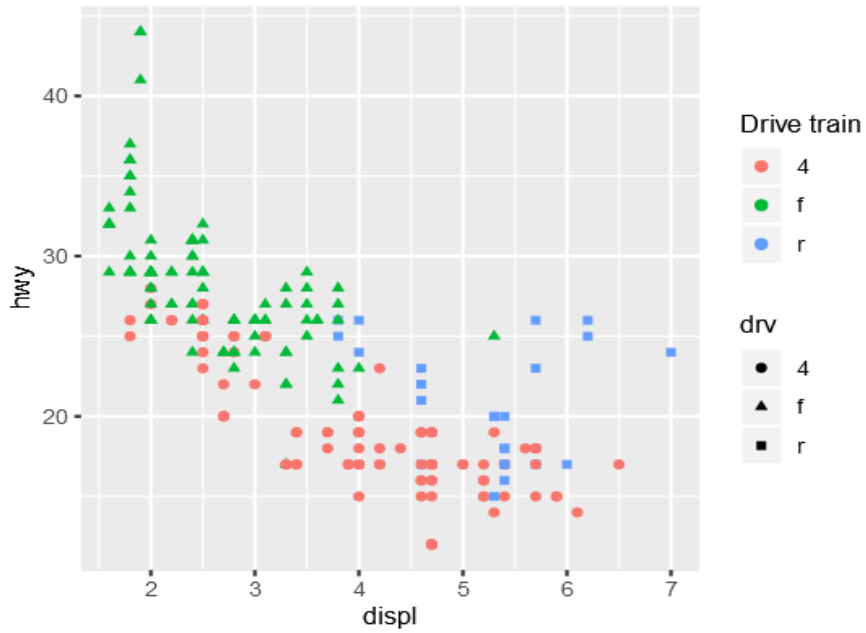
1. 아래의 code로 생성되는 자료를 이용하여 geom_smooth를 이용하지 않고, 아래의 그림과 같이 geom_smooth의 효과를 갖는 그림을 그리시오.

```
mod <- loess(hwy ~ displ, data = mpg)
smoothed <- data.frame(displ = seq(1.6, 7, length = 50))
pred <- predict(mod, newdata = smoothed, se = TRUE)
smoothed$hwy <- pred$fit
smoothed$hwy_lwr <- pred$fit - 1.96 * pred$se.fit
smoothed$hwy_upr <- pred$fit + 1.96 * pred$se.fit
base <- mpg %>% ggplot(aes(displ, hwy)) + geom_point()
base + geom_ribbon(data = smoothed, fill = 'grey', alpha = 0.4,  aes(ymin
= hwy_lwr, ymax = hwy_upr))+ geom_path(data = smoothed, col = 'blue', size
= 1, aes(displ, hwy))
```



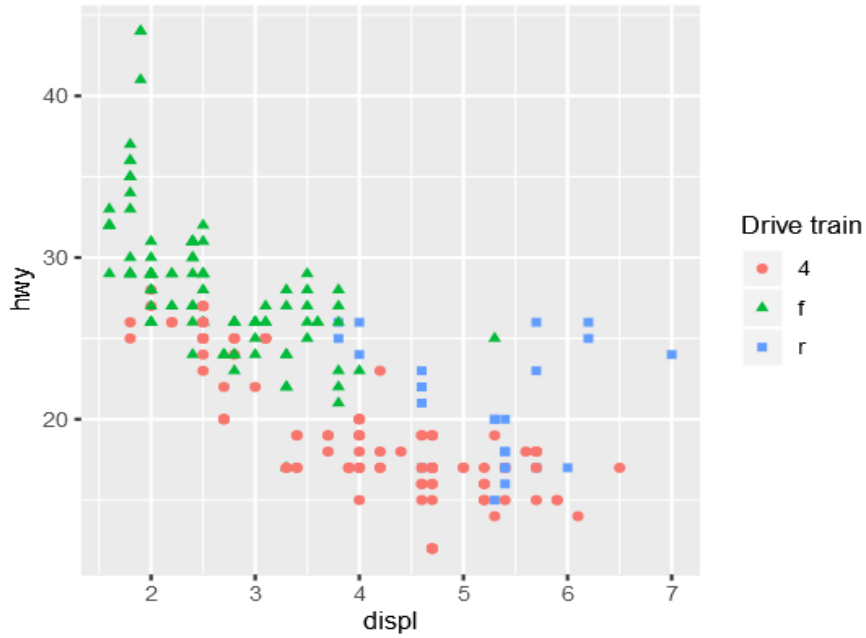
2. 아래의 코드에 대한 그림이다. 잘못된 점을 찾고, 올바른 그림을 그리시오.

```
ggplot(mpg, aes(displ, hwy)) + geom_point(aes(color = drv, shape = drv)) +
scale_color_discrete("Drive train")
```



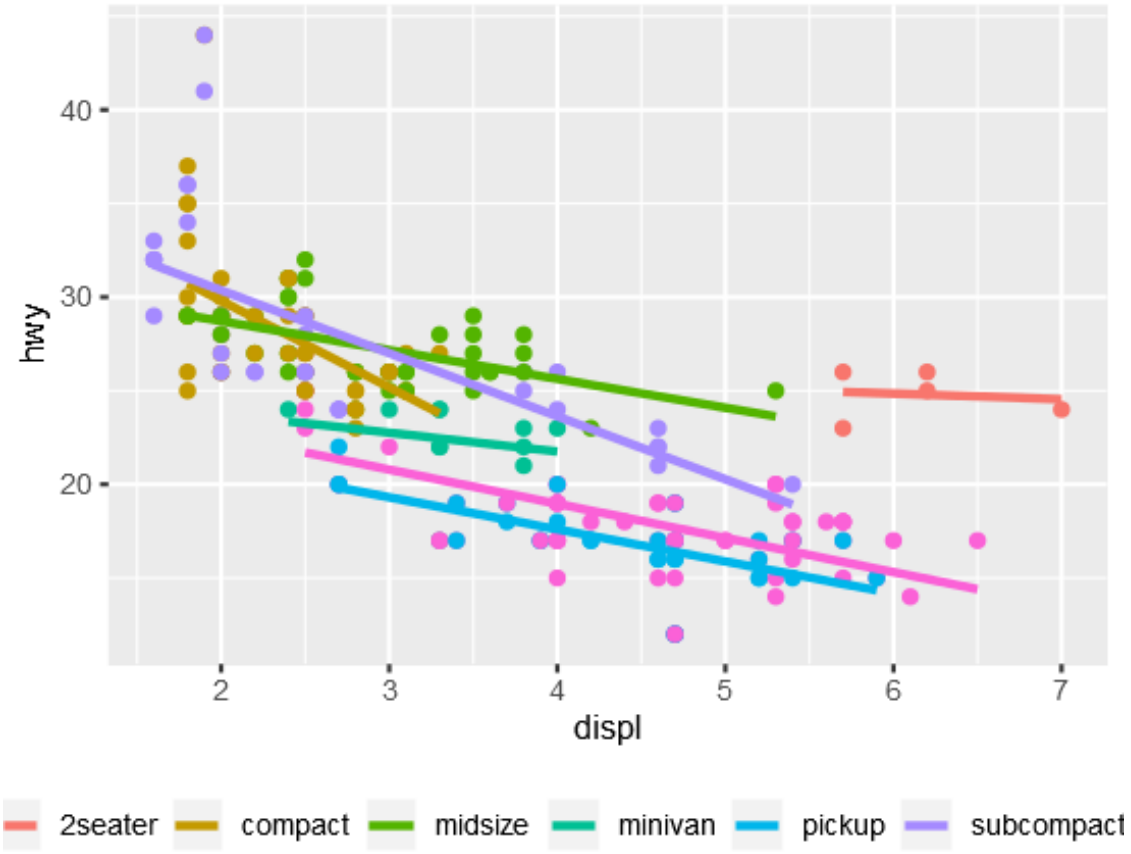
위 그림의 잘못된 점은 shape과 color를 보여주는 "Drive train" legend만 있는 것이 아니라 shape만 보여주는 "drv" legend가 공존한다는 것이다. 따라서, "drv" legend를 삭제해야 한다. 코드와 그림은 아래와 같다.

```
ggplot(mpg, aes(displ, hwy)) + geom_point(aes(color = drv, shape = drv)) +
scale_color_discrete("Drive train") + scale_shape_discrete("Drive train")
```



3. mpg 자료를 이용하여 아래의 그림을 그리는 code를 작성하시오.

```
library(modelr)

for(i in 1:length(unique(mpg$class))){
  assign(paste0("mod", i), lm(hwy ~ displ, data = subset(mpg, class == uni
que(mpg$class)[i])))
}

grid1 <-
 subset(mpg, class == unique(mpg$class)[1]) %>% select(displ, class) %>% g
ather_predictions(mod1)
grid2 <-
 subset(mpg, class == unique(mpg$class)[2]) %>% select(displ, class) %>% g
ather_predictions(mod2)
grid3 <-
 subset(mpg, class == unique(mpg$class)[3]) %>% select(displ, class) %>% g
ather_predictions(mod3)
grid4 <-
 subset(mpg, class == unique(mpg$class)[4]) %>% select(displ, class) %>% g
ather_predictions(mod4)
grid5 <-
 subset(mpg, class == unique(mpg$class)[5]) %>% select(displ, class) %>% g
ather_predictions(mod5)
grid6 <-
 subset(mpg, class == unique(mpg$class)[6]) %>% select(displ, class) %>% g
ather_predictions(mod6)
grid7 <-
 subset(mpg, class == unique(mpg$class)[7]) %>% select(displ, class) %>% g
ather_predictions(mod7)

set <- list(grid1, grid2, grid3, grid4, grid5, grid6, grid7)
grid <- Reduce('rbind', set)

mpg %>% ggplot(aes(displ, hwy, col = class)) + geom_point(show.legend = FA
LSE)+geom_line(data = grid, size = 1, aes(y = pred)) + theme(legend.positi
on = "bottom") + guides(color = guide_legend(title = " ", ncol = length(un
ique(mpg$class))))
```


hwy
40
30
20
2
3
4
5
6
7
displ
2seater
compact
midsize
minivan
pickup
subcompact

4. Midterm-EDAdata1-2020.csv 파일은 2016년 사망자 자료를 성별, 나이별(ageG), 사망원인 별로 정리한 자료이다. 사망원인에 대한 code는 Midterm-D56.xlsx에 있다. 이 자료를 이용하여 자료의 특성을 살펴보고 자료의 특성을 가장 잘 나타낸다고 생각하는 그림 한 장을 그리고 그 그림을 선택하게 된 이유를 명시하시오. (자료를 살펴보는 과정도 나타낼 것)

```
summary(df)
```

```
##     gender             ageG               D56                month       
##  Length:5164        Length:5164        Length:5164        Min.   : 1.000  
##  Class :character   Class :character   Class :character   1st Qu.: 3.000  
##  Mode  :character   Mode  :character   Mode  :character   Median : 6.000  
##                                                           Mean   : 6.454  
##                                                           3rd Qu.: 9.000  
##                                                           Max.   :12.000  
##      count         
##  Min.   :   1.00  
##  1st Qu.:   2.00  
##  Median :   5.00  
##  Mean   :  54.38  
##  3rd Qu.:  32.00  
##  Max.   :1450.00
```

```
dim(df)
```

```
## [1] 5164    5
```

위 자료는 2016년 사망자 자료를 성별, 나이별, 사망원인 그리고 달 별로 기록한 자료이다. gender, ageG, D56, month, 그리고 count로 5개의 변수와 5164개의 관측치로 이루어졌다.

```
# NA 확인
sum(is.na(df$gender))
```

```
## [1] 0
```

```
sum(is.na(df$ageG))
```

```
## [1] 35
```

```
sum(is.na(df$D56))
```

```
## [1] 0
```

```
sum(is.na(df$month))
```

```
## [1] 0
```

```
sum(is.na(df$count))
```

```
## [1] 0
```

NA는 ageG 변수에서 35개를 제외하고 없다.

```
summary(df$month)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   3.000   6.000   6.454   9.000  12.000
```

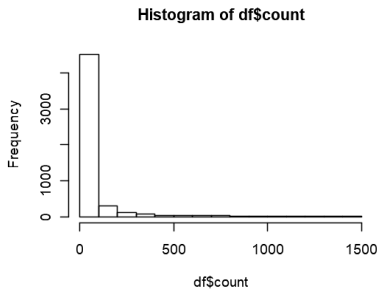
```
unique(df$month)
```

```
##  [1]  7  3  2  5 10 11  4 12  1  6  8  9
```

```
summary(df$count)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    2.00    5.00   54.38   32.00 1450.00
```

```
hist(df$count)
```



continuous 변수인 count와 month를 살펴보았다. month는 1~12월이 존재한다. count는 mean이 median 보다 큰 right skewed 분포를 보인다.

```
unique(df$gender)
```

```
## [1] "Female" "Male"
```

```
unique(df$ageG)
```

```
##  [1] "00-09"     "10 월 19 일" "20-29"     "30-39"     "40-49"
##  [6] "50-59"     "60-69"     "70-79"     "80-89"     NA
## [11] "over90"
```

```
unique(df$D56)
```

```
##  [1] "D-002" "D-003" "D-011" "D-012" "D-018" "D-024" "D-025" "D-026"
##  [9] "D-030" "D-034" "D-035" "D-037" "D-038" "D-039" "D-040" "D-042"
## [17] "D-047" "D-048" "D-049" "D-050" "D-051" "D-052" "D-056" "NONE"
## [25] "D-028" "D-053" "D-054" "D-055" "D-004" "D-005" "D-019" "D-027"
## [33] "D-029" "D-044" "D-045" "D-046" "D-016" "D-033" "D-043" "D-032"
## [41] "D-041" "D-031" "D-036" "D-007" "D-010" "D-020"
```

```
length(which(df$D56 == "NONE"))
```

```
## [1] 252
```

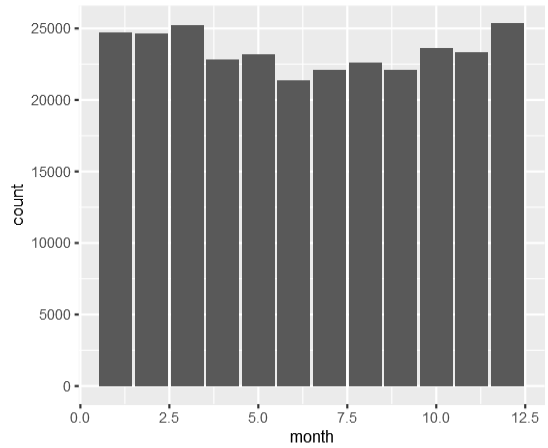
```
df[which(df$ageG=="10 월 19 일"), 'ageG'] <- "10-19"
unique(df$ageG)
```

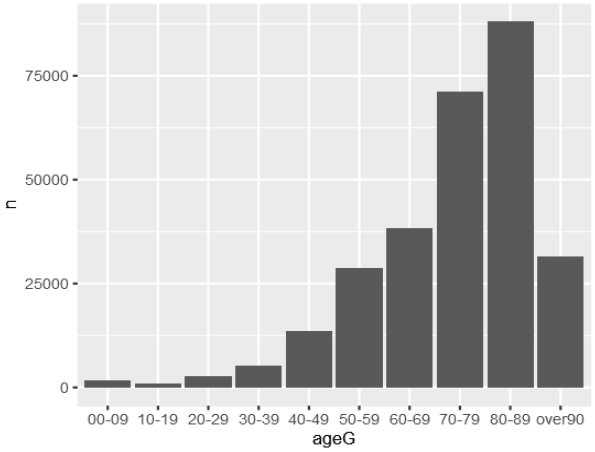
```
##  [1] "00-09"  "10-19"  "20-29"  "30-39"  "40-49"  "50-59"  "60-69"
##  [8] "70-79"  "80-89"  NA       "over90"
```

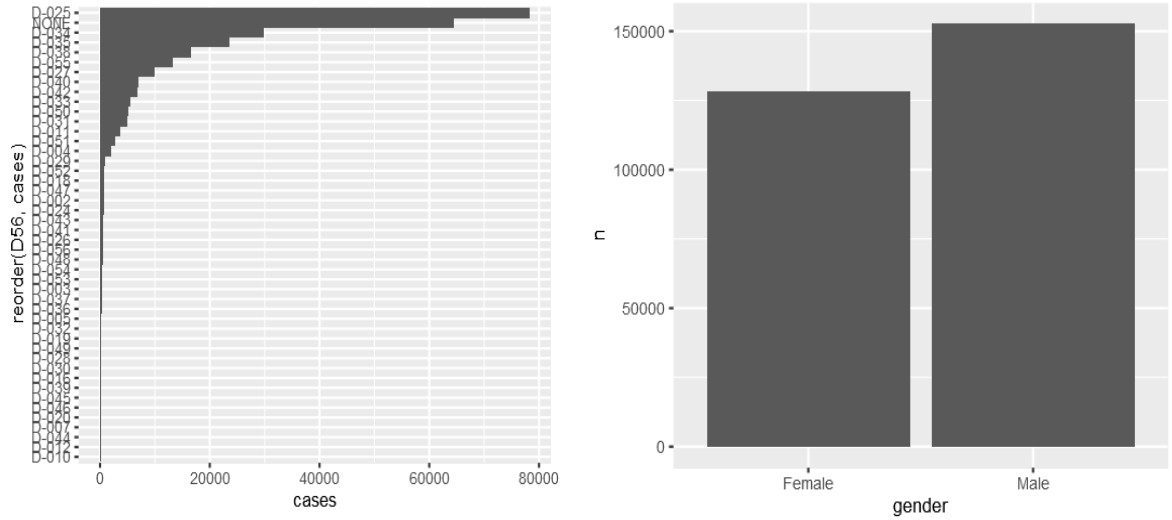
```
df %>% summarise(total = sum(count))

## # A tibble: 1 x 1
##    total
##    <dbl>
## 1 280827
```

Categorical 변수인 gender, ageG, 그리고 D56을 살펴보았다. ageG의 변수 중 10월 19일로 엑셀에서 발생한 오류가 발견되었다. 이를 10-19로 바꾸고 분석을 진행하였다. D56은 56개의 사망 원인 코드이다. 위 데이터에서는 45개의 코드와 NONE이 나타난다. NONE은 D56의 NA로 볼 수 있지만 본 데이터 분석에서는 NA로 보지 않고 사망 원인 미상이라고 판단하였다. 전체 사망자 수는 280,827명이다.

- [그림1]month별 사망자 수/ [그림2]ageG별 사망자 수/ [그림3]D56별 사망자 수/ [그림4]gender별 사망자 수

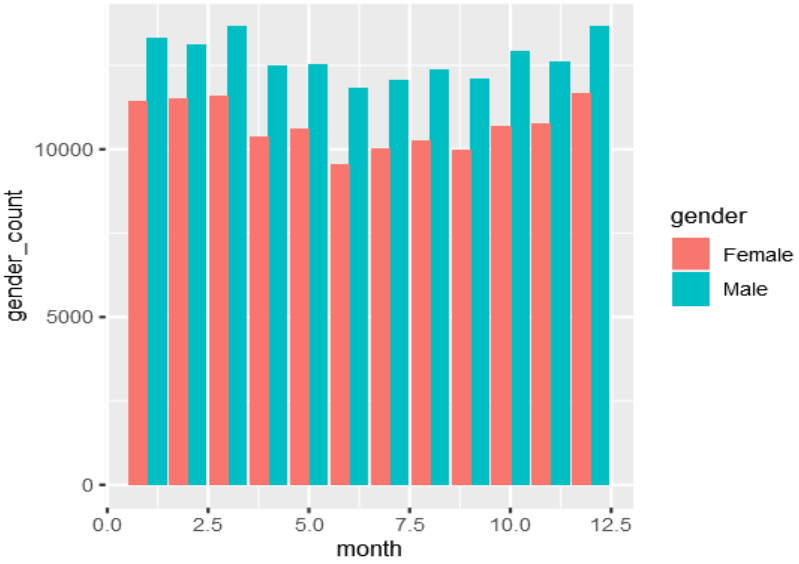






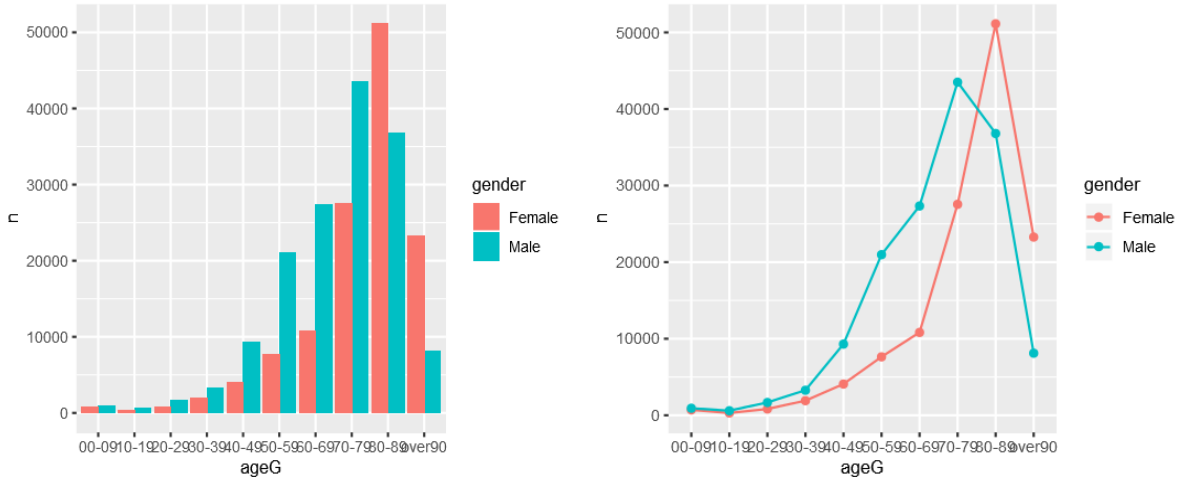
좌측 상단 month 별 사망자수 수 그래프는 V자 계곡 형상을 띄고 있는 것처럼 보인다. 여름에서 겨울로 갈수록 사망자 수가 증가하는 양상을 띈다. 우측 상단 ageG별 사망자 수 그래프는 00-09세에 높았다가 다시 감소하다가 20-29세부터 나이대가 증가함에 따라 높아지다 80-89세에 peak를 보이고 다시 떨어지는 양상을 보인다. 이때, ageG별 사망자 수 그래프는 위의 나머지 3개의 그래프와 달리 ageG의 NA값을 제거한 결과이다. 좌측 하단 D56별 사망자 수 그래프를 통해 사망자의 원인 상위 5위는 D-025, NONE, D-034, D-035, 3-038임을 알 수 있다. 이는 Midterm-D56.xlsx에 따르면, 악성신생물(Malignant neoplasms), NONE, 심장 질환(Heart diseases), 뇌혈관 질환(Cerebrovascular diseases), 폐렴(Pneumonia)에 해당한다. 우측 하단 gender별 사망자 수 그래프를 통해 여성 사망자 수 보다 남성 사망자 수가 더 많다는 것을 알 수 있다.

- [그림5]month별 gender별 사망자 수



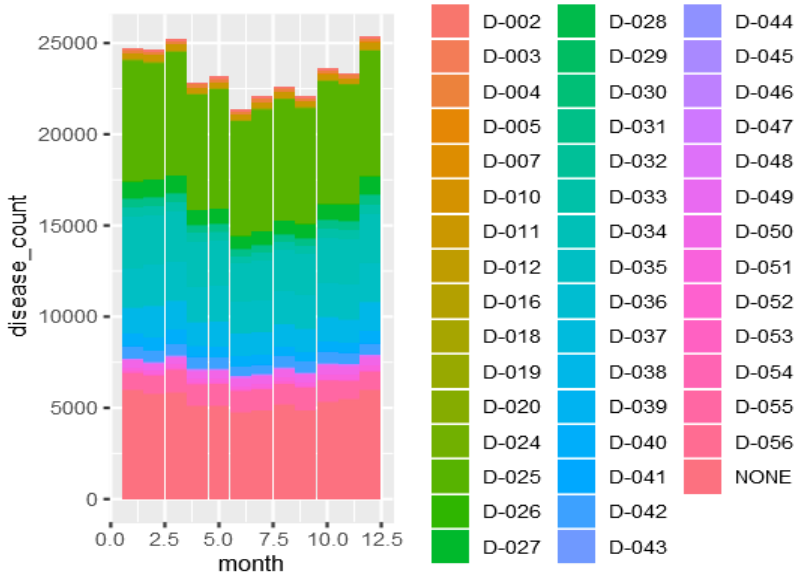
전체적으로 여성 사망자 수보다 남성 사망자 수가 더 크다. 또한, 여성과 남성 사망자 수 모두 1번의 month별 사망자 수와 같이 v자 계곡의 형상을 띄고 있다.

- [그림6]gender별 ageG별 month별 사망자 수



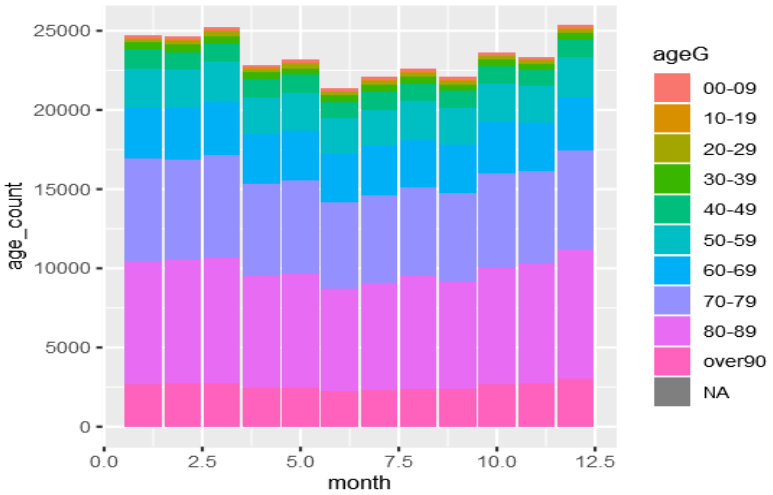
좌측은 bar chart로 우측은 꺾은 선으로 표현한 것이다. 남성의 사망자 수가 급격히 가팔라지는 나이대는 40-49세로 여성보다 낮다. 남성의 나이대가 40-49세부터 사망자 수가 급격히 가팔라져 70-79세에 사망자 수 peak를 보인다. 반면 여성은 나이대가 조금씩 높아짐에 따라 사망자 수가 상승세를 완만히 보이다가 70-79세에 급격히 증가하는 양상을 보이며 80-89세에서 사망자 수의 peak를 보인다. 이러한 여성 사망자 수의 나이대가 남성 사망자 수의 나이대보다 우측으로 skewed 분포는 꺾은 선 그래프에서 직관적으로 관찰할 수 있다.

- [그림7]month별 D56별 사망자 수



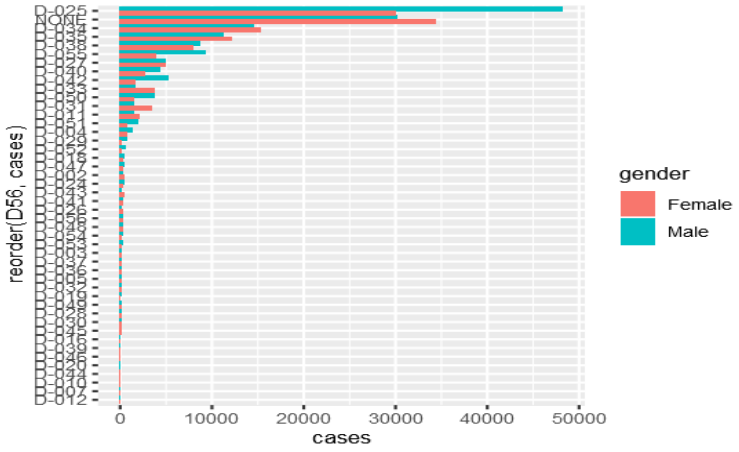
D56의 level의 개수는 46개로 다소 알아보기 어려운 그래프다. 위의 [그림2]D56별 사망자 수 그래프를 통해 D-025, D-034, NONE임을 유추해 볼 수 있으나 다른 category는 해석하기 어렵다. 그러나, 이 그래프를 통해 대부분의 질병이 [그림1]month별 사망자 수와 같이 V자 계곡 모양을 띔을 알 수 있다.

- [그림8]month별 ageG별 사망자 수



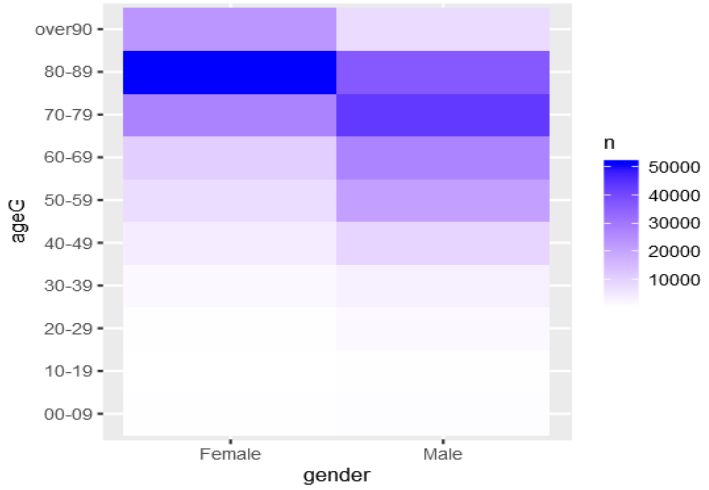
위의 그래프를 통해 나이대 별 사망자 수도 [그림1]month별 사망자 수와 같이 V자 계곡 모양을 띔을 알 수 있다.

- [그림9]gender별 D56별 사망자 수



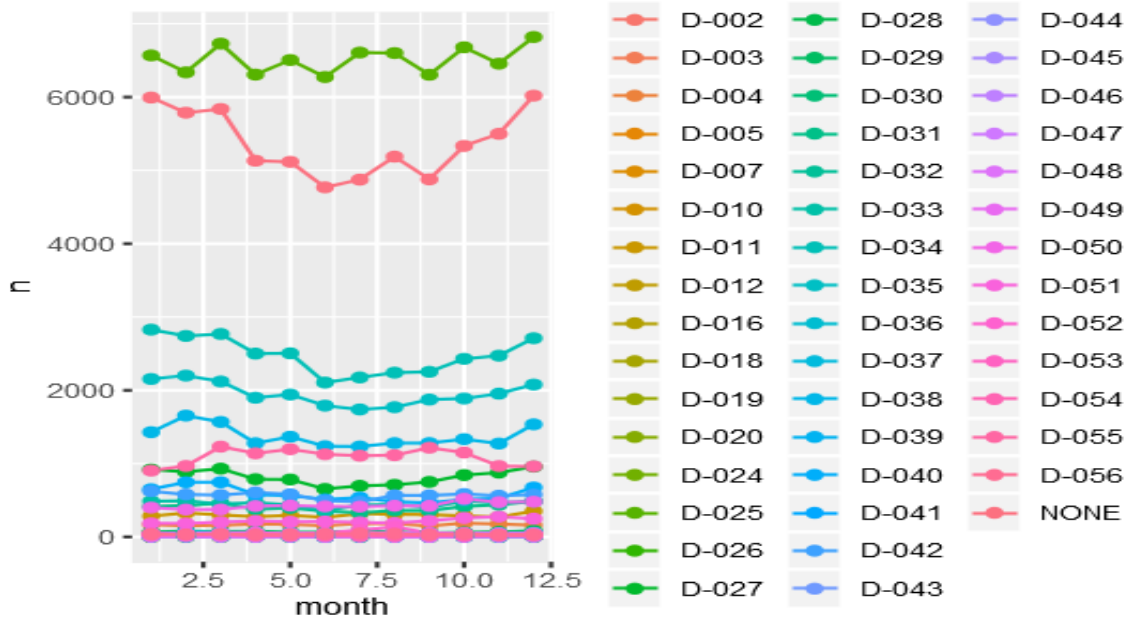
위 그래프를 통해 D-025(악성신생물(Malignant neoplasms))의 경우 남성의 사망자 수가 여성의 사망자 수보다 월등히 높다는 것을 알 수 있다. NONE의 경우 여성의 사망자 수가 남성의 사망자 수보다 높다. D-34(심장 질환(Heart diseases)) 그리고 D-35(뇌혈관 질환(Cerebrovascular diseases)) 또한 여성의 사망자 수가 남성의 사망자 수보다 높다. D-38(폐렴(Pneumonia)) 그리고 D-55(고의적 자해(자살) (Intentional self-harm))의 경우 남성의 사망자 수가 여성의 사망자 수보다 높다. 그러나, 사망 원인에 따른 사망자 수의 차이가 너무 크기 때문에 이 그래프를 통해서는 직관적으로 성별에 따라 사망 원인의 양상을 이해할 수 없다.

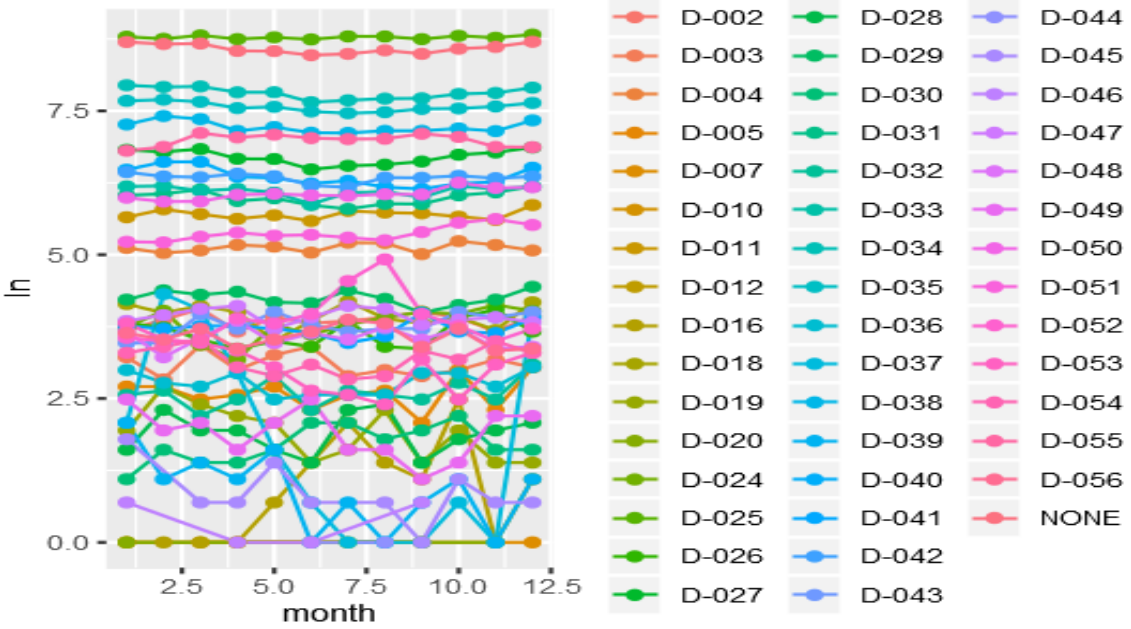
- [그림10]ageG별 gender별 사망자 수



위 그래프를 통해 성별에 관계 없이 나이대가 높아짐에 따라 사망자 수가 커짐을 알 수 있다. 여성의 경우 80-89세의 사망자 수가 가장 많다. 반면에 남성은 40-49세부터 여성에 비해 높은 사망자수를 보이며 점진적으로 사망자 수가 커지다가 70-79세에 가장 높은 것을 알 수 있다. 위 그래프는 [그림6]과 비슷한 양상을 보인다.

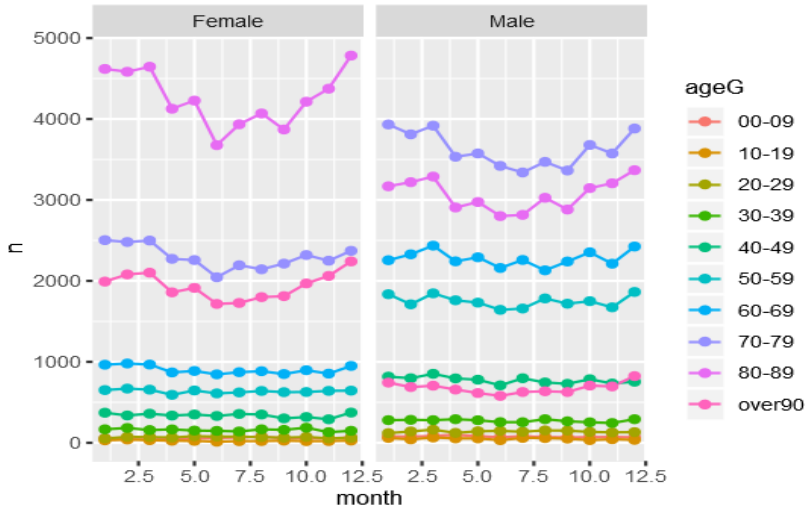
- [그림11]month별 D56별 그래프(상)/ [그림12] month별 D56별 그래프 log scaling(하)





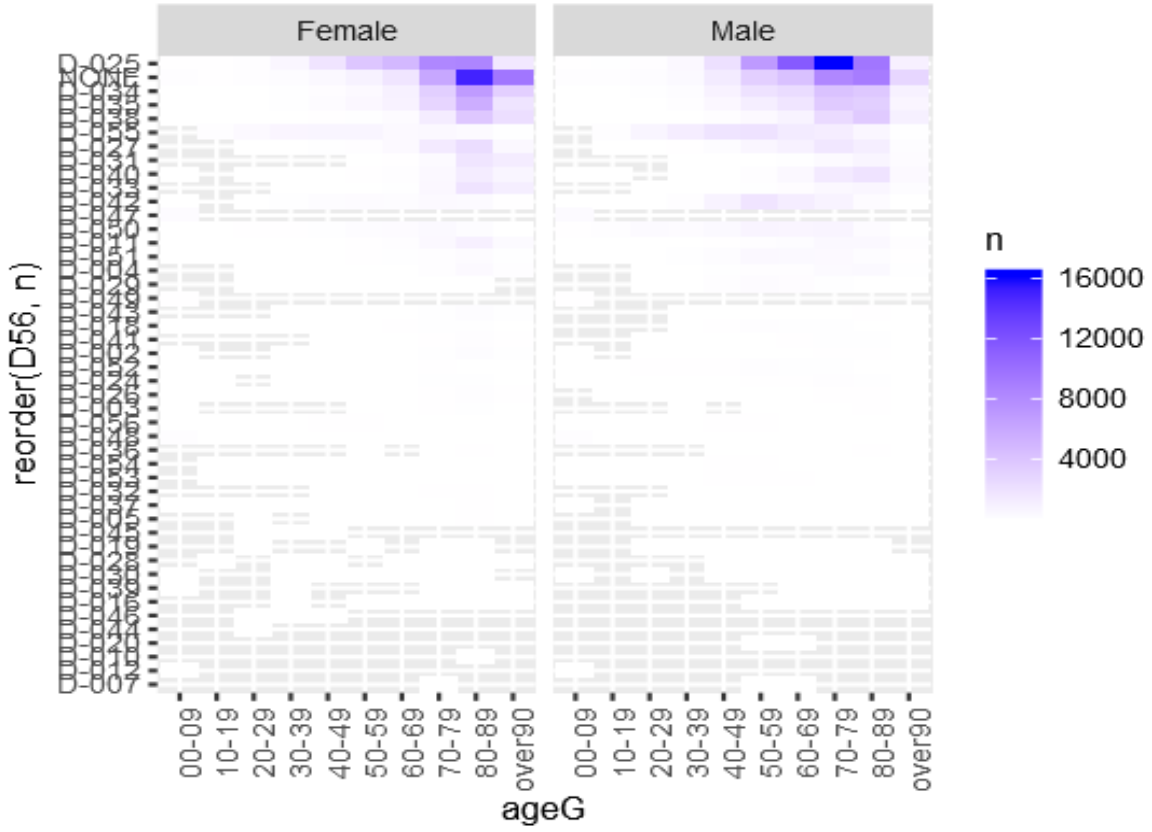
위 두 그래프를 통해 사망원인 별 사망자 수가 달 별로 영향을 받는지 알 수 있다. [그림11]의 경우 사망원인 별 사망자 수의 차가 너무 커 [그림12]에서 이를 log scale해주었다. 그 결과, [그림12]에서 특정 사망원인은 달 별로 사망자수가 달라짐을 알 수 있다. 그러나, 이 그래프의 levels가 너무 많아 어떤 사망원인인지 알아보기 쉽지 않다.

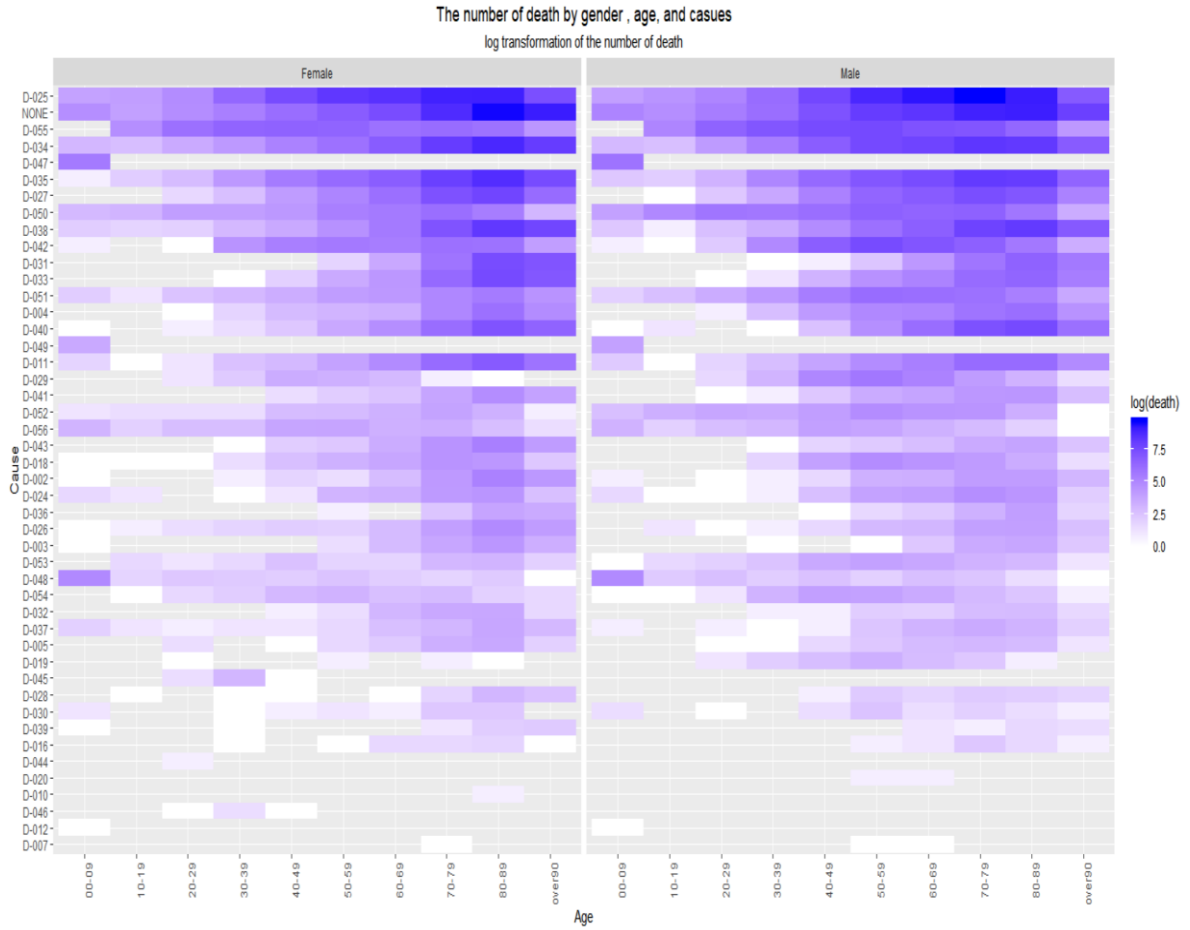
- [그림13]gender별 month별 ageG별 사망자 수



위 그래프는 [그림8]을 gender별로 비교하기 쉽게 만들어준다. 여성의 경우 [그림10]에서도 알 수 있듯이 80-89세에 가장 사망자 수가 많고 깊은 달 별로 V자 계곡 모양을 보여준다. 다음으로는 70-79세 그리고 over90이 뒤 따른다. 그 외의 나이대는 1000명 이하의 사망자 수를 보인다. 남성의 경우 [그림10]에서도 알 수 있듯이 70-79세에 가장 사망자 수가 많고 80-89세, 60-69세, 그리고 50-59세가 뒤따른다. 나머지 나이대에서 남성 사망자는 1000명 이하를 보인다.

- [그림14] ageG별, gender별, D56별 사망자 수(상)/ [그림15] [그림14]의 log scaling





[그림14]를 통해 ageG별, gender별, D56별 사망자 수를 파악하려고 했으나, 각 case별 사망자 수의 차이가 너무 커 해석이 쉽지 않았다. 따라서, [그림14]를 log scaling하여 [그림15]를 만들었다. [그림15]는 위의 [그림1]-[그림13]을 잘 함축하고 있어 Midterm-EDAdata1-2020.csv를 가장 잘 나타내는 그림으로 선택하였다. 여성의 사망자 수는 80-89세 NONE에서 가장 많이 나타나는 반면 남성 사망자 수는 70-79세 D-56(악성신생물(Malignant neoplasms))에서 가장 많이 나타난다. 그리고 사망자 수의 scale차이로 알 수 없었던 ageG별 D56별 사망자 수를 알 수 있다. 특히 00-09세 사망자 수가 40-49세 이전 나이대에서 비교적 많은 편인데, 남성 여성 사망자 모두 사망 원인으로 D-47(출생전후기에 기원한 특정 병태(Certain conditions originating in the perinatal period)) 그리고 D-48(선천 기형, 변형 및 염색체 이상(Congenital malformations, deformations and chromosomal abnormalities))이 높았다. 비교적 젊은 나이대의 남성 사망자 수가 높은 사망원인은 D-50(운수 사고(Transport accidents))이었다. 이와 같이 나이대, 성별로 사망자 수가 많은 사망원인을 비교해 볼 수 있다. 그러나 위 그림은 month별 사망자 수의 추이를 보여주지 못한다는 단점이 있다.

5. Midterm-EDAdata2-2020.csv 파일에는 16개의 변수가 있다. 이 자료에 숨어있는 정보를 찾아내시오. (숨어있는 정보를 찾아가는 과정에 사용한 code를 모두 제시해야합니다.)

```
rm(list=ls(all=TRUE))

# data load
df <- read_csv('./midtermdata/Midterm-EDAdata2-2020.csv')

## Warning: Missing column names filled in: 'X1' [1]

## Warning: Duplicated column names deduplicated: 'X1' => 'X1_1' [2]

## Parsed with column specification:
## cols(
##   X1 = col_double(),
##   X1_1 = col_double(),
##   X2 = col_double(),
##   X3 = col_double(),
##   X4 = col_double(),
##   X5 = col_double(),
##   X6 = col_double(),
##   X7 = col_double(),
##   X8 = col_double(),
##   X9 = col_double(),
##   X10 = col_double(),
##   X11 = col_double(),
##   X12 = col_double(),
##   X13 = col_double(),
##   X14 = col_double(),
##   X15 = col_double(),
##   X16 = col_double()
## )

# data 확인
names(df)

##  [1] "X1"   "X1_1" "X2"   "X3"   "X4"   "X5"   "X6"   "X7"   "X8"   "X9"
## [11] "X10"  "X11"  "X12"  "X13"  "X14"  "X15"  "X16"

summary(df)

##        X1            X1_1               X2
##  Min.   :    1   Min.   :-3.96789   Min.   :-1.7317151
##  1st Qu.: 3751   1st Qu.:-0.67826   1st Qu.:-0.8565850
##  Median : 7500   Median :-0.01137   Median : 0.0004448
##  Mean   : 7500   Mean   : 0.00000   Mean   : 0.0000000
##  3rd Qu.:11250   3rd Qu.: 0.68367   3rd Qu.: 0.8719026
##  Max.   :15000   Max.   : 3.96330   Max.   : 1.7270692
##        X3                 X4                 X5
##  Min.   :-3.445573   Min.   :-3.837054   Min.   :-3.68834
##  1st Qu.:-0.660734   1st Qu.:-0.681561   1st Qu.:-0.74135
##  Median :-0.000447   Median : 0.003444   Median : 0.01541
##  Mean   : 0.000000   Mean   : 0.000000   Mean   : 0.00000
```

```
##  3rd Qu.: 0.682132   3rd Qu.: 0.679206   3rd Qu.: 0.73879
##  Max.   : 3.980660   Max.   : 3.909845   Max.   : 3.68820
##        X6                  X7                  X8
##  Min.   :-1.729535   Min.   :-2.216450   Min.   :-4.44085
##  1st Qu.:-0.866752   1st Qu.:-0.786065   1st Qu.:-0.37902
##  Median :-0.005454   Median : 0.004614   Median : 0.05458
##  Mean   : 0.000000   Mean   : 0.000000   Mean   : 0.00000
##  3rd Qu.: 0.862294   3rd Qu.: 0.786019   3rd Qu.: 0.43800
##  Max.   : 1.733282   Max.   : 2.186460   Max.   : 5.15849
##        X9                 X10                 X11
##  Min.   :-2.228152   Min.   :-1.4097   Min.   :-1.7235119
##  1st Qu.:-0.781350   1st Qu.:-0.7353   1st Qu.:-0.8615051
##  Median : 0.002977   Median :-0.2218   Median :-0.0003596
##  Mean   : 0.000000   Mean   : 0.0000   Mean   : 0.0000000
##  3rd Qu.: 0.779801   3rd Qu.: 0.4747   3rd Qu.: 0.8684710
##  Max.   : 2.208837   Max.   : 7.6770   Max.   : 1.7294227
##       X12                 X13                 X14                 X15
##  Min.   :-1.7917   Min.   :-3.87342   Min.   :-1.4055   Min.   :-1.7933
##  1st Qu.:-0.5344   1st Qu.:-0.69216   1st Qu.:-0.7369   1st Qu.:-0.5352
##  Median : 0.7230   Median : 0.03723   Median :-0.2272   Median : 0.7229
##  Mean   : 0.0000   Mean   : 0.00000   Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.: 0.7230   3rd Qu.: 0.70816   3rd Qu.: 0.4852   3rd Qu.: 0.7229
##  Max.   : 0.7230   Max.   : 3.63084   Max.   : 7.4792   Max.   : 0.7229
##       X16
##  Min.   :-1.4156
##  1st Qu.:-0.8493
##  Median :-0.1684
##  Mean   : 0.0000
##  3rd Qu.: 0.7126
##  Max.   : 2.7858
```

```
for(i in 1:16){
  print(sum(is.na(df[[i]])))
}
```

```
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
## [1] 0
```

```
length(unique(df$X1)) # row name 으로 보임. 필요 없다고 판단
```

```
## [1] 15000
```

```
df <- df[,-1]
names(df)[1] <- "X1"
names(df)
```

```
##  [1] "X1"  "X2"  "X3"  "X4"  "X5"  "X6"  "X7"  "X8"  "X9"  "X10" "X11"
## [12] "X12" "X13" "X14" "X15" "X16"
```

```
str(df)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    15000 obs. of  16
variables:
##  $ X1 : num  -1.402 -2.036 1.761 0.659 -0.325 ...
##  $ X2 : num  -0.661 -1.08 -0.915 -0.761 -1.594 ...
##  $ X3 : num  -0.746 -0.474 0.631 1.746 0.49 ...
##  $ X4 : num  0.1935 -0.0257 0.1479 0.2776 0.4601 ...
##  $ X5 : num  -1.84 -1.84 -1.84 -1.84 -1.84 ...
##  $ X6 : num  -0.752 1.39 0.134 0.585 -0.267 ...
##  $ X7 : num  -1.09 1.45 -1.74 -1.81 1.13 ...
##  $ X8 : num  0.3785 0.1773 0.0889 0.0938 0.1429 ...
##  $ X9 : num  0.6 1.88 0.441 1.454 -0.139 ...
##  $ X10: num  -0.323 -0.713 -0.11 -0.503 -0.602 ...
##  $ X11: num  0.124 0.69 -1.032 1.112 -1.326 ...
##  $ X12: num  -0.534 -1.792 -1.792 -1.792 -1.792 ...
##  $ X13: num  0.05246 0.03879 0.02513 0.01147 -0.00219 ...
##  $ X14: num  1.9969 0.7189 -0.0046 -0.4075 -0.3955 ...
##  $ X15: num  -1.793 -0.535 -0.535 -0.535 -0.535 ...
##  $ X16: num  -0.446 -0.838 -0.718 0.401 0.124 ...
```

```
dim(df)
```

```
## [1] 15000    16
```

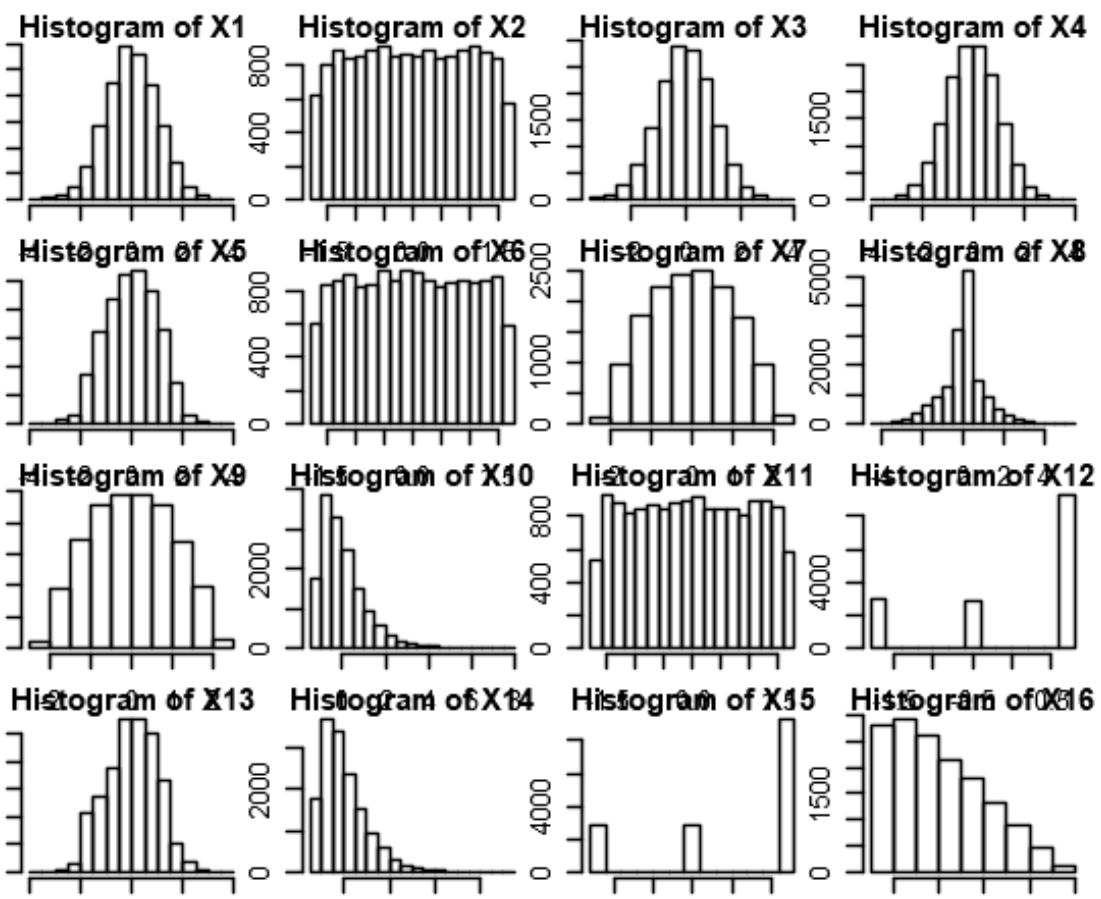
```
summary(df) # mean 값이 모두 0 이다. scale 데이터
```

```
##        X1                  X2                   X3
##  Min.   :-3.96789   Min.   :-1.7317151   Min.   :-3.445573
##  1st Qu.:-0.67826   1st Qu.:-0.8565850   1st Qu.:-0.660734
##  Median :-0.01137   Median : 0.0004448   Median :-0.000447
##  Mean   : 0.00000   Mean   : 0.0000000   Mean   : 0.000000
##  3rd Qu.: 0.68367   3rd Qu.: 0.8719026   3rd Qu.: 0.682132
##  Max.   : 3.96330   Max.   : 1.7270692   Max.   : 3.980660
##        X4                   X5                   X6
##  Min.   :-3.837054   Min.   :-3.68834   Min.   :-1.729535
##  1st Qu.:-0.681561   1st Qu.:-0.74135   1st Qu.:-0.866752
##  Median : 0.003444   Median : 0.01541   Median :-0.005454
##  Mean   : 0.000000   Mean   : 0.00000   Mean   : 0.000000
##  3rd Qu.: 0.679206   3rd Qu.: 0.73879   3rd Qu.: 0.862294
##  Max.   : 3.909845   Max.   : 3.68820   Max.   : 1.733282
##        X7                   X8                   X9
##  Min.   :-2.216450   Min.   :-4.44085   Min.   :-2.228152
##  1st Qu.:-0.786065   1st Qu.:-0.37902   1st Qu.:-0.781350
##  Median : 0.004614   Median : 0.05458   Median : 0.002977
```

```
##  Mean   : 0.000000   Mean   : 0.00000   Mean   : 0.000000
##  3rd Qu.: 0.786019   3rd Qu.: 0.43800   3rd Qu.: 0.779801
##  Max.   : 2.186460   Max.   : 5.15849   Max.   : 2.208837
##       X10                X11                X12
##  Min.   :-1.4097   Min.   :-1.7235119   Min.   :-1.7917
##  1st Qu.:-0.7353   1st Qu.:-0.8615051   1st Qu.:-0.5344
##  Median :-0.2218   Median :-0.0003596   Median : 0.7230
##  Mean   : 0.0000   Mean   : 0.0000000   Mean   : 0.0000
##  3rd Qu.: 0.4747   3rd Qu.: 0.8684710   3rd Qu.: 0.7230
##  Max.   : 7.6770   Max.   : 1.7294227   Max.   : 0.7230
##       X13                X14                X15                X16
##  Min.   :-3.87342   Min.   :-1.4055   Min.   :-1.7933   Min.   :-1.4156
##  1st Qu.:-0.69216   1st Qu.:-0.7369   1st Qu.:-0.5352   1st Qu.:-0.8493
##  Median : 0.03723   Median :-0.2272   Median : 0.7229   Median :-0.1684
##  Mean   : 0.00000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.: 0.70816   3rd Qu.: 0.4852   3rd Qu.: 0.7229   3rd Qu.: 0.7126
##  Max.   : 3.63084   Max.   : 7.4792   Max.   : 0.7229   Max.   : 2.7858
```

```r
par(mfrow = c(4,4))
par(mar=c(1,1,1,1))
for(i in 1:16){
  hist(df[[i]], main = paste0('Histogram of X', i), xlab = paste0('X',i))
}
```

[그림 16]

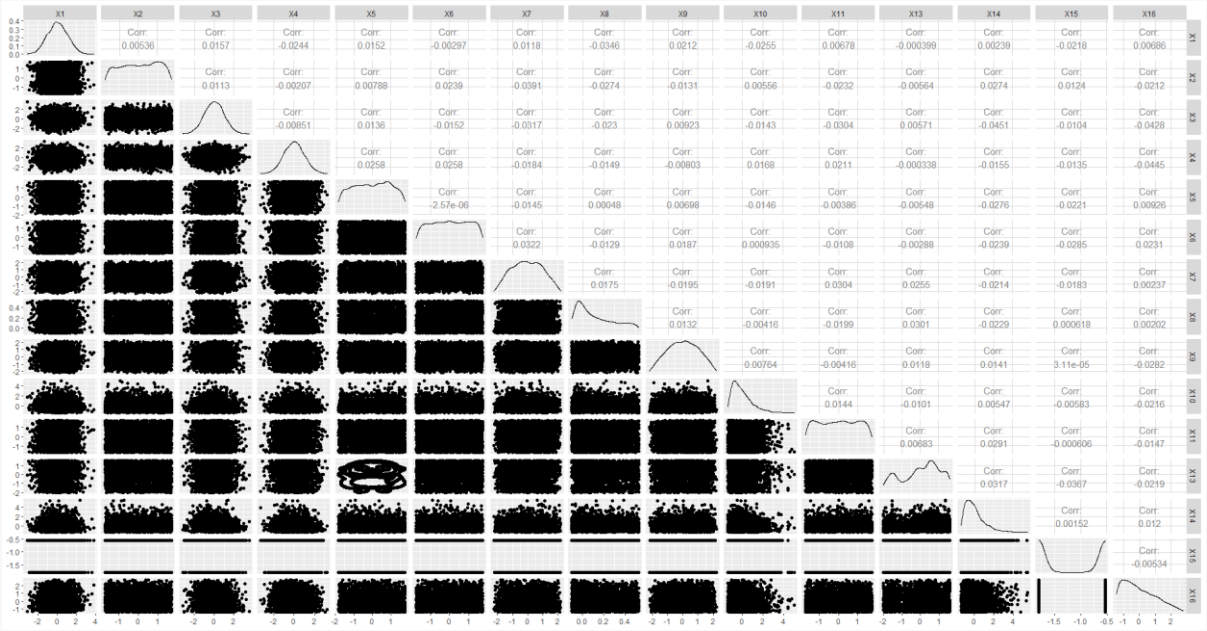


```r
par(mfrow = c(1,1))

# 반복되는 자료 : X12, X15
unique(df$X12)

## [1] -0.5343691 -1.7917081  0.7229699

unique(df$X15)

## [1] -1.7932606 -0.5351853  0.7228901

nest <- df %>% mutate(X12 = factor(X12))%>% group_by(X12) %>% nest
nest2 <- df %>%mutate(X15 = factor(X15))%>% group_by(X15) %>% nest
```

correlation 알아보기

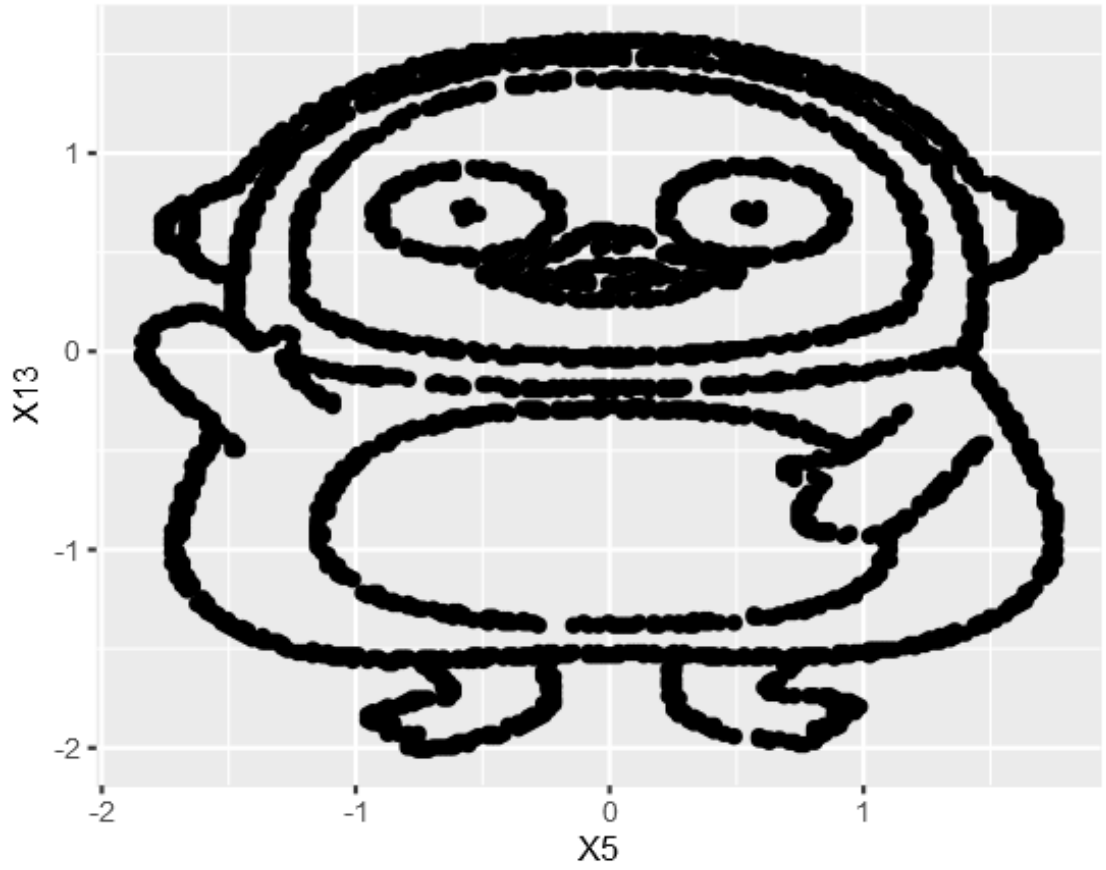
```r
library(GGally)
ggpairs(nest$data[[1]])
```

[그림 17]



```r
ggpairs(nest$data[[2]])
ggpairs(nest$data[[3]])

ggpairs(nest2$data[[1]])
ggpairs(nest2$data[[2]])
ggpairs(nest2$data[[3]])

nest$data[[1]] %>% ggplot(aes(X5, X13)) + geom_point()
```

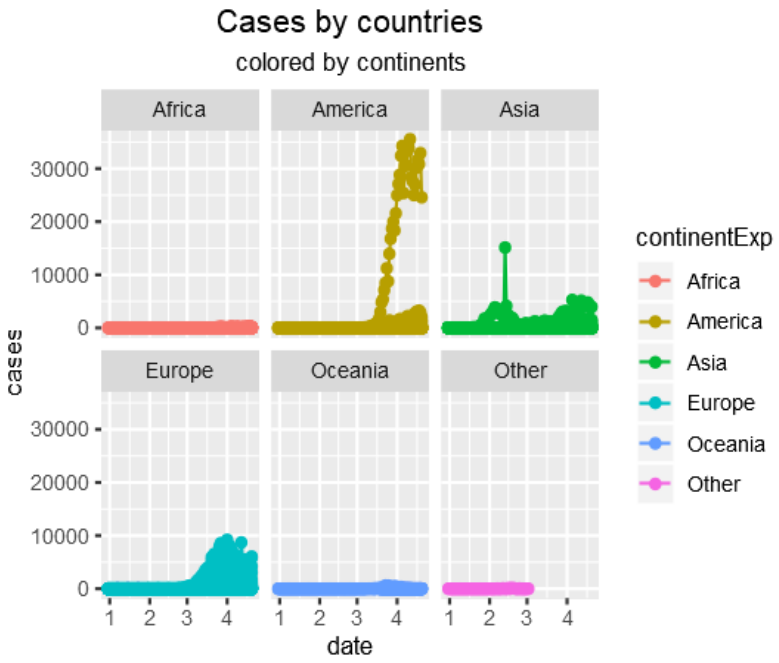
[그림 18]




```
nest$data[[2]] %>% ggplot(aes(X5, X13)) + geom_point()
nest$data[[3]] %>% ggplot(aes(X5, X13)) + geom_point()
nest2$data[[1]] %>% ggplot(aes(X5, X13)) + geom_point()
nest2$data[[2]] %>% ggplot(aes(X5, X13)) + geom_point()
nest2$data[[3]] %>% ggplot(aes(X5, X13)) + geom_point()
```

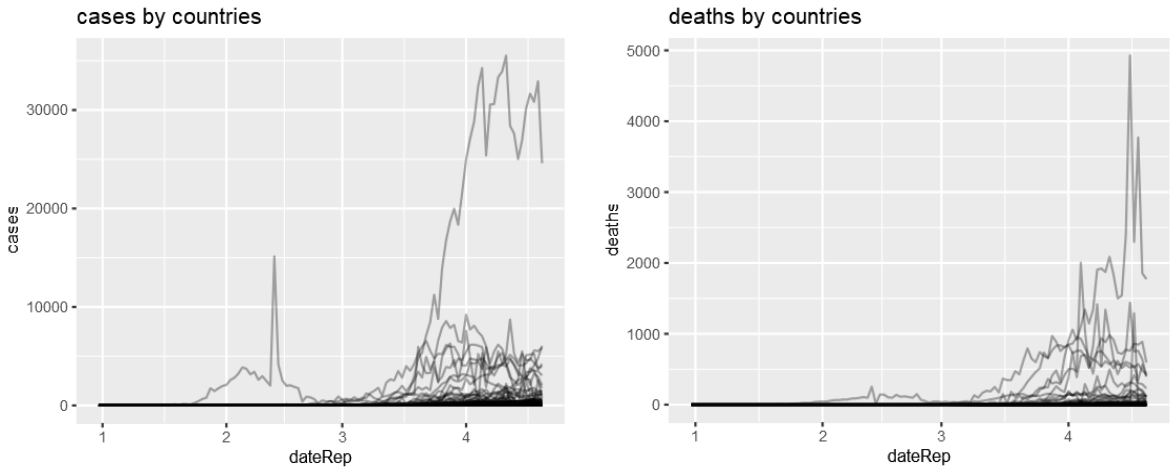
6. COVID-19-2020-04-20.csv 파일은 2019년 12월 31일부터 2020년 4월 20일까지의 전세계의 코로나 확진자와 사망자에 대한 자료이다. 이 자료를 이용하여 자료의 특성을 살펴보고 각 나라마다의 코로나 확진 추이의 차이를 살펴보시오.

- [그림19] 국가별 대륙별 확진자 추이



국가별 확진자 수의 차이가 크기 때문에 국가별로 해석하기는 어렵다. 그러나 대륙별 확진자 추이의 변화를 대략적으로 알 수 있다. America대륙은 3월 중순을 기점으로 확진자 수가 급격히 증가했으며, Asia는 2월부터 꾸준히 증가폭을 보이며, Europe은 America대륙 만큼은 아니지만 3월부터 확진자 수가 급격하게 증가하였다.

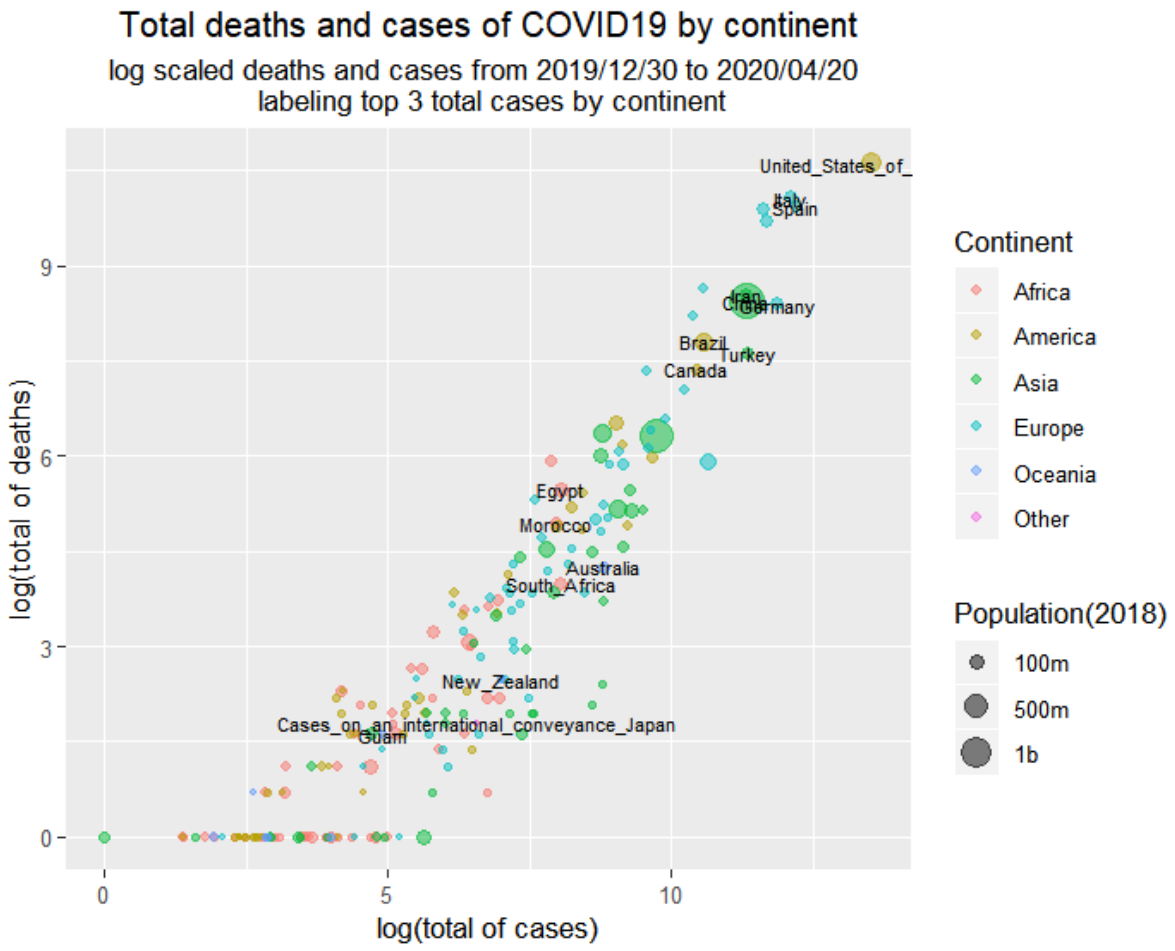
- [그림20] 국가별 확진자 수/ [그림21] 국가별 사망자 수



위 그래프를 통해 확진자 수의 급격한 증가가 일어난 시기는 2월과 3월말~4월 두 번이 있음을 알 수 있다. 2월에는 사망자 수의 증가 추이가 확진자 수의 증가 추이에 비해 미미하지만 3월말

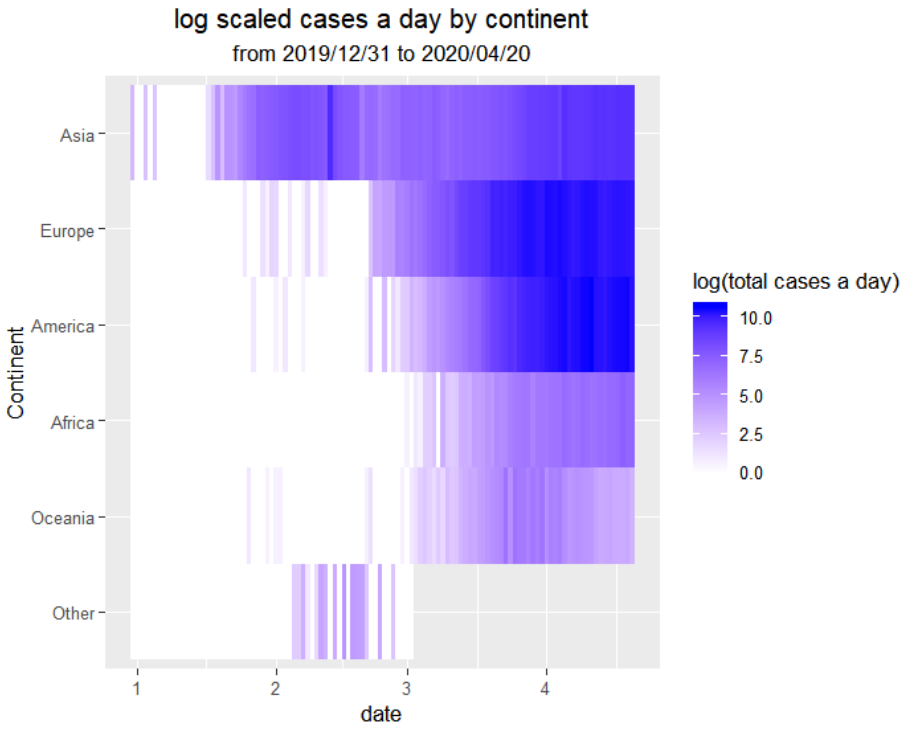
~4월은 확진자 수의 추이의 변동만큼 사망자 수의 추이도 비슷하게 움직인다.

- [그림22] 국가별 대륙별 log(인구수), log(확진자수), 그리고 log(사망자수)



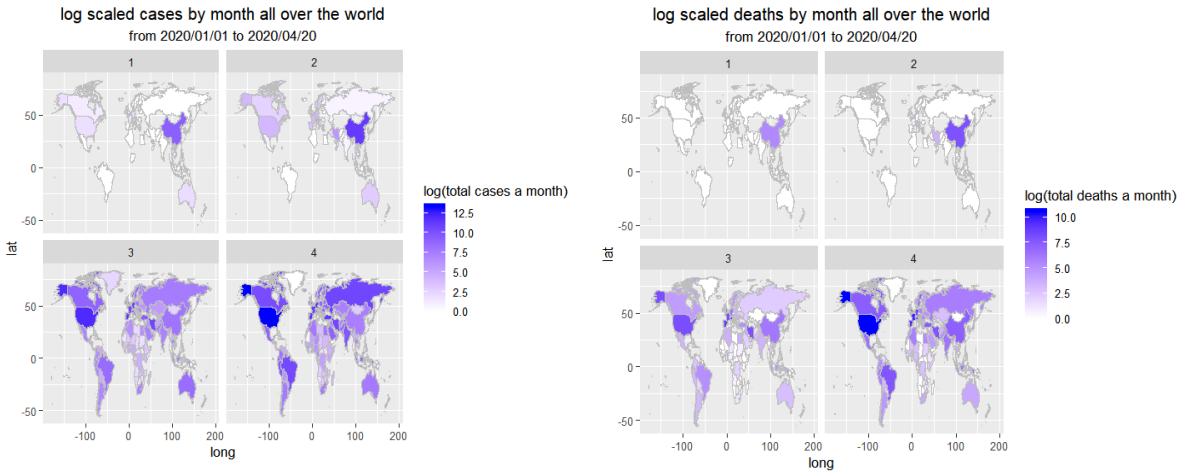
위 그래프는 각 국가별 2019/12/31 ~ 2020년 4월 20일까지의 COVID-19 확진자 수의 합, 사망자 수의 합, 인구를 log scale하여 각 국가별, 대륙별 인구수와 확진자 및 사망자수를 비교할 수 있다. 이때, 확진자 수의 합과 사망자 수의 합은 log scale하기 위해 값이 0 인 경우 1로 변경하고 그래프를 그렸다. 라벨된 국가명은 각 대륙별 확진자 수의 합 상위 3위에 속하는 국가들이다. America 대륙의 United_States_of_America가 가장 확진자 수의 합이 크다. 그 뒤를 Europe의 Italy와 Spain이 따른다. 각 대륙별 확진자 수의 합 상위 3위는 America 대륙의 경우 United_States_of_America, Brazil, Canada, Europe의 경우 Italy, Spain, 그리고 Germany이다. Asia의 경우 China, Iran, Turkey이며, Africa는 Egypt, Morocco, South_Africa이다. 마지막으로 Oceania는 Australia, New_Zealand, Guam이다. Other에는 일본 유람선이 포함되어 있다.

- [그림23] 대륙별 확진자 수 추이



위 그래프는 [그림19]에서 얻은 직관적인 해석을 더욱 선명하게 보여준다. Asia에서 확진자 수가 2월에 점차 증가하다가 3월에는 다시 감소 추세이다가 4월에 조금씩 증가한다. Europe은 3월부터 확진자 수가 증가하기 시작한다. America는 Europe에 바통을 이어 받아 3월 중순부터 증가하기 시작한다. Europe과 America의 증가와 함께 Africa 그리고 Oceania도 3월을 기점으로 차츰 증가하기 시작한다.

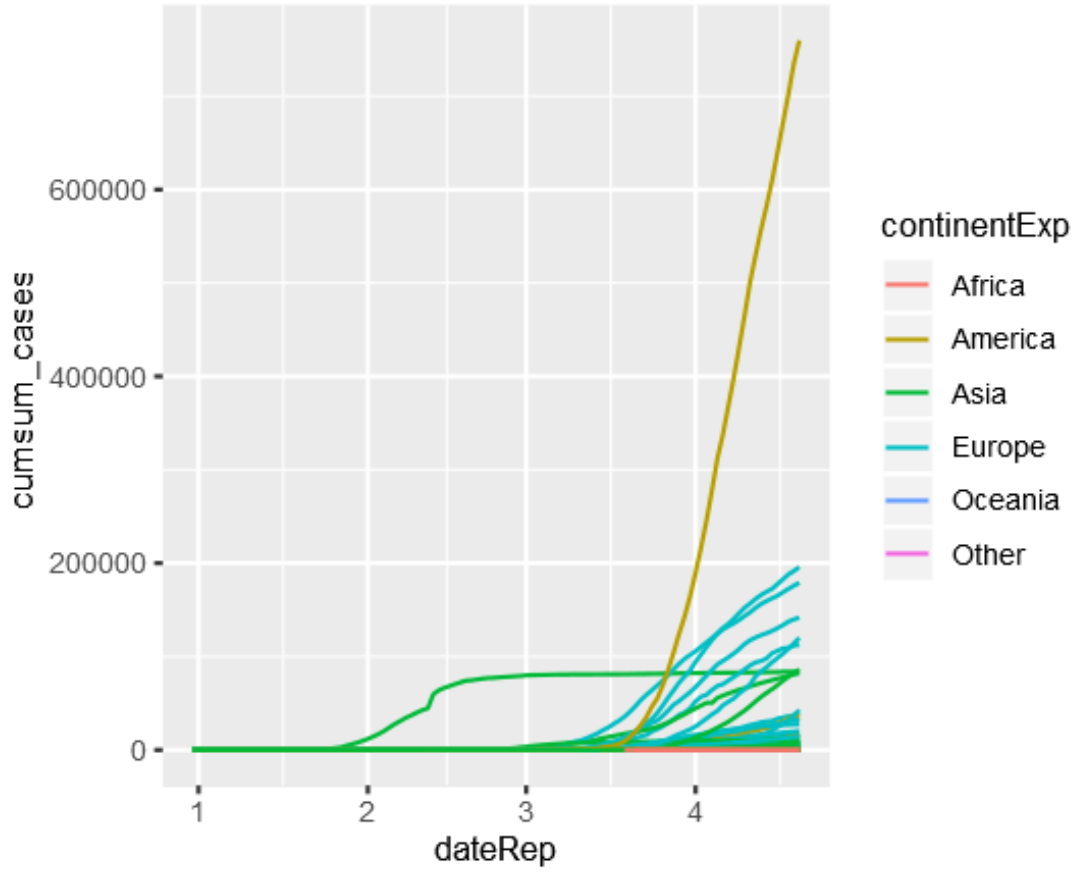
- [그림24] 전 세계 달별 확진자 수 합계 추이/ [그림25] 전 세계 달별 사망자 수 합계 추이



위 그래프를 통해 전세계 확진자 수 및 사망자 수 달별 합계 현황을 비교할 수 있다. 중국의 경우 1월~2월 확진자 수 증가가 많았다가 3~4월에 진전된다. 3월부터 전세계적으로 코로나가 퍼지기 시작한다. 3월 11일 세계보건기구(WHO)에서 코로나 판데믹 선언을 하였다. 특히 아메리카 대

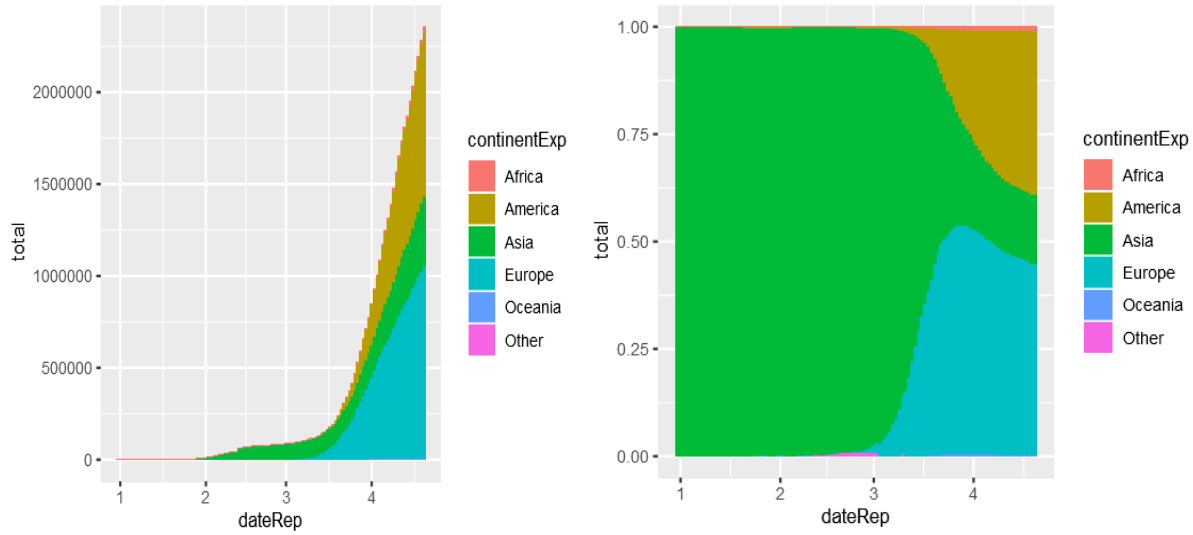
륙이 압도적인 증가폭을 보인다. 러시아 대륙, 남아메리카, 유럽, 그리고 오세아니아도 코로나 확진자 수가 증가한다. 다른 대륙에 비해 증가 폭이 작지만 1,2월에는 확진자 및 사망자 자료가 없어 그림이 없었던 아프리카 대륙도 확진자 수가 점차적으로 증가하는 추세를 보인다. 사망자 수도 확진자 수와 비슷한 증가폭을 보이지만 절대적인 수치는 작다.

- [그림26]대륙별 국가별 누적 확진자 수



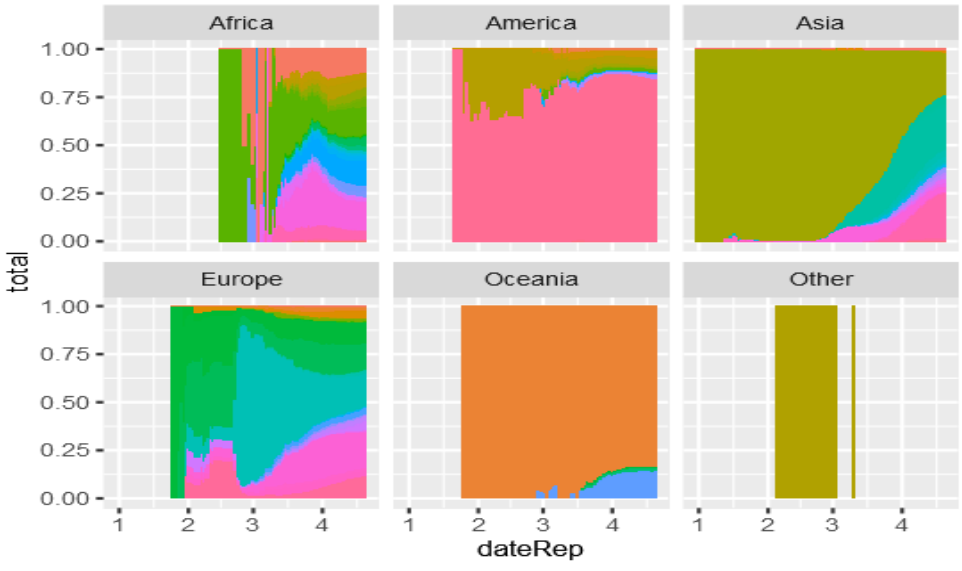
아시아의 한 국가는 2월에 크게 증가하였으나 3~4월 증가폭이 거의 없다. 반면 아메리카 대륙의 한 국가는 3월 중순을 기점으로 매우 가파른 속도로 누적 확진자 수가 증가한다. 유럽의 많은 국가들도 아메리카보다 조금 이른 3월을 기점으로 상승세를 보인다. 그에 따라 아프리카 대륙의 누적 확진자 수도 점차적으로 증가하는 것으로 볼 수 있다.

- [그림27] 누적 확진자 수 대륙별 비율/ [그림28] [그림27]의 전체 1비율



위 그래프를 통해 코로나 전세계 누적 확진자 수와 대륙별 확진자수 추이를 알 수 있다. 1~2월까지 아시아가 대부분의 확진자 수를 차지하고 있었다면 3월부터 유럽이 상승폭과 함께 유럽의 비중이 커지고 동시에 아메리카의 확진자수가 증가하면서 확진자 수 중 아시아인의 비율이 큰 폭으로 줄어든다.

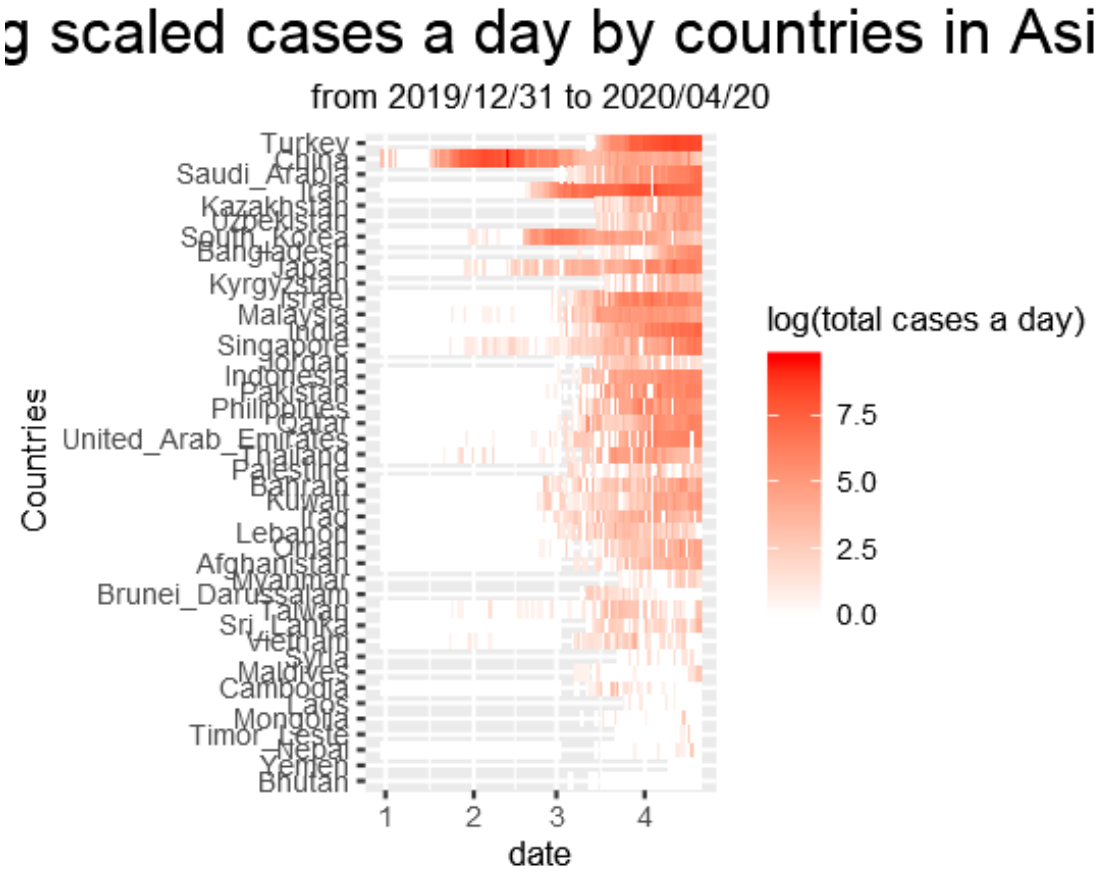
- [그림29] 각 대륙별 각 국가별 누적 확진자 수 비율



Asia는 2월까지 한 국가가 독식하는 양상에서 3월부터 다른 아시아 국가들의 확진자 수가 증가함에 따라 4개의 국가가 비슷한 비율을 갖는다. 아메리카는 2월부터 확진자 수가 생기고 한 국가가 독식하는 구조이다. Oceania도 비슷하다. Europe과 Africa는 비교적 다양한 국가들이 확진자 수 비중을 차지한다. 모든 대륙이 처음에는 한 국가가 1을 차지하고 있었지만 시간이 지남에 따라 확

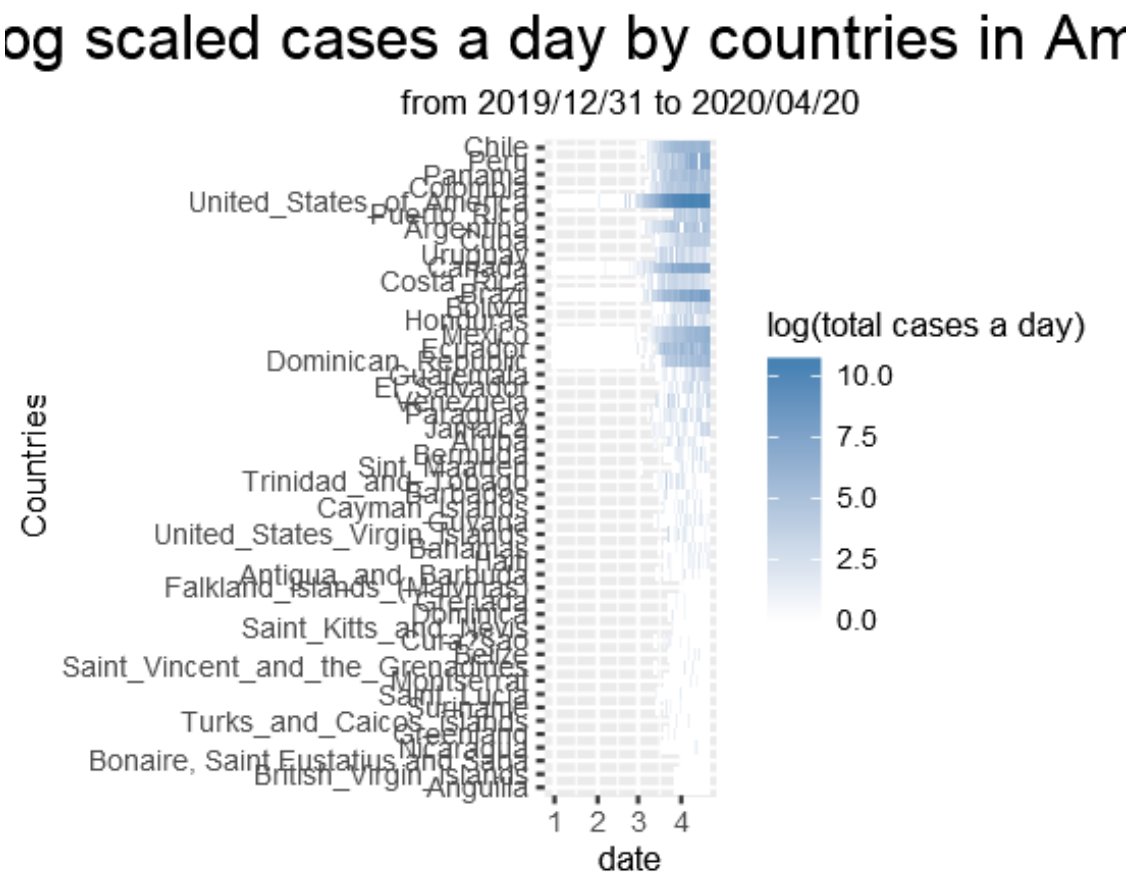
진자 수 비율이 나뉜다. 한 대륙내 다른 국가로의 감염의 양상으로 판단할 수 있다. 특히 이 양상은 유럽이 가장 빠른 것으로 보인다.

- [그림30] 아시아 대륙의 국가별 log scale된 확진자 수



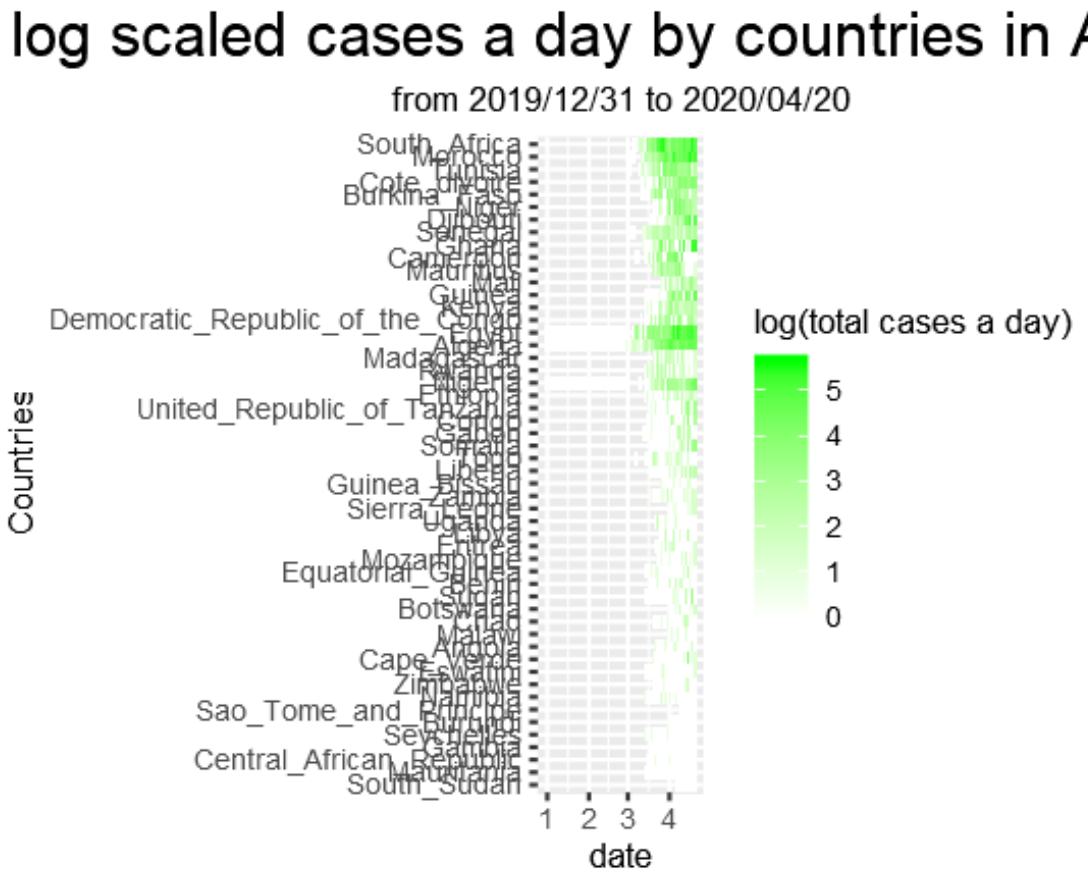
China는 1월부터 확진자 수가 증가하기 시작하여 2월에 가장 peak를 찍고 점점 상승폭이 감소한다. 2월 중순부터 South_Korea, Iran의 확진자 수 증가가 시작된다. Turkey는 3월부터 확진자수가 증가하기 시작한다. 아시아의 대부분 국가들은 3월부터 시작된다.

- [그림31] 아메리카 대륙의 국가별 log scale된 확진자 수



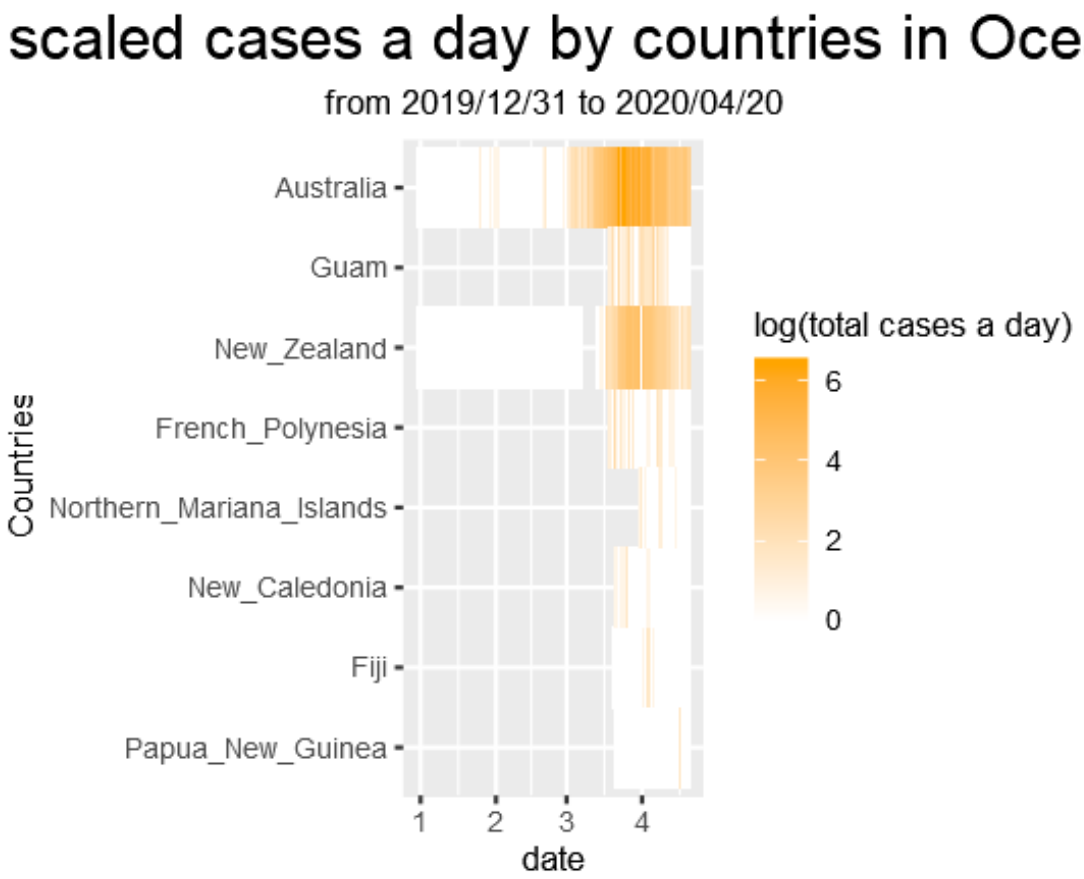
아메리카 대륙의 경우 United_States_of_America의 증가폭이 압도적이다. 이를 캐나다와 브라질이 뒤따른다. 나머지 국가는 아직은 크게 증가폭이 없다.

- [그림32] 아프리카 대륙의 국가별 log scale된 확진자 수



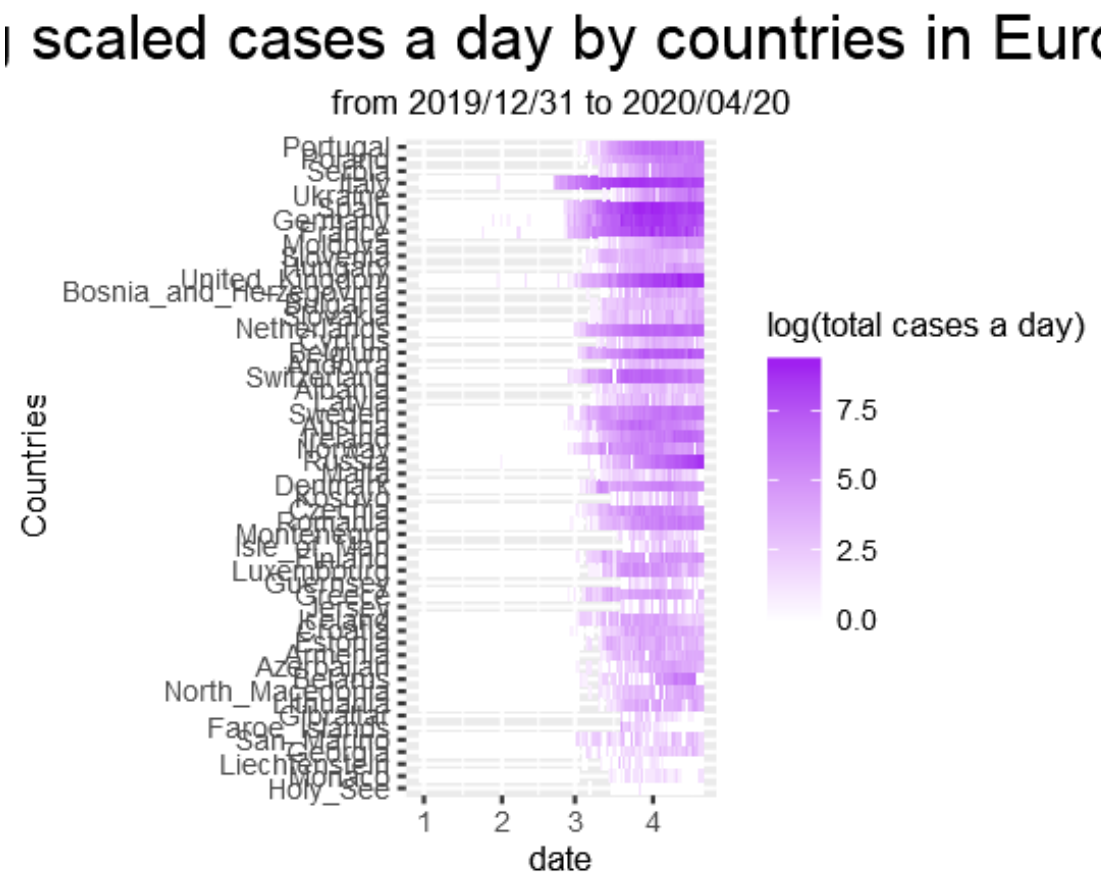
Africa대륙의 국가들의 확진자 수 증가 역시 3월부터 시작되었다. South_Africa와 Morocco 그리고 Egypt의 증가폭이 크다.

- [그림33] 오세아니아 대륙의 국가별 log scale된 확진자 수



오세아니아의 오스트레일리아와 뉴질랜드를 중심으로 확진자 수가 증가한다.

- [그림34] 유럽 대륙의 국가별 log scale된 확진자 수



유럽대륙은 Italy를 시작으로 확진자 수가 3월부터 증가하기 시작한다. Italy, Spain, Germany, France, Ukranina의 증가폭이 크고, Russia는 4월 중순부터 증가폭이 커지기 시작했다. 대부분의 국가들의 확진자 수 증가를 보여준다.