Final Project

박찬영

2024-11-22

```
library(tidyverse)
```

```
wd <- paste0(getwd(), "\\R\\2024\\FinalProject\\")
scf <- read_csv(paste0(wd, "scf_data.csv"))
scorecard <- read_csv(paste0(wd, "scorecard.csv"))
```

Topic 1 : College Scorecard data

`scorecard.csv`는 미국 교육부에서 제공하는 College Scorecard 의 교육기관별 데이터이다. College Scorecard 데이터는 각 대학의 분류, 학생지원, 학생들의 성과 등에 대한 정보를 제공한다. 이 데이터셋은 6484개의 대학과 3305개의 변수로 이루어져 있다.

주요 변수와 대략적인 설명은 아래와 같다. (자세한 설명은 https://collegescorecard.ed.gov의 Data Dictionary에서 확인할 수 있다.)

- `UNITID` : 대학의 고유 식별번호
- `INSTNM` : 대학의 이름
- `STABBR` : state 코드 (eg. AL: Alabama, CA: California, NY: New York)
- `PREDDEG` : 대학이 수여하는 주요 학위 (0:미분류, 1:자격증, 2:준학사, 3:학사, 4:대학원)
- `UGDS` : 학부 학생 수
- `ADM_RATE` : 입학률(입학을 허가받은 학생 수 / 지원한 학생 수)
- `MD_EARN_WNE_P6` : 입학 후 6년 뒤 급여의 중위값
- `PCT(25/75)_EARN_WNE_P6` : 입학 후 6년 뒤 급여의 (25/75) 백분위수 (eg. `PCT25_EARN_WNE_P6`)
- `DEBT_MDN` : 상환집단(상환을 시작한 학생) 중 대출 원금의 중위값
- `RPY_5YR_RT` : 상환집단 중 채무불이행 없이 5년간 대출 상환을 유지한 비율
- `FAMINC` : 입학 시 가구소득의 평균값

*** 본 분석에는 학부 학생 수가 800명 이상, 입학률이 0 초과, 수여하는 주요 학위가 학사학위인 대학만을 사용한다. 아래의 기준에 따라 데이터 전처리를 진행하고, data_cs로 저장하시오. [2점]**

- UNITID, INSTNM, STABBR은 character 변수로 저장한다.
- PREDDEG 변수는 factor형 변수로 저장한다.
- 이외의 주요 변수는 numeric 변수로 저장하되, 값 'PS'는 결측(NA)으로 처리한다.
- 학부 학생 수가 800명 이상, 입학률이 0 초과, 수여하는 주요 학위가 학사학위인 대학을 추출한다.

```
data_cs <- scorecard %>%
  select(
    UNITID, INSTNM, STABBR, PREDDEG, UGDS, ADM_RATE, MD_EARN_WNE_P6,
    PCT25_EARN_WNE_P6, PCT75_EARN_WNE_P6, DEBT_MDN, RPY_5YR_RT, FAMINC
  ) %>%
  mutate(UNITID = as.character(UNITID), PREDDEG = as.factor(PREDDEG)) %>%
  mutate(across(c(10, 11, 12), ~parse_number(., na = c("PS", "NA")))) %>%
  filter(UGDS >= 800, ADM_RATE > 0, PREDDEG == 3)

glimpse(data_cs)
```

```
## Rows: 1,237
## Columns: 12
## $ UNITID            <chr> "100654", "100663", "100706", "100724", "100751", "1~
## $ INSTNM            <chr> "Alabama A & M University", "University of Alabama a~
## $ STABBR            <chr> "AL", "AL", "AL", "AL", "AL", "AL", "AL", "AL", "AL"~
## $ PREDDEG           <fct> 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3~
## $ UGDS              <dbl> 5196, 12776, 6985, 3296, 31360, 3307, 25234, 968, 15~
## $ ADM_RATE          <dbl> 0.6840, 0.8668, 0.7810, 0.9660, 0.8006, 0.9223, 0.43~
## $ MD_EARN_WNE_P6    <dbl> 27851, 46572, 55610, 27453, 52233, 38216, 54629, 452~
## $ PCT25_EARN_WNE_P6 <dbl> 14130, 29464, 33999, 15271, 33353, 22285, 36022, 272~
## $ PCT75_EARN_WNE_P6 <dbl> 44543, 66959, 81066, 40900, 75004, 55734, 74667, 605~
## $ DEBT_MDN          <dbl> 16600, 15832, 13905, 17500, 17986, 13119, 17750, 160~
## $ RPY_5YR_RT        <dbl> 0.3311258, 0.5710578, 0.5883111, 0.2897054, 0.635813~
## $ FAMINC            <dbl> 32362.83, 51306.67, 61096.59, 31684.38, 91846.75, 41~
```

Question 1. (Admission rate) [15점]

각 대학의 입학률(경쟁률)에 대해 살펴보고자 한다. 아래 물음에 답하시오.

a. 입학률을 기준으로, 경쟁률이 가장 높은 상위 15개 대학을 출력하시오. [2점]

```r
data_cs %>%
  arrange(ADM_RATE) %>%
  head(15)
```

```
## # A tibble: 15 x 12
##    UNITID INSTNM  STABBR PREDDEG  UGDS ADM_RATE MD_EARN_WNE_P6 PCT25_EARN_WNE_P6
##    <chr>  <chr>   <chr>  <fct>   <dbl>    <dbl>          <dbl>             <dbl>
##  1 110404 Califo~ CA     3         982   0.0269         132140             68345
##  2 166027 Harvar~ MA     3        7973   0.0324          99572             54333
##  3 243744 Stanfo~ CA     3        7761   0.0368         102887             64169
##  4 190150 Columb~ NY     3        8902   0.0395          88535             48134
##  5 166683 Massac~ MA     3        4638   0.0396         131633             93246
##  6 130794 Yale U~ CT     3        6639   0.0457          81765             52868
##  7 217156 Brown ~ RI     3        7222   0.0506          79131             46303
##  8 144050 Univer~ IL     3        7511   0.0543          80870             48014
##  9 186131 Prince~ NJ     3        5527   0.057           87815             52741
## 10 198419 Duke U~ NC     3        6570   0.0635          85792             55723
## 11 182670 Dartmo~ NH     3        4412   0.0638          82541             46267
## 12 215062 Univer~ PA     3       10572   0.065           90555             55424
## 13 221999 Vander~ TN     3        7144   0.0667          73909             49418
## 14 167358 Northe~ MA     3       16172   0.068           78413             52640
## 15 216287 Swarth~ PA     3        1619   0.0693          56211             32936
## # i 4 more variables: PCT75_EARN_WNE_P6 <dbl>, DEBT_MDN <dbl>,
## #   RPY_5YR_RT <dbl>, FAMINC <dbl>
```

b. 대학이 가장 많이 소재한 상위 3개 주(state)를 출력하시오. [2점]

```r
data_cs %>%
  count(STABBR, sort = TRUE) %>%
  head(3)
```

```
## # A tibble: 3 x 2
##   STABBR     n
##   <chr>  <int>
## 1 NY       102
## 2 PA        91
## 3 CA        73
```

c. 입학률이 0.3 이하인 대학을 '명문(prestigious)대학'으로 정의한다. 명문대학이 가장 많이 소재한 상위 3개 주(state)를 출력하시오. [2점]

```r
data_cs %>%
  filter(ADM_RATE <= 0.3) %>%
  count(STABBR, sort = TRUE) %>%
  head(3)
```

```
## # A tibble: 3 x 2
##   STABBR     n
##   <chr>  <int>
## 1 CA        14
## 2 NY        12
## 3 MA        11
```

d. 캘리포니아에 위치한 명문대학의 이름을 모두 출력하시오. [2점]

```r
data_cs %>%
  filter(ADM_RATE <= 0.3, STABBR == "CA") %>%
  select(INSTNM)
```

```
## # A tibble: 14 x 1
##    INSTNM
##    <chr>
##  1 California Institute of Technology
##  2 University of California-Berkeley
##  3 University of California-Irvine
##  4 University of California-Los Angeles
##  5 University of California-San Diego
##  6 University of California-Santa Barbara
##  7 California Institute of the Arts
##  8 Claremont McKenna College
##  9 Harvey Mudd College
## 10 Pitzer College
## 11 Pomona College
## 12 Scripps College
## 13 University of Southern California
## 14 Stanford University
```

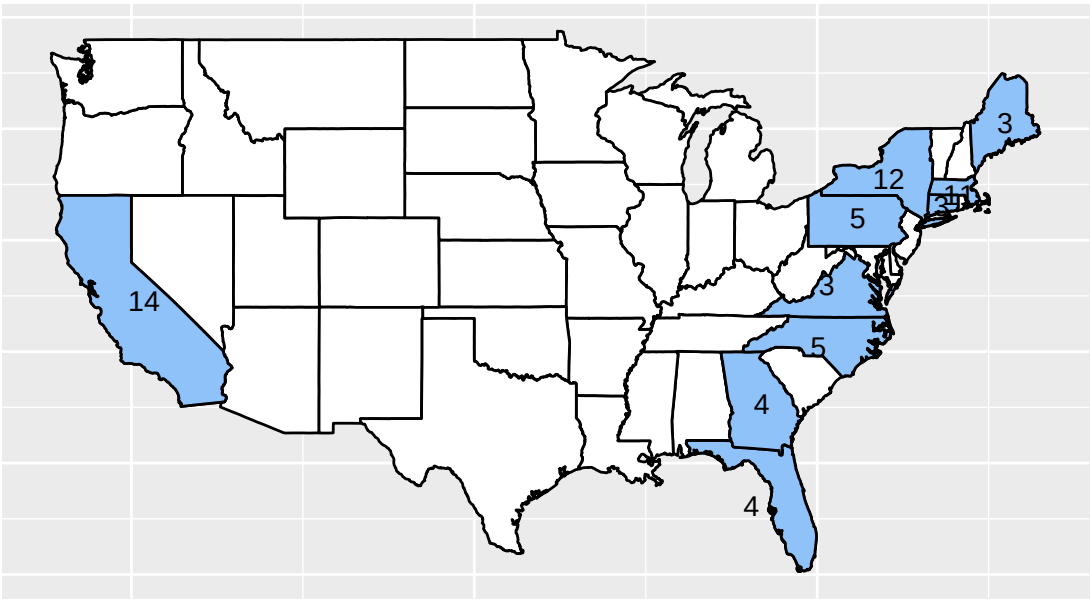
e. ggplot2 패키지의 `map_data('state')`는 미국의 각 주(state)에 대한 위도와 경도를 제공한다. 소재한 명문대학의 수가 3개 이상인 주에 대해서 명문대학의 수를 표시한 지도를 그리시오. (주어진 그래프는 일부에 대한 예시이다.) [7점]

- 주의 이름과 약어는 기본 패키지의 `state.name`과 `state.abb`을 참고한다. (Hint : Use `state_name = data.frame(region = state.name %>% tolower(), STABBR = state.abb)`)
- 주의 중심점은 위도와 경도 각각의 최대값과 최소값의 평균으로 간주한다.

- `map_data('state')`내 미국의 모든 주는 검은 선과 흰색 바탕으로 그린다. (Hint : Use `geom_polygon(aes(group=group))`)
- 해당 주를 지도 상의 색으로 표시하고 중심점에 명문대학의 수를 표기한다. (단, `label` 값을 직접 입력하는 경우 점수 미부여)
- 위도와 경도의 눈금 및 텍스트는 표시하지 않는다. (Hint : Use `theme(axis.ticks, axis.line, axis.text, axis.title)`)
- 그래프의 비율은 `coord_fixed(1.3)`을 이용한다.

```
namedoor <- data_cs %>%
  filter(ADM_RATE <= 0.3) %>%
  count(STABBR, sort = TRUE) %>%
  filter(n >= 3)

state_name <- data.frame(region = state.name %>% tolower(), STABBR = state.abb)

namedoor_map <- map_data("state") %>%
  inner_join(left_join(namedoor, state_name, by = "STABBR"), by = "region")

center <- namedoor_map %>%
  group_by(region) %>%
  summarise(
    long = mean(c(max(long), min(long))),
    lat = mean(c(max(lat), min(lat))),
    n = first(n)
  )



map_data("state") %>%
  ggplot() +
  geom_polygon(
    data = map_data("state"),
    aes(x = long, y = lat, group = group),
    fill = "white", color = "black"
  ) +
  geom_polygon(
    data = namedoor_map,
    aes(x = long, y = lat, group = group),
    fill = "#90c2fa", color = "black"
  ) +
  geom_text(data = center, aes(x = long, y = lat, label = n)) +
```

```
  theme(
    axis.ticks = element_blank(),
    axis.line = element_blank(),
    axis.title = element_blank(),
    axis.text = element_blank()
  ) +
  coord_fixed(1.3)
```

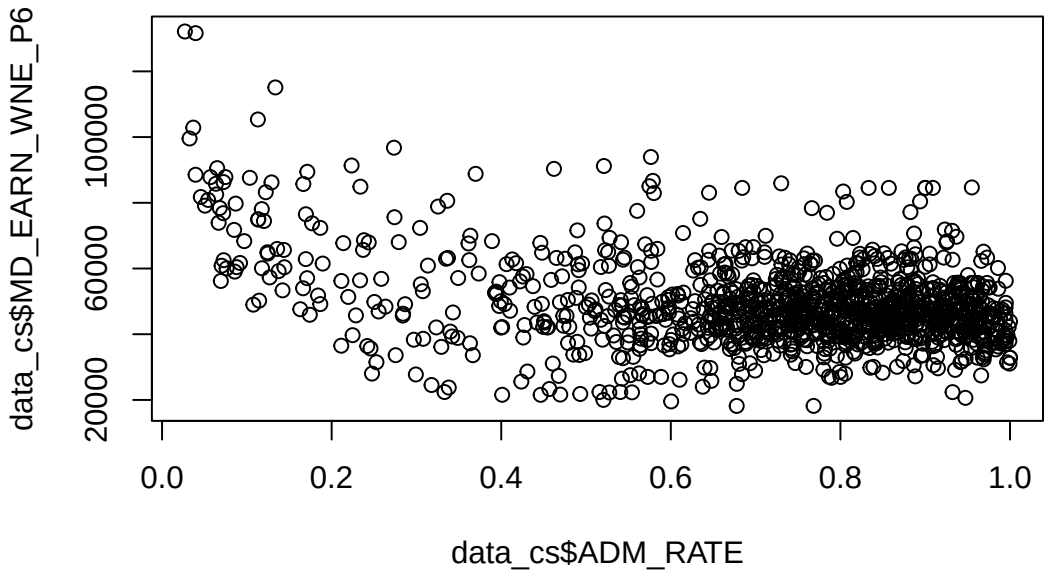


Question 2. (Admission rate and Earnings) [13점]

입학률과 학생들의 (입학 후 6년 뒤) 급여의 관계를 확인해보고자 한다. 아래 물음에 답하시오.

a. 입학률과 급여 중위값의 산점도를 그리시오. 두 변수 간에는 어떠한 관계가 있는가? [3점]

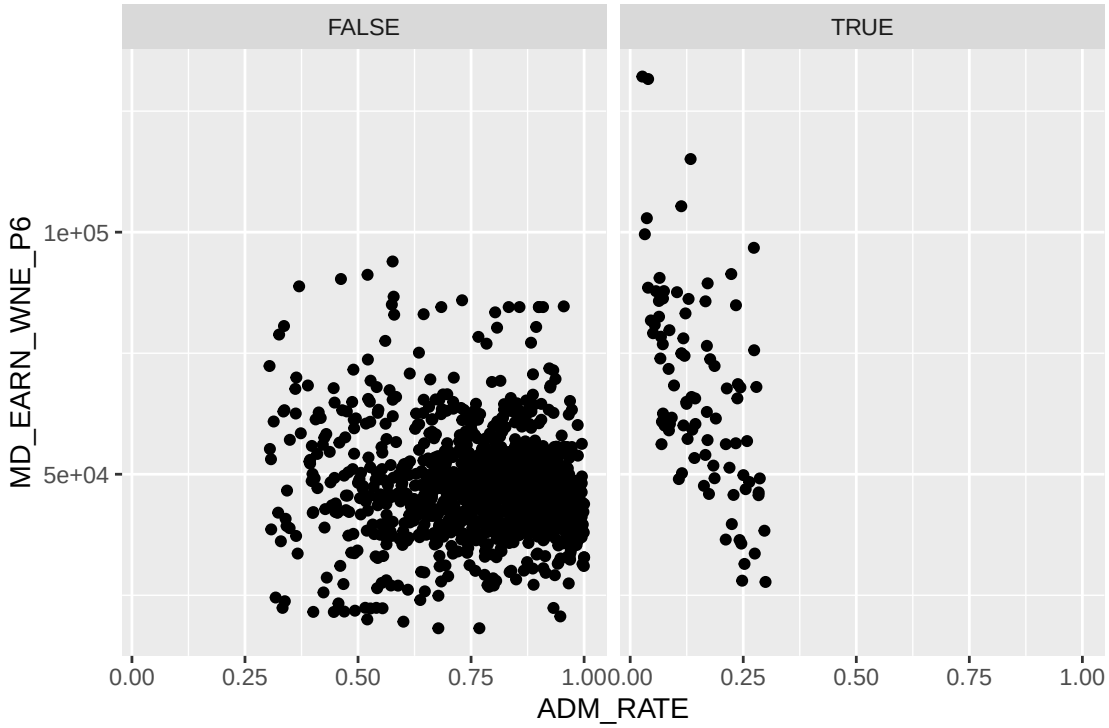
```
plot(data_cs$ADM_RATE, data_cs$MD_EARN_WNE_P6)
```



아주 약한 음의 상관관계를 보이는 것 같다.

b. 입학률과 급여 중위값의 산점도를 명문대학과 비명문대학으로 구분하여 그리시오. 두 산점도에 어떠한 차이가 나타나는가? [3점]

```
data_cs %>%
  mutate(namedoor = (ADM_RATE <= 0.3)) %>%
  ggplot(aes(ADM_RATE, MD_EARN_WNE_P6)) +
  geom_point() +
  facet_wrap(~namedoor)
```



명문대는 비명문대보다 음의 상관관계가 더 잘 보인다.

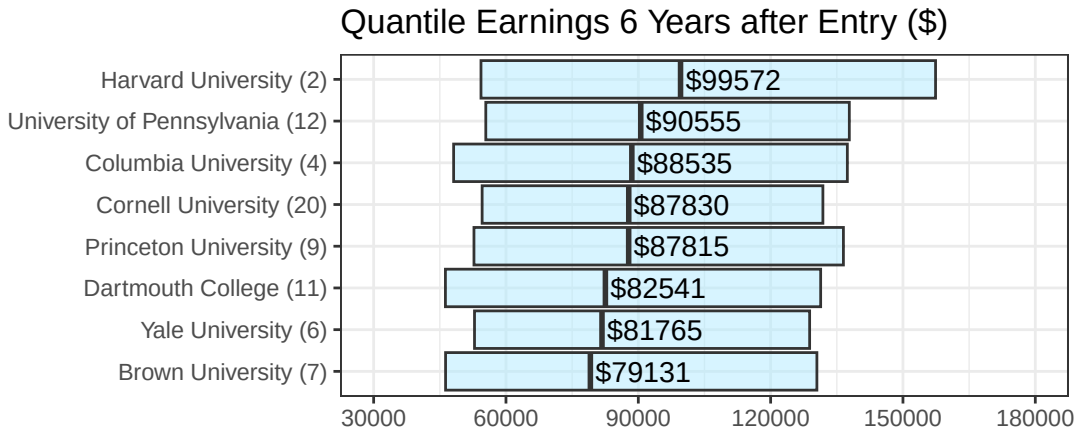
c. Ivy League 대학에 대해, 아래의 기준에 따라 급여의 분포를 비교하는 boxplot을 그리시오. (주어진 그래프는 일부에 대한 예시이다.) [7점]

- Ivy League는 8개 대학 Brown University, Columbia University, Cornell University, Dartmouth College, Harvard University, University of Pennsylvania, Princeton University, Yale University를 말한다.
- 급여의 사분위수를 box로 표시한다. (Hint : Use `geom_boxplot(mapping=aes(), stat='identity')`)
- 급여의 중위값 내림차순으로 그리며, 해당 대학의 급여 중위값($)을 해당 경계 옆에 표기한다.
- x축은 30000부터 180000까지 30000 단위로 표시하며, y축은 각 대학의 이름과 해당 대학의 ADM_RATE 순위를 [대학 이름 (순위)] 형태로 병기한다.
- 그래프의 제목은 'Quantile Earnings 6 Years after Entry ($)'이고, x축과 y축의 이름은 표시하지 않는다. 그래프의 색상 등은 가시성 있게 적절히 선택한다.
- 단, 대학의 이름은 위에 제시된 형태로 표기되어야 하며, 정규표현식을 활용하여 데이터 내의 표기형태를 찾을 수 있다.

```
ivyl <- c(
  "Brown University",
  "Columbia University in the City of New York",
  "Cornell University",
  "Dartmouth College",
  "Harvard University",
```

```
  "University of Pennsylvania",
  "Princeton University",
  "Yale University"
)



data_cs %>%
  mutate(rank = rank(ADM_RATE)) %>%
  filter(INSTNM %in% ivyl) %>%
  mutate(INSTNM = as.factor(str_replace_all(
    INSTNM,
    "Columbia University in the City of New York",
    "Columbia University"
  ))) %>%
  mutate(INSTNM = paste0(INSTNM, " (", rank, ")")) %>%
  mutate(INSTNM = fct_reorder(as.factor(INSTNM), MD_EARN_WNE_P6)) %>%
  ggplot(aes(x = INSTNM)) +
  geom_boxplot(aes(
    ymin = PCT25_EARN_WNE_P6,
    ymax = PCT75_EARN_WNE_P6,
    lower = PCT25_EARN_WNE_P6,
    middle = MD_EARN_WNE_P6,
    upper = PCT75_EARN_WNE_P6
  ), stat = "identity", fill = "#b1ebff", alpha = 0.5) +
  geom_text(
    aes(y = MD_EARN_WNE_P6 + 12000, label = paste0("$", MD_EARN_WNE_P6))
  ) +
  scale_y_continuous(
    limits = c(30000, 180000),
    breaks = seq(30000, 180000, 30000)
  ) +
  labs(
    title = "Quantile Earnings 6 Years after Entry ($)",
    y = "",
    x = ""
  ) +
  theme_bw() +
  theme(aspect.ratio = 0.47) +
```

```
  coord_flip()
```



Question 3. (Debt and Repayment) [10점]

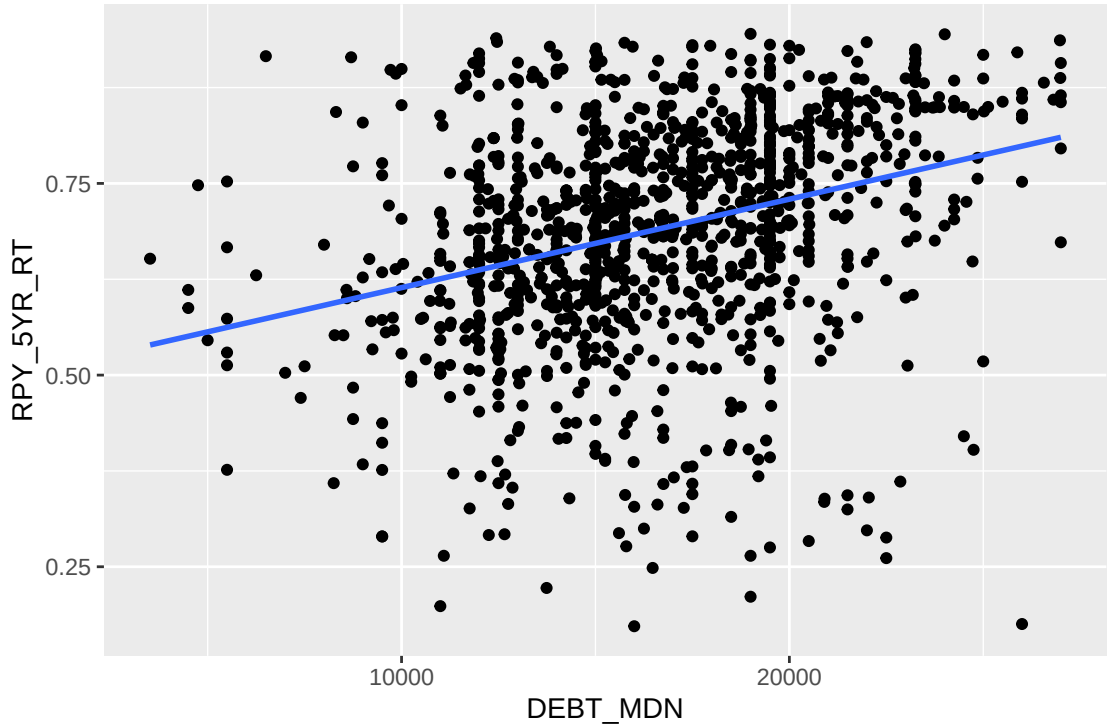
상환집단의 대출 원금(중위값)과 (5년 간) 상환 유지 비율의 관계를 확인하고자 한다. 물음에 답하시오.

a. 일반적으로 대출 원금이 낮다면 채무불이행 없이 대출상환을 꾸준히 유지하기 쉽다. 대출 원금과 상환 유지 비율의 산점도와 추세선을 그리고 결과를 해석하시오. [3점]

- 단, 추세선은 선형추세 `method = 'lm'`을 사용한다.

```
data_cs %>%
  ggplot(aes(DEBT_MDN, RPY_5YR_RT)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



어느정도 약한 양의 상관관계는 보이는 것 같다. 원금이 클수록 대출상환 유지비율이 크다는 것인데, 이는 예상과 다른 결과이다. 심슨의 역설일 수 있다.

b. 입학 시 평균 가구소득의 4분위수를 기준점으로 대학을 네 그룹으로 분류한 가구소득 그룹 FAMINC_GR을 정의한다. 대출원금과 상환 유지 비율의 산점도와 추세선을 가구소득 그룹에 따라 색으로 구분하여 그리시오. 어떤 관계가 나타나는가? [3점]

- 추세선은 선형추세 `method = 'lm'`을 사용한다.
- 그룹이 NA인 대학은 제외한다.

```
new_data_cs <- data_cs %>%
  filter(!is.na(FAMINC)) %>%
  mutate(FAMINC_GR = cut(
    FAMINC,
    breaks = quantile(FAMINC, c(0, 0.25, 0.5, 0.75, 1)) - c(5, 0, 0, 0, 0),
    labels = c("1", "2", "3", "4")
  ))
new_data_cs %>%
  ggplot(aes(DEBT_MDN, RPY_5YR_RT, color = FAMINC_GR)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

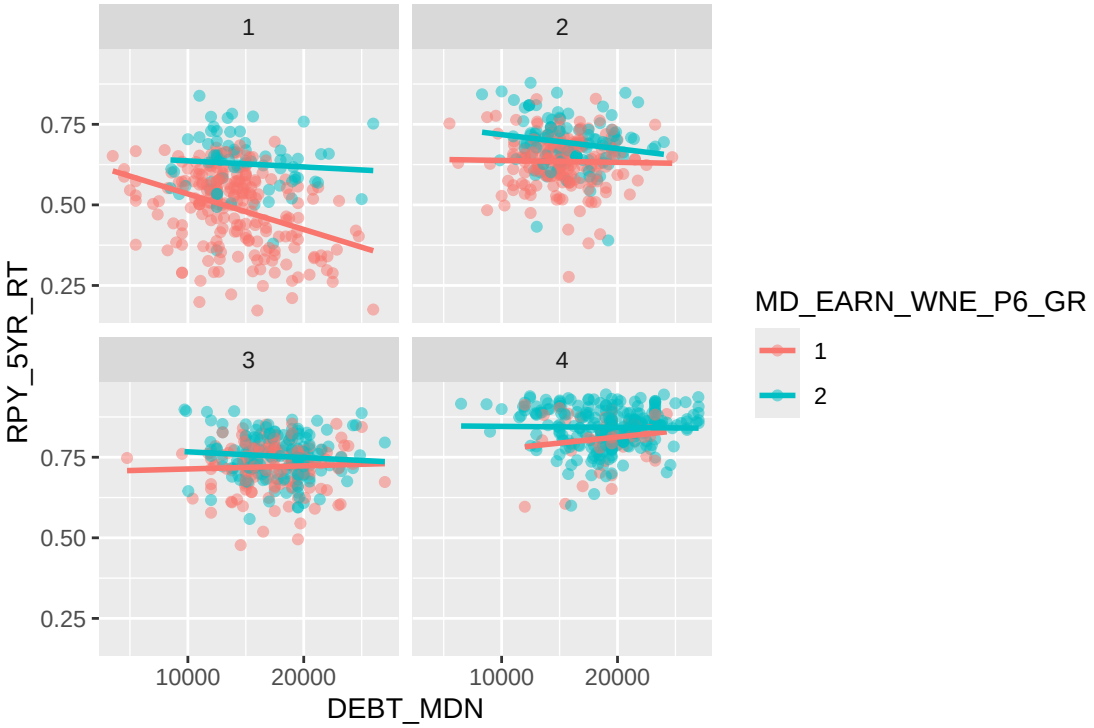
첫 예상대로 평균 가구소득이 중앙값보다 작은 두 그룹은 음의 상관관계를 보인다. 중앙값보다 큰 두 그룹은 상관관계를 거의 보이지 않는다. 심슨의 역설이 맞다. 또한 가구소득이 높을수록 대출상환율이 더 높다.

c. 중위 급여의 중위값을 기준점으로 대학을 두 그룹으로 분류한 급여 그룹 MD_EARN_WNE_P6_GR을 정의한다. facet_wrap()을 이용하여 각 가구소득 그룹에 대해, 대출 원금과 상환 유지 비율의 산점도와 추세선을 급여 그룹에 따라 색으로 구분하여 그리시오. 급여 그룹에 따른 분류로 어떠한 변화를 확인할 수 있는가? [4점]

- 추세선은 선형추세 `method = 'lm'`을 사용한다.
- 그룹이 NA인 대학은 제외한다.

```
new_data_cs %>%
  filter(!is.na(MD_EARN_WNE_P6)) %>%
  mutate(MD_EARN_WNE_P6_GR = as.factor(ifelse(
    MD_EARN_WNE_P6 > median(MD_EARN_WNE_P6),
    "2",
    "1"
  ))) %>%
  ggplot(aes(DEBT_MDN, RPY_5YR_RT, color = MD_EARN_WNE_P6_GR)) +
  geom_point(alpha = 0.5) +
  geom_smooth(method = "lm", se = FALSE) +
  facet_wrap(~FAMINC_GR)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



모든 가구소득 그룹에서 급여 중위값이 중앙값보다 크면 대출상환율이 더 크다. 급여가 높은 그룹은 가구소득이 1분위일때보다 2분위일때 더 상환을 못하는 추세를 보이고, 급여가 낮은 그룹은 1분위일때가 2분위일때보다 더 상환을 못하는 추세를 보인다. 또 가구소득 1분위이고 급여가 낮은 그룹외에는 음의 상관관계가 뚜렷하게 보이는 그룹은 많지 않다. 처음의 예측은 가구소득과 6년뒤 급여가 낮은 사람들에 대해서 적중한다. 나머지 그룹에 대해서는 확신할 수 없다. 가구소득이 클수록 6년뒤 급여가 큰 데이터가 많아지는 사실도 어느정도 확인할 수 있다.

Topic2 : 911 calls

Montgomery County in Pennsylvania의 911 calls (emergency calls) 에 대해 분석하고자 한다.

- `lat` : 위도(latitude)
- `lng` : 경도(longitude)
- `desc` : Description. Call에 대한 정보
- `zip` : zip 코드
- `title` : Call의 유형
- `timeStamp` : Call이 들어 온 시간
- `twp` : Township. County 내부 도시명
- `addr` : 상세 주소

Question 4. Dataset을 부르고, 다음 문제에 대한 코드를 작성하시오. (총 15점)

a. `e` 변수가 제거 가능한 근거를 제시하고 제거하시오. 또한 결측치가 있는 observation을 모두 제거하시오. (2점)

```
nine11 <- read_csv("F:\\data\\911.csv")
```

```
## Rows: 663522 Columns: 9
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): desc, title, twp, addr
## dbl  (4): lat, lng, zip, e
## dttm (1): timeStamp
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
summary(nine11$e)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       1       1       1       1       1       1
```

```
sum(is.na(nine11$e))
```

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

```
nine11_a <- nine11 %>% select(-e) %>% na.omit()
colSums(is.na(nine11_a))
```

```
##       lat       lng      desc       zip     title timeStamp       twp      addr
##         0         0         0         0         0         0         0         0
```

e 변수는 모든 값이 1인 변수이다. 이는 다른 관측치와 어떠한 관계나 영향 정보도주지 않아 삭제해도 된다. 그리고 NA값을 모두 제거하였다.

b. `title` 변수에는 emergency calls의 종류(`type`)와 상세 사유(`type_detail`)가 기록되어 있다. `title` 변수를 `:`을 기준으로 `type`, `type_detail`로 분리하시오. 또한 `type`의 종류별 observation 개수를 table로 제시하시오. (3점)

```
nine11_b <- nine11_a %>%
  separate_wider_delim(title, ": ", names = c("type", "type_detail"))

nine11_b %>% count(type)
```

```
## # A tibble: 3 x 2
##   type         n
##   <chr>    <int>
## 1 EMS     304785
## 2 Fire     88817
## 3 Traffic 189597
```

c. `type_detail`이 알파벳이 아닌 `" -"`로 끝나는 관찰값들이 있다. 시작과 끝이 알파벳이 되도록 `" -"`을 제거하시오. (3점)
ex) `"DISABLED VEHICLE -"` → `"DISABLED VEHICLE"`

```
nine11_b$type_detail <- str_replace_all(nine11_b$type_detail, " -$", "")
```

d. `desc` 변수는 `type`별로 다른 패턴을 가진다. `desc` 변수에는 `addr`, `twp`, `timeStamp` 변수의 정보가 들어 있고, `type`에 따라 `station`(emergency call을 받은 기관)에 대한 정보도 들어 있다. `desc` 변수로부터 `station` 변수를 만들어내시오. (7점)

- `station` 변수에는 "station" 이라는 string을 제외한, 알파벳과 숫자로만 이루어진 station의 코드가 들어있어야 한다. `desc` 변수에 `station` 정보가 없는 관찰값은 NA로 대체하시오.
- stringr cheatsheet을 참조하시오.

```
nine11_d <- nine11_b %>%
  mutate(station = str_extract(
    desc,
    "-?Station( |:)[a-zA-Z0-9]+;"
  )) %>%
  mutate(station = str_remove_all(station, "^-?Station(:| )|;"))

glimpse(nine11_d)
```

```
## Rows: 583,199
## Columns: 10
## $ lat         <dbl> 40.29788, 40.25806, 40.12118, 40.11615, 40.25347, 40.18211~
## $ lng         <dbl> -75.58129, -75.26468, -75.35198, -75.34351, -75.28324, -75~
## $ desc        <chr> "REINDEER CT & DEAD END;  NEW HANOVER; Station 332; 2015-1~
## $ zip         <dbl> 19525, 19446, 19401, 19401, 19446, 19044, 19426, 19438, 19~
## $ type        <chr> "EMS", "EMS", "Fire", "EMS", "EMS", "EMS", "EMS", "EMS", "~
## $ type_detail <chr> "BACK PAINS/INJURY", "DIABETIC EMERGENCY", "GAS-ODOR/LEAK"~
## $ timeStamp   <dttm> 2015-12-10 17:10:52, 2015-12-10 17:29:21, 2015-12-10 14:3~
## $ twp         <chr> "NEW HANOVER", "HATFIELD TOWNSHIP", "NORRISTOWN", "NORRIST~
## $ addr        <chr> "REINDEER CT & DEAD END", "BRIAR PATH & WHITEMARSH LN", "H~
## $ station     <chr> "332", "345", "STA27", "308A", "345", "352", "336", "344",~
```

Question 5. type 변수값마다 각각 type_detail 변수의 종류별 observation 수를 barplot으로 나타내고자 한다.

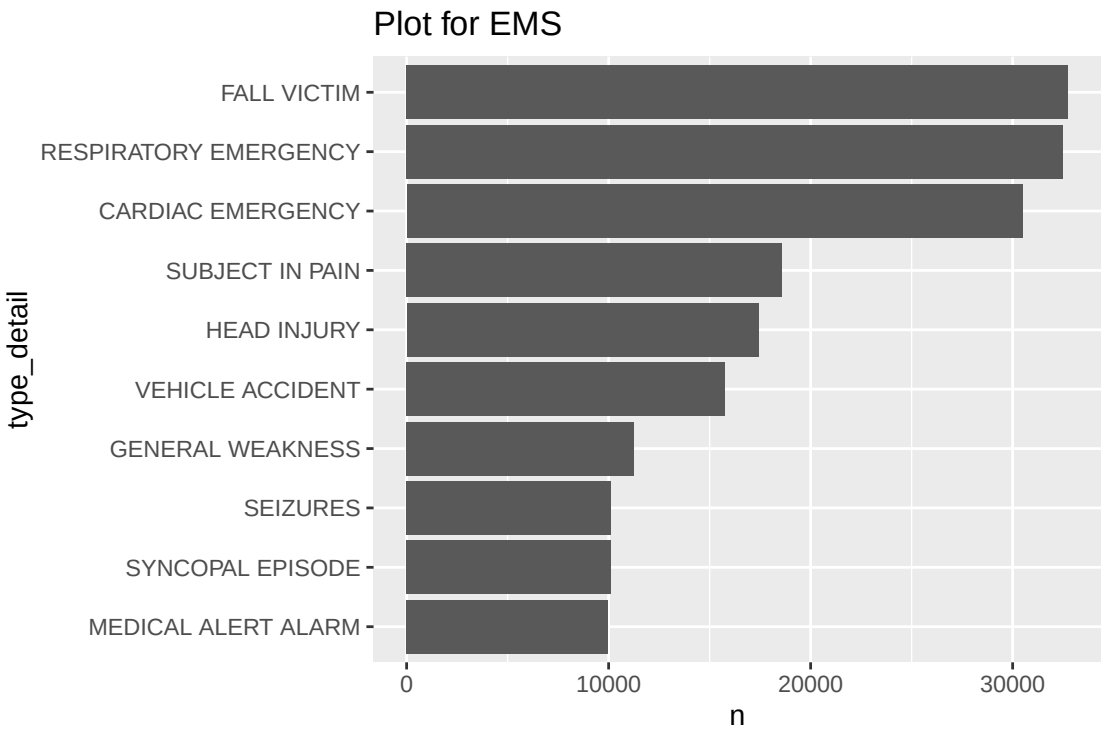
다음 조건에 맞는 코드를 작성하시오. (총 10점)

- plot들을 하나의 list에 저장하는 for문을 작성하시오. 이하 내용도 전부 이 for문 안에 작성한다. (4점)
- 각 plot마다, 관찰 수가 제일 많은 10개의 type_detail에 대해서만 출력하시오. (2점)
- bar의 길이를 내림차순으로 정렬하시오. (2점)

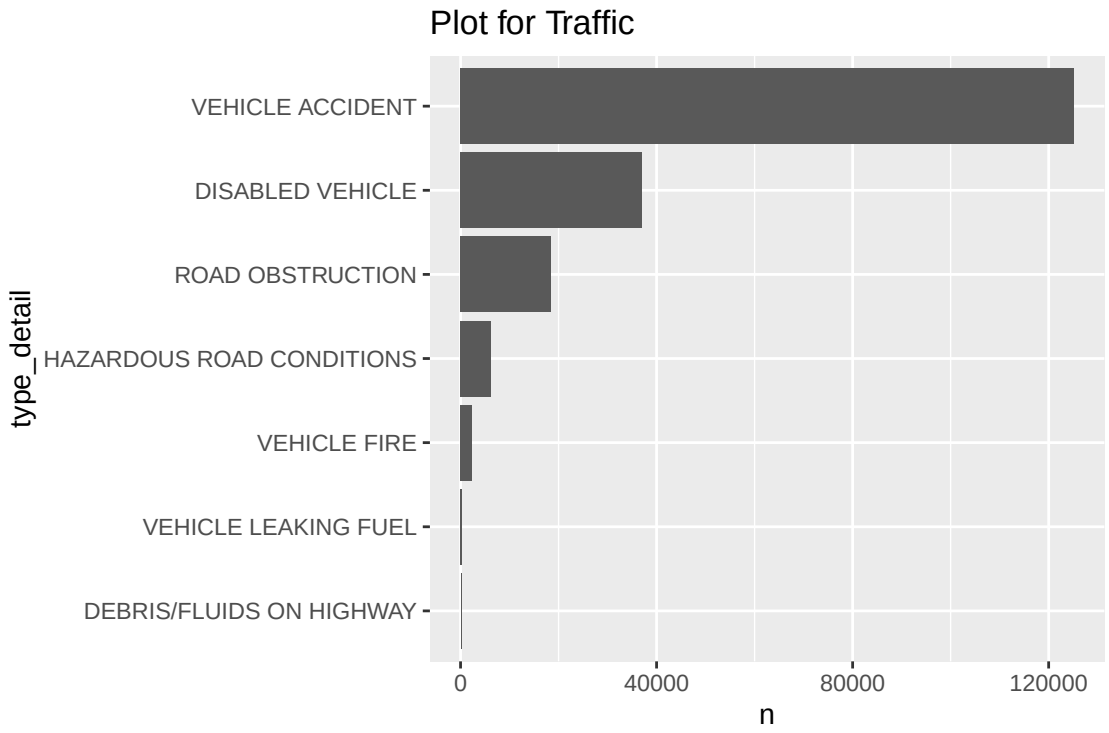
- 각 plot의 title은 “Plot for (type의 이름)”으로 한다. (2점)

- 예시 : type == “EMS”이면 title : “Plot for EMS”

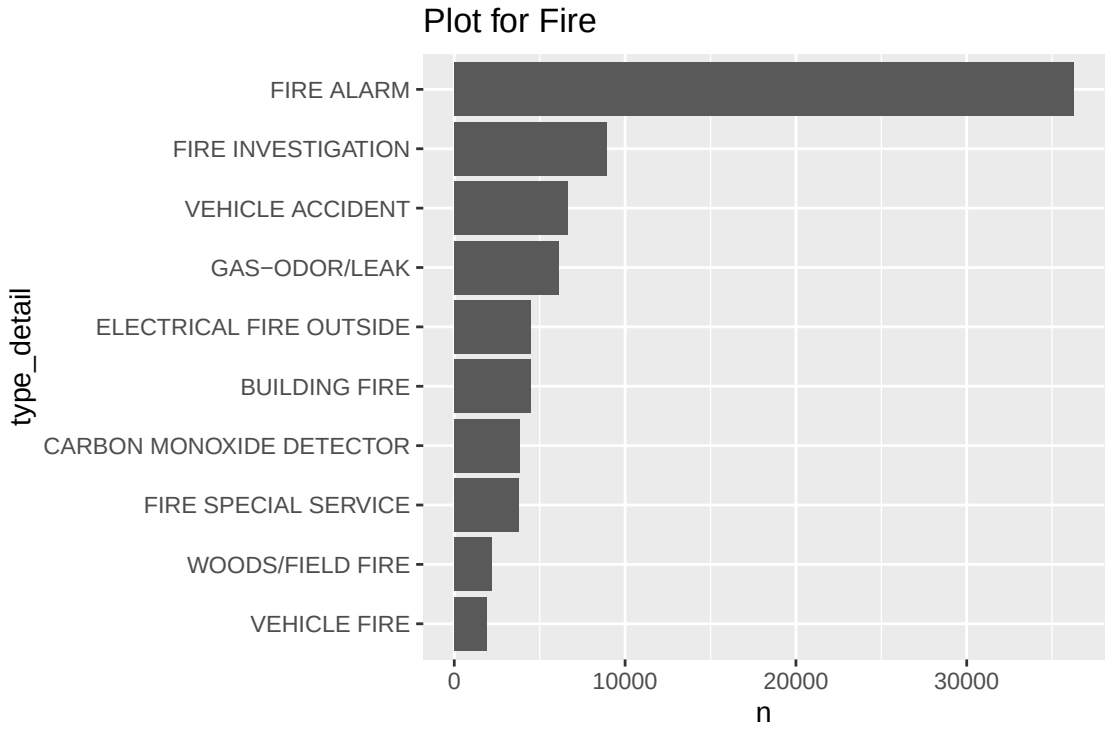
```
names <- c("EMS", "Traffic", "Fire")
i <- 1
plots <- vector("list", 3)
for (name in names){
  plots[[i]] <- nine11_d %>%
    filter(type == name) %>%
    count(type_detail, sort = TRUE) %>%
    head(10) %>%
    mutate(type_detail = fct_reorder(as.factor(type_detail), n)) %>%
    ggplot(aes(x = n, y = type_detail)) +
    geom_bar(stat = "identity") +
    labs(title = paste0("Plot for ", name))
  i <- i + 1
}

plots[[1]]
```



```
plots[[2]]
```




```
plots[[3]]
```



Question 6. Q4에서 얻은 dataset을 이용해 다음 물음에 답하시오. (총 15점)

a. `timeStamp` 변수를 이용해 `year`, `month`, `day`, `wday`, `hour` 변수를 만드시오. (3점) ex) `timeStamp` == "2015-12-10 17:10:52"인 경우 아래와 같이 생성. 요일은 한글 혹은 영어 `year` = 2015, `month` = 12, `day` = 10, `wday` = "목", `hour` = 17

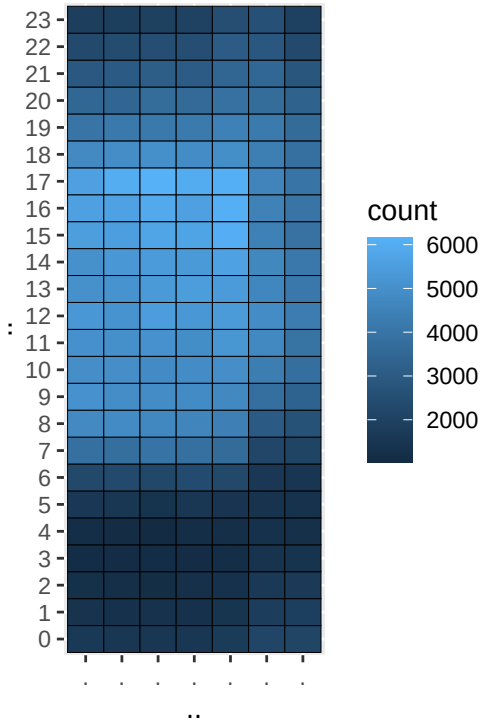
```
nine11_fin <- nine11_d %>%
  mutate(
    year = year(timeStamp),
    month = month(timeStamp),
    day = day(timeStamp),
    wday = factor(
      str_remove(weekdays(timeStamp), "요일"),
      c("월", "화", "수", "목", "금", "토", "일")
    ),
    hour = hour(timeStamp)
  )
glimpse(nine11_fin)
```

```
## Rows: 583,199
## Columns: 15
## $ lat         <dbl> 40.29788, 40.25806, 40.12118, 40.11615, 40.25347, 40.18211~
## $ lng         <dbl> -75.58129, -75.26468, -75.35198, -75.34351, -75.28324, -75~
## $ desc        <chr> "REINDEER CT & DEAD END;  NEW HANOVER; Station 332; 2015-1~
## $ zip         <dbl> 19525, 19446, 19401, 19401, 19446, 19044, 19426, 19438, 19~
## $ type        <chr> "EMS", "EMS", "Fire", "EMS", "EMS", "EMS", "EMS", "EMS", "~
## $ type_detail <chr> "BACK PAINS/INJURY", "DIABETIC EMERGENCY", "GAS-ODOR/LEAK"~
## $ timeStamp   <dttm> 2015-12-10 17:10:52, 2015-12-10 17:29:21, 2015-12-10 14:3~
## $ twp         <chr> "NEW HANOVER", "HATFIELD TOWNSHIP", "NORRISTOWN", "NORRIST~
## $ addr        <chr> "REINDEER CT & DEAD END", "BRIAR PATH & WHITEMARSH LN", "H~
## $ station     <chr> "332", "345", "STA27", "308A", "345", "352", "336", "344",~
## $ year        <dbl> 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015~
## $ month       <dbl> 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12~
## $ day         <int> 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10~
## $ wday        <fct> 목, 목, 목, 목, 목, 목, 목, 목, 목, 목, 목, 목, 목, 목, 목~
## $ hour        <int> 17, 17, 14, 16, 15, 16, 16, 16, 17, 16, 17, 17, 17, 17, 17~
```

b. 요일/시간의 조합별 신고수를 하나의 heatmap으로 나타내시오. 신고횟수가 많은 요일, 시간대를 자유롭게 제시하시오. (3점)

```
ggplot(nine11_fin, aes(
  x = wday, y = factor(hour)
```

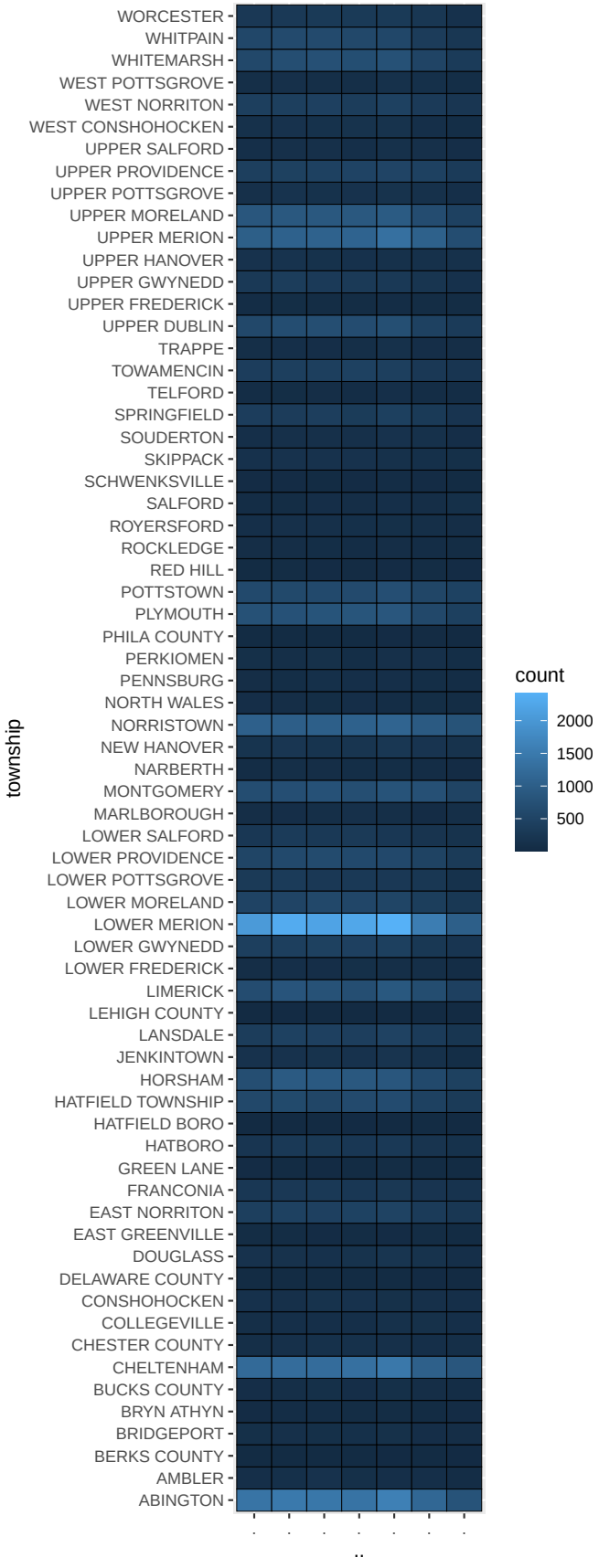
```
)) +
  geom_bin2d(binwith = c(1, 1), color = "black") +
  labs(x = "요일", y = "시간") +
  theme(aspect.ratio = 2.5)
```



금요일 17시! 확실하게 많다. 전체적으로 주말보다 평일 특히 금요일이 많고, 전체적으로 8시~17시가 신고가많다.

c. VEHICLE ACCIDENT은 type_detail의 종류 중 하나로, type==EMS,Traffic에서 등장한다. 요일/township의 조합별 VEHICLE ACCIDENT 발생 건수를 하나의 heatmap을 통해 나타내시오. VEHICLE ACCIDENT이 많이 발생하는 지역, 요일을 자유롭게 제시하시오. (4점)

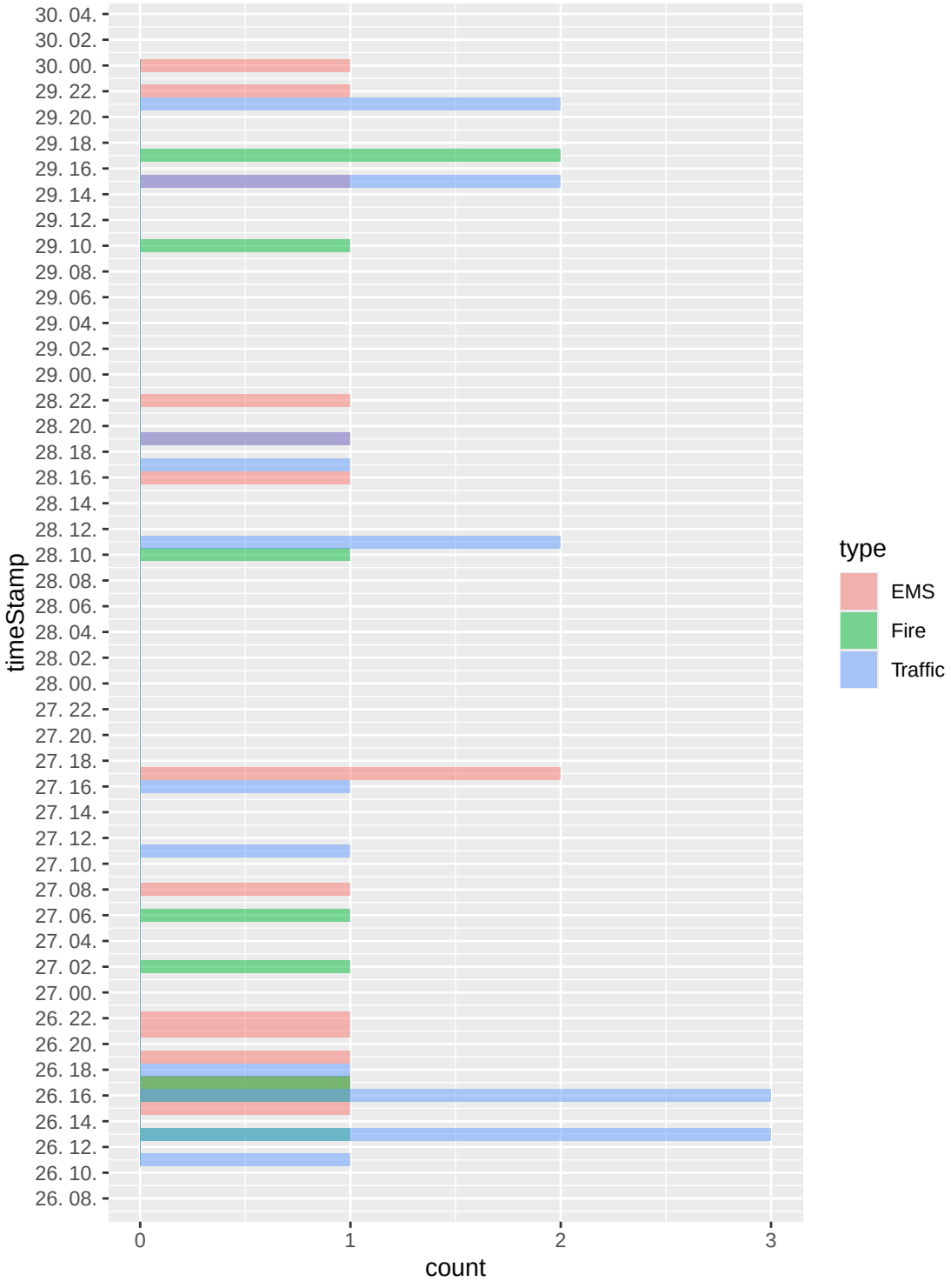
```
nine11_fin %>%
  filter(type_detail == "VEHICLE ACCIDENT") %>%
  ggplot(aes(x = wday, y = twp)) +
  geom_bin2d(color = "black") +
  labs(x = "요일", y = "township") +
  theme(aspect.ratio = 6)
```


township
WORCESTER
WHITPAIN
WHITEMARSH
WEST POTTSGROVE
WEST NORRITON
WEST CONSHOHOCKEN
UPPER SALFORD
UPPER PROVIDENCE
UPPER POTTSGROVE
UPPER MORELAND
UPPER MERION
UPPER HANOVER
UPPER GWYNEDD
UPPER FREDERICK
UPPER DUBLIN
TRAPPE
TOWAMENCIN
TELFORD
SPRINGFIELD
SOUDERTON
SKIPPACK
SCHWENKSVILLE
SALFORD
ROYERSFORD
ROCKLEDGE
RED HILL
POTTSTOWN
PLYMOUTH
PHILA COUNTY
PERKIOMEN
PENNSBURG
NORTH WALES
NORRISTOWN
NEW HANOVER
NARBERTH
MONTGOMERY
MARLBOROUGH
LOWER SALFORD
LOWER PROVIDENCE
LOWER POTTSGROVE
LOWER MORELAND
LOWER MERION
LOWER GWYNEDD
LOWER FREDERICK
LIMERICK
LEHIGH COUNTY
LANSDALE
JENKINTOWN
HORSHAM
HATFIELD TOWNSHIP
HATFIELD BORO
HATBORO
GREEN LANE
FRANCONIA
EAST NORRITON
EAST GREENVILLE
DOUGLASS
DELAWARE COUNTY
CONSHOHOCKEN
COLLEGEVILLE
CHESTER COUNTY
CHELTENHAM
BUCKS COUNTY
BRYN ATHYN
BRIDGEPORT
BERKS COUNTY
AMBLER
ABINGTON
count
2000
1500
1000
500

LOWER MERION에서 평일중 특히 금요일에 교통사고가 많이 일어난다.

d. 다음 기사는 2016년 Montgomery의 911 Service 손상과 관련된 기사이다. Link Cable cut으로 인해 2016-10-27 14:00:00 ~ 2016-10-28 05:18:00 동안 landline call에 문제가 생겼다. 위의 사건으로 인해, 해당 기간동안 신고 횟수의 변화가 있었는지 확인하고자 한다. 해당 기간 전후 3일간 신고량의 그래프를 type별로 그리고, cable cut의 영향이 있었는지 판단하시오. 한 plot 안에 type별 신고량이 색깔별로 나타나야 하며, cable cut 발생/복구 시간이 그래프에 나타나야 한다. (5점)

```
nine11_fin %>%
  filter(year == 2016, month == 10, day %in% 26:29, twp == "MONTGOMERY") %>%
  ggplot(aes(x = timeStamp)) +
  geom_histogram(
    binwidth = 3600,
    aes(fill = type),
    position = "identity",
    alpha = 0.5
  ) +
  scale_x_datetime(date_labels = "%d일 %H시", date_breaks = "2 hours") +
  coord_flip()
```



10/27 14시부터 10/28 5시까지 3건의 신고만이 들어왔다. EMS 2건 Traffic 1건이다. Fire 신고는 아예 들어오지 않았다. 15시간의 시간 범위를 잡았을 때 신고수가 3건인 것은 몽고메리 지역의 신고수가 적어서 충분히 일어날 수 있다. cable cut의 영향이 있었는지 알 수가 없다. cable cut 발생/복구 시간 또한 복구이후 신고가 한동안 들어오지 않았기에 확인할 수 없다.

Topic 3 : Survey of Consumer Finances

2022년에 미국에서 실시한 소비자 금융 조사에 대해 분석하고자 한다. 주요 변수에 대한 설명은 다음과 같다.

Variable

Attributes

yy1

Case ID

age

Age of reference person

income

Total amount of income of household

checking

Total value of checking accounts held by household

saving

Total value of savings accounts held by household

stocks

Total value of directly held stocks held by household

bond

Total value of directly held bonds held by household

vehic

Total value of all vehicles held by household

houses

Total value of primary residence of household

nnresre

Total value of net equity in nonresidential real estate held by household

late

Household had any late debt payments in last year

late60

Household had any debt payments more than 60 days past due in last year

mrthel

Total value of debt secured by the primary residence held by household

install

Total value of installment loans held by household

edn_inst

Total value of education loans held by household

veh_inst

Total value of vehicle loans held by household

housecl

Home-ownership category of household

own

have an owned vehicle

noccbal

Household does not carry a balance on credit cards

credit_score

Credit score of household (0~1000)

Question 7. 주어진 조건에 따라 자료를 정리하시오. [4점]

- 금융자산의 총 가치를 계산하여 `fin` 변수에 저장하시오.
- 비금융자산의 총 가치를 계산하여 `nfin` 변수에 저장하시오.
- 자산 총 가치를 계산하여 `asset` 변수에 저장하시오.
- 순자산가치를 계산하여 `networth` 변수에 저장하시오.
- 연령대를 구분하여 `age.grp` 변수에 저장하시오.
- `networth`, `fin`, `nfin`이 `income`의 몇 배인지 각각 계산하고 `*_multiple` 변수에 저장하시오.

```
new_scf <- scf %>%
  mutate(
    fin = checking + saving + stocks + bond,
    nfin = vehic + houses + nnresre,
    asset = fin + nfin,
    debt = install + edn_inst + veh_inst + mrthel,
    networth = asset - debt,
    networth_multiple = networth / income,
    fin_multiple = fin / income,
    nfin_multiple = nfin / income,
```

```
    age.grp = cut(
      age,
      c(0, 30, 35, 40, 45, 50, 55, 60, 67, 100),
      right = FALSE,
      labels = c(
        "zero", "one", "two", "three",
        "four", "six", "seven", "eigth", "ten"
      )
    )
  )
```

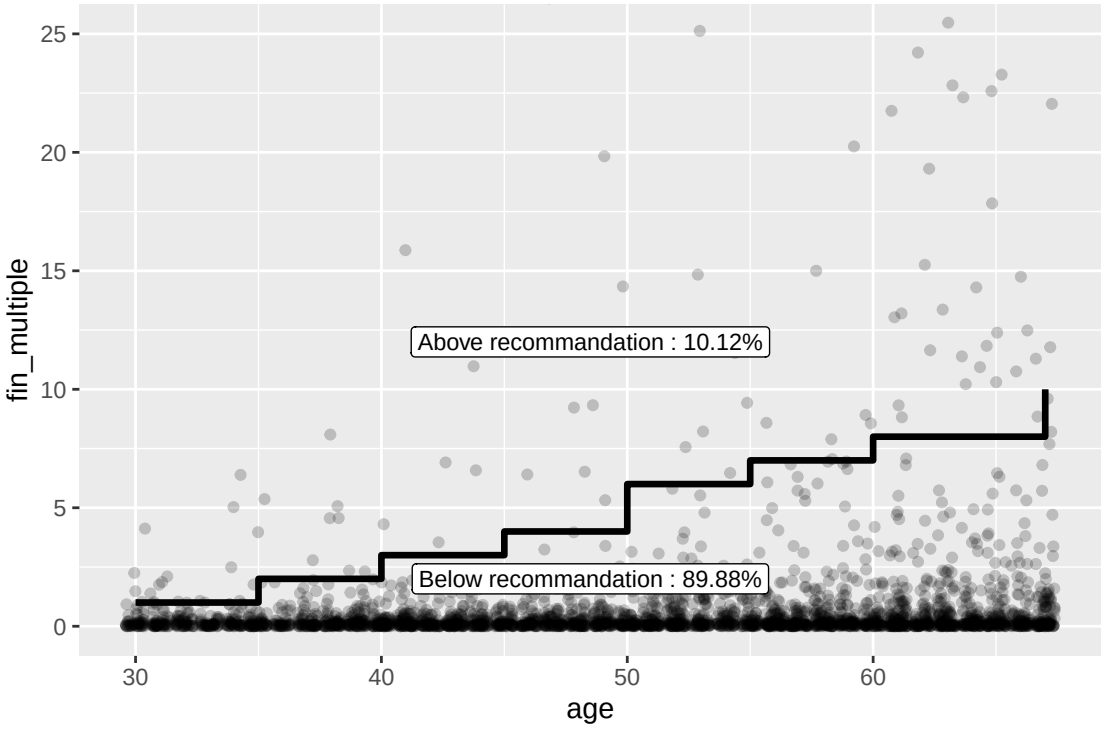
Question 8. 사회 생활을 활발히 하는 30 ~ 67세 인구의 노후 대비에 대해 분석하고자 한다. 조건에 따라 분석하고 물음에 답하시오. [8점]

- 아래의 그림은 연령별 노후 준비의 기준이다. 예를 들어, 50세에는 연봉 6배만큼의 자산이 있으면 노후 준비가 잘되어간다고 할 수 있다. 또한, 각 구간 안에서는 동일한 비율을 적용한다.
- 순자산, 금융자산, 비금융자산에 대하여 각각 제시된 기준보다 자산이 적은 사람과 많은 사람의 비율을 계산하여 예시와 같이 시각화하고 해석하여라.
- 실선은 그림에서 제시한 연령별 노후 준비 기준을 시각화한 것이다.

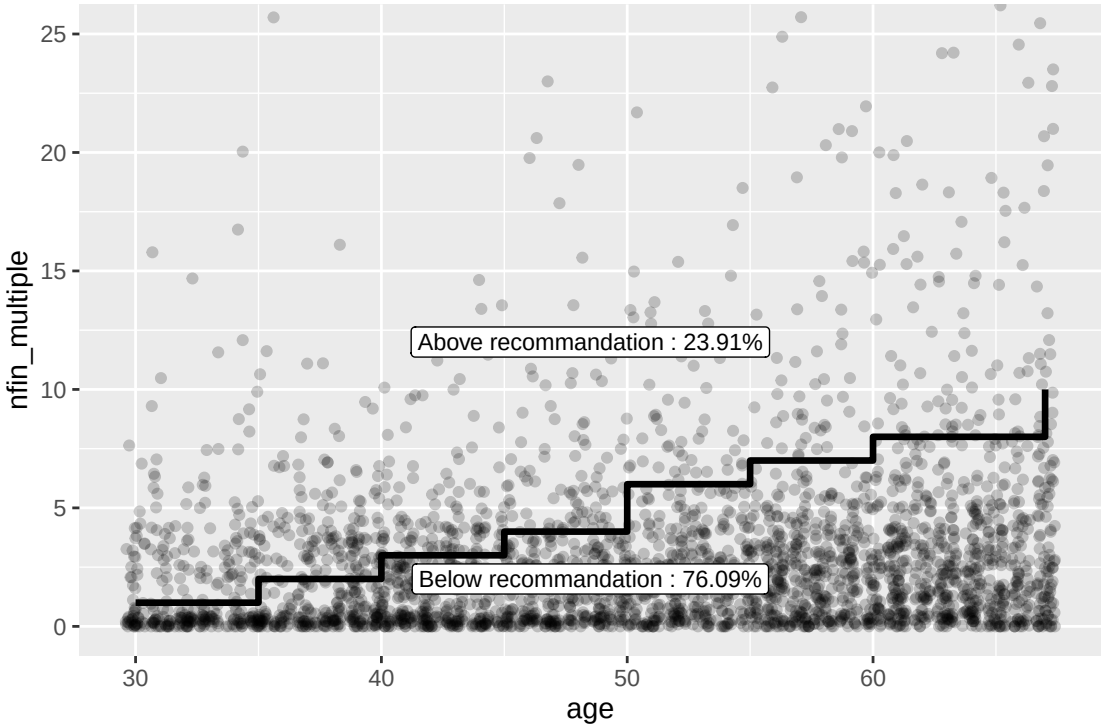
```
ddata <- tibble(
  multi = c(0, 1, 2, 3, 4, 6, 7, 8, 10),
  ages = c(0, 30, 35, 40, 45, 50, 55, 60, 67),
  age.grp = new_scf$age.grp %>% levels()
)

nohoo <- new_scf %>%
  left_join(ddata, by = "age.grp") %>%
  mutate(
    fin_nohoo = fin_multiple > multi,
    nfin_nohoo = nfin_multiple > multi,
    networth_nohoo = networth_multiple > multi
  ) %>%
  summarise(
    ab_fin_rt = round(mean(fin_nohoo) * 100, 2),
    bl_fin_rt = 100 - round(mean(fin_nohoo) * 100, 2),
    ab_nfin_rt = round(mean(nfin_nohoo) * 100, 2),
    bl_nfin_rt = 100 - round(mean(nfin_nohoo) * 100, 2),
    ab_networth_rt = round(mean(networth_nohoo) * 100, 2),
```

```
    bl_networth_rt = 100 - round(mean(networth_nohoo) * 100, 2)
  ) %>%
  t() %>%
  as_tibble %>%
  mutate(
    x = 48.5,
    y = rep(c(12, 2), 3),
    label = paste0(
      rep(c("Above ", "Below "), 2),
      "recommandation : ",
      V1,
      "%"
    )
  )


map2(
  c("fin_multiple", "nfin_multiple", "networth_multiple"),
  c(1, 3, 5),
  ~new_scf %>%
    filter(age >= 30, age <= 67) %>%
    ggplot(aes(x = age)) +
    geom_jitter(aes(y = .data[[..1]]), alpha = 0.2, height = 0) +
    geom_step(data = ddata[-1, ], aes(ages, multi), linewidth = 1.2) +
    geom_label(data = nohoo[c(..2, ..2 + 1), ], aes(
      x = x,
      y = y,
      label = label
    ), size = 3) +
    coord_cartesian(ylim = c(0, 25))
)
```

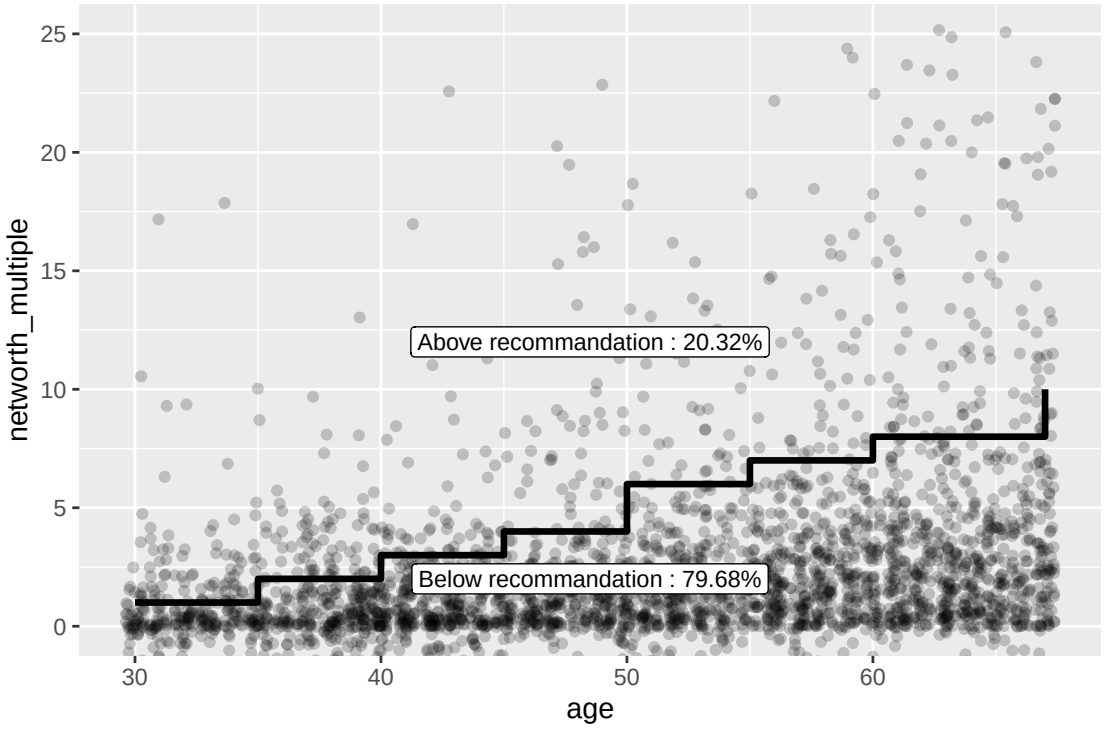
```
## [[1]]
```


fin_multiple
age
Above recommandation : 10.12%
Below recommandation : 89.88%
25
20
15
10
5
0
30
40
50
60

```
##
## [[2]]
```


nfin_multiple
age
Above recommandation : 23.91%
Below recommandation : 76.09%
25
20
15
10
5
0
30
40
50
60

```
##
## [[3]]
```



30~67세에 대해 소득대비 자산이 매우 높은(2000배) 사람을 제외하고 대부분의 표본이 있는 0배~25배를 시각화하였다. 금융자산과 비금융자산을 비교하였을 때, 보통 사람들은 금융자산보다 비금융자산을 통해서 노후 대비를 하고있음을 볼 수 있다. 금융자산으로 생각했을때 노후 대비가 된 사람은 10.12%, 비금융자산의 경우 23.91%였다. 순자산으로 보면 20.32%로 비금융보다 비율이 줄은 것을 알 수 있다.

Question 9. 신용 점수를 예측하기 위한 모형을 다음 순서 및 조건에 따라 만들고 물음에 답하시오. [15점]

- `sample()`을 이용하여 주어진 자료의 임의 표본 80%를 training set으로, 나머지 20%를 test set으로 설정하시오. [3점]
- training set에 대하여 다양한 선형 회귀 모형을 적합하시오. 단, 변수 선택 기준에 대해 서술하시오. [4점]
- test set에 대하여 예측한 `credit_score`와 관측된 `credit_score`의 SSE(Sum of squared error)가 가장 작은 모형을 선택하시오. [4점]
- 위 과정을 500번 반복한 후 가장 많이 선택된 모형을 신용 평가 모형으로 최종 선택하고 모형의 계수를 출력하시오. [4점]

```
map_dbl(
  new_scf %>% select(-age.grp),
  ~cor(., new_scf$credit_score)
) %>%
  abs() %>%
  sort(decreasing = TRUE)
```

```
##      credit_score              late            mrthel            late60
##       1.000000000       0.452563362       0.308841191       0.305647133
```

```
##              age              own          noccbal             debt
##      0.205196055      0.200340516      0.181384028      0.178859885
##         edn_inst          install         checking           income
##      0.133967178      0.075193458      0.067512922      0.061785512
##         networth         veh_inst             bond            asset
##      0.056673778      0.054161847      0.053746373      0.052147193
##          nnresre             nfin              fin           saving
##      0.050601327      0.049198930      0.046300106      0.041152678
##           stocks     fin_multiple networth_multiple           houses
##      0.037002486      0.031905952      0.031835643      0.025419990
##          housecl    nfin_multiple              yy1            vehic
##      0.022687418      0.018166166      0.011146585      0.006487035
```

```
fm1 <- credit_score ~ income + asset + debt + late
fm2 <- credit_score ~ age + income + debt + late + late60
fm3 <- credit_score ~ income + debt + asset + late + late60
fm4 <- credit_score ~ age + income + debt + asset + late + late60
fm5 <- credit_score ~ income + debt + asset + late + late60
fm6 <- credit_score ~ age + income + debt + late + late60 + own
fm7 <- credit_score ~ age + income + debt + late + late60 + noccbal + own
fm8 <- credit_score ~ age + income + debt +
  late + late60 + noccbal + own + asset

rs <- vector("double", 500)
for (i in 1:500) {
  train <- sample(seq_len(nrow(new_scf)), nrow(new_scf) * 0.8)
  train_set <- new_scf[train, ]
  test_set <- new_scf[-train, ]

  mod <- map(
    list(fm1, fm2, fm3, fm4, fm5, fm6, fm7, fm8),
    ~lm(., data = train_set)
  )

  rs[i] <- map_dbl(
    mod,
    ~mean((predict(., test_set) - test_set$credit_score)^2)
  ) %>%
    which.min()
}
```

```r
table(rs)
```

```
## rs
##   7   8
## 404  96
```

```r
mod_fin <- lm(fm7, data = new_scf)
summary(mod_fin)
```

```
##
## Call:
## lm(formula = fm7, data = new_scf)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -359.52  -47.28   -2.73   43.22  644.21
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.111e+02  5.076e+00 140.096   <2e-16 ***
## age          1.037e+00  7.255e-02  14.297   <2e-16 ***
## income       1.045e-06  9.690e-08  10.782   <2e-16 ***
## debt        -1.639e-05  8.609e-07 -19.042   <2e-16 ***
## late        -1.418e+02  4.595e+00 -30.854   <2e-16 ***
## late60      -1.262e+01  6.915e+00  -1.825    0.068 .
## noccbal     -6.122e+01  2.421e+00 -25.286   <2e-16 ***
## own          4.901e+01  3.399e+00  14.418   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 75.59 on 4450 degrees of freedom
## Multiple R-squared:  0.3929, Adjusted R-squared:  0.3919
## F-statistic: 411.4 on 7 and 4450 DF,  p-value: < 2.2e-16
```

각 변수와 credit_score의 표본상관계수를 통하여 계수의 절댓값이 큰 late, late60, age, own, noccbal, debt, income, asset으로 다양하게 모델을 만들어보았다. 각 모델들은 age, late60, asset을 넣다 빼며 만들고, 점점 변수의 수를 늘려서도 만들었다. credit_score ~ age + income + debt + late + late60 + noccbal + own 모델이 500번중 많이 뽑히므로 최종모델로 선택하였다.

Question 10. 어느 은행에서 은퇴자를 대상으로 비금융자산에 대해 담보 대출을 실시하고자 한다. 다음의 조건들을 참고하여 고객 정보를 입력하면 대출 가능 여부, 대출 한도, 금리를 산출하는 함수를 작성하여라. 채점 시 여러 조합의 입력값들에 대해 테스트하여

점수가 부여됨. [13점]

- `age…` 을 설정 가능한 입력값으로 할 것. 또한, 신용 평가에 필요한 추가적인 변수들도 함수의 입력값으로 설정하시오. `asset`, `income`, `debt`, `late`
- 입력값의 형식이나 길이가 잘못 주어졌을 경우 에러 메세지를 출력하게 하시오.
- 나이가 67세 이하 or 채무 연체 기록 존재 or 부채가 $1000000 이상인 경우 대출이 불가하다. 이 경우 다음과 같은 메세지가 출력되도록 하시오. `Loans are not available.`
- 함수의 결과값은 다음과 같은 메세지가 출력되도록 하시오. `Loans are available. Credit line : (?)$, Interest rate : (?)%`
- 대출 한도는 비금융자산의 80%로 산정한다.
- 금리는 Q9에서 선택한 모형을 이용하여 Input으로 들어오는 고객의 정보를 통해 예측한 신용 점수를 바탕으로 산정하며 기준은 다음 표와 같다.

Credit Score

Interest rate

900 ~ 1000

1%

800 ~ 899

2%

700 ~ 799

3%

600 ~ 699

4%

500 ~ 599

5%

400 ~ 499

6%

300 ~ 399

7%

200 ~ 299

8%

100 ~ 199

9%

0 ~ 99

10%

```
library(purrr)
loans <- function(
  age, income, debt, late, late60, noccbal, own, nfin, mod = mod_fin
) {
  ln <- length(c(age, income, debt, late, late60, noccbal, own, nfin))
  if (ln != 8) {
    stop("입력된 값들의 길이가 맞지않습니다. (단일 수치만 입력가능)")
  }
  isn <- map_dbl(
    c(age, income, debt, late, late60, noccbal, own, nfin), ~!is.numeric(.)
  )
  isl <- map_dbl(
    c(late, late60, noccbal, own), ~!(. %in% c(0, 1))
  )
  if (sum(isn + isl) != 0) {
    stop("올바른 형식이 아닙니다.")
  }
  if (!(age >= 67 && late == 0 && debt <= 1000000)) {
    return("Loans are not available.")
  }
  isp <- map_dbl(
    c(age, income, debt, late, late60, nfin), ~!(. >= 0)
  )
  if (sum(isp) != 0) {
    stop("양수만 입력하세요.")
  }
  my_data <- data.frame(
    age = age,
    income = income,
    debt = debt,
    late = late,
    late60 = late60,
    noccbal = noccbal,
    own = own
  )
  cs <- predict(mod, my_data)
```

```
  if (cs >= 1000) {
    interest <- 1
  } else if (cs <= 0) {
    interest <- 10
  } else {
    interest <- 10 - (cs %/% 100)
  }
  return(
    paste0(
      "Loans are available. Credit line : ", nfin * 0.8,
      "$, Interest rate : ", interest, "%"
    )
  )
}
loans(c(70, 28), 36448, 291800, 0, 0, 1, 1, 778000)
```

```
## Error in loans(c(70, 28), 36448, 291800, 0, 0, 1, 1, 778000): 입력된 값들의 길이가 맞지않습니다. (단일 수치
```

```
loans("adf", 464576, 0, 1, 0, 1, 0, 1398986)
```

```
## Error in loans("adf", 464576, 0, 1, 0, 1, 0, 1398986): 올바른 형식이 아닙니다.
```

```
loans(70, 36448, 291800, 5, 0, 1, 1, 778000)
```

```
## Error in loans(70, 36448, 291800, 5, 0, 1, 1, 778000): 올바른 형식이 아닙니다.
```

```
loans(70, 36448, -291800, 0, 0, 1, 1, 778000)
```

```
## Error in loans(70, 36448, -291800, 0, 0, 1, 1, 778000): 양수만 입력하세요.
```

```
loans(68, 464576, 0, 1, 0, 1, 0, 1398986)
```

```
## [1] "Loans are not available."
```

```
loans(46, 224830, 549800, 0, 0, 0, 1, 639800)
```

```
## [1] "Loans are not available."
```

```
loans(79, 5277876, 1380000, 0, 0, 1, 1, 10005200)
```

```
## [1] "Loans are not available."
```

```
loans(70, 36448, 291800, 0, 0, 1, 1, 778000)
```

```
## [1] "Loans are available. Credit line : 622400$, Interest rate : 3%"
```

Topic 4 : Behavioral Risk Factor Surveillance System (BRFSS)

Behavioral Risk Factor Surveillance System(BRFSS)은 미국 CDC가 18세 이상 성인을 대상으로 전화 설문조사를 실시하여 건강 관련 정보를 수집하는 프로그램이다. 이 조사에는 나이, 인종, 성별, 소득, 교육 수준 등의 인구통계학적 변수와 만성질환 여부, 위험 행동, 의료 서비스 접근성 등 다양한 건강 관련 질문이 포함된다. 매년 약 40만 명의 응답자가 참여하며, 수집된 데이터는 모든 주에서 공통적으로 사용되는 핵심 문항(core component)과 일부 주에서만 선택적으로 포함되는 선택 모듈(optional modules)로 구성된다. 본 프로젝트에서는 2023년 데이터를 활용하였으며, 자세한 설명은 공식 사이트에서 확인할 수 있다.

`brfss2023.csv`는 이 프로젝트를 위해 추려진 변수들을 포함하고 있는 데이터셋이며, 첨부된 코드북(`brfss2023_codebook.html`)에 각 변수명에 대한 설명이 기재되어 있다. 따라서 프로젝트 수행 시 해당 파일에서 변수명으로 검색하여 필요한 정보를 얻을 수 있다. (`X_*` 변수들은 `X`를 제외한 `_*`로 검색해야 한다.) 또한, `states.csv`는 주별 FIPS 코드를, `modules2023.csv`는 선택 모듈별로 참여 주에 대한 데이터셋이다.

Question 11. (States and Optional Modules)

a. BRFSS 데이터를 불러온 후 조건에 맞게 변형하고, 결과를 출력하여라. [2점]

- `brfss2023.csv` 데이터를 불러와 변수 `brfss`에 저장한다.
- `states.csv`를 이용하여 state FIPS code를 나타내는 `X_STATE` 대신 이름을 나타내는 `state` 변수를 첫 번째 열로 추가한다.
- 다섯 개의 행을 랜덤하게 출력하라.

```
brfss <- read_csv(paste0(wd, "brfss2023.csv"))
```

```
## Rows: 433323 Columns: 23
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (23): X_STATE, X_AGE_G, X_SEX, X_RACEPRV, X_INCOMG1, X_EDUCAG, SOMALE, S...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
modules <- read_csv(paste0(wd, "modules.csv"))
```

```
## Rows: 30 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): module, states
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
states <- read_csv(paste0(wd, "states.csv"))
```

```
## Rows: 52 Columns: 2
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): state
## dbl (1): value
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
new_brfss <- brfss %>%
  left_join(states, by = c("X_STATE" = "value")) %>%
  dplyr::select(state, everything(), -X_STATE)

slice_sample(new_brfss, n = 5)
```

```
## # A tibble: 5 x 23
##   state      X_AGE_G X_SEX X_RACEPRV X_INCOMG1 X_EDUCAG SOMALE SOFEMALE TRNSGNDR
##   <chr>        <dbl> <dbl>     <dbl>     <dbl>    <dbl>  <dbl>    <dbl>    <dbl>
## 1 New Mexico       6     2         3         2        2     NA        2        4
## 2 Utah             3     1         8         3        3      2       NA        4
## 3 Virgin Is~       4     2         2         2        3     NA        2        4
## 4 North Dak~       6     2         1         2        3     NA        2        4
## 5 Minnesota        6     2         8         4        4     NA       NA       NA
## # i 14 more variables: X_MICHD <dbl>, ADDEPEV3 <dbl>, X_RFSMOK3 <dbl>,
## #   ACEHURT1 <dbl>, ACETOUCH <dbl>, ACETTHEM <dbl>, ACEHVSEX <dbl>,
## #   ACESWEAR <dbl>, ACEDEPRS <dbl>, ACEDRINK <dbl>, ACEDRUGS <dbl>,
## #   ACEPRISN <dbl>, ACEDIVRC <dbl>, ACEPUNCH <dbl>
```

b. 선택 모듈은 일부 주에서만 채택하여 사용하므로, 관심 있는 연구 질문에 답하기 위해 특정 모듈을 사용한 주가 어디인지 알아내고 싶다고 하자. `modules.csv`를 이용해 다음 조건을 만족하는 함수를 작성하고 해당하는 결과를 출력하여라. [5점]

- 입력값으로 길이 1 이상의 모듈명 벡터를 받는다.
- 주어진 모듈들을 모두 채택하여 설문에 사용한 주들의 이름 벡터를 반환(`return`)한다.
- 이때, 존재하지 않는 모듈(들)이 입력으로 주어지면, 그 모듈명을 모두 포함한 다음과 같은 메시지와 함께 error가 발생하도록 한다. `"The following module(s) do not exist: <all non-existent module names>"`.
- (예시 1), (예시 2) 각각에 해당하는 입력값으로 함수를 실행하여라. 이때, (예시 2)는 에러를 발생시키므로 R Markdown에서 해당 code chunk만 `error=TRUE`로 설정하여 에러가 발생하여도 성공적으로 knit가 될 수 있도록 한다.

예시 1. (valid)

예시 2. (error)

```
library(stringr)
library(dplyr)
```

```r
get_states_by_modules <- function(v, data = modules) {
  if (!((diff <- setdiff(v, data$module)) |> length()) == 0) {
    stop(
      paste0(
        "The following module(s) do not exist: ",
        paste0("'", diff, "'", collapse = ", "),
        "."
      )
    )
  }
  my_data <- data.frame(module = v) %>% left_join(data)
  return(Reduce(intersect, str_split(my_data$states, ", ")))
}

states_ace <- get_states_by_modules(
  c(
    "Adverse Childhood Experiences",
    "Sexual Orientation and Gender Identity (SOGI)"
  )
)
```

```
## Joining with `by = join_by(module)`
```

```r
states_ace
```

```
## [1] "Delaware"     "Georgia"      "Missouri"     "Nevada"       "New Jersey"
## [6] "Rhode Island" "Virginia"
```

```r
get_states_by_modules(c(
  "Adverse Childhood Experiences",
  "Depression",
  "This module doesn't exist!"
))
```

```
## Error in get_states_by_modules(c("Adverse Childhood Experiences", "Depression", : The following modul
```

Question 12. (Data Preprocessing)

Adverse Childhood Experience(ACE)는 개인의 건강에 장기적으로 부정적 영향을 끼칠 수 있는 것으로 알려져 있다. 이러한 현상을 2023년 BRFSS 데이터를 통해 확인해보고자 한다. 특히 관심있는 건강 관련 반응변수는 심혈관 질환 발병 여부이다.

참고. About Adverse Childhood Experiences

선택 모듈 'Adverse Childhood Experiences'를 사용한 Nevada 주로 한정하여 분석을 진행하고자 한다. 다음과 같이 Nevada

주의 데이터만 필터링한 df 변수를 생성하자.

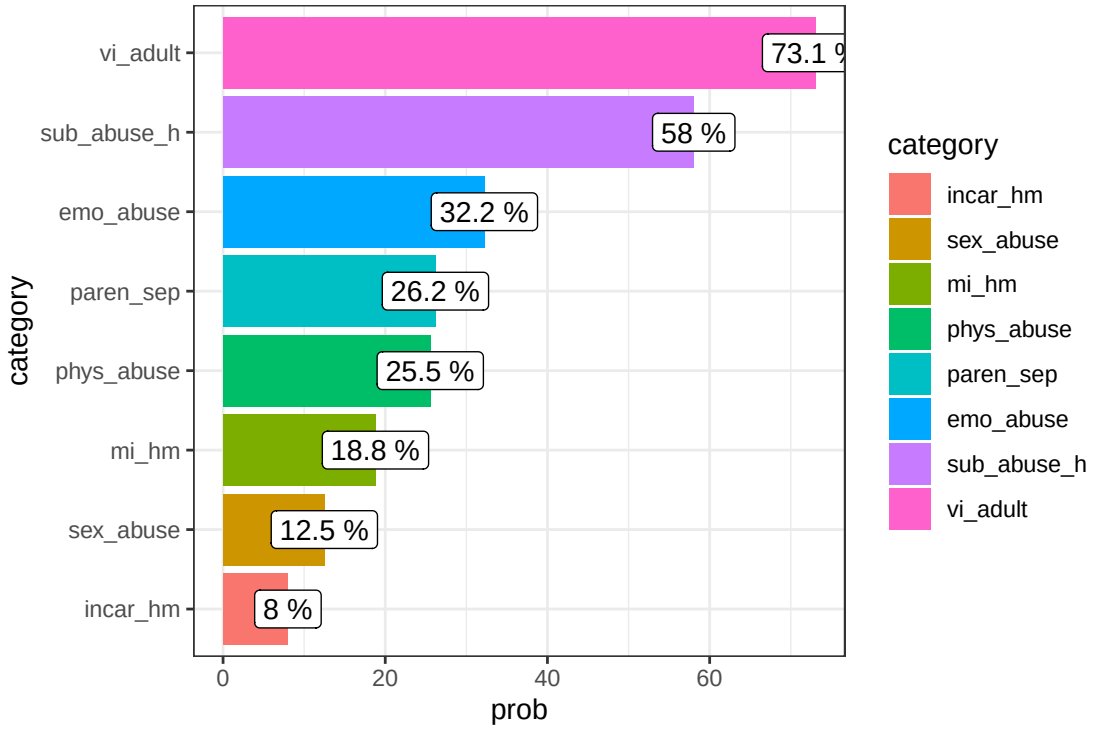
```
df <- brfss |>
  filter(state == "Nevada") |>
  select(-c(state, X_STATE))
```

a. ACE를 아래 표와 같이 8개의 세부 카테고리로 분류하자. 다음 조건에 맞게 ACE 문항들을 8개의 세부 카테고리로 정리하고, 그 결과를 시각화하여라. [6점]

- Scoring 기준을 따라 phys_abuse, sex_abuse 등의 8개의 이진형 변수를 만든다.
- "Don't know/Not Sure", "Refused"와 missing 모두 0으로 처리한다.
- 세부 카테고리 별로 발생 빈도를 나타내는 막대 그래프를 그린다. 이때, 막대들을 발생 빈도 순으로 정렬하고, 각 막대 위에 발생 빈도를 백분율(%) 단위로 label한다. 빈도를 계산할 때 NA 값은 무시한다. (Hint: `mean(. na.rm=TRUE)`)

```
get_binary_freq <- function(v) {
  if (v %in% c(1, 7, 8, 9, NA)) {
    return(0)
  } else {
    return(1)
  }
}
get_binary_yn <- function(v) {
  if (v %in% c(2, 7, 8, 9, NA)) {
    return(0)
  } else {
    return(1)
  }
}

df <- new_brfss %>%
  filter(state == "Nevada") %>%
  select(-state) %>%
  mutate(
    phys_abuse = map_dbl(ACEHURT1, get_binary_freq),
    sex_abuse = map_dbl(ACETOUCH + ACETTHEM + ACEHVSEX - 2, get_binary_freq),
    emo_abuse = map_dbl(ACESWEAR, get_binary_freq),
    mi_hm = map_dbl(ACEDEPRS, get_binary_yn),
    sub_abuse_h = map_dbl(ACEDRINK + ACEDRUGS - 1, get_binary_yn),
    incar_hm = map_dbl(ACEPRISN, get_binary_yn),
    vi_adult = map_dbl(ACEPUNCH, get_binary_yn),
    paren_sep = map_dbl(ACEDIVRC, get_binary_yn)
```

```
  )



df %>%
  summarise_at(vars(-(1:22)), ~mean(.) * 100) %>%
  pivot_longer(everything(), names_to = "category", values_to = "prob") %>%
  mutate_at("category", ~fct_reorder(as_factor(.), prob)) %>%
  ggplot(aes(x = category, y = prob)) +
  geom_bar(aes(fill = category), stat = "identity") +
  geom_label(aes(label = paste(round(prob, 1), "%"))) +
  coord_flip() +
  theme_bw()
```



b. ACE score를 모든 세부 카테고리 점수의 합으로 정의하자. ACE score는 0에서 8까지의 값을 가질 수 있다. 또한, ACE score가 3 이상이면 "high", 0~2이면 "low"로 정의하자. [3점]

- ACE score를 나타내는 `ace_score` 열을 추가한다.
- ACE score를 이산화하여 3 이상이면 "high", 0~2이면 "low"의 값을 가지는 factor `ace_hl` 열을 생성한다. 이때, factor level의 순서는 "low", "high"로 지정한다.
- 각 ACE 레벨(`ace_hl`)별 count를 출력한다.

```
df_b <- df %>%
  mutate(ace_score = rowSums(df[, -(1:22)])) %>%
```

```
  mutate(ace_hl = cut(
    ace_score,
    breaks = c(-Inf, 2, Inf),
    labels = c("low", "high")
  ))

df_b %>% count(ace_hl)
```

```
## # A tibble: 2 x 2
##   ace_hl     n
##   <fct>  <int>
## 1 low     1385
## 2 high    1265
```

c. 심혈관 질환(Cardiovascular disease, 이하 CVD) 발병 여부에 대한 변수 `X_MICHD`를 다음 조건에 따라 적절히 변환하고, 결과를 확인하여라. [2점]

- 1 = Yes, 0 = No인 이진형 numeric 벡터 `cvd` 열을 생성한다.
- missing은 그대로 `NA`로 처리한다.
- `1`, `0`, `NA` 각각의 개수와 발병률(non-NA 응답 중 1의 비율)을 출력한다.

```
df_c <- df_b %>%
  mutate(cvd = (2 - X_MICHD))
df_c %>% count(cvd)
```

```
## # A tibble: 3 x 2
##     cvd     n
##   <dbl> <int>
## 1     0  2413
## 2     1   210
## 3    NA    27
```

```
mean(df_c$cvd, na.rm = TRUE)
```

```
## [1] 0.080061
```

1은 210, 0은 2413, NA는 27개이다. 발병률은 0.08이다.

d. 한편, CVD는 특정 조건의 사람들에게서 더 흔하게 발병할 수 있다. 관련된 몇 가지 주요 인구통계학적 변수는 성별, 나이, 인종, 교육 수준이다. 해당 변수들을 다음 조건에 맞게 factor로 변환하여라. [4점]

- 기존 변수(Variable)에서 조건에 맞는 새로운 factor 변수(New variable)를 생성하는데, 몇몇 변수는 기존 변수와 다른 factor level을 갖고 있다는 점을 유의한다. (예. White, Black, Hispanic 이외의 인종은 모든 Other로 합친다.)
- Factor level은 표의 Levels에 기재된 순서를 보존해야 한다.

- "Don't know/Not Sure", "Refused"와 missing 모두 NA로 처리한다.

분석에 활용할 모든 변수를 적절하게 변환했으므로, 변환된 변수(cvd, ace_hl, sex, age, race, educ) 외의 변수는 제거한다.

```
df_fin <- df_c %>%
  mutate(
    sex = factor(X_SEX, labels = c("MALE", "FEMALE")),
    age = factor(
      X_AGE_G,
      labels = c("18-24", "25-34", "35-44", "45-54", "55-64", "65-")
    ),
    race = factor(
      X_RACEPRV,
      labels = c("White", "Black", rep("Other", 5), "Hispanic")
    ) %>%
      fct_relevel(c("White", "Black", "Hispanic", "Other")),
    educ = factor(
      ifelse(X_EDUCAG == 9, NA, X_EDUCAG),
      labels = c(
        "Less than HS",
        rep("Graduated HS", 2),
        "Graduated college"
      )
    )
  ) %>%
  select(cvd, ace_hl, sex, age, race, educ)
glimpse(df_fin)
```

```
## Rows: 2,650
## Columns: 6
## $ cvd    <dbl> 1, 0, 0, NA, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0,~
## $ ace_hl <fct> high, low, low, low, high, high, high, low, high, low, low, hig~
## $ sex    <fct> MALE, MALE, FEMALE, MALE, MALE, FEMALE, FEMALE, MALE, MALE, FEM~
## $ age    <fct> 65-, 55-64, 65-, 65-, 65-, 65-, 45-54, 55-64, 55-64, 65-, 65-, ~
## $ race   <fct> White, White, White, White, White, White, White, White, White, ~
## $ educ   <fct> Graduated HS, Graduated college, Graduated college, Graduated H~
```

Question 13. (Modelling)

a. ACE와 CVD 연관성을 단순하게 파악해보자. Low ACE score 그룹에 비해 high ACE score 그룹에서 CVD의 발병 빈도가 증가하는가? 이 결과에 대해 어떻게 생각하는가? [2점]

```r
df_fin %>%
  summarise(rt = mean(cvd, na.rm = TRUE), .by = ace_hl)
```

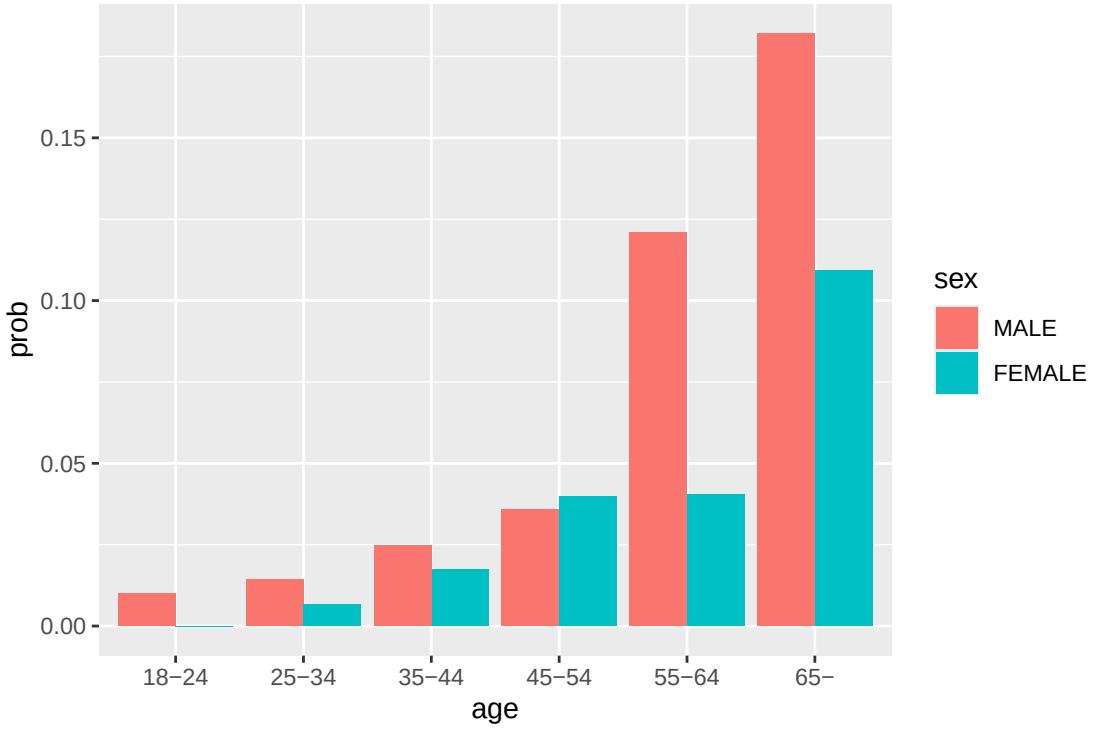
```
## # A tibble: 2 x 2
##   ace_hl     rt
##   <fct>   <dbl>
## 1 high   0.0766
## 2 low    0.0833
```

오히려 high그룹이 발병빈도가 더 낮다. 예측과 달라 조금 당황스럽긴하다. 하지만 인구통계학적 변수를 추가하였을 때 ace의 설명력이 높아져 다른 결과가 나올 것 같다.

b. 인구통계학적 변수들이 관심 반응변수인 CVD 발병 여부와 어떤 관련이 있는지 알아보자. [6점]

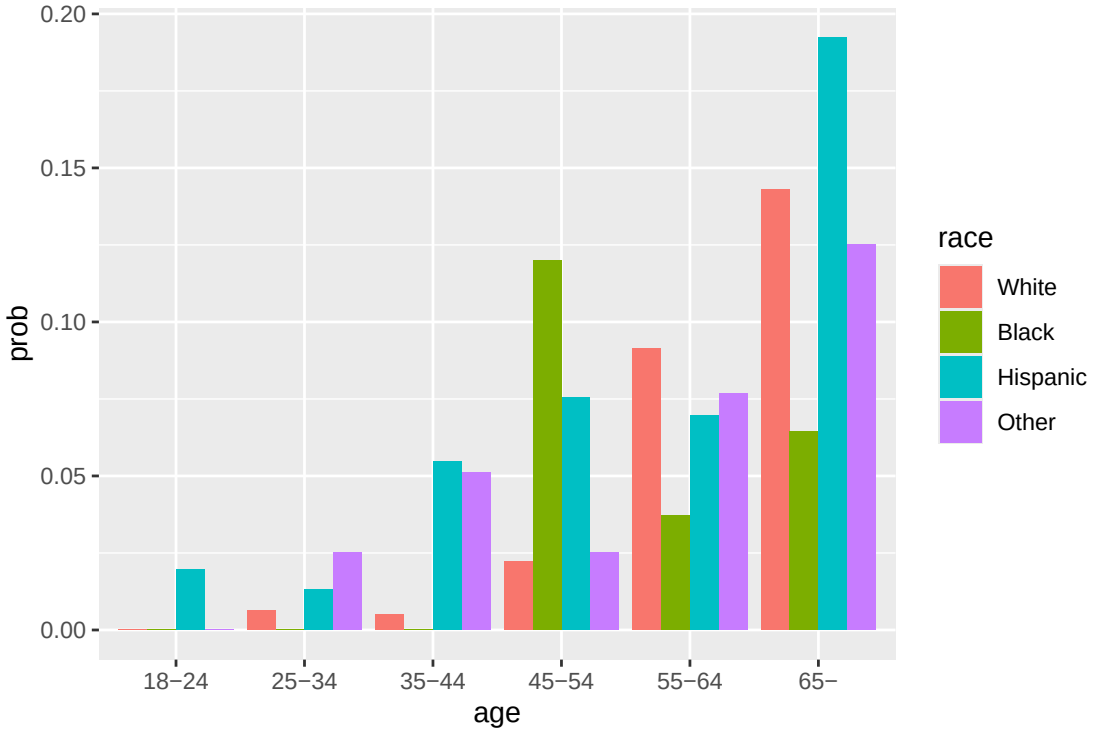
- 각 인구통계학적 변수에 대해, factor 레벨에 따라 CVD 발병률이 어떻게 변화하는지 그래프 또는 표를 통해 탐색하고 결과를 논하여라.
- 두 개의 인구학적 변수들의 조합(변수1, 변수2)에 대해 CVD 발병률을 확인하고자 한다. x축을 변수1, y축을 CVD 발병률로 하는 막대 그래프를 변수2 레벨에 따라 색으로 구분하여 그려라. 각 그래프가 드러내는 경향성을 논하여라. (Hint: 총 6개의 그래프를 그려야 한다.)

```r
df_fin %>%
  summarise(prob = mean(cvd, na.rm = TRUE), .by = c(sex, age)) %>%
  ggplot(aes(y = prob, x = age, fill = sex)) +
  geom_bar(stat = "identity", position = "dodge")
```



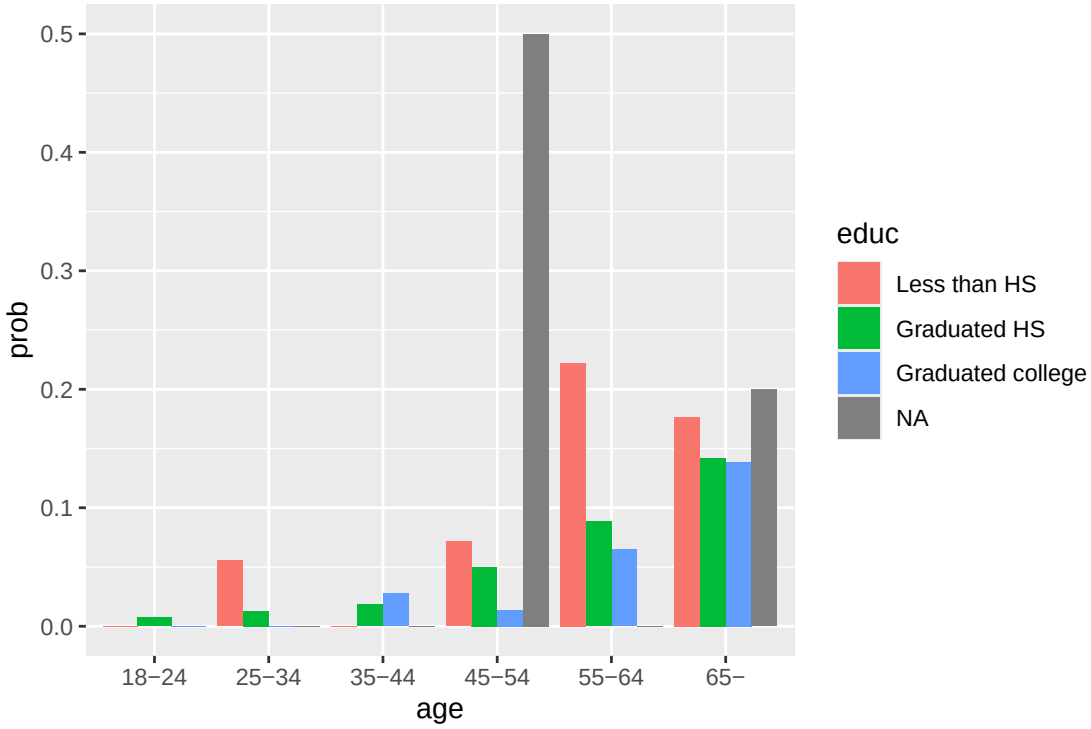
나이가 많을수록 발병률이 높아지는 추세를 보이고 있고, 남성의 나이에따른 발병률이 여성보다 빠르게 증가한다.

```
df_fin %>%
  summarise(prob = mean(cvd, na.rm = TRUE), .by = c(race, age)) %>%
  ggplot(aes(y = prob, x = age, fill = race)) +
  geom_bar(stat = "identity", position = "dodge")
```



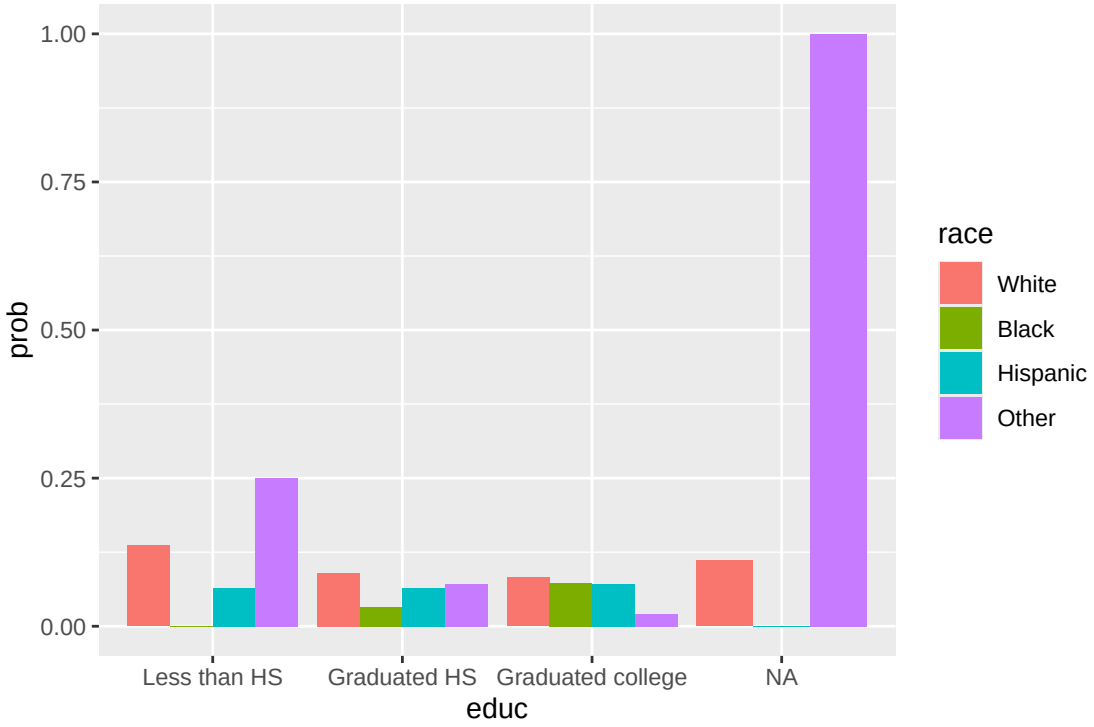
역시 나이에 따라 발병률이 오른다. 백인은 나이에 따라 발병률이 오르는게 잘 보이는데, 나머지 변수는 확실하지는 않지만 비슷한 경향성을 보인다. 각 인종끼리의 비교는 나이대마다 다 다르다.

```
df_fin %>%
  summarise(prob = mean(cvd, na.rm = TRUE), .by = c(educ, age)) %>%
  ggplot(aes(y = prob, x = age, fill = educ)) +
  geom_bar(stat = "identity", position = "dodge")
```



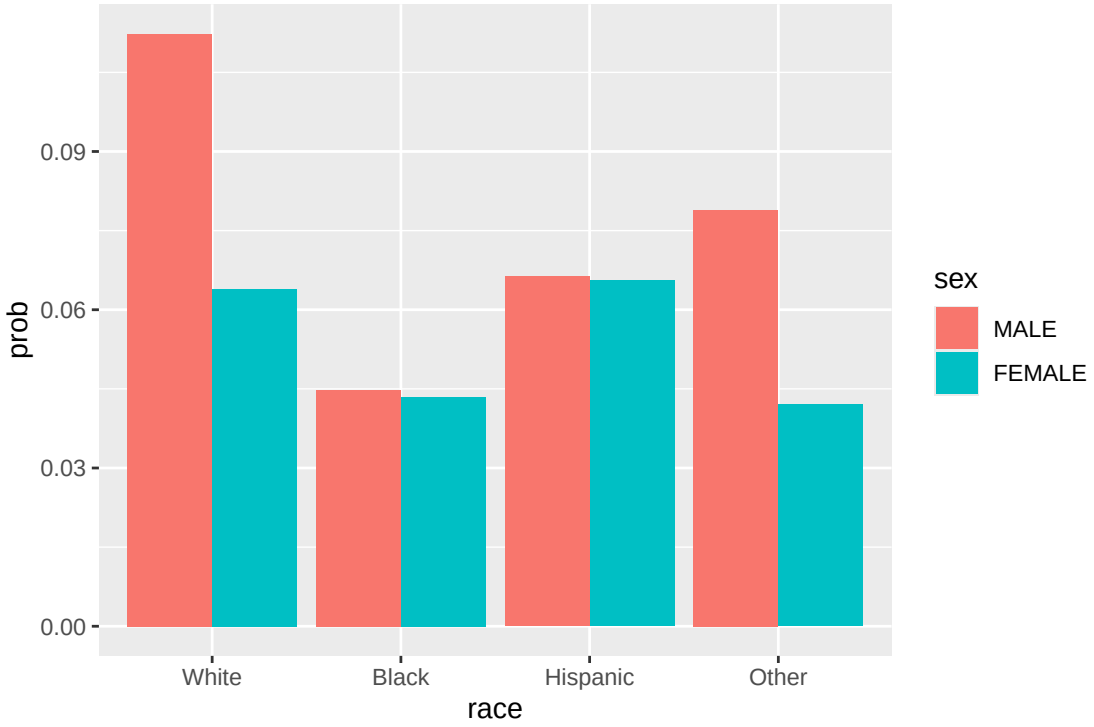
NA값을 제외하고 생각해보자. 모든 나이대그룹에서 고등학교 미졸업이 제일 발병률이 높다. 다른 학력들은 위에서와 같이 나이에 따라 우상향하는 경향을 볼 수 있다. 고등학교 미졸업이 55-64 나이대에서 높다가 65-에서 없는데… cvd로 죽었을 가능성도 있다.

```
df_fin %>%
  summarise(prob = mean(cvd, na.rm = TRUE), .by = c(educ, race)) %>%
  ggplot(aes(y = prob, x = educ, fill = race)) +
  geom_bar(stat = "identity", position = "dodge")
```



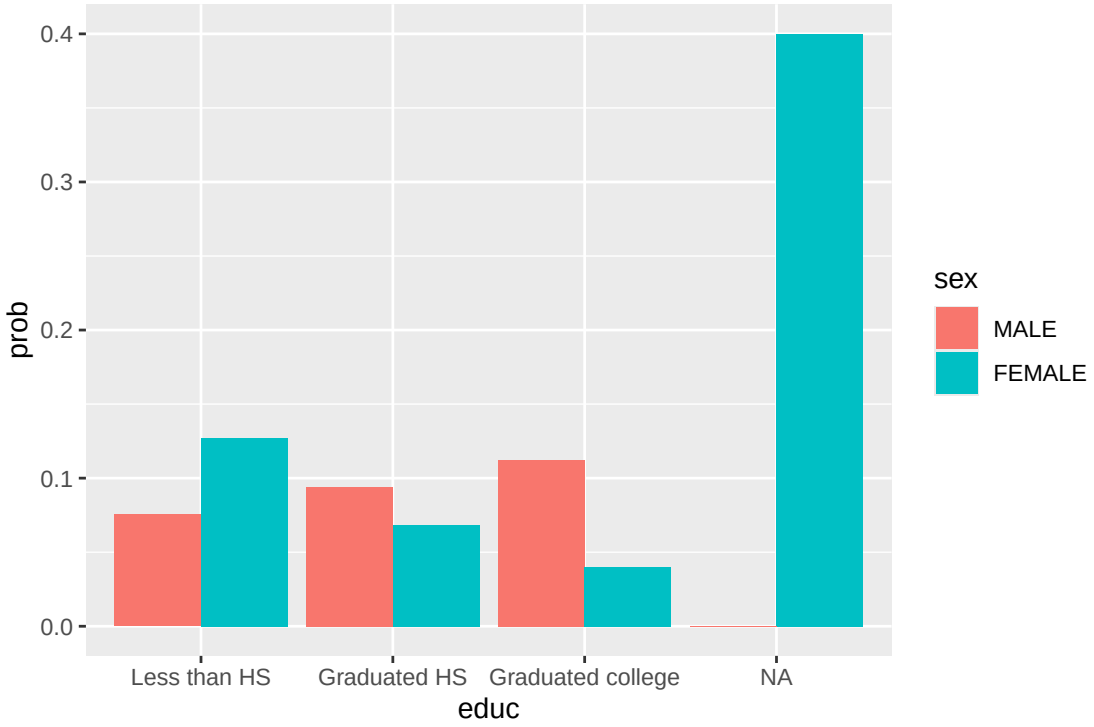
NA는 제외하고 보자 위의 결과와같이 Less than HS가 비율이 높다. 각 인종에 대해서는 위에서와 같이 각 그룹마다 우세가 다 다르다.

```
df_fin %>%
  summarise(prob = mean(cvd, na.rm = TRUE), .by = c(sex, race)) %>%
  ggplot(aes(y = prob, x = race, fill = sex)) +
  geom_bar(stat = "identity", position = "dodge")
```



모든 인종에대해 남성이 여성보다 더 높은 발병률을 보인다. 인종중에는 white가 대체적으로 발병률이 높은 편이다.

```
df_fin %>%
  summarise(prob = mean(cvd, na.rm = TRUE), .by = c(sex, educ)) %>%
  ggplot(aes(y = prob, x = educ, fill = sex)) +
  geom_bar(stat = "identity", position = "dodge")
```



남성은 학력이 높을수록 발병률이 높고 여성은 그 반대이다. 통합하여 보면 Less than HS가 발병률이 높은 것을 알 수 있다. 성별과 학력사이에 발병률에 미치는 어떠한 관계가 있음을 알 수 있다.

c. ACE가 CVD 발병에 미치는 영향을 통계적으로 모델링하고자 한다. 이때 반응변수는 이진형이므로 로지스틱 회귀 모델을 적합하는 것이 적절하다. 앞선 결과를 토대로 적절한 로지스틱 회귀 모델을 적합하고 결과를 분석하여라. [4점]

- 앞의 결과를 토대로 모델 적합 시 인구통계학적 변수 역시 포함해야 하는 이유를 서술하여라.
- `glm()`을 이용하여 로지스틱 회귀 모델을 적합하고 결과를 간단하게 서술하여라.

```
mod <- glm(cvd ~ ., data = df_fin, family = binomial)
summary(mod)
```

```
##
## Call:
## glm(formula = cvd ~ ., family = binomial, data = df_fin)
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -4.75695    1.07773  -4.414 1.02e-05 ***
## ace_hlhigh             0.16423    0.15099   1.088 0.276732
## sexFEMALE             -0.69147    0.15316  -4.515 6.34e-06 ***
## age25-34               0.69329    1.16024   0.598 0.550147
## age35-44               1.44002    1.07549   1.339 0.180591
```

```
## age45-54                2.00613    1.04783   1.915 0.055549 .
## age55-64                2.94283    1.02153   2.881 0.003966 **
## age65-                  3.63423    1.01372   3.585 0.000337 ***
## raceBlack              -0.34374    0.43739  -0.786 0.431928
## raceHispanic            0.43271    0.26509   1.632 0.102621
## raceOther              -0.06736    0.30791  -0.219 0.826836
## educGraduated HS       -0.36000    0.35443  -1.016 0.309770
## educGraduated college -0.53867    0.36467  -1.477 0.139641
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1451.6  on 2611  degrees of freedom
## Residual deviance: 1295.1  on 2599  degrees of freedom
##   (38 observations deleted due to missingness)
## AIC: 1321.1
##
## Number of Fisher Scoring iterations: 7
```

4.13.b의 결과를 보았을때 각 인구통계학적 변수가 cvd에 미치는 영향이 꽤나 분명하고, 이를 모델링에 넣으면 ace_hl이 cvd에 미치는 영향이 더 정확하게 구별되어 나타날 수 있다. 그래서 인구통계학적 변수를 넣어서 모델링을 해야한다.

위의 로지스틱 회귀분석의 결과를 보면, 유의확률이 0.05보다 낮아서 유의한 변수는 다음과 같다. sexFEMALE, age55-64, age65- 성별과 나이가 cvd에 미치는 영향이 유의하다는 것을 알 수 있다. 성별은 여성이면 발병률이 낮은 경향성이고, 나이는 많을수록 발병률이 높은 경향성이다. ace_hl은 유의확률 0.28로 유의하지 않다, 하지만 단순히 그룹화해서 비교할때와는 다르게 회귀계수가 0.164로 양수라서 ace_hl이 high이면 발병률의 로그 오즈를 더 높이는 것으로 된다. 인구통계학적 변수를 넣음으로 ace_hl이 미치는 영향이 더 잘 드러났다고 볼 수 있다.

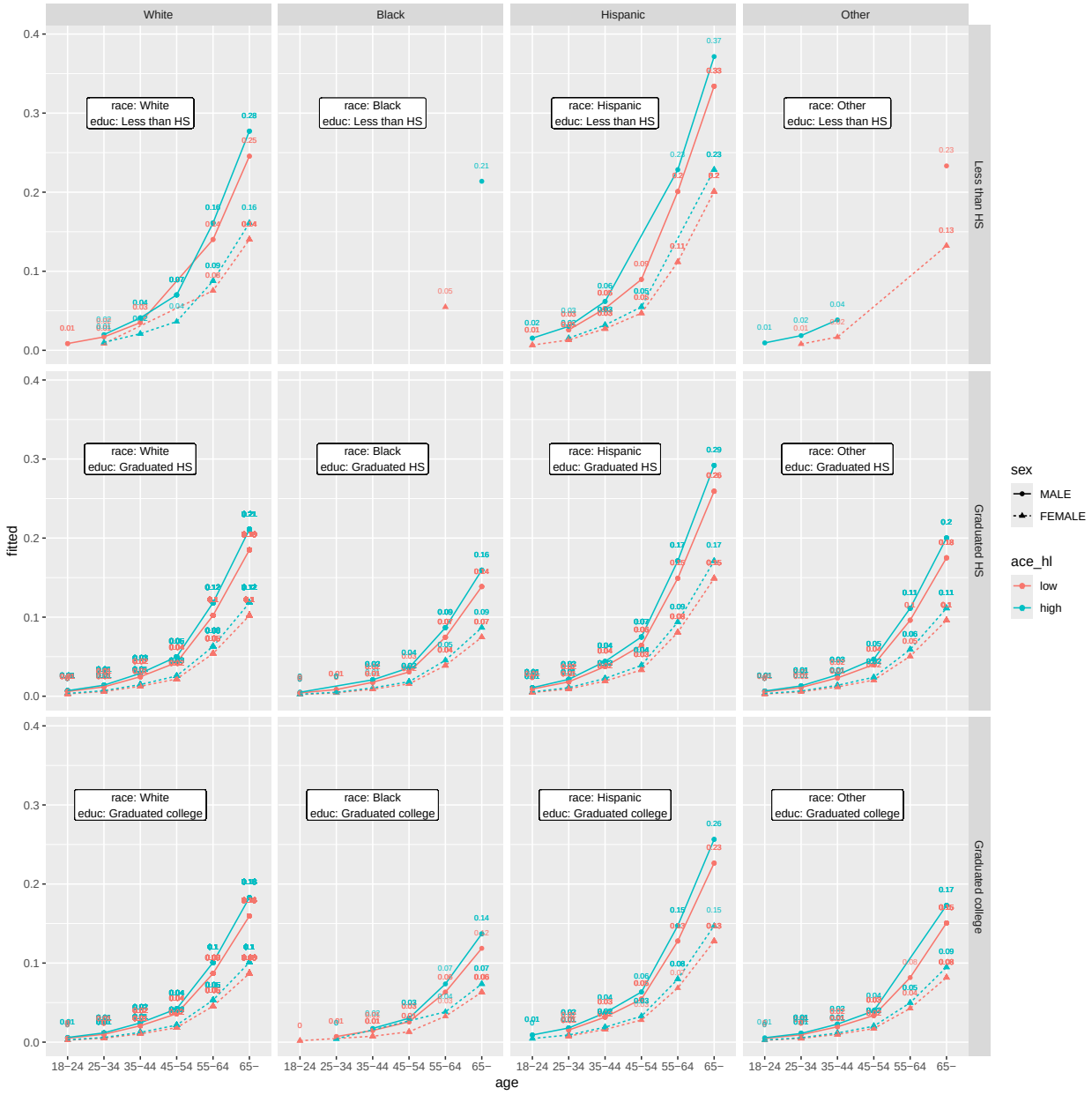
d. 적합된 로지스틱 회귀 모델이 주어진 변수 값에 따라 CVD 발병 확률을 어떻게 예측하는지 시각화를 통해 알아보자. [6점]

- x축을 age로, y축을 CVD 발병 확률 예측값으로 선 그래프를 그리는데, ace_hl은 색으로, sex는 선의 종류(linetype)로 구분하고, facet_grid()를 이용해 (educ, race) 조합별로 12개의 플롯을 생성한다. 이해를 돕기 위해 적절한 label과 legend를 추가한다. (Hint: 주어진 변수 값에서 CVD 발병 확률의 예측값은 predict(., type="response")로 구할 수 있다.)
- 그림을 바탕으로 적합된 로지스틱 회귀 모형이 예측하는 ACE와 CVD 발병의 관계를 설명하여라.

```
df_fin %>%
  na.omit() %>%
  mutate(fitted = predict(mod, type = "response")) %>%
  ggplot(
    aes(x = age, y = fitted, group = interaction(ace_hl, sex))
```

```
  ) +
  geom_point(aes(shape = sex, color = ace_hl)) +
  geom_line(aes(linetype = sex, color = ace_hl)) +
  geom_text(
    aes(label = round(fitted, 2), color = ace_hl),
    size = 2,
    nudge_y = 0.02
  ) +
  geom_label(aes(x = "35-44", y = 0.3, label = paste0(
    "race: ", race,
    "\neduc: ", educ
  )), size = 3) +
  facet_grid(educ ~ race)
```

```
## `geom_line()`: Each group consists of only one observation.
## i Do you need to adjust the group aesthetic?
```



거의 대부분의 plot에서 파란 점선이 빨간 점선보다 위, 파란 실선이 빨간 실선보다 위임을 알 수 있다. 그래프로 보았을때 ace_hl이 high인 데이터들이 low인 데이터보다 발병률의 예측값이 높음을 알 수 있다. 회귀모델을 통하여 보았을 때 ace_hl이 high이면 발병률이 높다고 예측한다. 모든 그룹에서 나이가 많아질수록 발병률의 예측값은 높아진다. 또한 성별로 보았을때 실선이 점선보다 항상 위에 있어서, 남자이면 발병률의 예측값이 높아짐을 알 수 있다. 남자, 나이, ace_hl에 따른 예측치의 차이가 확실하게 보이는 시각화이다. educ와 race에대한 예측치에 차이는 이 시각화에서 명확히 드러나지는 않는 것 같다.