VADeaths

coop711

Tidy Data

깔끔한(tidy) 데이터를 만드는 방법에 대하여 알아본다. 사용되는 데이터는 R에 내장되어 있는 VADeaths 이다. 이 데이터의 구조는 5세 간격의 연령대를 행의 이름으로 하고, 장소(Rural, Urban)와 성별(Male, Female)의 조합을 열의 이름으로 갖는 행렬임을 알 수 있다.

```
VADeaths
```

```
##       Rural Male Rural Female Urban Male Urban Female
## 50-54       11.7          8.7       15.4          8.4
## 55-59       18.1         11.7       24.3         13.6
## 60-64       26.9         20.3       37.0         19.3
## 65-69       41.0         30.9       54.6         35.1
## 70-74       66.0         54.3       71.1         50.0
```

```
str(VADeaths)
```

```
##  num [1:5, 1:4] 11.7 18.1 26.9 41 66 8.7 11.7 20.3 30.9 54.3 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : chr [1:5] "50-54" "55-59" "60-64" "65-69" ...
##   ..$ : chr [1:4] "Rural Male" "Rural Female" "Urban Male" "Urban Female"
```

Base R 의 도구 활용

왜 이 데이터가 깔끔하지(tidy) 않은지 생각해 보자. 데이터를 어떻게 표현해야 깔끔한 것인지 최종 결과물과 비교한다.

c()는 행렬 구조로 표현한 VADeaths를 기다란 하나의 벡터로 나타낸다. 이렇게 만든 한 줄의 벡터를 Rates에 옮겨 넣는다.

보통 ordered()가 아닌 factor()를 사용하는 경우가 많은데 연령이라는 변수의 특성을 감안하면 단순히 명목형이 아니고 엄연히 순서가 있기 때문에ordered()를 사용하는 것이 적절하다.

```
Rates <- c(VADeaths)  ##
N <- length(Rates) ## `Rates`     `N`    .
Age <- ordered(rownames(VADeaths)) #            ,           .
Age <- rep(ordered(rownames(VADeaths)), #           . `length.out = `      .
           length.out = N)
Place <- gl(2, 10, N,
            labels = c("Rural", "Urban"))
        #  ,        10      `factor`
Gender <- gl(2, 5, N, #    5
             labels = c("Male", "Female"))
data.frame(Age, Place, Gender, Rates) #
```

```
##      Age Place Gender Rates
## 1  50-54 Rural   Male  11.7
## 2  55-59 Rural   Male  18.1
## 3  60-64 Rural   Male  26.9
## 4  65-69 Rural   Male  41.0
## 5  70-74 Rural   Male  66.0
## 6  50-54 Rural Female   8.7
## 7  55-59 Rural Female  11.7
## 8  60-64 Rural Female  20.3
## 9  65-69 Rural Female  30.9
## 10 70-74 Rural Female  54.3
## 11 50-54 Urban   Male  15.4
## 12 55-59 Urban   Male  24.3
## 13 60-64 Urban   Male  37.0
## 14 65-69 Urban   Male  54.6
## 15 70-74 Urban   Male  71.1
## 16 50-54 Urban Female   8.4
## 17 55-59 Urban Female  13.6
## 18 60-64 Urban Female  19.3
## 19 65-69 Urban Female  35.1
## 20 70-74 Urban Female  50.0
```

```
VADeaths_df <- data.frame(Age, Place, Gender, Rates) #         R
VADeaths_df #
```

```
##      Age Place Gender Rates
## 1  50-54 Rural   Male  11.7
## 2  55-59 Rural   Male  18.1
## 3  60-64 Rural   Male  26.9
## 4  65-69 Rural   Male  41.0
## 5  70-74 Rural   Male  66.0
## 6  50-54 Rural Female   8.7
## 7  55-59 Rural Female  11.7
## 8  60-64 Rural Female  20.3
## 9  65-69 Rural Female  30.9
## 10 70-74 Rural Female  54.3
## 11 50-54 Urban   Male  15.4
## 12 55-59 Urban   Male  24.3
## 13 60-64 Urban   Male  37.0
## 14 65-69 Urban   Male  54.6
## 15 70-74 Urban   Male  71.1
## 16 50-54 Urban Female   8.4
## 17 55-59 Urban Female  13.6
## 18 60-64 Urban Female  19.3
## 19 65-69 Urban Female  35.1
## 20 70-74 Urban Female  50.0
```

```
str(VADeaths_df) #
```

```
## 'data.frame':    20 obs. of  4 variables:
##  $ Age   : Ord.factor w/ 5 levels "50-54"<"55-59"<..: 1 2 3 4 5 1 2 3 4 5 ...
##  $ Place : Factor w/ 2 levels "Rural","Urban": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Gender: Factor w/ 2 levels "Male","Female": 1 1 1 1 1 2 2 2 2 2 ...
##  $ Rates : num  11.7 18.1 26.9 41 66 8.7 11.7 20.3 30.9 54.3 ...
```

VADeaths를 table구조로 변환하고, as.data.frame을 적용할 수도 있으나 Place와 Gender를 다시 분리하여야 함.

```
as.data.frame(as.table(VADeaths))
```

```
##     Var1         Var2 Freq
## 1  50-54   Rural Male 11.7
## 2  55-59   Rural Male 18.1
## 3  60-64   Rural Male 26.9
## 4  65-69   Rural Male 41.0
## 5  70-74   Rural Male 66.0
## 6  50-54 Rural Female  8.7
## 7  55-59 Rural Female 11.7
## 8  60-64 Rural Female 20.3
## 9  65-69 Rural Female 30.9
## 10 70-74 Rural Female 54.3
## 11 50-54   Urban Male 15.4
## 12 55-59   Urban Male 24.3
## 13 60-64   Urban Male 37.0
## 14 65-69   Urban Male 54.6
## 15 70-74   Urban Male 71.1
## 16 50-54 Urban Female  8.4
## 17 55-59 Urban Female 13.6
## 18 60-64 Urban Female 19.3
## 19 65-69 Urban Female 35.1
## 20 70-74 Urban Female 50.0
```

혹은 한 번에

```
as.data.frame.table(VADeaths)
```

```
##     Var1         Var2 Freq
## 1  50-54   Rural Male 11.7
## 2  55-59   Rural Male 18.1
## 3  60-64   Rural Male 26.9
## 4  65-69   Rural Male 41.0
## 5  70-74   Rural Male 66.0
## 6  50-54 Rural Female  8.7
## 7  55-59 Rural Female 11.7
## 8  60-64 Rural Female 20.3
## 9  65-69 Rural Female 30.9
## 10 70-74 Rural Female 54.3
## 11 50-54   Urban Male 15.4
## 12 55-59   Urban Male 24.3
## 13 60-64   Urban Male 37.0
## 14 65-69   Urban Male 54.6
## 15 70-74   Urban Male 71.1
## 16 50-54 Urban Female  8.4
## 17 55-59 Urban Female 13.6
## 18 60-64 Urban Female 19.3
## 19 65-69 Urban Female 35.1
## 20 70-74 Urban Female 50.0
```

```r
df <- as.data.frame.table(VADeaths)
df$Var2
```

```
##  [1] Rural Male   Rural Male   Rural Male   Rural Male   Rural Male
##  [6] Rural Female Rural Female Rural Female Rural Female Rural Female
## [11] Urban Male   Urban Male   Urban Male   Urban Male   Urban Male
## [16] Urban Female Urban Female Urban Female Urban Female Urban Female
## Levels: Rural Male Rural Female Urban Male Urban Female
```

```r
Var2_str <- strsplit(as.character(df$Var2), split = " ")
Place <- sapply(Var2_str, function(x) x[1])
Gender <- sapply(Var2_str, function(x) x[2])
data.frame(Age = df$Var1, Place, Gender, Rates = df$Freq)
```

```
##      Age Place Gender Rates
## 1  50-54 Rural   Male  11.7
## 2  55-59 Rural   Male  18.1
## 3  60-64 Rural   Male  26.9
## 4  65-69 Rural   Male  41.0
## 5  70-74 Rural   Male  66.0
## 6  50-54 Rural Female   8.7
## 7  55-59 Rural Female  11.7
## 8  60-64 Rural Female  20.3
## 9  65-69 Rural Female  30.9
## 10 70-74 Rural Female  54.3
## 11 50-54 Urban   Male  15.4
## 12 55-59 Urban   Male  24.3
## 13 60-64 Urban   Male  37.0
## 14 65-69 Urban   Male  54.6
## 15 70-74 Urban   Male  71.1
## 16 50-54 Urban Female   8.4
## 17 55-59 Urban Female  13.6
## 18 60-64 Urban Female  19.3
## 19 65-69 Urban Female  35.1
## 20 70-74 Urban Female  50.0
```

tidyverse를 이용한 방법

다음 코드를 차례대로 실행하면서 어떤 흐름이 잡히는 지 살펴보시오.

경고문의 `Conflicts ...`이하는 `R Base`에 있는 `filter()`나 `lag()`함수를 사용하려면 구체적으로 `stats::filter()`나 `stats::lag()`라고 하여야 한다는 것을 의미한다.

`gather()` 버전

```r
library(tidyverse) # `tidyverse`          .              .
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
```

```
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.2     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.4
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
VADeaths_tbl <- VADeaths %>% #       `tibble`     .
  as_tibble() %>% #      `tibble`    . `tbl_df()`          .
  mutate(Age = row.names(VADeaths)) %>% #
  gather(key = Place_Gender, # `Age`          `key, value`             .
         value = Rates,
         -Age) %>%
  separate(Place_Gender, c("Place", "Gender"), # `Place_Gender`  `Place`  `Gender`   .
           sep = " ") %>%
  mutate(Age = ordered(Age), # `Age`, `Place`, `Gender`
         Place = factor(Place),
         Gender = factor(Gender,  # `Gender`  `level = `                    .
                         levels = c("Male", "Female"))) # , `Female` 1, `Male` 2  .
VADeaths_tbl # `tibble`
```

```
## # A tibble: 20 x 4
##    Age   Place Gender Rates
##    <ord> <fct> <fct>  <dbl>
##  1 50-54 Rural Male    11.7
##  2 55-59 Rural Male    18.1
##  3 60-64 Rural Male    26.9
##  4 65-69 Rural Male    41
##  5 70-74 Rural Male    66
##  6 50-54 Rural Female   8.7
##  7 55-59 Rural Female  11.7
##  8 60-64 Rural Female  20.3
##  9 65-69 Rural Female  30.9
## 10 70-74 Rural Female  54.3
## 11 50-54 Urban Male    15.4
## 12 55-59 Urban Male    24.3
## 13 60-64 Urban Male    37
## 14 65-69 Urban Male    54.6
## 15 70-74 Urban Male    71.1
## 16 50-54 Urban Female   8.4
## 17 55-59 Urban Female  13.6
## 18 60-64 Urban Female  19.3
## 19 65-69 Urban Female  35.1
## 20 70-74 Urban Female  50
```

```
str(VADeaths_tbl) #    .
```

```
## tibble [20 x 4] (S3: tbl_df/tbl/data.frame)
##  $ Age   : Ord.factor w/ 5 levels "50-54"<"55-59"<..: 1 2 3 4 5 1 2 3 4 5 ...
##  $ Place : Factor w/ 2 levels "Rural","Urban": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Gender: Factor w/ 2 levels "Male","Female": 1 1 1 1 1 2 2 2 2 2 ...
##  $ Rates : num [1:20] 11.7 18.1 26.9 41 66 8.7 11.7 20.3 30.9 54.3 ...
```

이 과정을 순서대로 살펴보면, 먼저 행렬 구조를 tibble형식으로 변환하고,

```
VADeaths %>%
  as_tibble()
```

```
## # A tibble: 5 x 4
##   `Rural Male` `Rural Female` `Urban Male` `Urban Female`
##          <dbl>          <dbl>        <dbl>          <dbl>
## 1         11.7            8.7         15.4            8.4
## 2         18.1           11.7         24.3           13.6
## 3         26.9           20.3         37             19.3
## 4         41             30.9         54.6           35.1
## 5         66             54.3         71.1           50
```

Age 변수 생성

```
VADeaths %>%
  as_tibble() %>%
  mutate(Age = rownames(VADeaths))
```

```
## # A tibble: 5 x 5
##   `Rural Male` `Rural Female` `Urban Male` `Urban Female` Age
##          <dbl>          <dbl>        <dbl>          <dbl> <chr>
## 1         11.7            8.7         15.4            8.4 50-54
## 2         18.1           11.7         24.3           13.6 55-59
## 3         26.9           20.3         37             19.3 60-64
## 4         41             30.9         54.6           35.1 65-69
## 5         66             54.3         71.1           50   70-74
```

Age 를 제외한 변수를 key, value 쌍으로 정리하면서 새로운 변수명 부여, Age의 새로운 위치에 유의

```
VADeaths %>%
  as_tibble() %>%
  mutate(Age = rownames(VADeaths)) %>%
  gather(key = Place_Gender,
         value = Rates,
         -Age)
```

```
## # A tibble: 20 x 3
##    Age   Place_Gender Rates
##    <chr> <chr>        <dbl>
##  1 50-54 Rural Male    11.7
##  2 55-59 Rural Male    18.1
##  3 60-64 Rural Male    26.9
##  4 65-69 Rural Male    41
##  5 70-74 Rural Male    66
##  6 50-54 Rural Female   8.7
##  7 55-59 Rural Female  11.7
##  8 60-64 Rural Female  20.3
##  9 65-69 Rural Female  30.9
## 10 70-74 Rural Female  54.3
## 11 50-54 Urban Male    15.4
```

```
## 12 55-59 Urban Male    24.3
## 13 60-64 Urban Male    37
## 14 65-69 Urban Male    54.6
## 15 70-74 Urban Male    71.1
## 16 50-54 Urban Female   8.4
## 17 55-59 Urban Female  13.6
## 18 60-64 Urban Female  19.3
## 19 65-69 Urban Female  35.1
## 20 70-74 Urban Female  50
```

`Place_Gender`를 `Place`와 `Gender`로 분리. `sep =`의 사용 방법에 유의.

```
VADeaths %>%
  as_tibble() %>%
  mutate(Age = rownames(VADeaths)) %>%
  gather(key = Place_Gender,
         value = Rates,
         -Age) %>%
  separate(Place_Gender, c("Place", "Gender"),
           sep = " ")
```

```
## # A tibble: 20 x 4
##    Age   Place Gender Rates
##    <chr> <chr> <chr>  <dbl>
##  1 50-54 Rural Male    11.7
##  2 55-59 Rural Male    18.1
##  3 60-64 Rural Male    26.9
##  4 65-69 Rural Male    41
##  5 70-74 Rural Male    66
##  6 50-54 Rural Female   8.7
##  7 55-59 Rural Female  11.7
##  8 60-64 Rural Female  20.3
##  9 65-69 Rural Female  30.9
## 10 70-74 Rural Female  54.3
## 11 50-54 Urban Male    15.4
## 12 55-59 Urban Male    24.3
## 13 60-64 Urban Male    37
## 14 65-69 Urban Male    54.6
## 15 70-74 Urban Male    71.1
## 16 50-54 Urban Female   8.4
## 17 55-59 Urban Female  13.6
## 18 60-64 Urban Female  19.3
## 19 65-69 Urban Female  35.1
## 20 70-74 Urban Female  50
```

각 구성요소를 특성에 맞게 변환. `Gender`의 경우 `levels =`를 설정하는 이유에 대하여 생각해 볼 것.

```
VADeaths %>%
  as_tibble() %>%
  mutate(Age = rownames(VADeaths)) %>%
  gather(key = Place_Gender,
         value = Rates,
         -Age) %>%
```

```
  separate(Place_Gender, c("Place", "Gender"),
           sep = " ") %>%
  mutate(Age = ordered(Age),
         Place = factor(Place),
         Gender = factor(Gender,
                         levels = c("Male", "Female")))
```

```
## # A tibble: 20 x 4
##    Age   Place Gender Rates
##    <ord> <fct> <fct>  <dbl>
##  1 50-54 Rural Male    11.7
##  2 55-59 Rural Male    18.1
##  3 60-64 Rural Male    26.9
##  4 65-69 Rural Male    41
##  5 70-74 Rural Male    66
##  6 50-54 Rural Female   8.7
##  7 55-59 Rural Female  11.7
##  8 60-64 Rural Female  20.3
##  9 65-69 Rural Female  30.9
## 10 70-74 Rural Female  54.3
## 11 50-54 Urban Male    15.4
## 12 55-59 Urban Male    24.3
## 13 60-64 Urban Male    37
## 14 65-69 Urban Male    54.6
## 15 70-74 Urban Male    71.1
## 16 50-54 Urban Female   8.4
## 17 55-59 Urban Female  13.6
## 18 60-64 Urban Female  19.3
## 19 65-69 Urban Female  35.1
## 20 70-74 Urban Female  50
```

pivot_longer() 버전

```
VADeaths %>%
  as_tibble(rownames = "Age") %>%
  pivot_longer(
    cols = -Age,
    names_to = c("Place", "Gender"),
    names_sep = " ",
    values_to = "Rates"
  ) %>%
  mutate(
    Age = ordered(Age),
    Place = factor(Place),
    Gender = factor(Gender, levels = c("Male", "Female"))
  )
```

```
## # A tibble: 20 x 4
##    Age   Place Gender Rates
##    <ord> <fct> <fct>  <dbl>
##  1 50-54 Rural Male    11.7
```

```
##  2 50-54 Rural Female   8.7
##  3 50-54 Urban Male    15.4
##  4 50-54 Urban Female   8.4
##  5 55-59 Rural Male    18.1
##  6 55-59 Rural Female  11.7
##  7 55-59 Urban Male    24.3
##  8 55-59 Urban Female  13.6
##  9 60-64 Rural Male    26.9
## 10 60-64 Rural Female  20.3
## 11 60-64 Urban Male    37
## 12 60-64 Urban Female  19.3
## 13 65-69 Rural Male    41
## 14 65-69 Rural Female  30.9
## 15 65-69 Urban Male    54.6
## 16 65-69 Urban Female  35.1
## 17 70-74 Rural Male    66
## 18 70-74 Rural Female  54.3
## 19 70-74 Urban Male    71.1
## 20 70-74 Urban Female  50
```

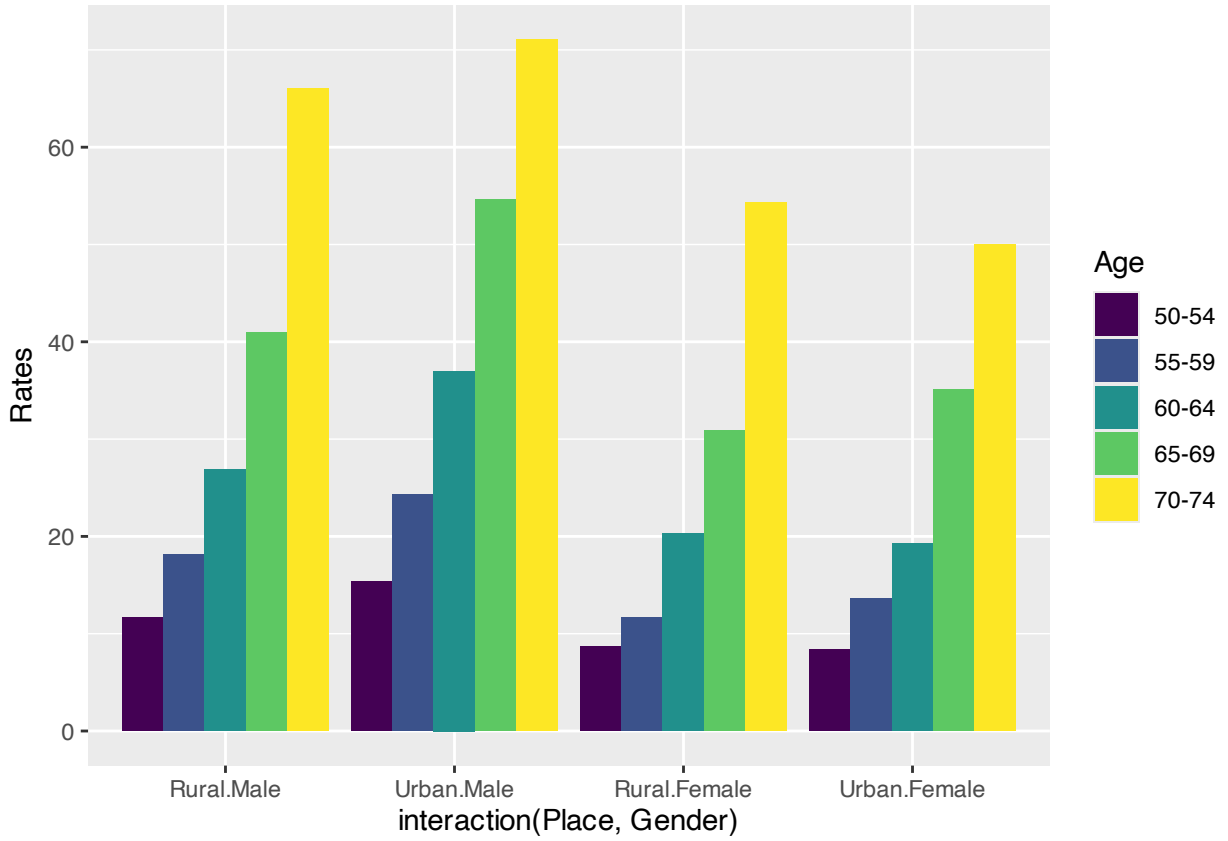
Plots

이 데이터 프레임을 시각적으로 ggplot()을 이용하여 표현하는 방법에 대하여 생각해 보자. 먼저 기본 함수들을 이용하여 생성한 VADeaths_df를 이용하여 그려보면, data = VADeaths_df로 설정하고, aes()의 x = 에는 장소(Place)와 성별(Gender)의 조합인 농촌남성(Rural.Male), 도시남성(Urban.Male), 농촌여성(Rural.Female), 도시여성(Urban.Female)을 interaction(Place, Gender)로 나타낸다. y =에는 사망률(Rates)을, 각 연령대(Age)를 막대의 색깔(fill =)로 구분한다.

막대그래프로 표현하기 위하여 geom_bar()를 사용하였는데, 가장 간단한 형식으로 나타내었다. 추가 정보나 보다 세부적인 표현은 다음에 다루기로 한다.

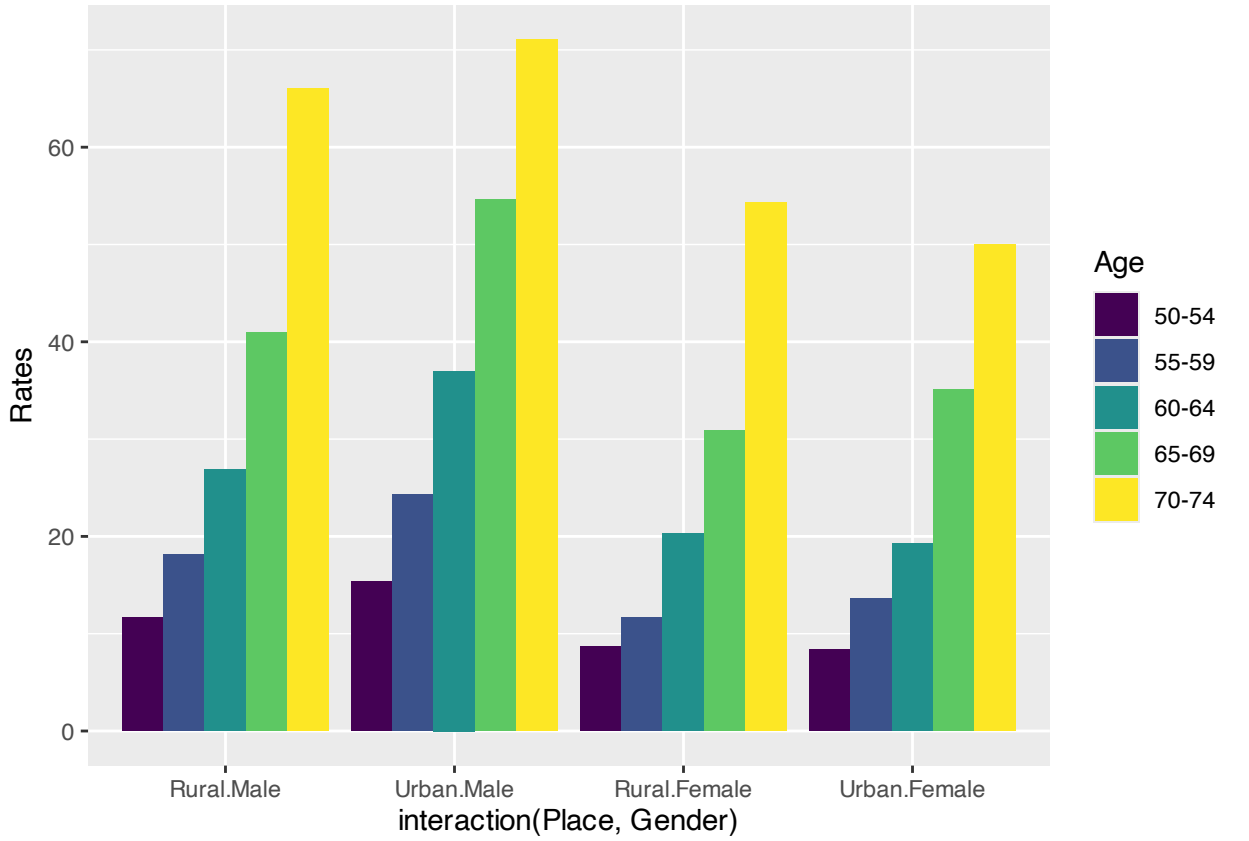
도시남성들의 사망률이 전 연령대에서 고르게 가장 높게 나타나는 반면, 도시 여성들은 대부분의 연령대에서 사망률이 낮게 나타나고 있다. 도시에 사는 남성들 …

```
ggplot(data = VADeaths_df,
              mapping = aes(x = interaction(Place, Gender),
                            y = Rates,
                            fill = Age)) +
geom_bar(stat = "identity",
         position = position_dodge())
```



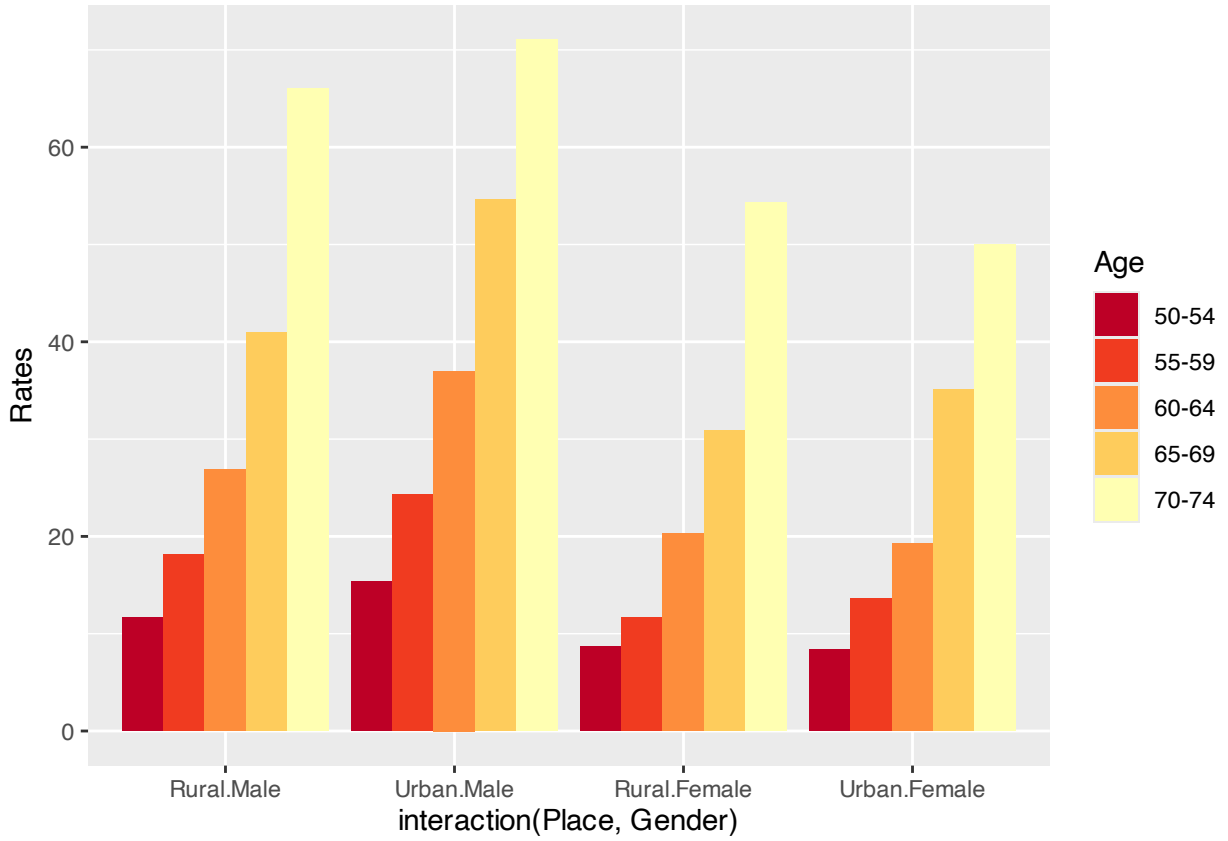
동일한 내용을 VADeaths_tbl로 그리면,

```
ggplot(data = VADeaths_tbl,
              mapping = aes(x = interaction(Place, Gender),
                            y = Rates,
                            fill = Age)) +
geom_bar(stat = "identity",
         position = position_dodge())
```



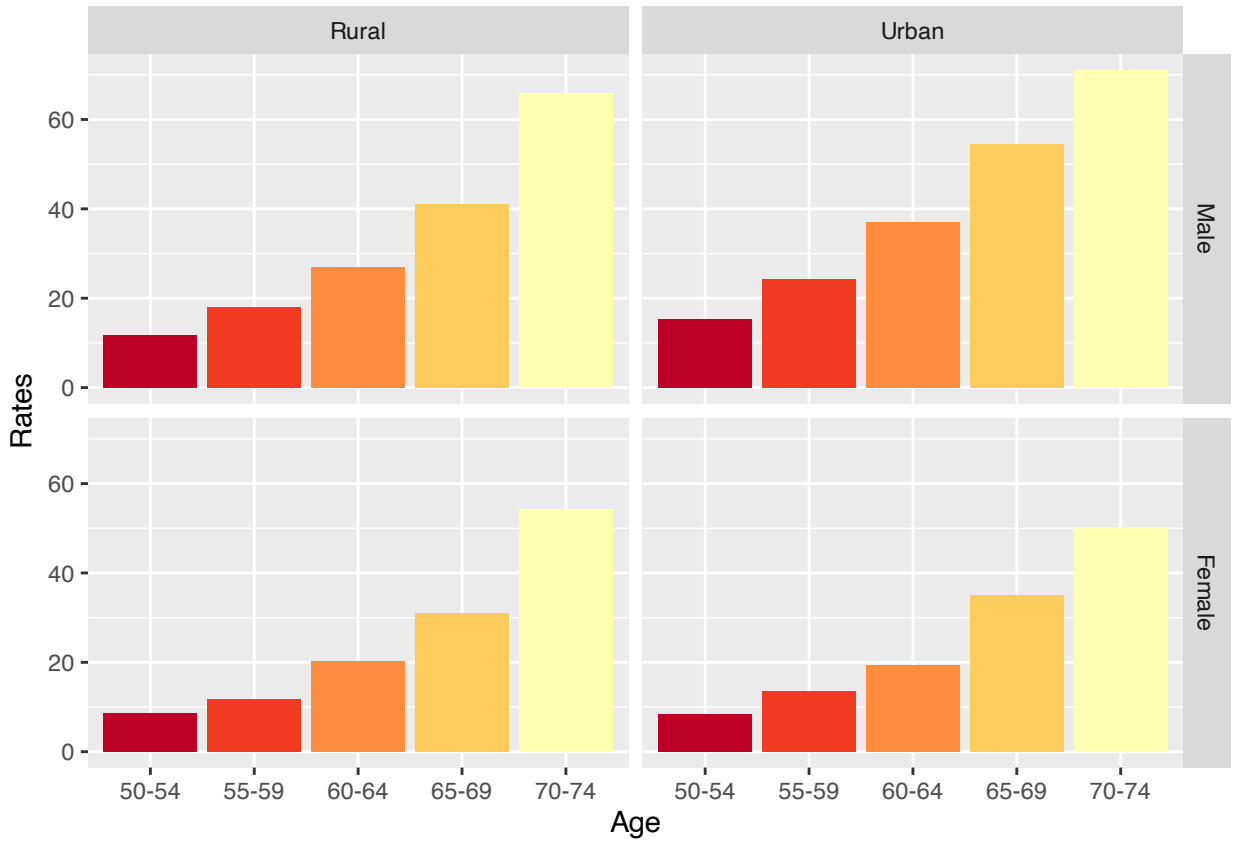
막대의 색깔을 Sequential 팔렛뜨 계열(`scale_fill_brewer` 도움말 참조)의 색깔 중 연령대의 변화에 맞도록 조정하면,

```
ggplot(data = VADeaths_tbl,
              mapping = aes(x = interaction(Place, Gender),
                            y = Rates,
                            fill = Age)) +
geom_bar(stat = "identity",
         position = position_dodge()) +
scale_fill_brewer(palette = "YlOrRd",
                  direction = -1)
```



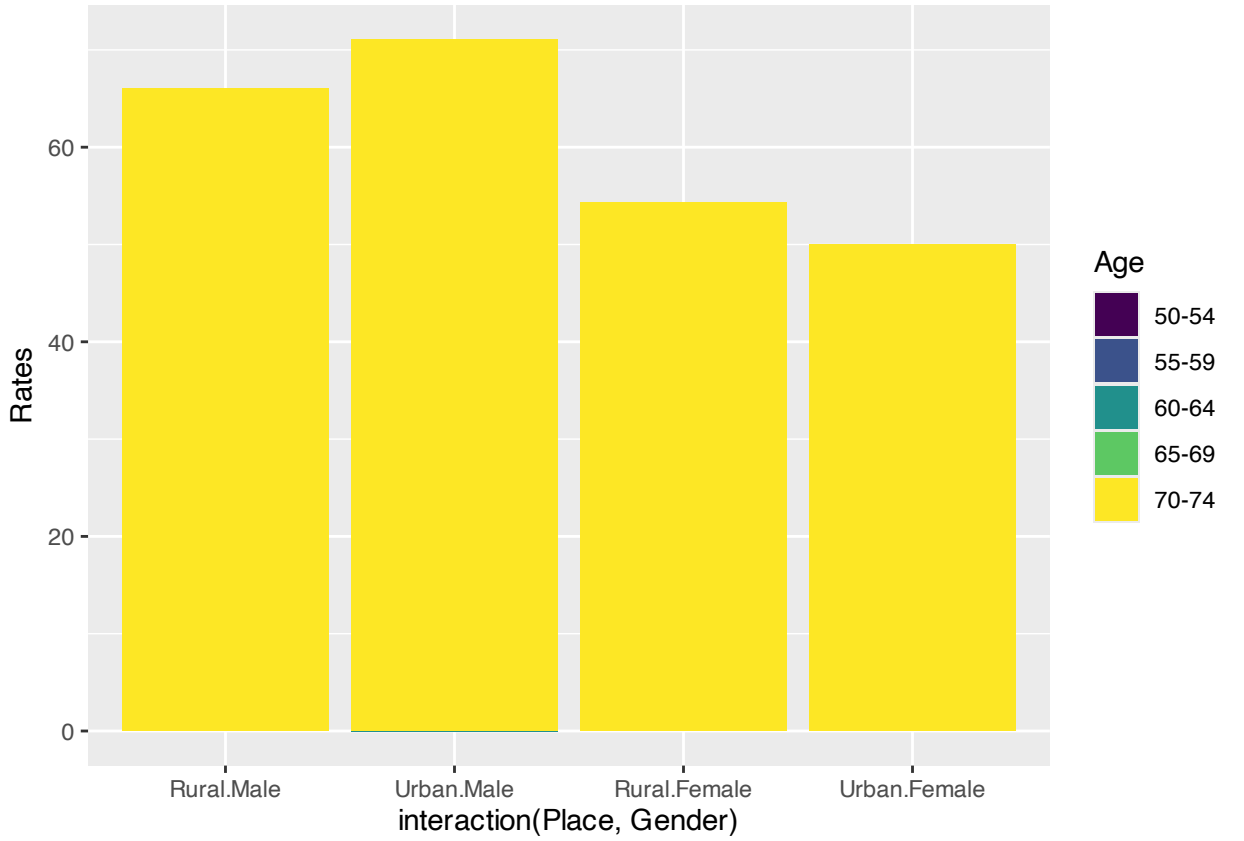
facet_grid를 이용하여 패널로 구분하여 나타내면,

```
ggplot(data = VADeaths_tbl,
              mapping = aes(x = Age,
                            y = Rates,
                            fill = Age)) +
geom_bar(stat = "identity",
         position = position_dodge()) +
scale_fill_brewer(guide = "none",
                  palette = "YlOrRd",
                  direction = -1) +
facet_grid(Gender ~ Place)
```

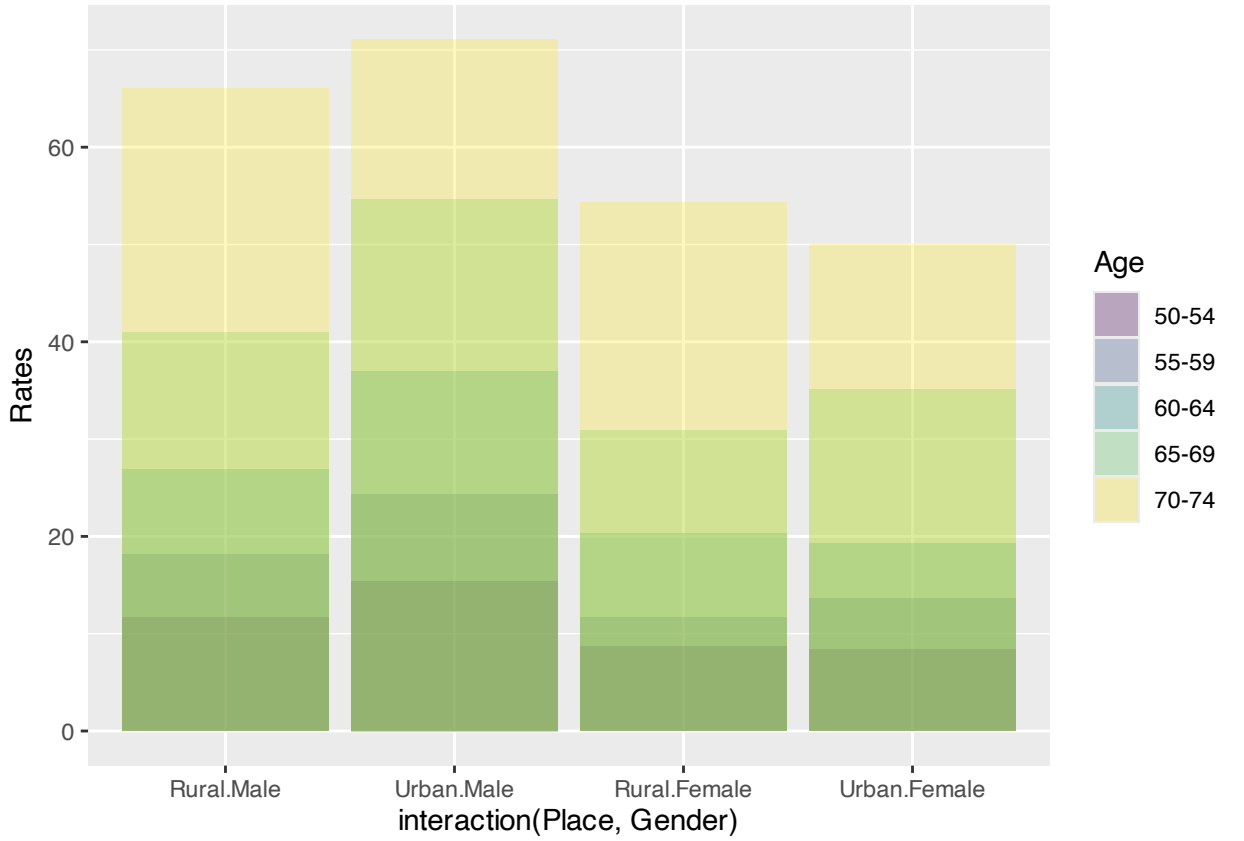



```
position = "identity"
```

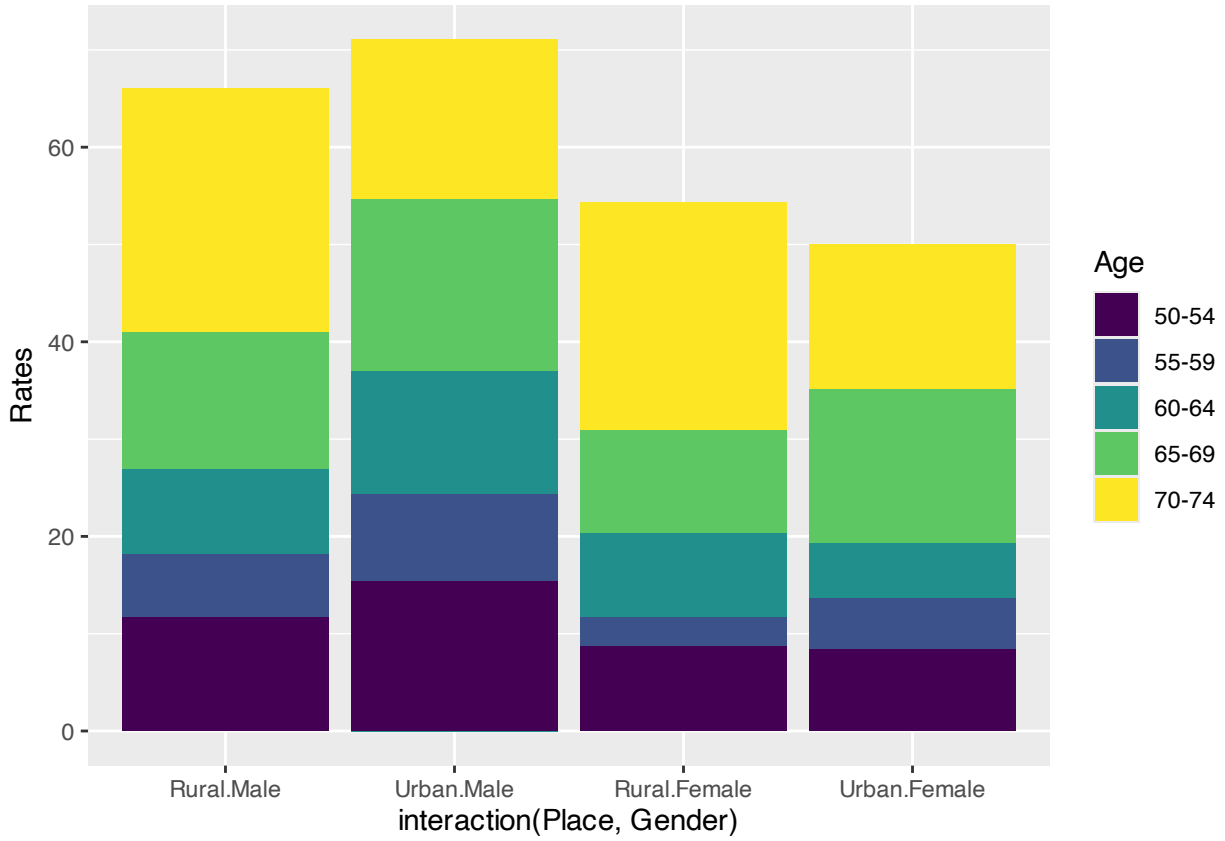
```
ggplot(data = VADeaths_df,
              mapping = aes(x = interaction(Place, Gender),
                            y = Rates,
                            fill = Age)) +
geom_bar(stat = "identity",
         position = "identity")
```




```
ggplot(data = VADeaths_df,
              mapping = aes(x = interaction(Place, Gender),
                            y = Rates,
                            fill = Age)) +
geom_bar(stat = "identity",
         position = "identity",
         alpha = 0.3)
```




```
ggplot(data = VADeaths_df[c(5:1, 10:6, 16:11, 20:16), ],
              mapping = aes(x = interaction(Place, Gender),
                            y = Rates,
                            fill = Age)) +
geom_bar(stat = "identity",
         position = "identity")
```



Comments

학습한 내용에 대하여 가급적 상세하게 기술합니다.