1장 데이터 사전 준비

1. 개요 및 소개
2. 예측 대상 변수(Target) 생성
3. 상수 제거 및 Null 보정
4. 날짜 변수에 대해 경과일수 변수 생성
5. Summary 및 Transpose

1. 개요 및 소개

■ 데이터 소개

- **health.csv는 국내 한 손해보험 회사의 건강보험 가입 고객들로부터 임의 추출된 18,269명의 데이터임**
- **22개 변수들로 구성됨**

1. Provance: 담당 지역
2. 건강기대치: 보험유지 가능성(점수가 높을수록 보험유지 가능성은 높아짐)
3. 재갱신여부: 1(갱신), 0(갱신하지 않음)
4. 상품옵션: 실손보험(건강상품), 특약추가를 나타냄, 기본, 중급, 고급의 3분류
5. 교육: 고졸이하, 전문대, 학사
6. 고용상태: 1인 기업도 포함, 고용, 비고용
7. 성별: 0(남성), 1(여성)
8. 소득 B안: 연소득(단위: 원)
9. 거주지역: 도시근교, 도심, 시외지역
10. 결혼유무: 기혼, 미혼
11. 월별 납입금액 B안: 자동차 제외 총 월 납입보험료(단위: 만원)
12. 마지막 클레임 후 현재까지의 기간 B안: 마지막 의료비 청구 후 현재까지의 기간(단위 월)

■ 데이터 소개

13. 상품보유기간: 실손보험 가입 기간(단위 월)
14. 불만표현횟수
15. 이용서비스: 건강상품외 이용서비스: 자동차, 약관대출 등
16. 고객타입: 개인, 법인
17. 고객등급: 개인1, 개인2, 개인3, 법인1, 법인2, 법인3
18. 갱신인센티브: 갱신을 하기 위한 할인, 포인트, 사은품, 없음
19. 판매채널: 자사조직, 콜센터, 인터넷, GA
20. 총 지불금액 B안: 총 지불된 보험금(실손, 건강상품에 한함)
21. 집형태: 일반, 상
22. 집크기: 대 중, 소(가격에 의함)
23. 나이: 20대, 30대, ...

1. 개요 및 소개

■ 데이터 가져오기

```
> library(tidyverse)
> health_raw <- read_csv( "dataupdated.csv", locale=locale('ko', encoding='euc-kr'))
> health = health_raw %>%
          rename(retention = "건강 기대치",
                 renewal= 재갱신여부,                          # 1(갱신), 0(갱신하지 않음)
                 product_type = 상품옵션,                      # 기본, 중급, 고급의 3분류
                 edu = 교육,                                   # 고졸이하, 전문대, 학사
                 work = 고용상태,                              # 고용, 비고용
                 gender = 성별,                                # 0(남성), 1(여성)
                 income =  "소득 B안",                         # 연소득 (단위: 원)
                 residence = 거주지역,                         # 도시근교, 도심, 시외지역
                 marrage = 결혼유무,                           # 결혼상태: 기혼, 미혼, 무응답
                 monthly_premium = "월별 납입금액 B안",        # 월납입 보험료(자동차보험제외)
                 period_claim ="마지막 클레임 후 현재까지의 기간 B안", # (단위 월)
                 period_keep = "상품보유기간",                 # 실손보험 가입 기간(단위 월)
                 n_complaint = 불만표현횟수,                   # 불만표현횟수
                 n_product  = "이용 서비스",                   # 건강상품외 이용서비스: 자동차, 약관대출 등
                 customer_type =  고객타입,                    # 개인, 법인
                 customer_grade = 고객등급,                    # 개인1, 개인2, 개인3, 법인1, 법인2, 법인3
                 incentive =갱신인센티브,                      # 갱신을 하기 위한 할인, 포인트, 사은품, 없음
```

■ 데이터 가져오기

```
                channel = 판매체널,                          # 자사조직, 콜센터, 인터넷, GA
                claim_size = "총 지불금액 B안",              # 총 지불된 보험금(실손, 건강상품에 한함)
                house_type = "집 형태",                      # 일반, 상
                house_size = "집 크기",                      # 대 중, 소(가격에 의함)
                age = 나이)                                  # 20대, 30대, ......
> str(health)
spec_tbl_df [18,268 x 24] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ ID              : num [1:18268] 1 2 3 4 5 6 7 8 9 10 ...
 $ Provance        : chr [1:18268] "경상남북도" "전라남북도" "강원도" "서울경기" ...
 $ retention       : num [1:18268] 1.2 5.8 8.7 6.4 1.3 6.9 4.4 6.2 9.6 6.1 ...
 $ renewal         : num [1:18268] 0 0 0 0 0 1 1 0 1 0 ...
 $ product_type    : chr [1:18268] "기본" "중급" "고급" "기본" ...
 $ edu             : chr [1:18268] "학사" "학사" "학사" "학사" ...
 $ work            : chr [1:18268] "고용" "비고용" "고용" "비고용" ...
 $ gender          : num [1:18268] 0 0 0 1 1 0 0 1 1 0 ...
 $ income          : num [1:18268] 51360000 0 44510000 0 40010000 ...
 $ residence       : chr [1:18268] "도시근교" "도시근교" "도시근교" "도시근교" ...
 $ marrage         : chr [1:18268] "기혼" "미혼" "기혼" "기혼" ...
 $ monthly_premium: num [1:18268] 60 90 100 100 70 60 60 90 70 90 ...
 $ period_claim    : num [1:18268] 31 12 17 17 11 13 0 0 12 16 ...
 $ period_keep     : num [1:18268] 5 42 38 65 44 94 13 68 3 7 ...
```

1. 개요 및 소개

■ 데이터 가져오기

```
$ n_complaint    : num [1:18268] 0 0 0 0 0 0 0 0 0 0 ...
$ n_product      : chr [1:18268] "1" "4이상" "2" "4이상" ...
$ customer_type  : chr [1:18268] "법인" "개인" "개인" "법인" ...
$ customer_grade : chr [1:18268] "법인3" "개인3" "개인3" "법인2" ...
$ incentive      : chr [1:18268] "할인" "포인트" "할인" "할인" ...
$ channel        : chr [1:18268] "자사조직" "자사조직" "자사조직" "콜센터" ...
$ claim_size     : num [1:18268] 330 1030 490 460 120 140 290 320 440 370 ...
$ house_type     : chr [1:18268] "일반" "일반" "일반" "일반" ...
$ house_size     : chr [1:18268] "중" "중" "중" "중" ...
$ age            : num [1:18268] 20 50 80 40 30 60 40 50 80 50 ...
- attr(*, "spec")=
 .. cols(
 ..   ID = col_double(),
 ..   Provance = col_character(),
 ..   `건강 기대치` = col_double(),
 ..   재갱신여부 = col_double(),
 ..   상품옵션 = col_character(),
 ..   교육 = col_character(),
 ..   고용상태 = col_character(),
 ..   성별 = col_double(),
 ..   `소득 B안` = col_double(),
```

■ 데이터 가져오기

```
..   거주지역 = col_character(),
..   결혼유무 = col_character(),
..   `월별 납입금액 B안` = col_double(),
..   `마지막 클레임 후 현재까지의 기간 B안` = col_double(),
..   상품보유기간 = col_double(),
..   불만표현횟수 = col_double(),
..   `이용 서비스` = col_character(),
..   고객타입 = col_character(),
..   고객등급 = col_character(),
..   갱신인센티브 = col_character(),
..   판매체널 = col_character(),
..   `총 지불금액 B안` = col_double(),
..   `집 형태` = col_character(),
..   `집 크기` = col_character(),
..   나이 = col_double()
.. )
- attr(*, "problems")=<externalptr>
```

3. 날짜 변수에 대해 경과 일수 변수 생성

■ 날짜 관련 기본 함수

- 날짜는 **1970-01-01** 이후의 일수**(number of days)**로 표현되며, 그 이전의 날짜는 음의 값으로 나타냄

문자열 함수	설명
Sys.Date()	• 현재의 시스템 날짜를 “4자리수 년-월-일“ 형식의 문자형으로 반환
Sys.time()	• 현재의 시스템 날짜를 “4자리수 년-월-일 시:분:초 KST” 형식의 문자형으로 반환
date()	• 현재의 시스템 날짜를 “요일 월 일 시:분:초 4자리수 년” 형식의 문자형으로 반환
as.Date(x, “format”)	• 문자열을 주어진 형식에 맞게 읽어 “%Y-%m-%d” 형식의 날짜로 전환해 줌 • “format” =“%Y-%m-%d” , “%Y/%m/%d” • “%Y-%m-%d” 형식으로 반환함
strptime(x, “format”)	• 문자로부터 날짜-시간을 POSIXlt의 형식으로 전환해줌. • 기본값 “format” =“%Y-%m-%d %H:%M:%S”
strftime(x, “format”)	• 시간을 문자로 변환
ISOdatetime()	• 수치 표현을 날짜/시간으로 전환하는 함수

3. 날짜 변수에 대해 경과 일수 변수 생성

■ 날짜 관련 기본 함수

- **format: %d, %a, %A, %m, %b, %B, %y, %Y**

기호	의미	예시
%d	일 숫자(0-31)	01-31
%a %A	요일 약어 요일	Mon Monday
%m	월(00-12)	00-12
%b %B	월 약어 월	Jan January
%y %Y	2자리 년 4자리 년	07 2007

3. 날짜 변수에 대해 경과 일수 변수 생성

■ 날짜 관련 기본 함수

```
> dates.1 = as.Date(c("2016-02-04", "2016-02-10", "2016-02-15"))
> class(dates.1)
[1] "Date "
> diff(dates.1)
Time differences in days
[1] 6 5
> days = dates.1[2] - dates.1[1]; days  # 두 날짜 사이의 일 수 계산
> Time difference of 6 days
> today = Sys.Date()              # 오늘 날짜, 시간을 저장
> format(today, format= "%B %d %Y")
[1] "1월 21 2021"
> format(today, format= "%b %d %Y")
[1] "1 21 2021 "
> date.chr = c("02/04/2016", "02/10/2016")          # 특정 형식으로 지정된 날짜 읽어 오기
> date.new = as.Date(date.chr, format= "%m/%d/%Y")    # 저장된 형식을 format= 으로 지정
> date.new
[1] "2016-02-04" "2016-02-10 "
> class(date.new)
[1] "Date "
> date.chr2 = as.character(date.new); date.chr2
 [1] "2016-02-04" "2016-02-10 "
> class(date.chr2)
[1] "character"
```

3. 날짜 변수에 대해 경과 일수 변수 생성

■ lubridate 패키지 함수

- **ymd(), hms(), ymd_hms(), year(), month(), dat(), wday(), hour(), minutes(), second()**

문자열 함수	설명
now()	• 현재의 시스템 날짜를 “4자리수 년-월-일 시:분:초 KST” 형식의 문자형으로 반환
ymd(x), mdy(x), dmy(x), ydm(x), dym(x), myd(x)	• x: 바꾸고자 하는 문자형 또는 수치형 날짜 객체 • y: 년, m: 월, d: 일 • “%Y-%m-%d” 형식으로 반환함
hms(x), hm(x), ms(x), ymd_hms(x), ymd_hm(x), ymd_h(x), mdy_hms(x)…	• x: 바꾸고자 하는 문자형 또는 수치형 날짜 객체로서 각 함수의 글자 순서에 따름 • ““%Y-%m-%d H:M:S UTC” 형식의 문자형으로 반환 UTC(Coordinated Universal Time)는 세계 공통의 협정 시간(영국)
year(x), month(x), day(x), wday(x), hour(x), mininute(x), second(x)	• x: 바꾸고자 하는 문자형 또는 수치형 날짜 객체로서 년, 월 일, 요일, 시, 분, 초를 반환함

3. 날짜 변수에 대해 경과 일수 변수 생성

■ lubridate 패키지 함수

```
> ymd_hms(20230813100138, "2023-08-13 10 1 38", "2023/08/13 10,1,38", "2023 08 13 10, 01 38",
          "2023-8, 13 10:01:38", "202308-13 10::01::38", "20238-13 10:01:::38")
[1] "2023-08-13 10:01:38 UTC" "2023-08-13 10:01:38 UTC" "2023-08-13 10:01:38 UTC"
[4] "2023-08-13 10:01:38 UTC" "2023-08-13 10:01:38 UTC" "2023-08-13 10:01:38 UTC"
[7] "2023-08-13 10:01:38 UTC"
> date_time <- now(); date_time     # 현재의 R 시스템 날짜-시간
[1] "2021-03-31 16:19:56 KST"
> year(date_time)                   # 년 반환
[1] 2021
> month(date_time)                  # 월 반환
[1] 3
> day(date_time)                    # 일 반환
[1] 31
> wday(date_time,label=T)           # 요일 반환, label=T 옵션을 주면 팩터로 변환
[1] 수
Levels: 일 < 월 < 화 < 수 < 목 < 금 < 토
> hour(date_time)                   # 시 반환
[1] 16
> minute(date_time)                 # 분 반환
[1] 19
> second(date_time)                 # 초 반환
[1] 56.40385
```

4. Summary 및 Transpose

■ **lubridate 패키지 함수**

2장 데이터 탐색

1. 데이터 탐색 개요 및 소개
2. 변수 속성 정의
3. 변수별 정보 가치(Information Value) 산출

1. 데이터 탐색 개요 및 소개

■ 데이터 탐색

- **머신 러닝을 하기 위해 데이터를 수집하고 R 데이터 구조로 데이터를 로딩 한 이후는 데이터를 자세히 관찰하는 데이터 탐색 단계임**
- **데이터의 특징과 예시를 탐색하고 데이터를 고유하게 만들어줄 특성을 알게 해줌**
- **데이터를 잘 이해하게 될수록 학습 문제를 머신 러닝 모델에 잘 매칭할 수 있음**
- **데이터 탐색의 대상**
  - 변수들에 대한 평균, 표준편차, 최댓값, 최솟값, 상관계수 등 변수의 분포 또는 특성을 구하여 살펴봄
  - 그래프를 이용하여 시각적으로 표현하여 데이터의 구조를 이해함
  - 각 변수들의 분포, 이상점 등을 파악함
  - 변수들 간의 상관성에 대한 정보를 파악함
  - 데이터 값이 실제 가능한 범위의 값인지 벗어난 값인지를 통해 입력 오류를 파악함
  - 통계적 분석을 하기 전에 데이터의 내용과 특성을 정확히 아는 것이 매우 중요함

1. 데이터 탐색의 개요 및 소개

■ 데이터 구조 탐색

- **데이터셋을 조사할 때 물어봐야 할 첫 번째 질문 중 하나는 데이터셋이 어떻게 구성되어 있는 가임**
- **출처가 제공하는 데이터 사전(data dictionary)이 있는지 확인함**
  - 데이터 사전은 데이터셋의 특징을 설명하는 문서임
- **데이터 사전이 없다면 직접 작성하는 것이 필요함**
- **str() 함수를 통해 데이터 프레임, 벡터, 리스트 같은 R 데이터의 구조를 파악함**
  - 데이터 사전의 기초 윤관을 생성할 때 str()함수를 활용함
  - 관측 개수, 변수의 수 확인
  - 각 변수가 숫자, 문자 변수인지 확인
  - 각 변수가 가지는 값들이 무엇인지 확인

1. 데이터 탐색의 개요 및 소개

■ 데이터 구조 탐색

```
> str(health)
spec_tbl_df [18,268 x 24] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ ID              : num [1:18268] 1 2 3 4 5 6 7 8 9 10 ...
 $ Provance        : chr [1:18268] "경상남북도" "전라남북도" "강원도" "서울경기" ...
 $ retention       : num [1:18268] 1.2 5.8 8.7 6.4 1.3 6.9 4.4 6.2 9.6 6.1 ...
 $ renewal         : num [1:18268] 0 0 0 0 0 1 1 0 1 0 ...
 $ product_type    : chr [1:18268] "기본" "중급" "고급" "기본" ...
 $ edu             : chr [1:18268] "학사" "학사" "학사" "학사" ...
 $ work            : chr [1:18268] "고용" "비고용" "고용" "비고용" ...
 $ gender          : num [1:18268] 0 0 0 1 1 0 0 1 1 0 ...
 $ income          : num [1:18268] 51360000 0 44510000 0 40010000 ...
 $ residence       : chr [1:18268] "도시근교" "도시근교" "도시근교" "도시근교" ...
 $ marrage         : chr [1:18268] "기혼" "미혼" "기혼" "기혼" ...
 $ monthly_premium: num [1:18268] 60 90 100 100 70 60 60 90 70 90 ...
 $ period_claim    : num [1:18268] 31 12 17 17 11 13 0 0 12 16 ...
 $ period_keep     : num [1:18268] 5 42 38 65 44 94 13 68 3 7 ...
 $ n_complaint     : num [1:18268] 0 0 0 0 0 0 0 0 0 0 ...
 $ n_product       : chr [1:18268] "1" "4이상" "2" "4이상" ...
 $ customer_type   : chr [1:18268] "법인" "개인" "개인" "법인" ...
 $ customer_grade : chr [1:18268] "법인3" "개인3" "개인3" "법인2" ...
 $ incentive       : chr [1:18268] "할인" "포인트" "할인" "할인" ...
```

1. 데이터 탐색의 개요 및 소개

■ 데이터 구조 탐색

```
$ channel       : chr [1:18268] "자사조직" "자사조직" "자사조직" "콜센터" ...
$ claim_size    : num [1:18268] 330 1030 490 460 120 140 290 320 440 370 ...
$ house_type    : chr [1:18268] "일반" "일반" "일반" "일반" ...
$ house_size    : chr [1:18268] "중" "중" "중" "중" ...
$ age           : num [1:18268] 20 50 80 40 30 60 40 50 80 50 ...
```

2. 변수 속성 정의

■ 전체 변수 탐색

```
> summary(health)
       ID          Provance            retention        renewal       product_type
 Min.   :   1   Length:18268       Min.   :-0.3   Min.   :0.0000   Length:18268
 1st Qu.:4568   Class :character   1st Qu.: 2.5   1st Qu.:0.0000   Class :character
 Median :9134   Mode  :character   Median : 5.0   Median :0.0000   Mode  :character
 Mean   :6851                      Mean   : 5.0   Mean   :0.1432
 3rd Qu.:9134                      3rd Qu.: 7.5   3rd Qu.:0.0000
 Max.   :9134                      Max.   :10.3   Max.   :1.0000
     edu                work                gender          income
 Length:18268       Length:18268       Min.   :0.00   Min.   :        0
 Class :character   Class :character   1st Qu.:0.00   1st Qu.:        0
 Mode  :character   Mode  :character   Median :0.00   Median : 37390000
                                       Mean   :0.49   Mean   : 42552844
                                       3rd Qu.:1.00   3rd Qu.: 68380000
                                       Max.   :1.00   Max.   :134700000
  residence           marrage          monthly_premium   period_claim   period_keep
 Length:18268       Length:18268       Min.   : 60.00   Min.   : 0.0   Min.   : 0.00
 Class :character   Class :character   1st Qu.: 70.00   1st Qu.: 6.0   1st Qu.:24.00
 Mode  :character   Mode  :character   Median : 80.00   Median :14.0   Median :48.00
                                       Mean   : 94.69   Mean   :15.1   Mean   :48.06
                                       3rd Qu.:110.00   3rd Qu.:23.0   3rd Qu.:71.00
                                       Max.   :320.00   Max.   :36.0   Max.   :99.00
```

2. 변수 속성 정의

■ 수치 변수 탐색

```
n_complaint      n_product          customer_type      customer_grade
Min.   :0.0000   Length:18268       Length:18268       Length:18268
1st Qu.:0.0000   Class :character   Class :character   Class :character
Median :0.0000   Mode  :character   Mode  :character   Mode  :character
Mean   :0.3844
3rd Qu.:0.0000
Max.   :5.0000
 incentive          channel            claim_size      house_type
Length:18268       Length:18268       Min.   :   0.0   Length:18268
Class :character   Class :character   1st Qu.: 270.0   Class :character
Mode  :character   Mode  :character   Median : 390.0   Mode  :character
                                      Mean   : 446.6
                                      3rd Qu.: 580.0
                                      Max.   :3040.0
 house_size            age
Length:18268       Min.   :20.0
Class :character   1st Qu.:30.0
Mode  :character   Median :50.0
```

2. 변수 속성 정의

■ 수치 변수 탐색

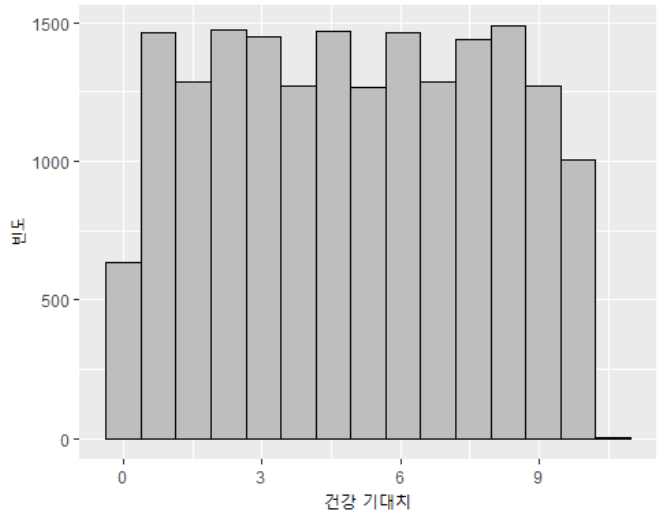
- **summary()를 이용한 요약 통계 확인**
- **평균과 중앙값에 의한 중심 경향 측정**
- **사분위수와 다섯숫자 요약(five number summary)에 의한 산포도(퍼짐 정도) 측정**
  - 다섯숫자 요약: 최솟값, 1사분위, 중앙값, 3사분위, 최댓값
  - 최솟값과 최댓값의 차이인 범위: range(), diff() 함수 이용
  - 사분위수 범위(interquartile range, IQR): IQR() 함수 이용
  - 분위수 계산: quantile() 함수 이용
- **수치변수 시각화**
  - 수치변수의 시각화는 데이터를 진단하는 데 도움이 됨
  - 상자그림(boxplot)과 상자수염그림(box and whiskers plot)

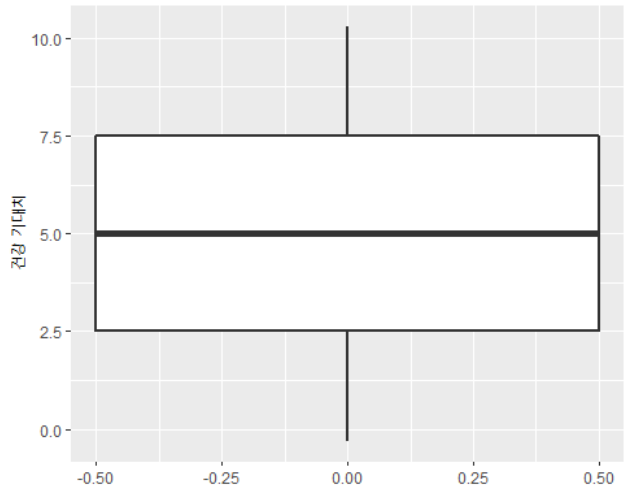
2. 변수 속성 정의

■ 수치 변수 탐색

- 건강기대치 **(retention)**

```
>  summary(health$retention)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   -0.3     2.5     5.0     5.0     7.5    10.3
> health %>% ggplot(aes(x=retention)) +
  geom_histogram(aes(y=..count..), bins=15, color= 'black' , fill= 'green') +
  xlab( "건강 기대치") + ylab( "빈도")
> health %>% ggplot(aes(y=retention)) +
           geom_boxplot(width = 1, lwd=1, outlier.color = "red") +
           ylab( "건강 기대치")
```





2. 변수 속성 정의

■ 수치 변수 탐색

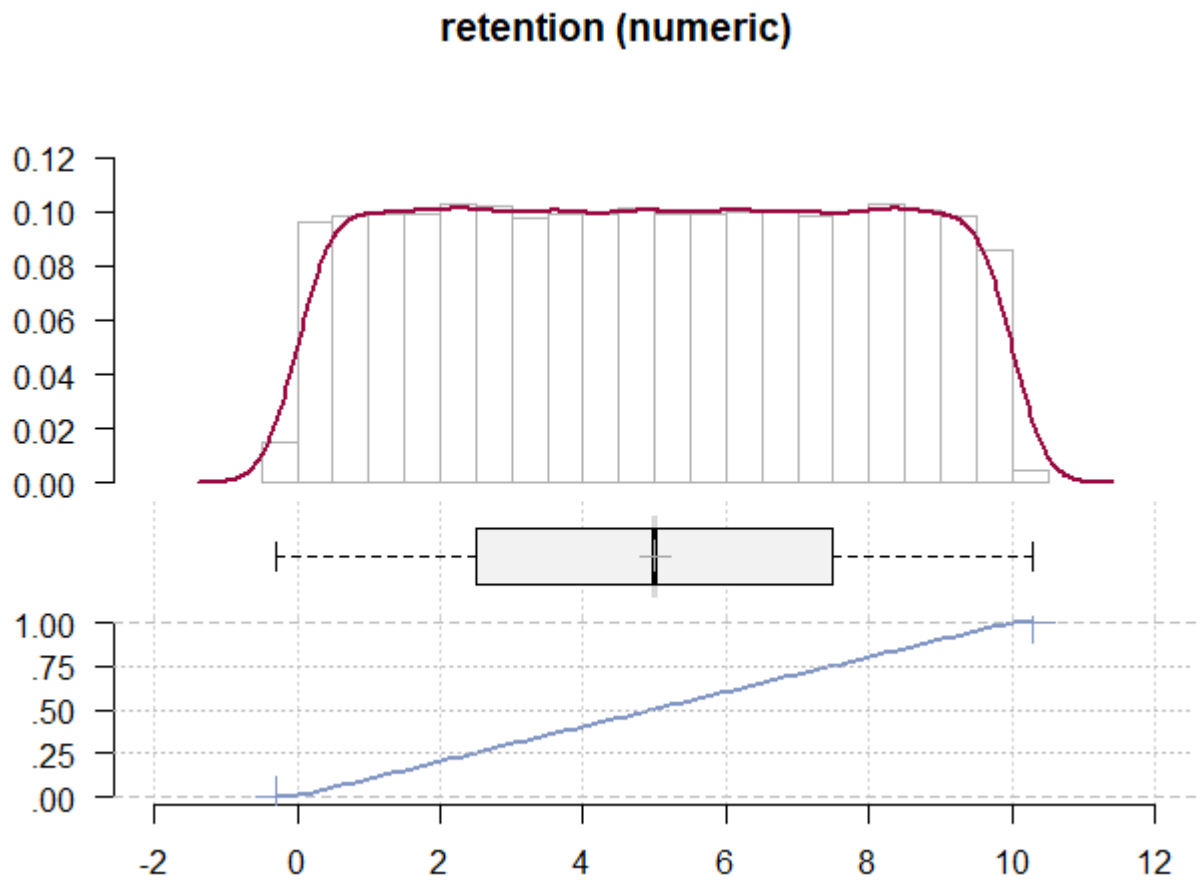
- 건강기대치 **(retention)**

```
> library(DescTools)
> health %>% with(Desc(retention))
-------------------------------------------------------------------------------

health$retention (numeric)

  length       n    NAs  unique     0s  mean  meanCI'
  18'268  18'268      0     107     89  5.00    4.96
          100.0%   0.0%          0.5%          5.04

     .05     .10     .25  median    .75   .90     .95
    0.50    1.00    2.50    5.00   7.50  9.00    9.50

   range      sd  vcoef     mad   IQR  skew    kurt
   10.60    2.89   0.58    3.71  5.00  0.00   -1.20

lowest : -0.30 (10), -0.20 (17), -0.10 (22), 0.0 (89), 0.1 (148)
highest: 9.9 (176), 10.0 (86), 10.1 (23), 10.2 (16), 10.3 (3)

' 95%-CI (classic)
```

2. 변수 속성 정의

■ 수치 변수 탐색

- 건강기대치 **(retention)**

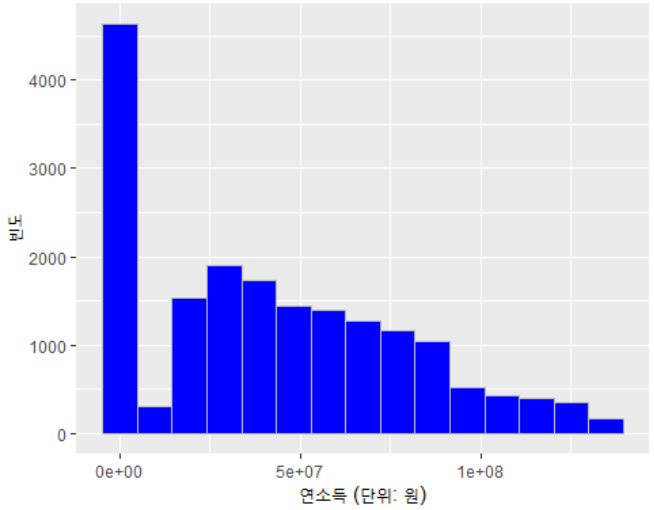


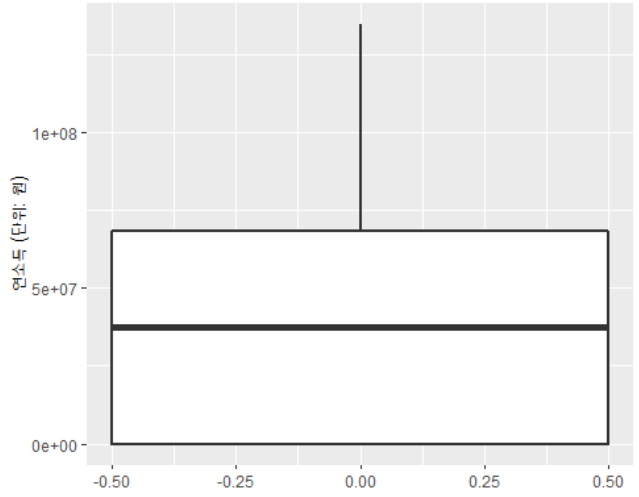
2. 변수 속성 정의

■ 수치 변수 탐색

- 연소득 **(income)**

```
> summary(health$income)
   Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
      0          0   37390000   42552844   68380000  134700000
> health %>% ggplot(aes(x=income)) +
             geom_histogram(aes(y=..count..), bins=15, color= 'grey', fill= 'blue') +
             xlab( "연소득 (단위: 원)") + ylab( "빈도")
> health %>% ggplot(aes(y=retention)) +
             geom_boxplot(width = 1, lwd=1, outlier.color = "red") +
             ylab( "연소득 (단위: 원)")
```





2. 변수 속성 정의

■ 수치 변수 탐색

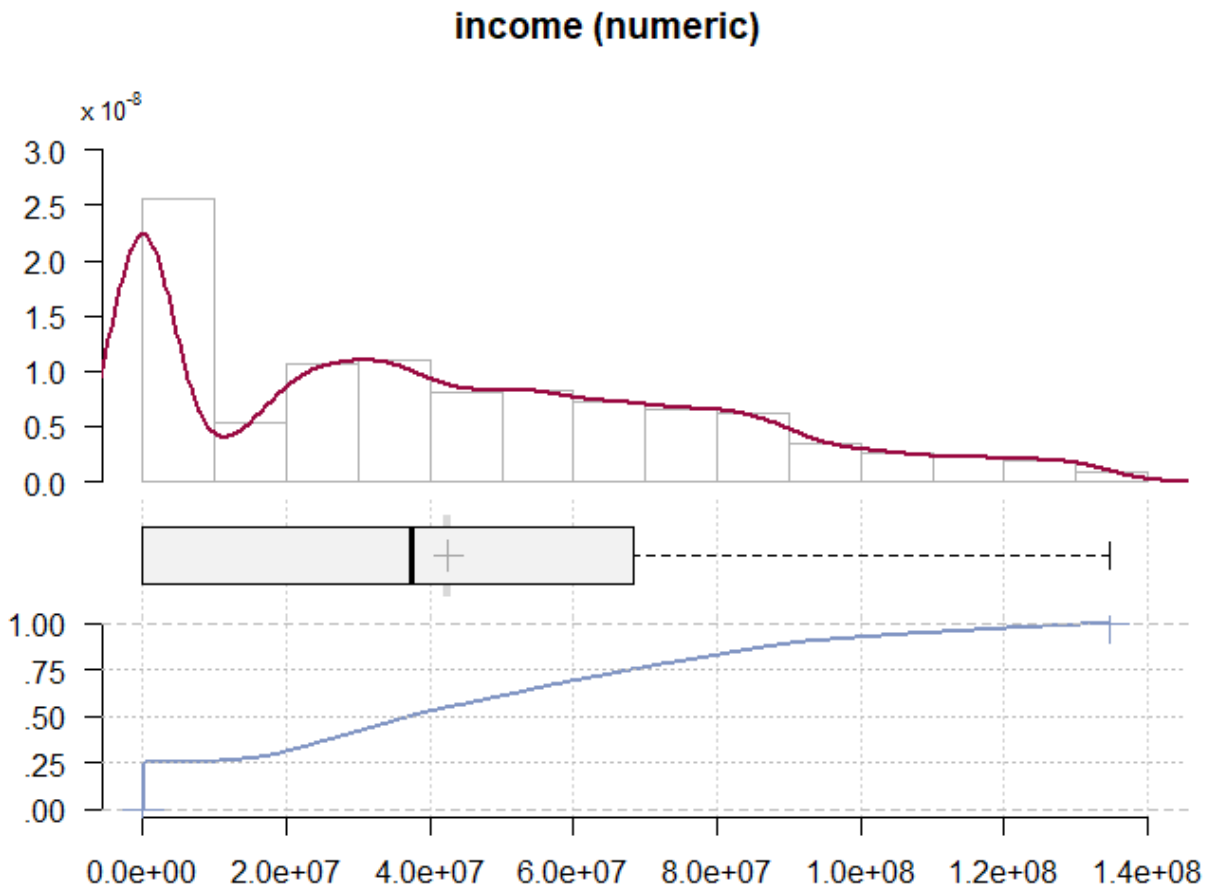
▪ 연소득 (income)

```
> health %>% with(Desc(income))
──────────────────────────────────────────────────────────────────────────────────────────────────
income (numeric)

     length         n     NAs    unique        0s      mean    meanCI'
     18'268    18'268       0     7'053     4'634  4.26e+07  4.20e+07
               100.0%    0.0%                25.4%            4.31e+07

        .05       .10     .25    median       .75       .90       .95
       0.00      0.00    0.00  3.74e+07  6.84e+07  9.23e+07  1.11e+08

      range        sd   vcoef       mad       IQR      skew      kurt
   1.35e+08  3.59e+07    0.84  4.67e+07  6.84e+07      0.52     -0.62

lowest : 0.0 (4'634), 9'160'000.0, 9'190'000.0, 9'220'000.0 (2), 9'260'000.0
highest: 1.35e+08 (7), 1.35e+08 (2), 1.35e+08, 1.35e+08 (2), 1.35e+08

heap(?): remarkable frequency (25.4%) for the mode(s) (= 0)

' 95%-CI (classic)
```

2. 변수 속성 정의

■ 수치 변수 탐색

- 연소득 **(income)**

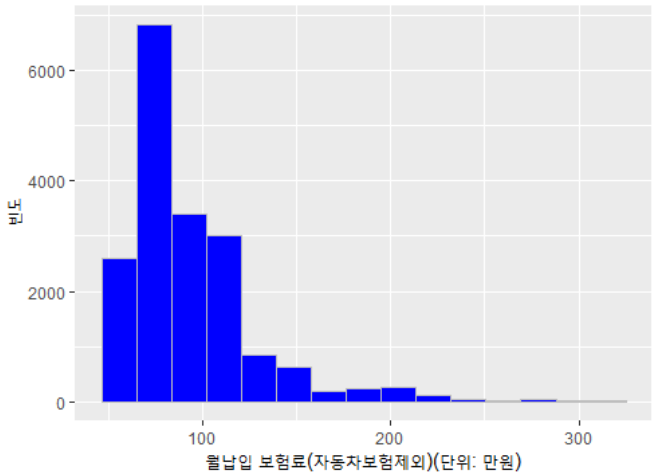


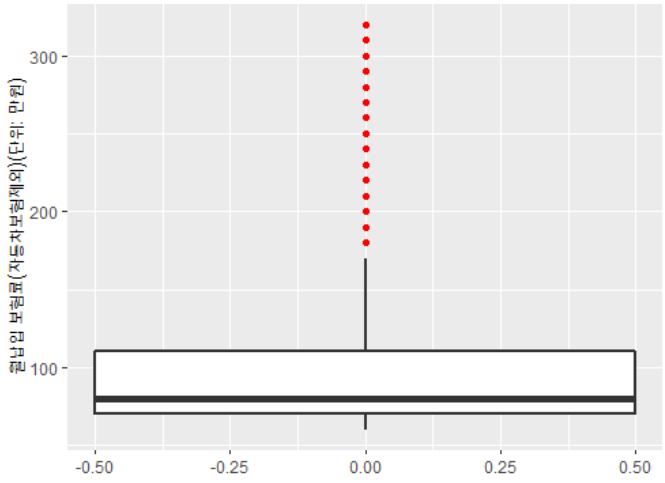
2. 변수 속성 정의

■ 수치 변수 탐색

- **월납입 보험료(자동차보험제외, 단위: 만원) (monthly_premium)**

```
> summary(health$monthly_premium)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  60.00   70.00   80.00   94.69  110.00  320.00
> health %>% ggplot(aes(x=monthly_premium)) +
             geom_histogram(aes(y=..count..), bins=15, color= 'grey', fill= 'blue') +
             xlab( "월납입 보험료(자동차보험제외)(단위: 만원)") + ylab( "빈도")
> health %>% ggplot(aes(y=monthly_premium)) +
             geom_boxplot(width = 1, lwd=1, outlier.color = "red") +
             ylab( "월납입 보험료(자동차보험제외)(단위: 만원)")
```





2. 변수 속성 정의

■ 수치 변수 탐색

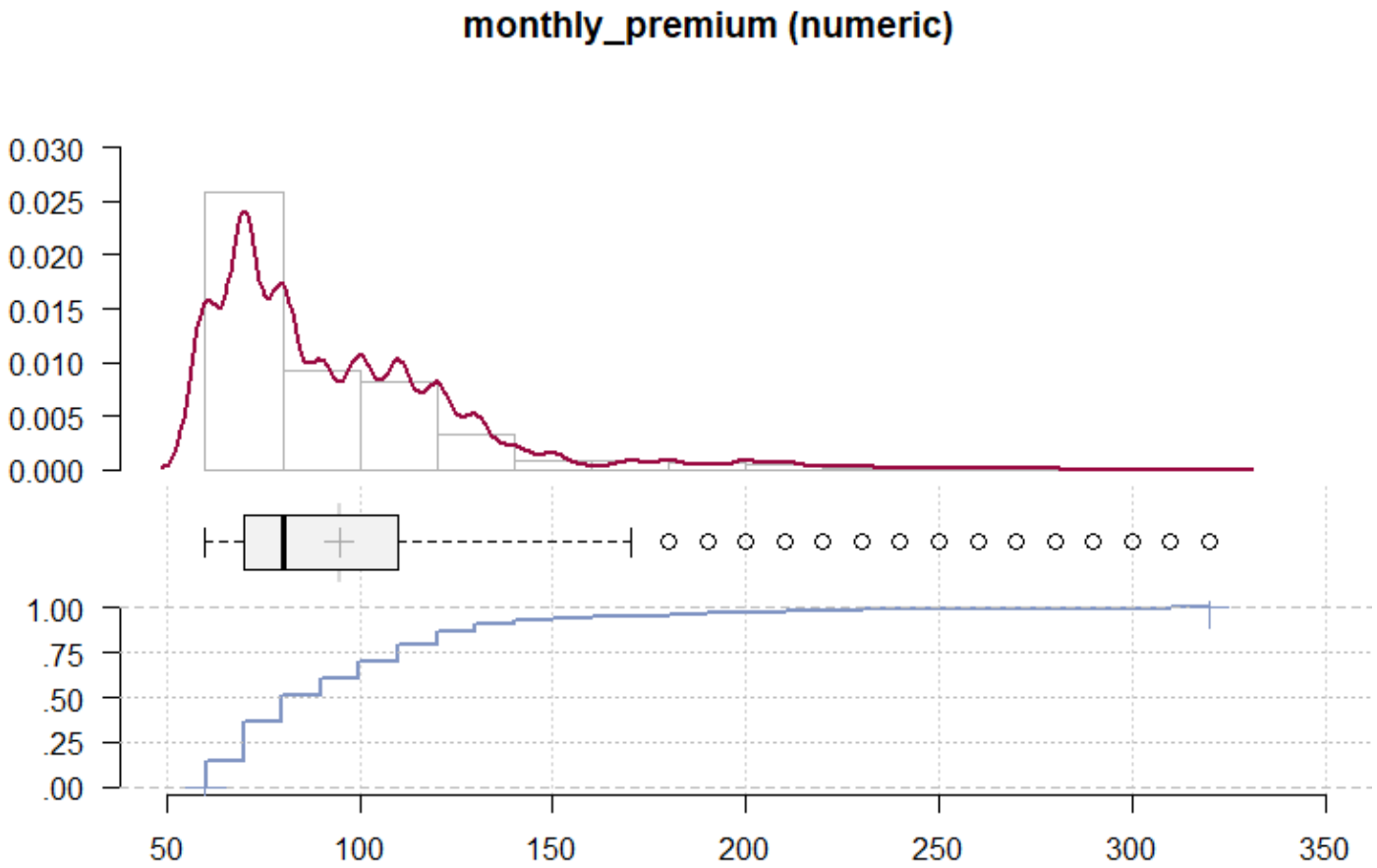
- **월납입 보험료(자동차보험제외, 단위: 만원) (monthly_premium)**

```
> health %>% with(Desc(monthly_premium))
------------------------------------------------------------------------------------------------------------
monthly_premium (numeric)

  length       n    NAs  unique      0s    mean  meanCI'
  18'268  18'268      0      27       0   94.69   94.18
          100.0%   0.0%             0.0%           95.21

     .05     .10     .25  median     .75     .90     .95
   60.00   60.00   70.00   80.00  110.00  130.00  170.00

   range      sd   vcoef     mad     IQR    skew    kurt
  260.00   35.69    0.38   29.65   40.00    2.09    6.23

lowest : 60.0 (2'585), 70.0 (4'006), 80.0 (2'821), 90.0 (1'633), 100.0 (1'759)
highest: 280.0 (15), 290.0 (9), 300.0 (9), 310.0 (13), 320.0 (7)

heap(?): remarkable frequency (21.9%) for the mode(s) (= 70)

' 95%-CI (classic)
```

2. 변수 속성 정의

■ 수치 변수 탐색

- **월납입 보험료(자동차보험제외, 단위: 만원) (monthly_premium)**

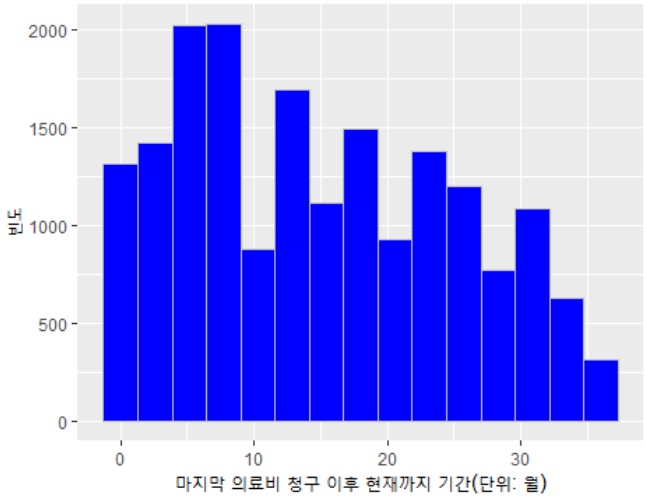


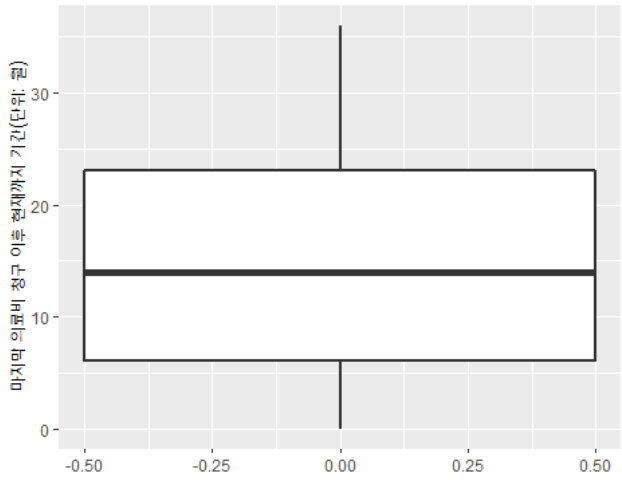
2. 변수 속성 정의

■ 수치 변수 탐색

- 마지막 의료비 청구 이후 현재까지 기간(단위: 월) **(period_claim)**

```
> summary(health$monthly_premium)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  60.00   70.00   80.00   94.69  110.00  320.00
> health %>% ggplot(aes(x=period_claim)) +
             geom_histogram(aes(y=..count..), bins=15, color= 'grey', fill= 'blue') +
             xlab( "마지막 의료비 청구 이후 현재까지 기간(단위: 월)") + ylab( "빈도")
> health %>% ggplot(aes(y=period_claim)) +
             geom_boxplot(width = 1, lwd=1, outlier.color = "red") +
             ylab( "마지막 의료비 청구 이후 현재까지 기간(단위: 월)")
```





2. 변수 속성 정의

■ 수치 변수 탐색

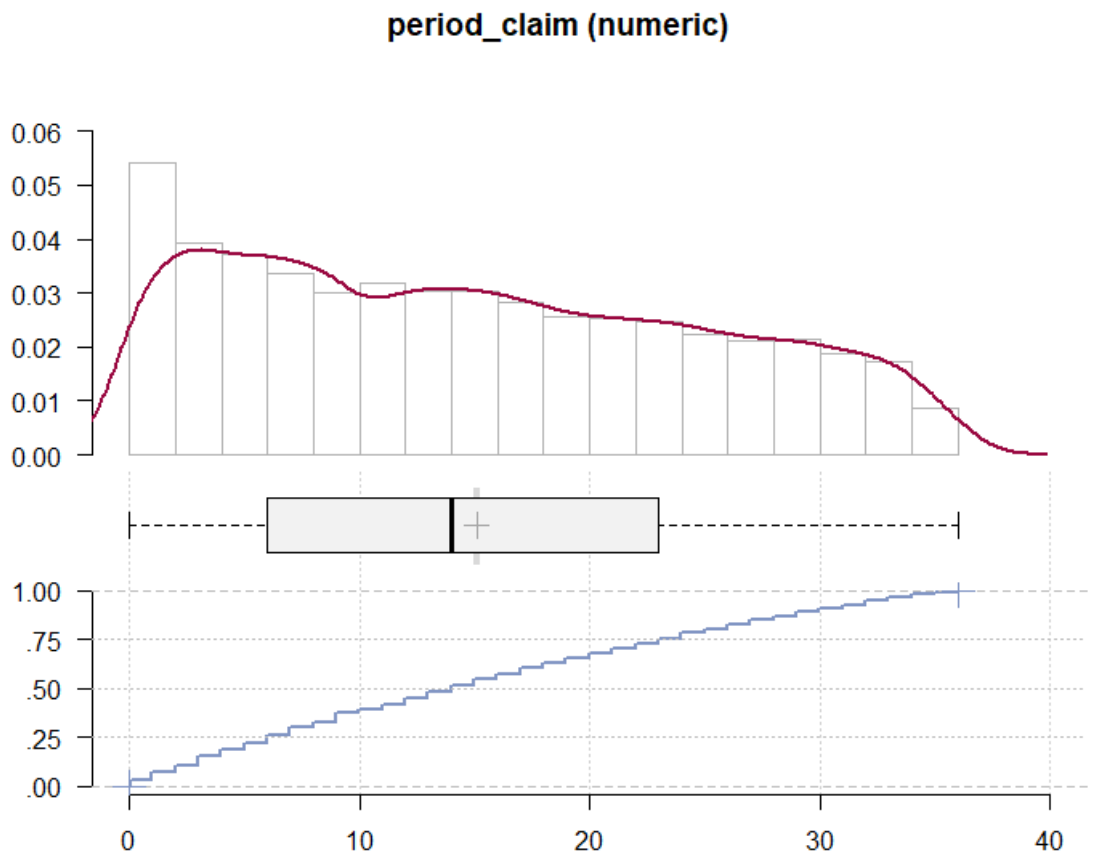
- 마지막 의료비 청구 이후 현재까지 기간(단위: 월) **(period_claim)**

```
> health %>% with(Desc(period_claim))
-------------------------------------------------------------------------------
period_claim (numeric)

  length       n    NAs  unique     0s   mean  meanCI'
  18'268  18'268      0      37    628  15.10   14.95
          100.0%   0.0%           3.4%         15.24

     .05     .10     .25  median    .75    .90     .95
    1.00    2.00    6.00   14.00  23.00  30.00   33.00

   range      sd   vcoef     mad     IQR   skew    kurt
   36.00   10.10    0.67   11.86   17.00   0.29   -1.07

lowest : 0.0 (628), 1.0 (690), 2.0 (658), 3.0 (762), 4.0 (670)
highest: 32.0 (369), 33.0 (307), 34.0 (321), 35.0 (169), 36.0 (142)

' 95%-CI (classic)
```

2. 변수 속성 정의

■ 수치 변수 탐색

- 마지막 의료비 청구 이후 현재까지 기간(단위: 월) **(period_claim)**

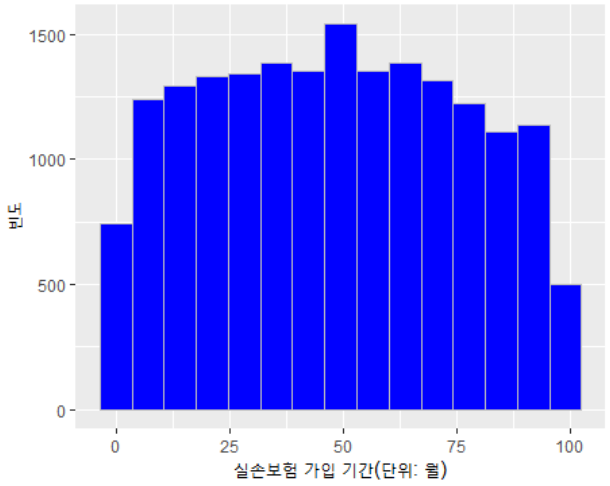


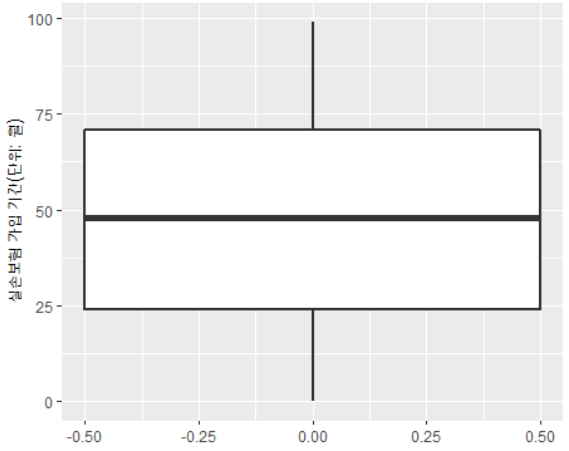
2. 변수 속성 정의

■ 수치 변수 탐색

- 실손보험 가입 기간(단위 월) **(period_keep)**

```
> summary(health$period_keep)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00   24.00   48.00   48.06   71.00   99.00
> health %>% ggplot(aes(x=period_keep)) +
             geom_histogram(aes(y=..count..), bins=15, color= 'grey', fill= 'blue') +
             xlab( "실손보험 가입 기간(단위: 월)") + ylab( "빈도")
> health %>% ggplot(aes(y=period_keep)) +
             geom_boxplot(width = 1, lwd=1, outlier.color = "red") +
             ylab( "실손보험 가입 기간(단위: 월)")
```





2. 변수 속성 정의

■ 수치 변수 탐색

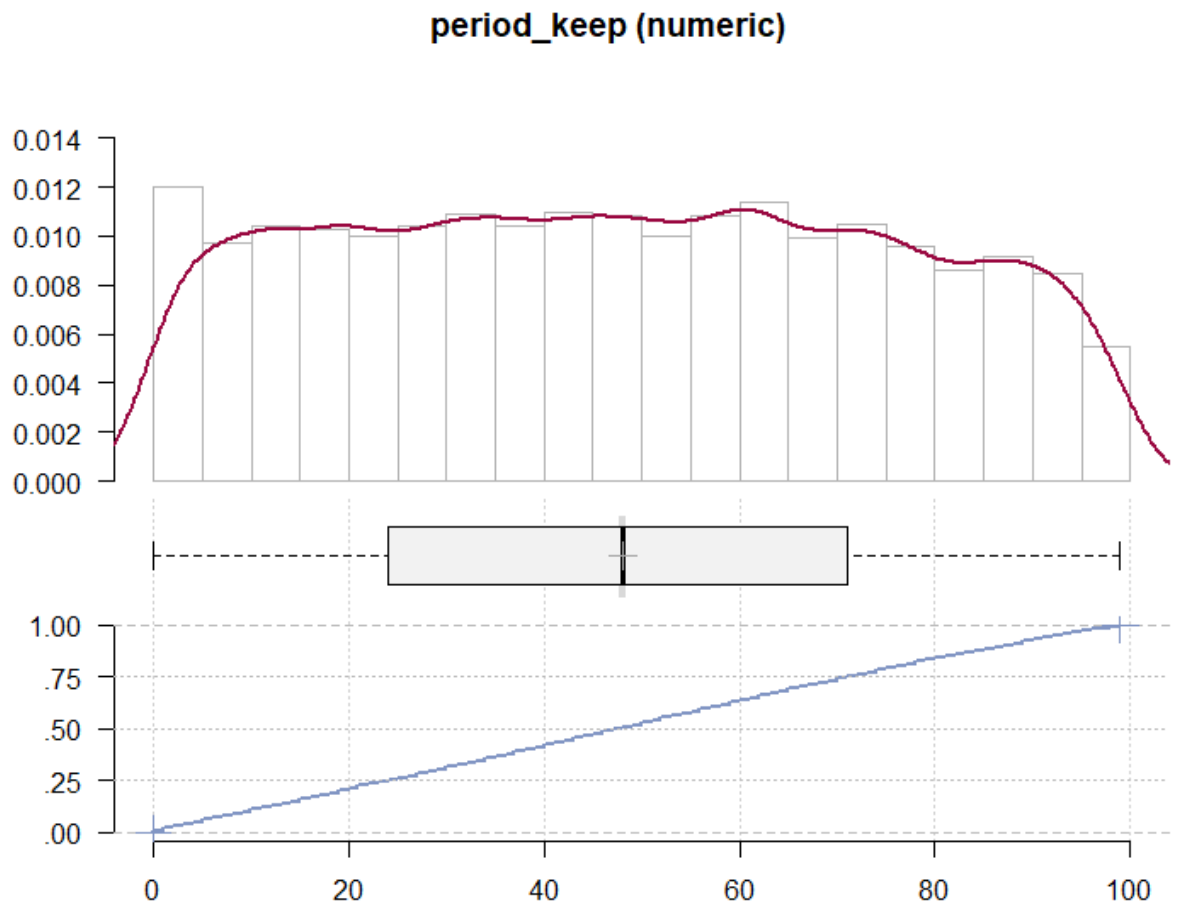
- 실손보험 가입 기간(단위 월) **(period_keep)**

```
> health %>% with(Desc(period_keep))
------------------------------------------------------------------------------------------------------------------------

period_keep (numeric)

  length       n    NAs  unique     0s   mean  meanCI'
  18'268  18'268      0     100    166  48.06   47.66
          100.0%   0.0%           0.9%          48.47

     .05     .10     .25  median    .75    .90     .95
    4.00   10.00   24.00   48.00  71.00  87.00   93.00

   range      sd   vcoef     mad    IQR   skew    kurt
   99.00   27.91    0.58   35.58  47.00   0.04   -1.13

lowest : 0.0 (166), 1.0 (170), 2.0 (178), 3.0 (228), 4.0 (182)
highest: 95.0 (154), 96.0 (134), 97.0 (104), 98.0 (108), 99.0 (156)

' 95%-CI (classic)
```

2. 변수 속성 정의

■ 수치 변수 탐색

- 실손보험 가입 기간(단위 월) **(period_keep)**

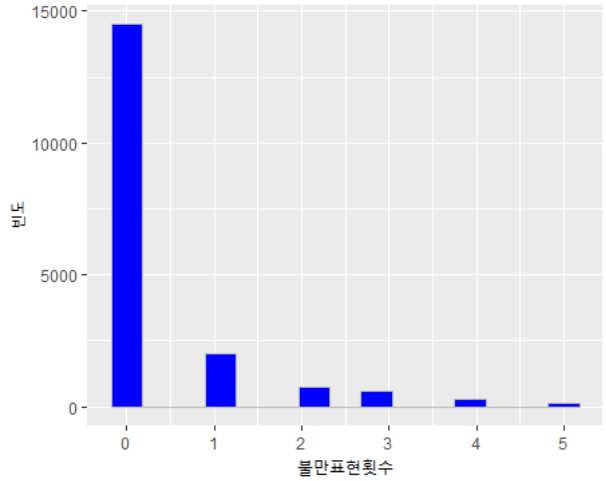


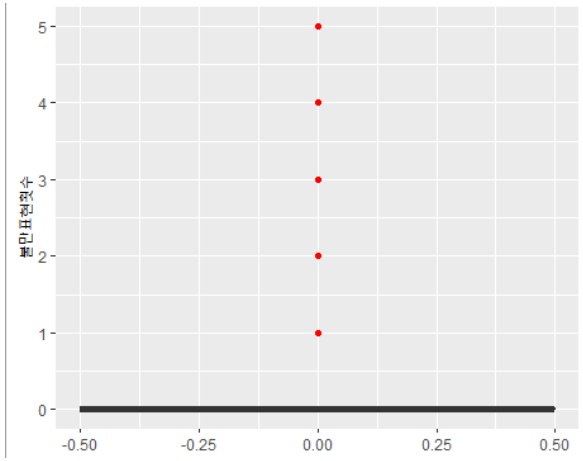
2. 변수 속성 정의

■ 수치 변수 탐색

▪ 불만표현횟수 (n_complaint)

```
> summary(health$n_complaint)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0000  0.0000  0.0000  0.3844  0.0000  5.0000
> health %>% ggplot(aes(x=health$n_complaint)) +
             geom_histogram(aes(y=..count..), bins=15, color= 'grey', fill= 'blue') +
             xlab( "불만표현횟수") + ylab( "빈도")
> health %>% ggplot(aes(y=health$n_complaint)) +
             geom_boxplot(width = 1, lwd=1, outlier.color = "red") +
             ylab( "불만표현횟수")
```





2. 변수 속성 정의

■ 수치 변수 탐색

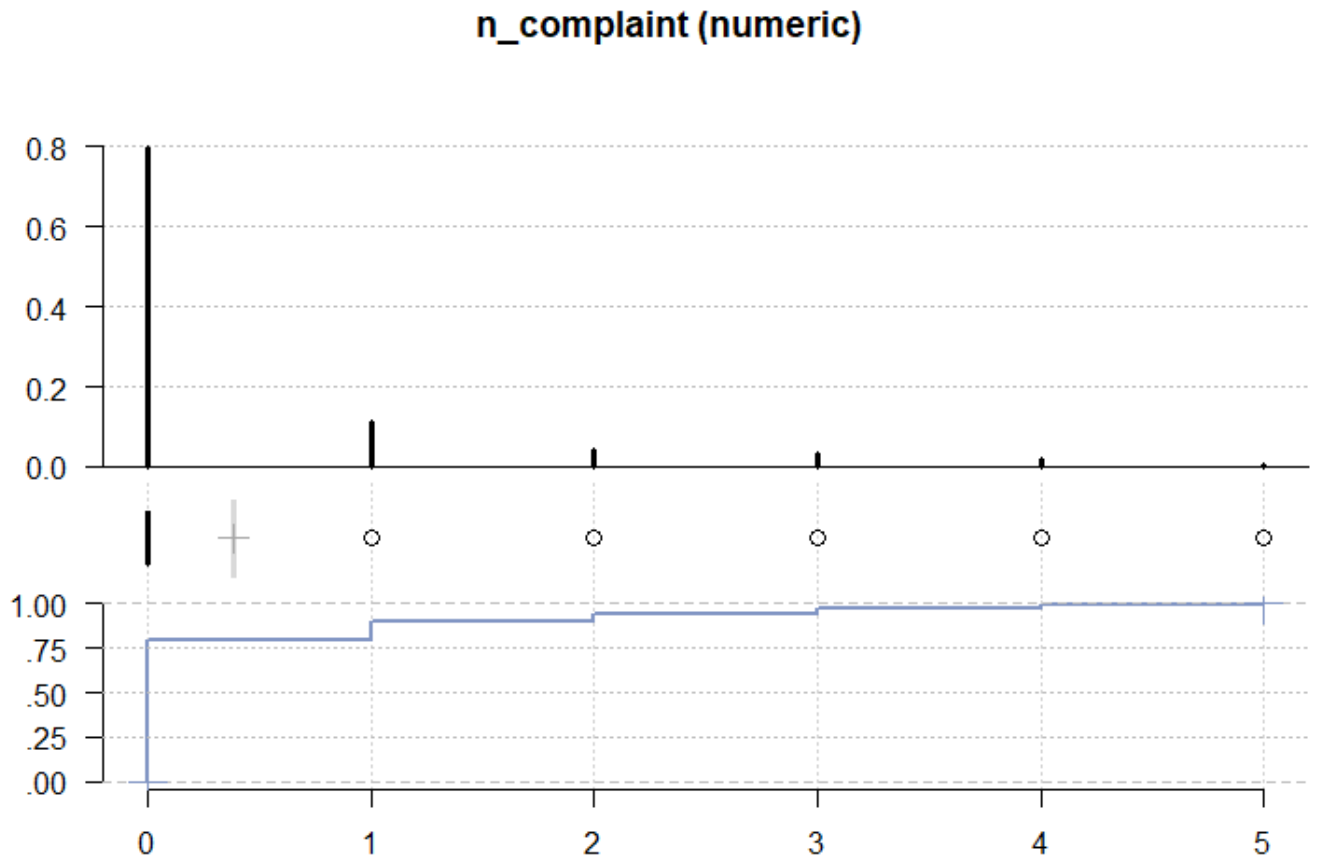
- 불만표현횟수 **(n_complaint)**

```
> health %>% with(Desc(n_complaint))
-------------------------------------------------------------------------------------------------------------
n_complaint (numeric)

  length      n    NAs  unique     0s  mean  meanCI'
  18'268 18'268      0       6 14'504  0.38    0.37
         100.0%   0.0%         79.4%          0.40

     .05     .10     .25  median     .75   .90     .95
    0.00    0.00    0.00    0.00    0.00  1.00    3.00

   range      sd   vcoef     mad     IQR  skew    kurt
    5.00    0.91    2.37    0.00    0.00  2.78    7.74

   level    freq   perc  cumfreq  cumperc
1      0 14'504  79.4%   14'504    79.4%
2      1  2'022  11.1%   16'526    90.5%
3      2    748   4.1%   17'274    94.6%
4      3    584   3.2%   17'858    97.8%
5      4    298   1.6%   18'156    99.4%
6      5    112   0.6%   18'268   100.0%
' 95%-CI (classic)
```

2. 변수 속성 정의

■ 수치 변수 탐색

- 불만표현횟수 **(n_complaint)**

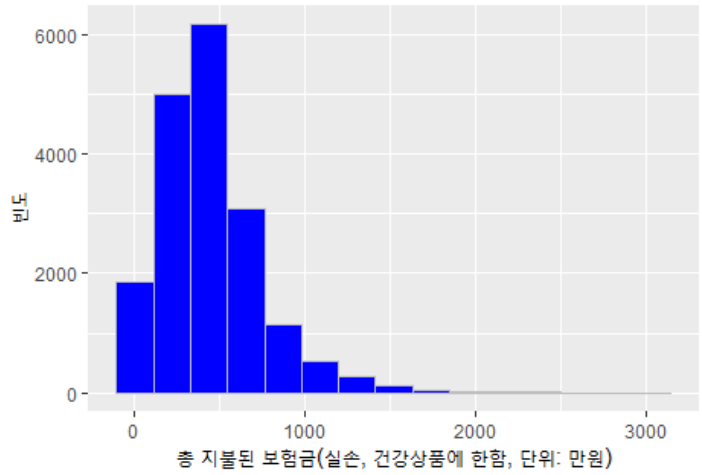


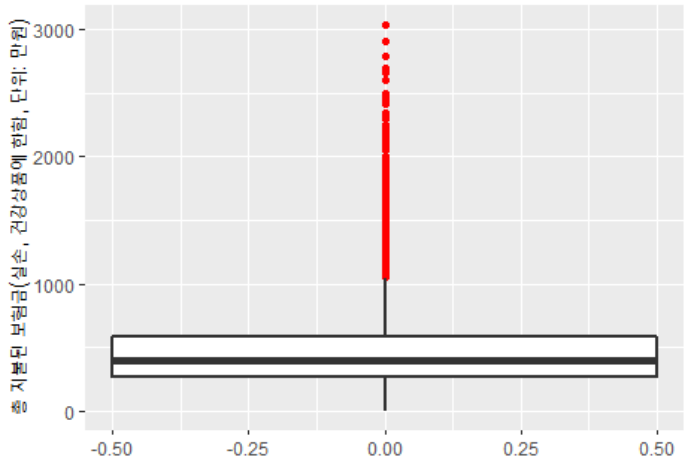
2. 변수 속성 정의

■ 수치 변수 탐색

- **총 지불된 보험금(실손, 건강상품에 한함, 단위: 만원) (claim_size)**

```
> summary(health$claim_size)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    0.0   270.0   390.0   446.6   580.0  3040.0
> health %>% ggplot(aes(x=health$claim_size)) +
             geom_histogram(aes(y=..count..), bins=15, color= 'grey', fill= 'blue') +
             xlab( "총 지불된 보험금(실손, 건강상품에 한함, 단위: 만원)") + ylab( "빈도")
> health %>% ggplot(aes(y=health$claim_size)) +
             geom_boxplot(width = 1, lwd=1, outlier.color = "red") +
             ylab( "총 지불된 보험금(실손, 건강상품에 한함, 단위: 만원)")
```





2. 변수 속성 정의

■ 수치 변수 탐색

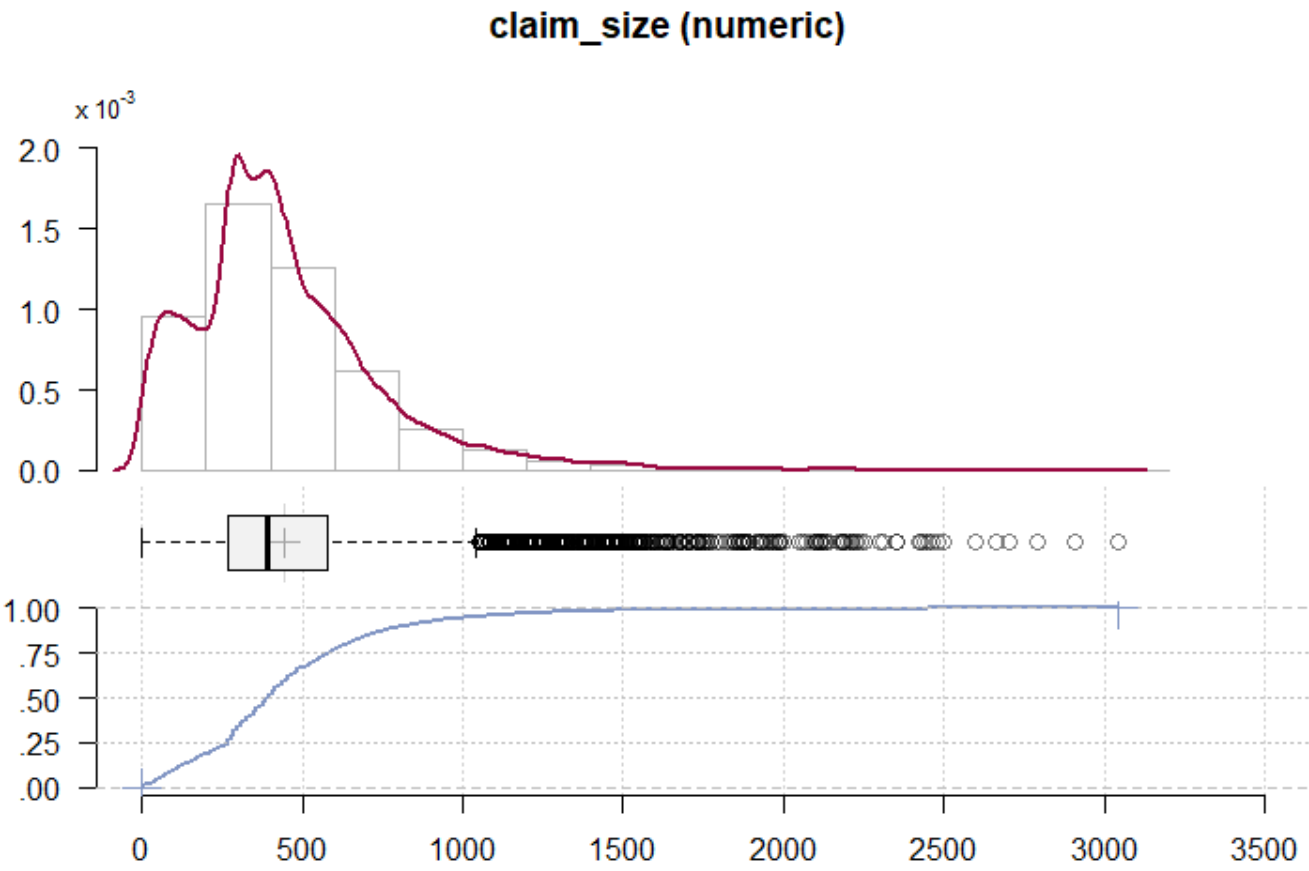
- 총 지불된 보험금(실손, 건강상품에 한함, 단위: 만원) **(claim_size)**

```
> health %>% with(Desc(claim_size))
──────────────────────────────────────────────────────────────────────────────

claim_size (numeric)

     length         n       NAs    unique        0s      mean     meanCI'
     18'268    18'268         0       232        79    446.64     442.21
               100.0%      0.0%              0.4%               451.07

        .05       .10       .25    median       .75       .90        .95
      50.00    100.00    270.00    390.00    580.00    810.00   1'020.00

      range        sd     vcoef       mad       IQR      skew       kurt
   3'040.00    305.58      0.68    237.22    310.00      1.65       5.25

lowest : 0.0 (79), 10.0 (165), 20.0 (160), 30.0 (132), 40.0 (208)
highest: 2'660.0, 2'700.0, 2'790.0, 2'910.0, 3'040.0

' 95%-CI (classic)
```

2. 변수 속성 정의

■ 수치 변수 탐색

- 총 지불된 보험금(실손, 건강상품에 한함, 단위: 만원) **(claim_size)**



2. 변수 속성 정의

■ 범주형 변수 탐색

- **summary()**를 이용한 요약 통계 확인
- 범주의 수 및 비율 등의 분포 확인
  - table()
  - proportion()
- 범주형 변수 시각화
  - 범주들의 빈도와 비중을 확인하는 데 도움이 됨

2. 변수 속성 정의

■ 범주형 변수 탐색

- 재갱신 여부 **(renewal)**

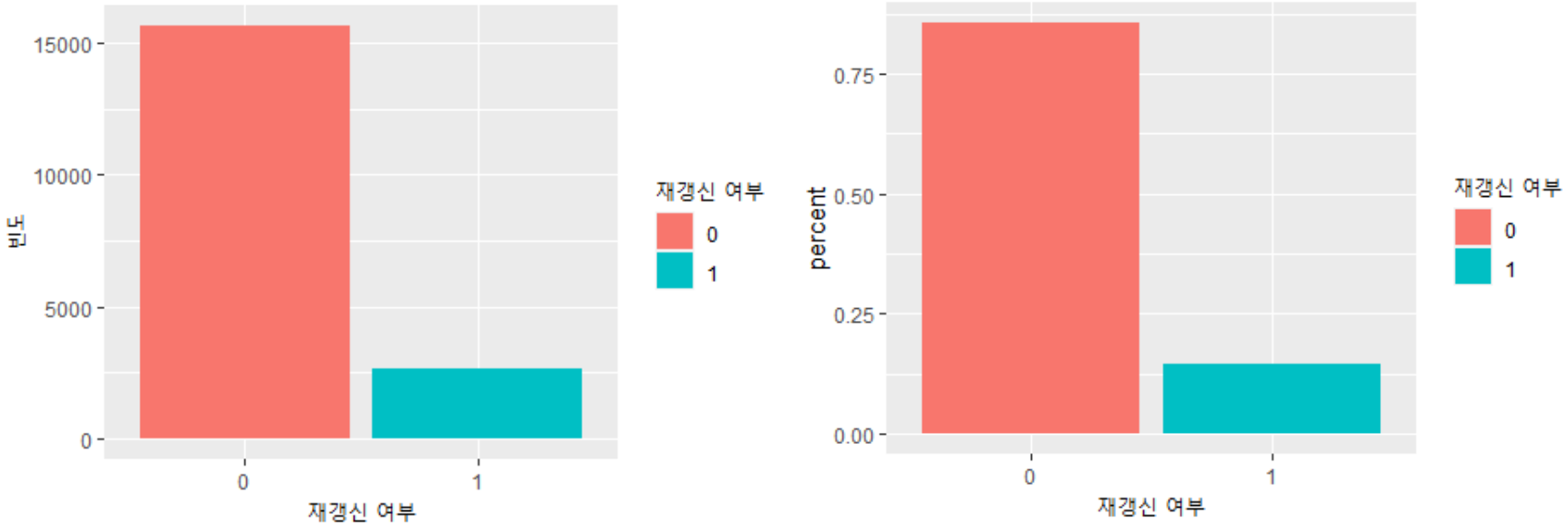
```
> options(digits=4)
> health$renewal = factor(health$renewal)
> summary(health$renewal)
    0     1
15652  2616
> table(health$renewal)

    0     1
15652  2616

> proportions(table(health$renewal))

0      1
0.8568 0.1432

> health %>% with(table(renewal))/NROW(health)
renewal
 0      1
0.8568 0.1432
```

2. 변수 속성 정의

■ 범주형 변수 탐색

- 재갱신 여부 **(renewal)**

```
> health %>% ggplot(aes(x=renewal, fill=renewal)) +
             geom_bar() +
             xlab( "재갱신 여부") + ylab( "빈도") + labs(fill = "재갱신 여부")
> health %>% ggplot(aes(x=renewal, fill=renewal)) +
             geom_bar(aes(y = (..count..)/sum(..count..))) +
             labs(x= "재갱신 여부 “, y= "percent “, fill = "재갱신 여부")
```



2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 담당지역 (Provance)

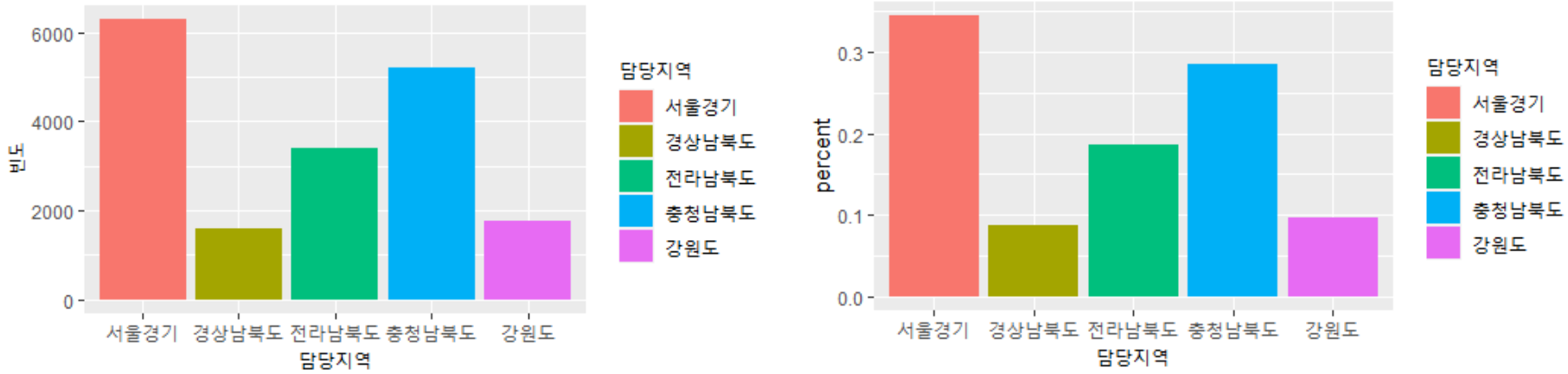
```
> summary(health$Provance)
   Length     Class      Mode
    18268 character character

> table(health$Provance)

    강원도 경상남북도   서울경기 전라남북도 충청남북도
      1764       1596       6300       3406       5202
> health$Provance = factor(health$Provance, levels=c("서울경기", "경상남북도",
                                                     "전라남북도", "충청남북도", "강원도"))
> table(health$Provance)

  서울경기 경상남북도 전라남북도 충청남북도     강원도
      6300       1596       3406       5202       1764
> health %>% with(table(Provance))/NROW(health)
Provance
  서울경기 경상남북도 전라남북도 충청남북도     강원도
   0.34487    0.08737    0.18645    0.28476    0.09656
```

2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 담당지역 (Provance)

```
> health %>% ggplot(aes(x=Provance, fill=Provance)) +
             geom_bar() +
             labs(x = "담당지역", y= "빈도", fill = "담당지역")

> health %>% ggplot(aes(x=Provance, fill=Provance)) +
             geom_bar(aes(y = (..count..)/sum(..count..))) +
             labs(x = "담당지역", y= "percent", fill = "담당지역")
```



2. 변수 속성 정의

■ 범주형 변수 탐색

- 상품 옵션 **(product_type)**

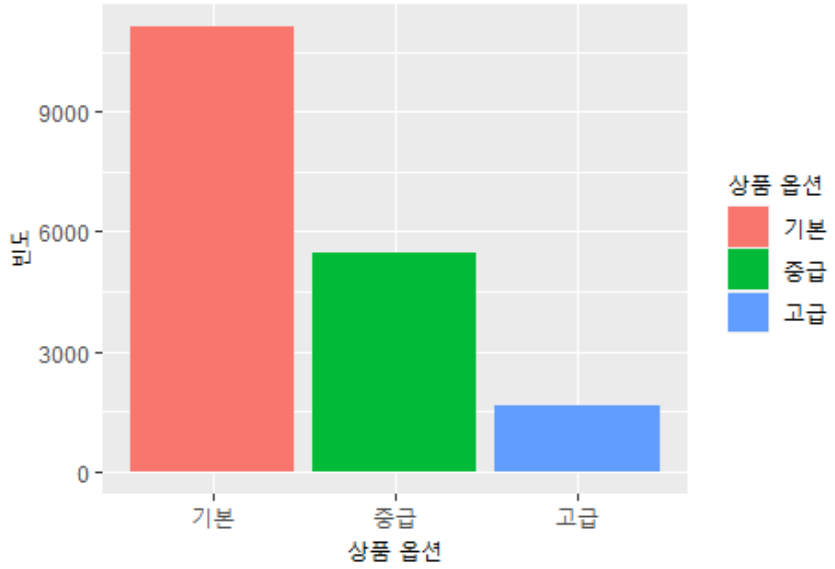
```
> summary(health$product_type)
   Length     Class      Mode
    18268 character character
> health$product_type = factor(health$product_type, levels=c( "기본", "중급", "고급"))
> summary(health$product_type)
 기본  중급  고급
11136  5484  1648
> table(health$product_type)

 기본  중급  고급
11136  5484  1648
> proportions(table(health$product_type))

 기본    중급    고급
0.60959 0.30020 0.09021
> health %>% with(table(product_type))/NROW(health)
product_type
 기본    중급    고급
0.60959 0.30020 0.09021
```

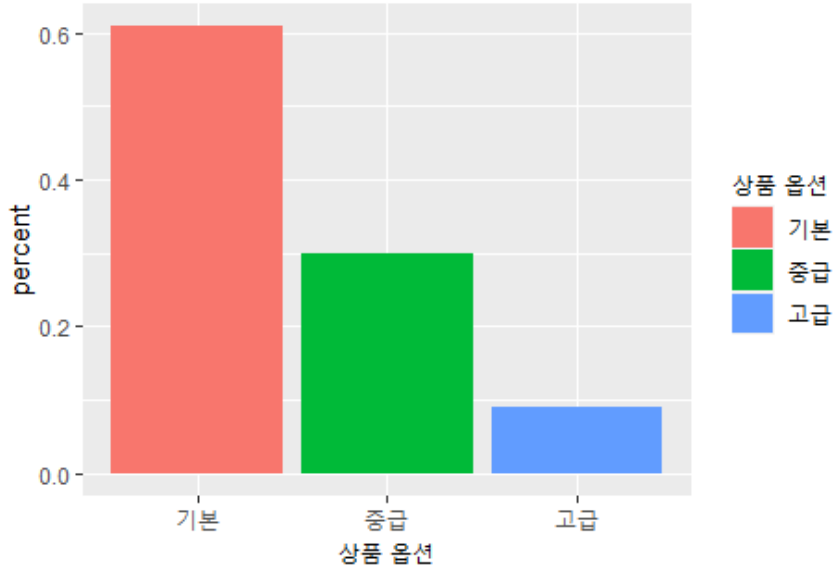
2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 상품 옵션 (product_type)

```
> health %>% ggplot(aes(x=product_type, fill=product_type)) +
           geom_bar() +
           xlab( "상품 옵션") + ylab( "빈도") + labs(fill = "상품 옵션")
> health %>% ggplot(aes(x=renewal, fill=product_type)) +
           geom_bar(aes(y = (..count..)/sum(..count..))) +
           xlab( "상품 옵션") + ylab( "percent") + labs(fill = "상품 옵션")
```





2. 변수 속성 정의

■ 범주형 변수 탐색

- 교육 정도 **(edu)**

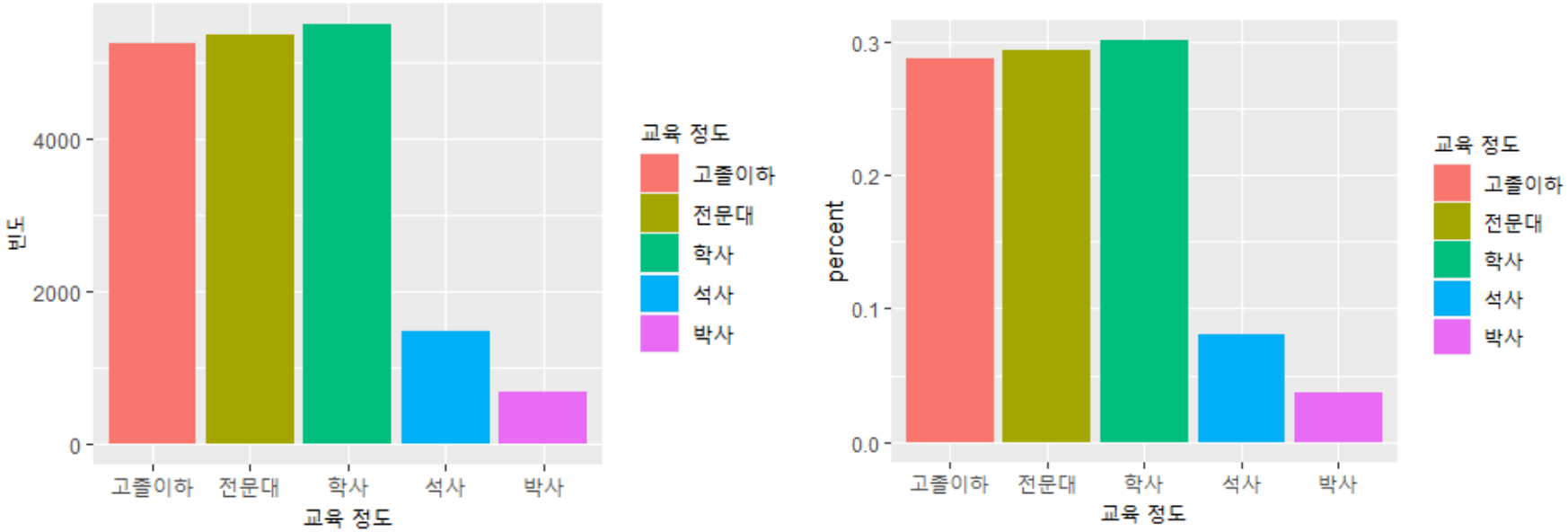
```
> summary(health$edu)
   Length     Class      Mode
    18268 character character
> table(health$edu)

고졸이하      박사      석사    전문대      학사
    5244       684      1482      5362      5496
> health$edu = factor(health$edu, levels=c( "고졸이하", "전문대", "학사", "석사", "박사"))
> table(health$edu)

고졸이하    전문대      학사      석사      박사
    5244      5362      5496      1482       684

> health %>% with(table(edu))/NROW(health)
edu
고졸이하    전문대      학사      석사      박사
 0.28706   0.29352   0.30085   0.08113   0.03744
```

2. 변수 속성 정의

■ 범주형 변수 탐색

- 교육 정도 **(edu)**

```
> health %>% ggplot(aes(x=edu, fill=edu)) +
           geom_bar() +
           xlab( "교육 정도") + ylab( "빈도") + labs(fill = "교육 정도")
health %>% ggplot(aes(x=edu, fill=edu)) +
           geom_bar(aes(y = (..count..)/sum(..count..))) +
           xlab( "교육 정도") + ylab( "percent") + labs(fill = "교육 정도")
```



2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 고용상태 (work)

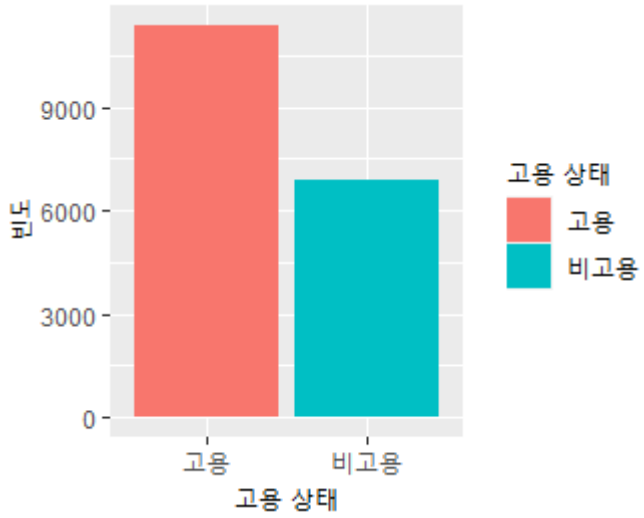
```
> table(health$work)

  고용 비고용
 11396    6872

> health$marrage = factor(health$work)

> health %>% with(table(work))/NROW(health)
work
  고용 비고용
0.6238 0.3762
```

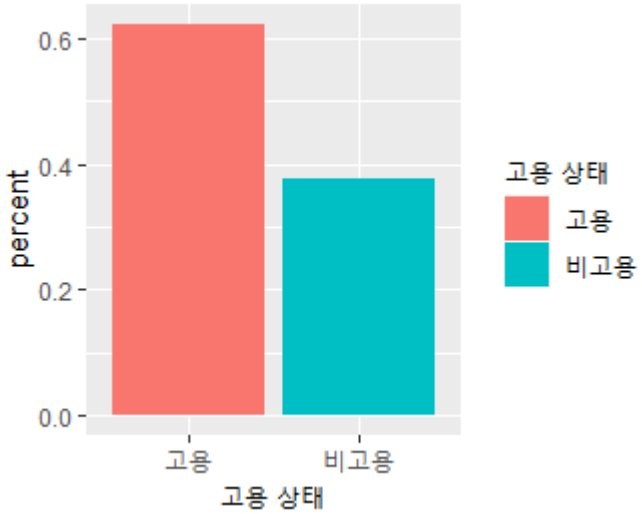
2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 고용상태 (work)

```
> health %>% ggplot(aes(x=work, fill=work)) +
             geom_bar() +
            labs(x= "고용 상태", y="빈도", fill = "고용 상태")
> health %>% ggplot(aes(x=work, fill=work)) +
             geom_bar(aes(y = (..count..)/sum(..count..))) +
             labs(x = "고용 상태", y = "percent", fill = "고용 상태")
```





2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 성별 (gender)

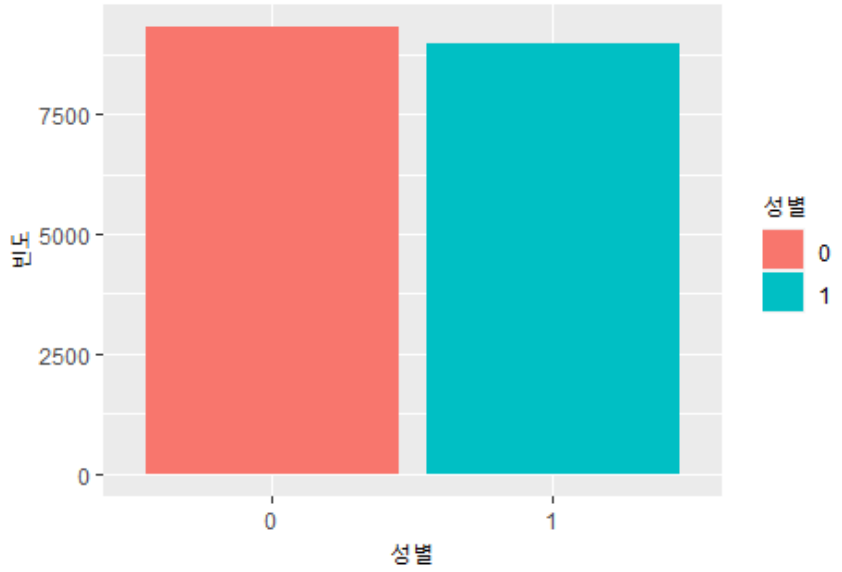
```
> summary(health$gender)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00    0.00    0.00    0.49    1.00    1.00
> health$gender = factor(health$gender, levels=c(0, 1))
> table(health$gender)

   0    1
9316 8952

> health %>% with(table(gender))/NROW(health)
gender
   0    1
0.51 0.49
```

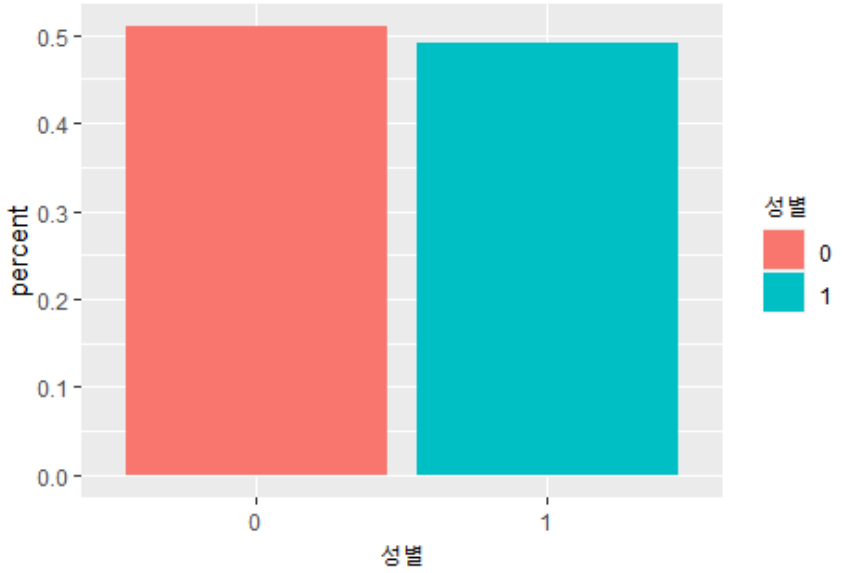
2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 성별 (gender)

```
> health %>% ggplot(aes(x=gender, fill=gender)) +
           geom_bar() +
           xlab( "성별") + ylab( "빈도") + labs(fill = "성별")
> health %>% ggplot(aes(x=gender, fill=gender)) +
           geom_bar(aes(y = (..count..)/sum(..count..))) +
           xlab( "성별") + ylab("percent") + labs(fill = "성별")
```





2. 변수 속성 정의

■ 범주형 변수 탐색

- 거주지역 **(residence)**

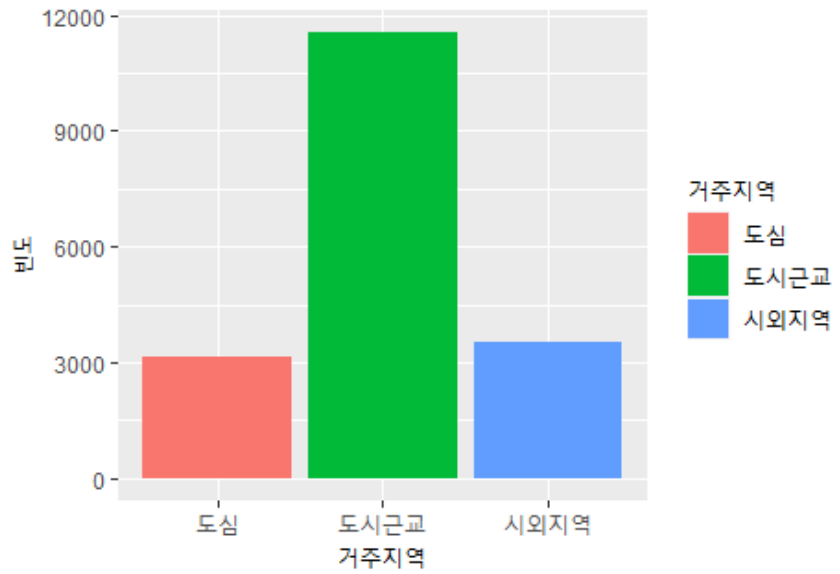
```
> table(health$residence)

도시근교      도심 시외지역
   11558      3164      3546
> health$residence = factor(health$residence, levels=c( "도심", "도시근교", "시외지역"))
> table(health$residence)

    도심 도시근교 시외지역
    3164    11558     3546

> health %>% with(table(residence))/NROW(health)
residence
    도심 도시근교 시외지역
  0.1732   0.6327   0.1941
```

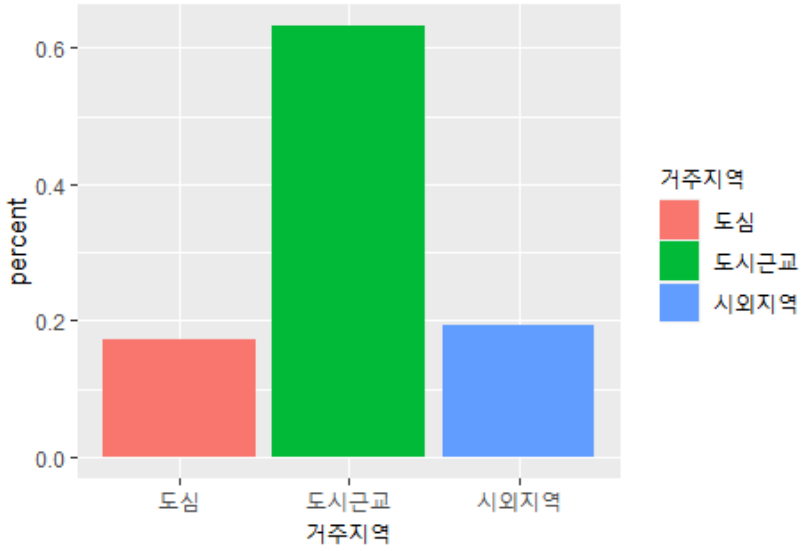
2. 변수 속성 정의

■ 범주형 변수 탐색

- 거주지역 **(residence)**

```
> health %>% ggplot(aes(x=residence, fill=residence)) +
             geom_bar() +
             labs(x= "거주지역", y = "빈도 “, fill = "거주지역")
> health %>% ggplot(aes(x=residence, fill=residence)) +
             geom_bar(aes(y = (..count..)/sum(..count..))) +
             labs(x = "거주지역", y = "percent “, fill = "거주지역")
```





2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 결혼 상태 (marrage)

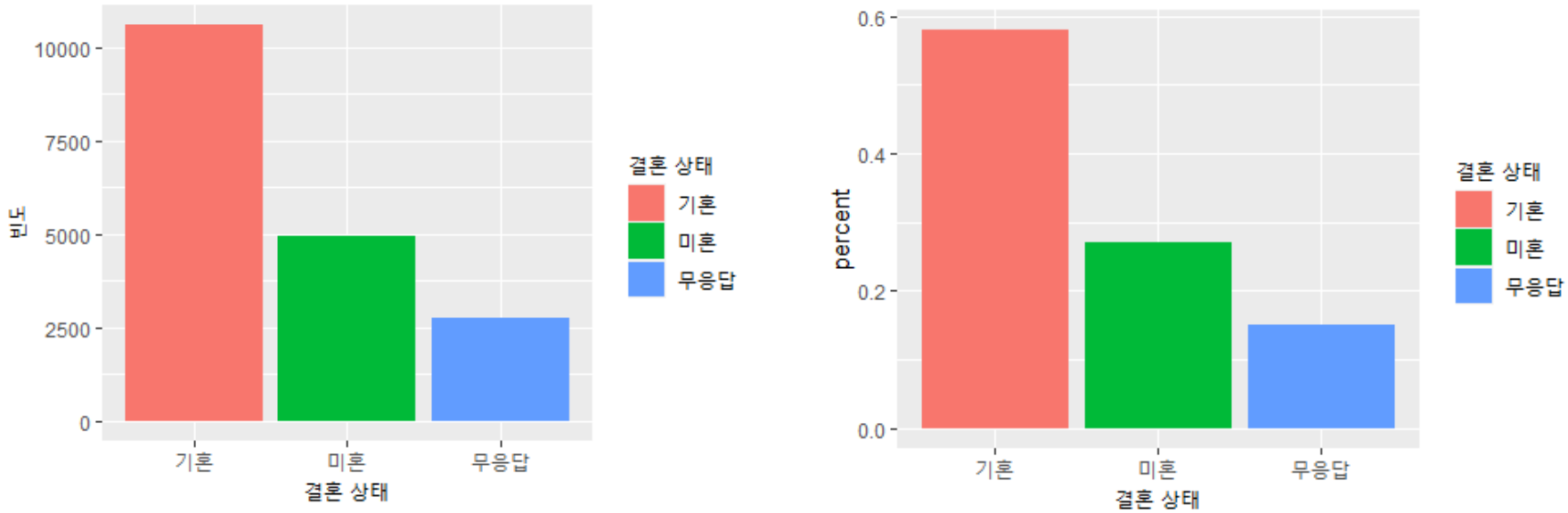
```
> table(health$marrage)

  기혼 무응답   미혼
 10596   2738   4934
> health$marrage = factor(health$marrage, levels=c( "기혼", "미혼", "무응답"))
> table(health$marrage)

  기혼   미혼 무응답
 10596   4934   2738
> health %>% with(table(marrage))/NROW(health)
marrage
  기혼   미혼 무응답
0.5800 0.2701 0.1499
```

2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 결혼 상태 (marrage)

```
> health %>% ggplot(aes(x=marrage, fill=marrage)) +
  geom_bar() +
  labs(x= "결혼 상태", y= "빈도 “, fill = "결혼 상태")
> health %>% ggplot(aes(x=marrage, fill=marrage)) +
  geom_bar(aes(y = (..count..)/sum(..count..))) + labs() +
  labs(x = "결혼 상태", y = "percent “, fill = "결혼 상태")
```



2. 변수 속성 정의

■ 범주형 변수 탐색

- 불만표현횟수 (n_complaint)

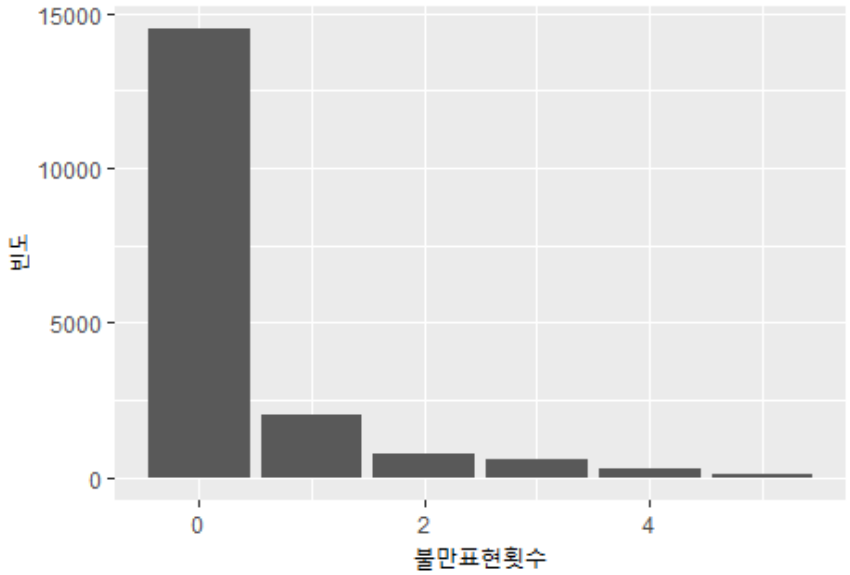
```
> mode(health$n_complaint)
[1] "numeric “

> table(health$n_complaint)

    0     1     2     3     4     5
14504  2022   748   584   298   112

> health %>% with(table(n_complaint))/NROW(health)
n_complaint
       0        1        2        3        4        5
0.793957 0.110685 0.040946 0.031968 0.016313 0.006131
```

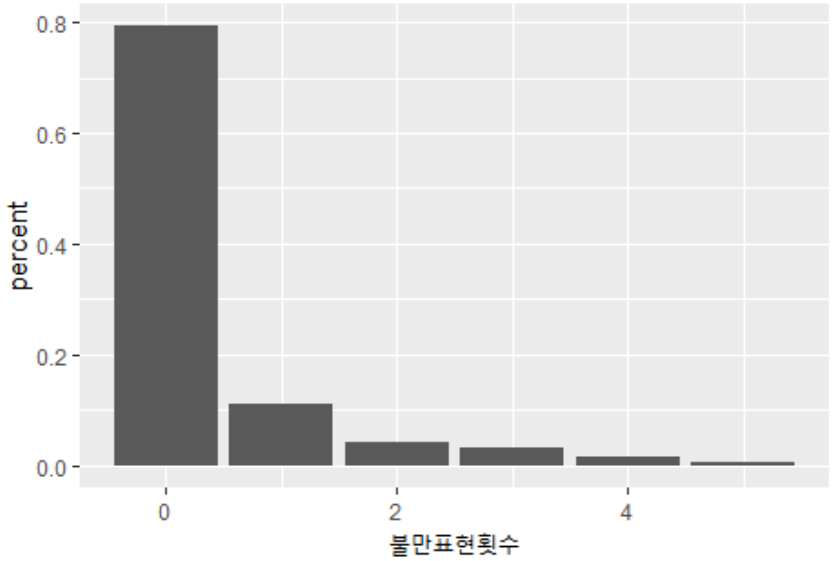
2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 불만표현횟수 (n_complaint)

```
> health %>% ggplot(aes(x=n_complaint)) +
             geom_bar() +
             labs(x= "불만표현횟수", y= "빈도")
> health %>% ggplot(aes(x=n_complaint)) +
             geom_bar(aes(y = (..count..)/sum(..count..))) +
             labs(x = "불만표현횟수", y = "percent")
```





2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 이용 서비스 수 (n_product)

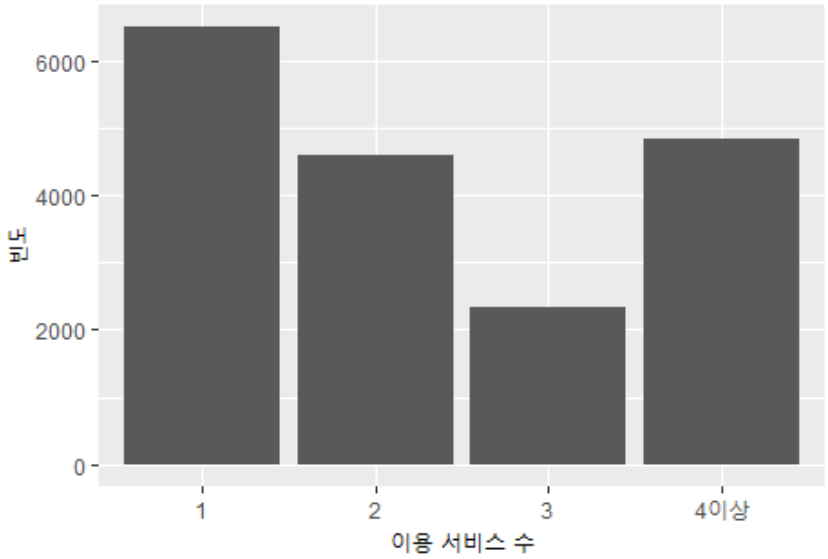
```
> table(health$n_product)

    1     2     3 40이상
 6502  4588  2336  4842
> proportions(table(health$n_product))

     1      2      3  40이상
0.3559 0.2511 0.1279 0.2651

> health %>% with(table(n_product))/NROW(health)
n_product
     1      2      3  40이상
0.3559 0.2511 0.1279 0.2651
```

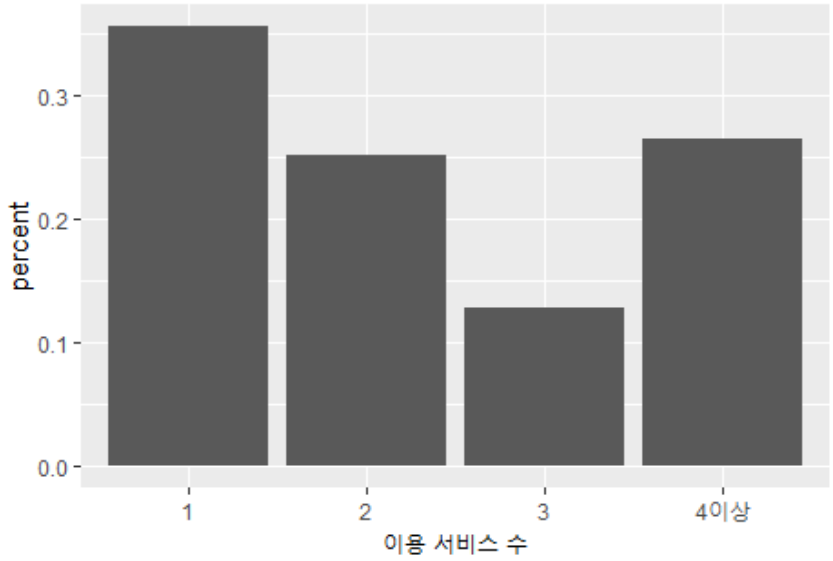
2. 변수 속성 정의

■ 범주형 변수 탐색

- 이용 서비스 수 **(n_product)**

```
> health %>% ggplot(aes(x=n_product)) +
           geom_bar() +
           labs(x= "이용 서비스 수", y= "빈도")
> health %>% ggplot(aes(x=n_product)) +
             geom_bar(aes(y = (..count..)/sum(..count..))) +
             labs(x = "이용 서비스 수", y = "percent")
```





2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 고객타입 (customer_type)

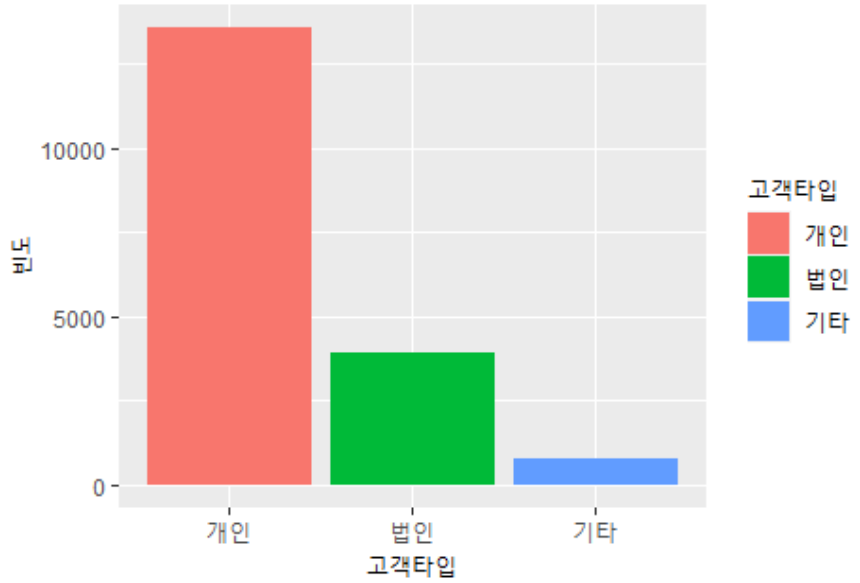
```
> table(health$customer_type)

 개인  기타  법인
13576   756  3936

> health$customer_type = factor(health$customer_type, levels=c( "개인", "법인", "기타"))

> table(health$customer_type)

 개인  법인  기타
13576  3936   756

> health %>% with(table(customer_type))/NROW(health)
customer_type
   개인     법인     기타
0.74316 0.21546 0.04138
```

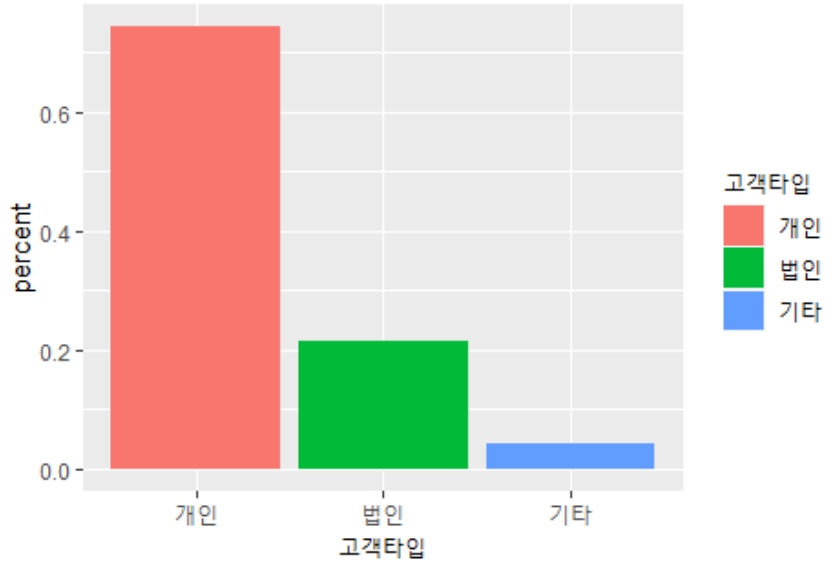
2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 고객타입 (customer_type)

```
> health %>% ggplot(aes(x=customer_type, fill=customer_type)) +
             geom_bar() +
             labs(x= "고객타입", y= "빈도", fill = "고객타입")
> health %>% ggplot(aes(x=customer_type, fill=customer_type)) +
             geom_bar(aes(y = (..count..)/sum(..count..))) +
             labs(x = "고객타입", y = "percent", fill = "고객타입")
```





2. 변수 속성 정의

■ 범주형 변수 탐색

- 고객등급 **(customer_grade)**

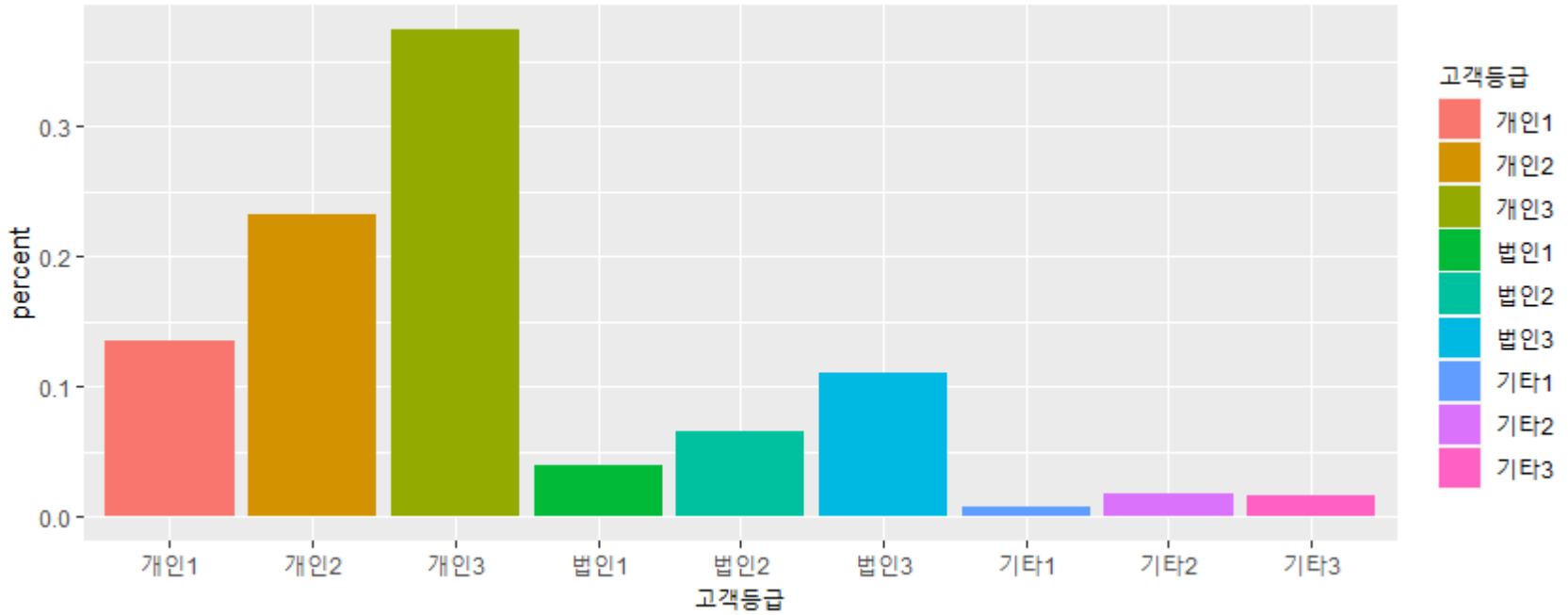
```
> table(health$customer_grade)

개인1 개인2 개인3 기타1 기타2 기타3 법인1 법인2 법인3
 2480  4244  6852   132   328   296   718  1190  2028

> health$customer_grade = factor(health$customer_grade,
                                 levels=c("개인1", "개인2", "개인3", "법인1", "법인2", "법인3" ,
                                          "기타1", "기타2", "기타3"))
> table(health$customer_grade)

개인1 개인2 개인3 법인1 법인2 법인3 기타1 기타2 기타3
 2480  4244  6852   718  1190  2028   132   328   296

> health %>% with(table(customer_grade))/NROW(health)
customer_grade
   개인1    개인2    개인3    법인1    법인2    법인3    기타1    기타2    기타3
0.135757 0.232319 0.375082 0.039304 0.065141 0.111014 0.007226 0.017955 0.016203
```

2. 변수 속성 정의

■ 범주형 변수 탐색

- 고객등급 **(customer_grade)**

```
> health %>% ggplot(aes(x=customer_grade, fill=customer_grade)) +
             geom_bar(aes(y = (..count..)/sum(..count..))) +
             labs(x = "고객등급", y = "percent", fill = "고객등급")
```



2. 변수 속성 정의

■ 범주형 변수 탐색

- 갱신인센티브 **(incentive)**

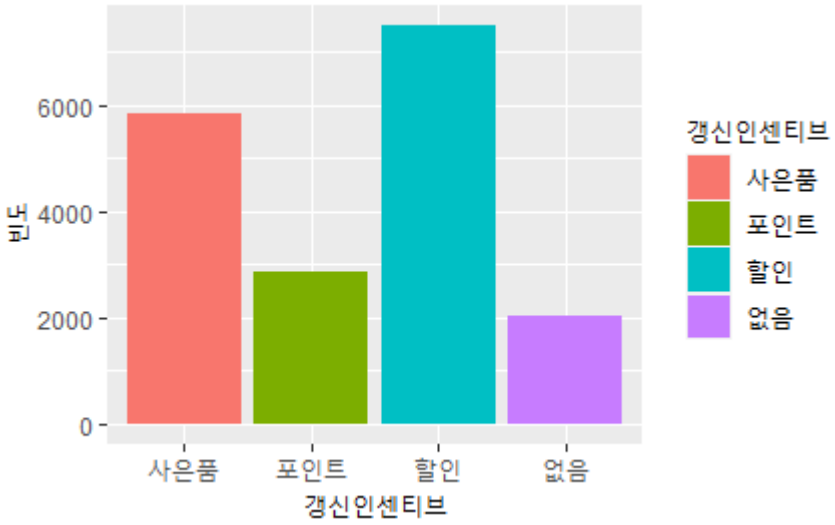
```
> table(health$incentive)

사은품    없음 포인트    할인
  5852    2048    2864    7504

> health$incentive = factor(health$incentive, levels=c("사은품", "포인트", "할인", "없음"))

> table(health$incentive)

사은품 포인트    할인    없음
  5852    2864    7504    2048

> health %>% with(table(incentive))/NROW(health)
incentive
사은품 포인트    할인    없음
0.3203 0.1568 0.4108 0.1121
```

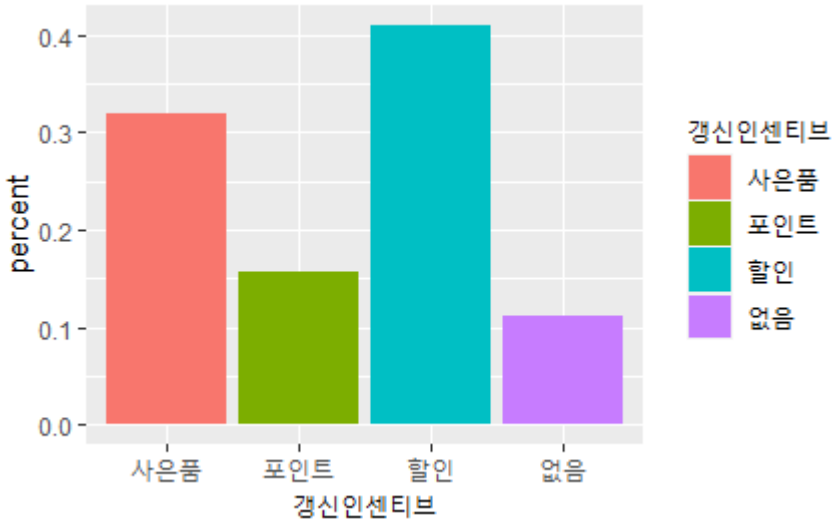
2. 변수 속성 정의

■ 범주형 변수 탐색

- 갱신인센티브 **(incentive)**

```
> health %>% ggplot(aes(x=incentive, fill=incentive)) +
             geom_bar() +
             labs(x= "갱신인센티브", y= "빈도", fill = "갱신인센티브")
> health %>% ggplot(aes(x=incentive, fill=incentive)) +
             geom_bar(aes(y = (..count..)/sum(..count..))) +
             labs(x = "갱신인센티브", y = "percent", fill = "갱신인센티브")
```





2. 변수 속성 정의

■ 범주형 변수 탐색

- 판매채널 **(channel)**

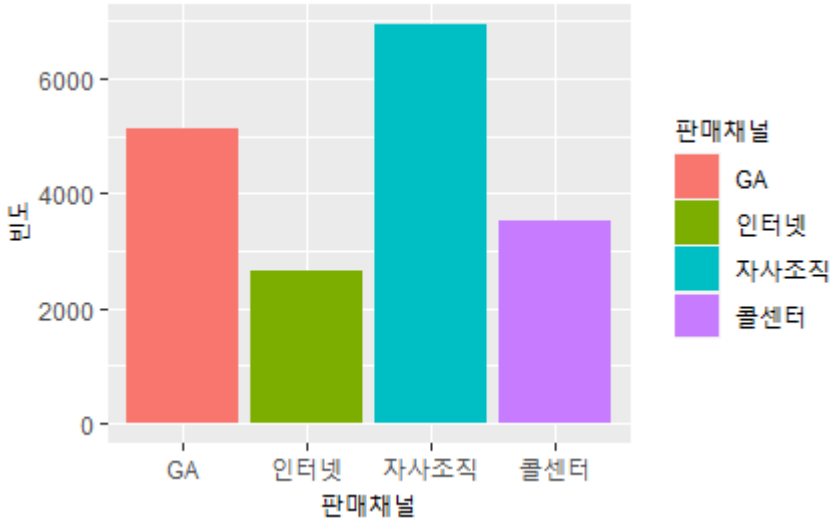
```
> table(health$channel)

      GA   인터넷 자사조직   콜센터
    5134     2650     6954     3530
> health$channel = factor(health$channel, levels=c( "자사조직", "GA", "콜센터", "인터넷"))

> table(health$channel)

자사조직       GA   콜센터   인터넷
    6954     5134     3530     2650
> health %>% with(table(channel))/NROW(health)
channel
자사조직       GA   콜센터   인터넷
  0.3807   0.2810   0.1932   0.1451
```

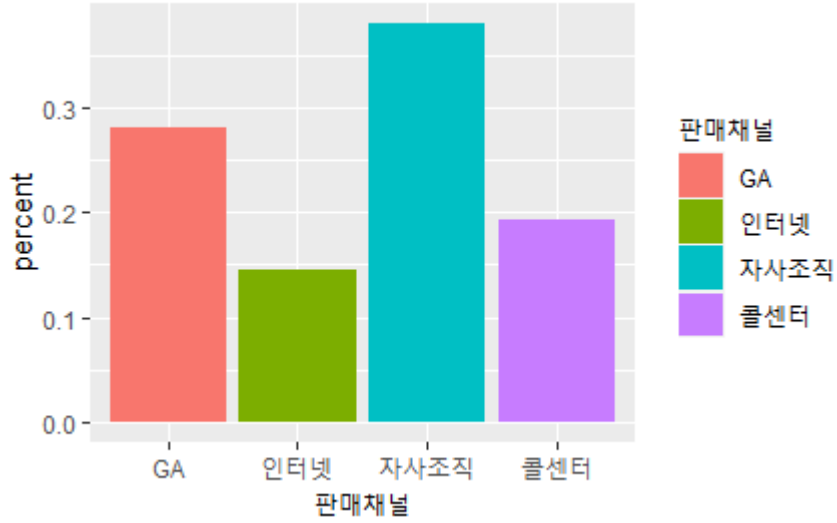
2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 판매채널 (channel)

```
> health %>% ggplot(aes(x=channel, fill=channel)) +
             geom_bar() +
             labs(x= "판매채널", y= "빈도", fill = "판매채널")
> Health %>% ggplot(aes(x=channel, fill=channel)) +
             geom_bar(aes(y = (..count..)/sum(..count..))) +
             labs(x = "판매채널", y = "percent", fill = "판매채널")
```





2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 집 형태 (house_type)

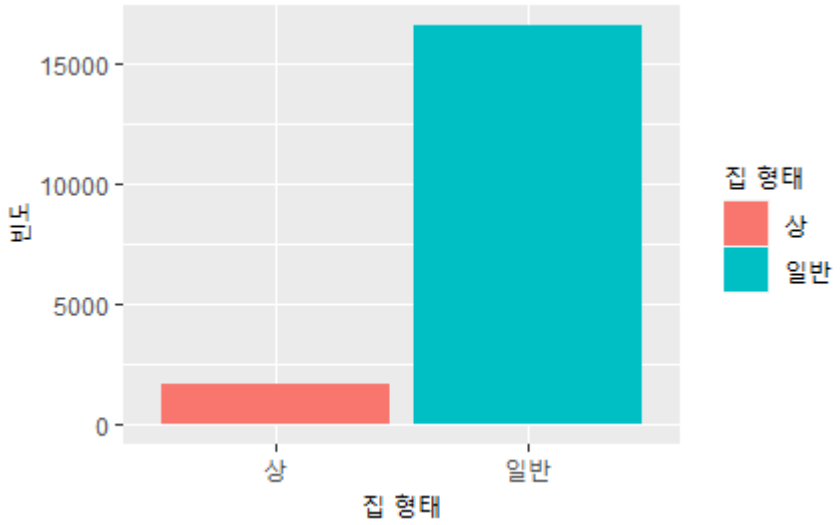
```
> table(health$house_type)

   상  일반
 1662 16606

> health$house_type = factor(health$house_type, levels=c( "상", "일반"))

> health %>% with(table(house_type))/NROW(health)
house_type
      상     일반
0.09098 0.90902
```

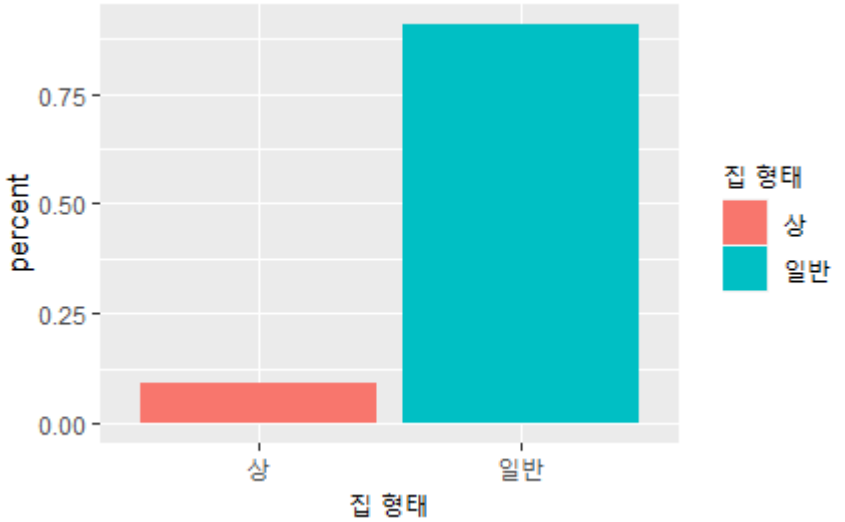
2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 집 형태 (house_type)

```
> health %>% ggplot(aes(x=house_type, fill=house_type)) +
             geom_bar() +
             labs(x= "집 형태", y= "빈도", fill = "집 형태")
> health %>% ggplot(aes(x=house_type, fill=house_type)) +
             geom_bar(aes(y = (..count..)/sum(..count..))) +
             labs(x = "집 형태", y = "percent", fill = "집 형태")
```





2. 변수 속성 정의

■ 범주형 변수 탐색

- 집 크기 **(house_size)**

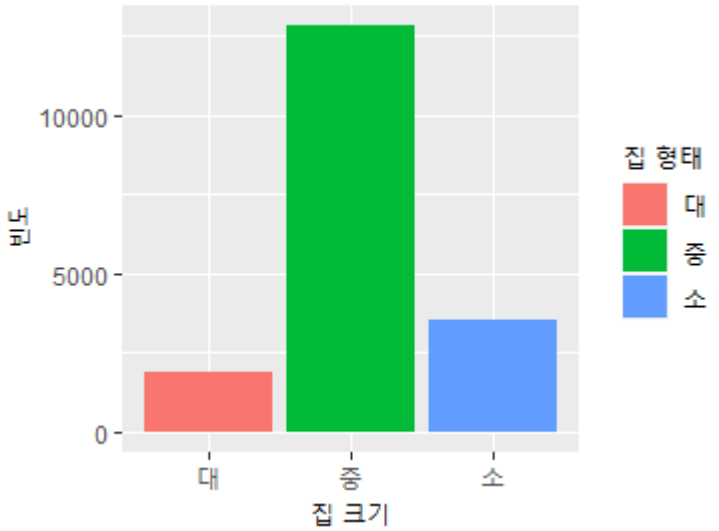
```
> table(health$house_size)

   대    소    중
 1892  3528 12848

> health$house_type = factor(health$house_size, levels=c( "대", "중", "소"))

> table(health$house_type)

   대    중    소
 1892 12848  3528

> health %>% with(table(house_size))/NROW(health)
house_size
    대     소     중
0.1036 0.7033 0.1931
```

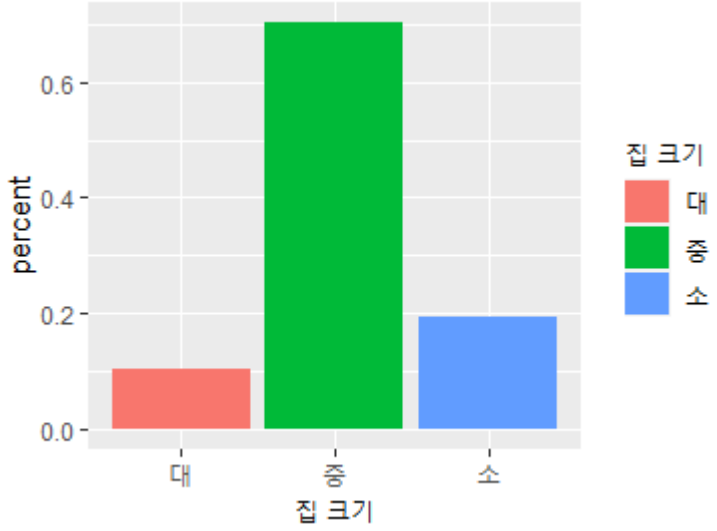
2. 변수 속성 정의

■ 범주형 변수 탐색

- 집 크기 **(house_size)**

```
> health %>% ggplot(aes(x=house_size, fill=house_size)) +
             geom_bar() +
             labs(x= "집 크기", y= "빈도", fill = "집 형태")
> health %>% ggplot(aes(x=house_size, fill=house_size)) +
             geom_bar(aes(y = (..count..)/sum(..count..))) +
             labs(x = "집 크기", y = "percent", fill = "집 크기")
```





2. 변수 속성 정의

■ 범주형 변수 탐색

- 나이 **(age)**

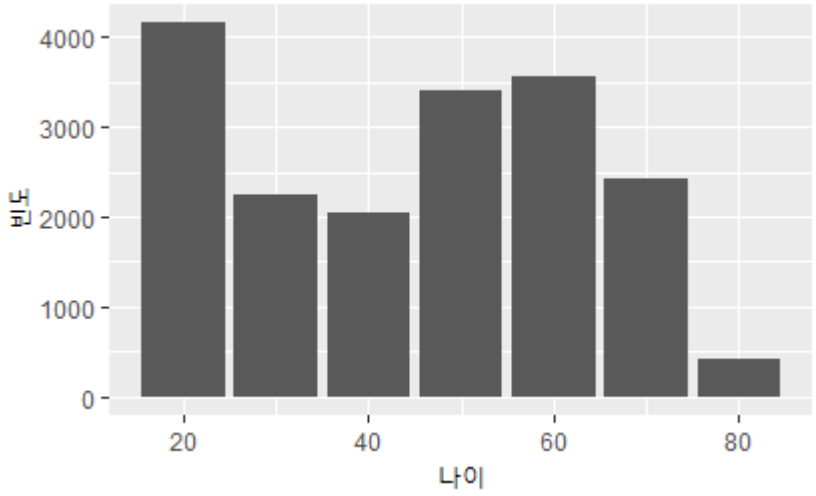
```
> table(health$age)

  20   30   40   50   60   70   80
4162 2246 2042 3396 3566 2434  422

> health$channel = factor(health$age)

> health %>% with(table(age))/NROW(health)
age
    20     30     40     50     60     70     80
0.2278 0.1229 0.1118 0.1859 0.1952 0.1332 0.0231
```

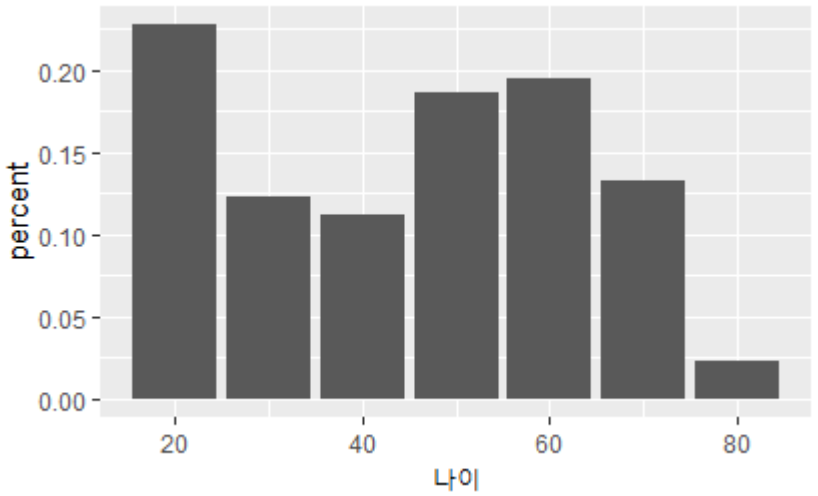
2. 변수 속성 정의

■ 범주형 변수 탐색

▪ 나이 (age)

```
> health %>% ggplot(aes(x=age, fill=age)) +
             geom_bar() +
             labs(x= "나이", y= "빈도", fill = "나이")
> health %>% ggplot(aes(x=age, fill=age)) +
             geom_bar(aes(y = (..count..)/sum(..count..))) +
             labs(x = "나이", y = "percent", fill = "나이")
```





3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

- **WOE (Weight of Evidence)**
  - 각 Attribute에서의 Goods, Bad의 비율차이를 측정하여 Attribute의 영향력 측정
  - WOE= ln(%Good/%Bad)
  - High Positive ⇒ Low Risk, High Negative ⇒ High Risk
- **Information Value(IV)**
  - Characteristic 전체의 영향력 측정
  - ∑[(%Good – %Bad) × {ln(%Good/%Bad)}]
  - ~ 0.02 : Unpredicted, 0.02 ~ 0.1 : Weak, 0.1 ~ 0.3 : Medium, 0.3 ~ : Strong
  - 0.5이상의 Information Value를 보이는 경우, Over-fitting의 위험성이 크므로 모델에서 제외함

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

- 예제

Range	Bins	Non events	Events	% of Non-Events	% of Events	WOE	IV
0-50	1	197	20	5.4%	5.9%	-0.0952	0.0005
51-100	2	450	34	12.3%	10.1%	0.2002	0.0045
101-150	3	492	39	13.4%	11.5%	0.1522	0.0029
151-200	4	597	51	16.3%	15.1%	0.0774	0.0009
201-250	5	609	54	16.6%	16.0%	0.0401	0.0003
251-300	6	582	55	15.9%	16.3%	-0.0236	0.0001
301-350	7	386	41	10.5%	12.1%	-0.1405	0.0022
351-400	8	165	23	4.5%	6.8%	-0.4123	0.0095
>401	9	184	21	5.0%	6.2%	-0.2123	0.0025
	Total	3662	338				0.0234

- WOE= [ln(%Good/%Bad)]
  = ln (0.134 / 0.115) = 0.1522

- $\text{Information Value} = 0.0234$

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

▪ iv.mult {woe}

- 데이터프레임에 정의된 변수들(문자, 수치형, 펙터형)에 대해 IV를 계산해 줌

함수		의미
iv.mult(df, y, summary = FALSE, vars = NULL, verbose = FALSE, rcontrol = NULL)	df	• 데이터프레임
	y	• 이항 값을 같는 컬럼
	summary	• summary=TRUE일 때 변수에 대한 IV를 제공함. 결과는 IV 순으로 큰 것부터 정리됨
	vars	• 변수들의 리스트. 지정되지 않으면 모든 변수들에 대해 진행
	verbose	• TRUE일 때 부가적인 정보 제공. debugging할 때 용이
	rcontrol	• rpart tree 생성 시에 사용되는 추가 항목

▪ iv.plot.summary {woe}

- 변수들에 대한 요약된 IV 그림을 보여 줌
- iv.plot.summary(iv)

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

```
> library("woe")
> iv_all=iv.mult(df=data.frame(health), y= "renewal", summary=TRUE)
Information Value 0
Information Value 0
Information Value 0.08
Information Value 0
Information Value 0.01
Information Value 0.01
Information Value 0
Information Value 0.17
Information Value 0.12
Information Value 0.09
Information Value 0
Information Value 0.01
Information Value 0.03
Information Value 0
Information Value 0.02
Information Value 0
Information Value 0.01
Information Value 0.47
Information Value 0.09
Information Value 0.1
Information Value 0
Information Value 0.02
Information Value 0
```

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

```
> iv_all
          Variable InformationValue Bins ZeroBins  Strength
1        incentive        4.694e-01    4        1    Strong
2           income        1.718e-01    4        0   Average
3        residence        1.236e-01    3        0   Average
4       claim_size        9.635e-02    5        0      Weak
5          channel        9.298e-02    4        0      Weak
6          marrage        8.981e-02    3        0      Weak
7        retention        8.009e-02    6        0      Weak
8      period_keep        3.375e-02    3        0      Weak
9       house_size        2.158e-02    3        0      Weak
10       n_product        1.712e-02    4        0 Wery weak
11            work        1.199e-02    2        0 Wery weak
12   period_claim        1.049e-02    2        0 Wery weak
13             edu        9.585e-03    5        0 Wery weak
14 customer_grade        7.744e-03    9        0 Wery weak
15      house_type        1.565e-03    2        0 Wery weak
16   customer_type        1.479e-03    3        0 Wery weak
17        Provance        3.962e-04    5        0 Wery weak
18          gender        1.575e-04    2        0 Wery weak
```

3. 변수별 정보 가치(Information Value) 산출

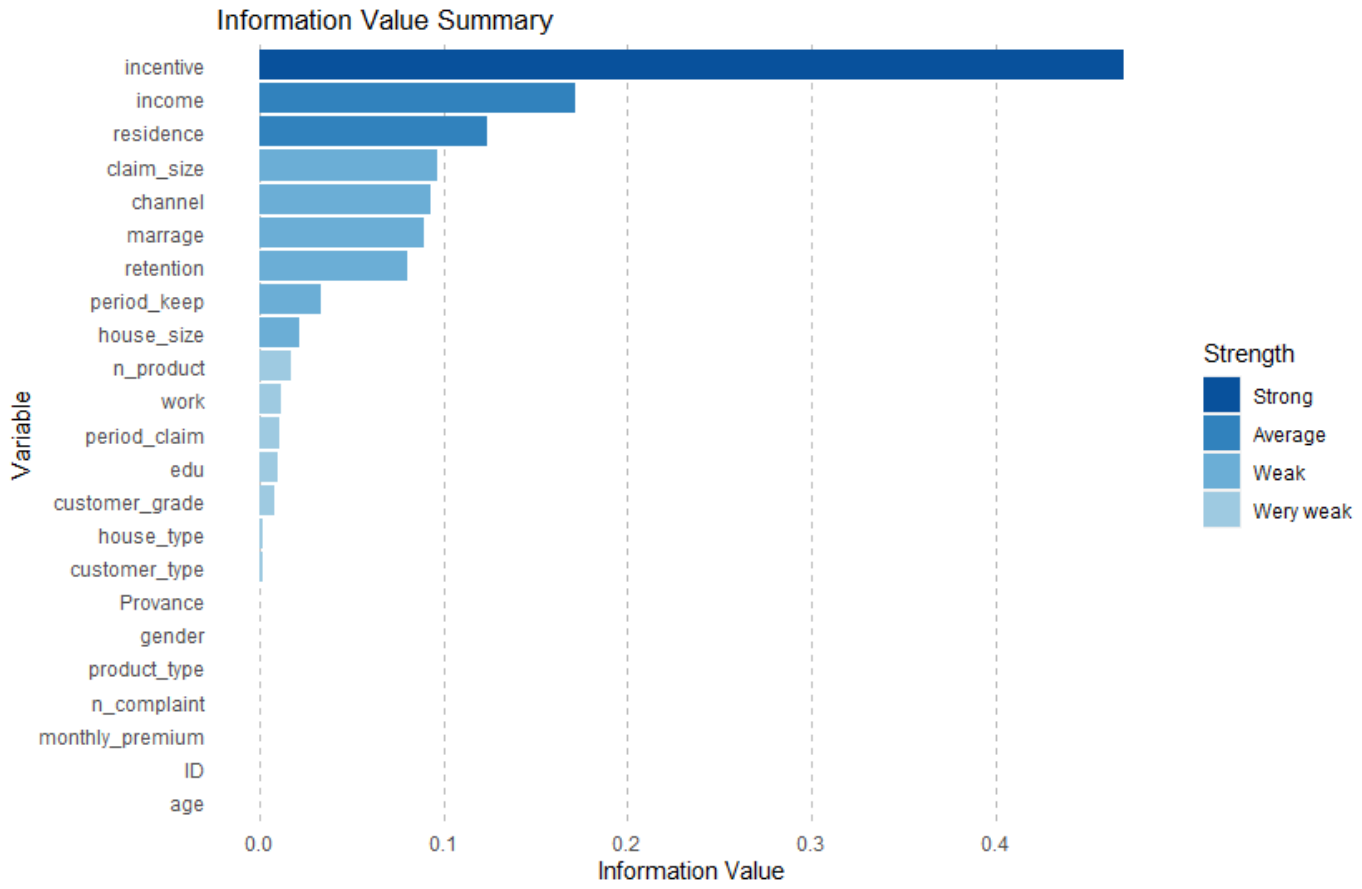
■ 정보 가치(Information Value)

19	product_type	5.448e-05	3	0	Wery weak
20	n_complaint	0.000e+00	1	0	Wery weak
21	monthly_premium	0.000e+00	1	0	Wery weak
22	age	0.000e+00	1	0	Wery weak
23	ID	0.000e+00	1	0	Wery weak

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

```
> iv.plot.summary(iv_all)
```



3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

```
> iv.mult(df=data.frame(health), y= "renewal", vars=c("incentive"))     # 모형 선택변수 고려함
Information Value 0.47
[[1]]
   variable  class outcome_0 outcome_1  pct_0   pct_1   odds     woe      miv
1 incentive 사은품      4484      1368 0.2865 0.52294 0.5478 -0.6018 0.142295
2 incentive 포인트      2804        60 0.1791 0.02294 7.8108  2.0555 0.321092
3 incentive   할인      6316      1188 0.4035 0.45413 0.8886 -0.1181 0.005978
4 incentive   없음      2048         0 0.1308 0.00000 1.0000  0.0000 0.000000

> iv.mult(df=data.frame(health), y= "renewal", vars=c("income"))       # 모형 선택변수 고려함 (구간은 추후 결정)
Information Value 0.17
[[1]]
  variable               class outcome_0 outcome_1  pct_0  pct_1   odds      woe      miv
1   income          (;9395000)      4249       396 0.2715 0.1514 1.7933  0.58407 0.070142
2   income  <9395000;25080000)      1476       534 0.0943 0.2041 0.4620 -0.77226 0.084815
3   income <25080000;35120000)      1658       390 0.1059 0.1491 0.7105 -0.34173 0.014747
4   income         <35120000;)      8269      1296 0.5283 0.4954 1.0664  0.06428 0.002114
                                                                                    sql
1                          when income < 9395000 then 0.584072458107117
2  when income >= 9395000 AND income < 25080000 then -0.772257104552991
3 when income >= 25080000 AND income < 35120000 then -0.341731674181666
4                         when income >= 35120000 then 0.064278714056733
```

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

```
> iv.mult(df=data.frame(health), y= "renewal", vars=c("residence"))     # 모형 선택변수 고려함
Information Value 0.12
[[1]]
 variable    class outcome_0 outcome_1  pct_0  pct_1   odds     woe     miv
1 residence     도심      2888       276 0.1845 0.1055 1.7489  0.5590 0.04416
2 residence 도시근교      9542      2016 0.6096 0.7706 0.7911 -0.2344 0.03773
3 residence 시외지역      3222       324 0.2059 0.1239 1.6621  0.5081 0.04166

> iv.mult(df=data.frame(health), y= "renewal", vars=c("claim_size"))    # 모형 선택변수 고려함 (구간 추후 결정)
Information Value 0.1
[[1]]
    variable     class outcome_0 outcome_1  pct_0  pct_1   odds      woe      miv
1 claim_size    (;255)      3875       384 0.2476 0.1468 1.6866  0.52271 0.052680
2 claim_size <255;415)      4719       870 0.3015 0.3326 0.9066 -0.09809 0.003048
3 claim_size <415;495)      1804       474 0.1153 0.1812 0.6361 -0.45240 0.029829
4 claim_size <495;695)      2767       546 0.1768 0.2087 0.8470 -0.16605 0.005303
5 claim_size    <695;)      2487       342 0.1589 0.1307 1.2154  0.19507 0.005493
                                                                      sql
1                         when claim_size < 255 then 0.522706118446507
2 when claim_size >= 255 AND claim_size < 415 then -0.0980932906754556
3  when claim_size >= 415 AND claim_size < 495 then -0.452397891825721
4  when claim_size >= 495 AND claim_size < 695 then -0.166052266711193
5                        when claim_size >= 695 then 0.195069435987586
```

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

```
> iv.mult(df=data.frame(health), y= "renewal", vars=c("channel"))   # 모형 선택변수로 고려함
Information Value 0.09
[[1]]
  variable    class outcome_0 outcome_1  pct_0  pct_1   odds     woe      miv
1  channel 자사조직      5622       1332 0.3592 0.5092 0.7054 -0.3489 0.052337
2  channel       GA      4546        588 0.2904 0.2248 1.2922  0.2563 0.016833
3  channel   콜센터      3146        384 0.2010 0.1468 1.3693  0.3143 0.017037
4  channel   인터넷      2338        312 0.1494 0.1193 1.2524  0.2251 0.006777

> iv.mult(df=data.frame(health), y= "renewal", vars=c("marrage"))  # 모형 선택변수로 고려함
Information Value 0.09
[[1]]
  variable  class outcome_0 outcome_1  pct_0  pct_1   odds      woe     miv
1  marrage   기혼      9204       1392 0.5880 0.5321 1.1051  0.09994 0.00559
2  marrage   미혼      4358        576 0.2784 0.2202 1.2645  0.23471 0.01367
3  marrage 무응답      2090        648 0.1335 0.2477 0.5391 -0.61792 0.07055
```

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

```
> iv.mult(df=data.frame(health), y= "renewal", vars=c("retention"))  # 모형 선택변수 고려함 (구간 추후 결정)
Information Value 0.08
[[1]]
   variable       class outcome_0 outcome_1  pct_0   pct_1   odds      woe       miv
1 retention     (;1.15)      1738       360 0.1110 0.13761 0.8069 -0.21457 5.702e-03
2 retention <1.15;2.25)      1800       214 0.1150 0.08180 1.4058  0.34061 1.131e-02
3 retention <2.25;3.35)      1614       380 0.1031 0.14526 0.7099 -0.34265 1.444e-02
4 retention <3.35;5.55)      3448       586 0.2203 0.22401 0.9834 -0.01672 6.212e-05
5 retention <5.55;6.85)      2198       188 0.1404 0.07187 1.9541  0.66991 4.593e-02
6 retention     <6.85;)      4854       888 0.3101 0.33945 0.9136 -0.09037 2.650e-03
                                                                                  sql
1                       when retention < 1.15 then -0.214565996378032
2   when retention >= 1.15 AND retention < 2.25 then 0.340613658109191
3  when retention >= 2.25 AND retention < 3.35 then -0.342652674643751
4 when retention >= 3.35 AND retention < 5.55 then -0.0167224285466427
5   when retention >= 5.55 AND retention < 6.85 then 0.669908901380404
6                     when retention >= 6.85 then -0.0903656274588373
```

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

```
> iv.mult(df=data.frame(health), y= "renewal", vars=c("period_keep"))
Information Value 0.03
[[1]]
     variable       class outcome_0 outcome_1  pct_0  pct_1  odds     woe      miv
1 period_keep     (;37.5)      6214       912 0.3970 0.3486 1.139  0.1300 0.006289
2 period_keep <37.5;72.5)      5636      1176 0.3601 0.4495 0.801 -0.2219 0.019851
3 period_keep     <72.5;)      3802       528 0.2429 0.2018 1.203  0.1852 0.007608
                                                                      sql
1                          when period_keep < 37.5 then 0.129967829125448
2 when period_keep >= 37.5 AND period_keep < 72.5 then -0.221896526193215
3                         when period_keep >= 72.5 then 0.185233968588899

> iv.mult(df=data.frame(health), y= "renewal", vars=c("house_size"))
Information Value 0.02
[[1]]
    variable class outcome_0 outcome_1   pct_0  pct_1   odds      woe       miv
1 house_size    대      1556       336 0.09941 0.1284 0.7740 -0.25619 0.0074367
2 house_size    중     10964      1884 0.70049 0.7202 0.9726 -0.02773 0.0005463
3 house_size    소      3132       396 0.20010 0.1514 1.3219  0.27906 0.0135975
```

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

```
>  iv.mult(df=data.frame(health), y= "renewal", vars=c("n_product"))
Information Value 0.02
[[1]]
   variable class outcome_0 outcome_1  pct_0  pct_1   odds      woe       miv
1 n_product     1      5470       1032 0.3495 0.3945 0.8859 -0.12117 0.0054551
2 n_product     2      3904        684 0.2494 0.2615 0.9539 -0.04715 0.0005679
3 n_product     3      2072        264 0.1324 0.1009 1.3118  0.27137 0.0085377
4 n_product 4이상      4206        636 0.2687 0.2431 1.1053  0.10012 0.0025630

>  iv.mult(df=data.frame(health), y= "renewal", vars=c("work"))
Information Value 0.01
[[1]]
  variable  class outcome_0 outcome_1  pct_0 pct_1   odds      woe      miv
1     work   고용      9884       1512 0.6315 0.578 1.0926  0.08853 0.004737
2     work 비고용      5768       1104 0.3685 0.422 0.8732 -0.13557 0.007253
```

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

```
> iv.mult(df=data.frame(health), y= "renewal", vars=c("period_claim"))
Information Value 0.01
[[1]]
      variable   class outcome_0 outcome_1  pct_0  pct_1   odds      woe      miv
1 period_claim (;24.5)     12135       2136 0.7753 0.8165 0.9495 -0.05179 0.002135
2 period_claim <24.5;)      3517        480 0.2247 0.1835 1.2246  0.20263 0.008351
                                                  sql
1 when period_claim < 24.5 then -0.0517933526725974
2  when period_claim >= 24.5 then 0.202625257818832

> iv.mult(df=data.frame(health), y= "renewal", vars=c("edu"))
Information Value 0.01
[[1]]
  variable    class outcome_0 outcome_1   pct_0   pct_1   odds      woe      miv
1      edu 고졸이하      4560       684 0.29134 0.26147 1.1142  0.10817 0.003231
2      edu   전문대      4546       816 0.29044 0.31193 0.9311 -0.07136 0.001533
3      edu     학사      4740       756 0.30284 0.28899 1.0479  0.04680 0.000648
4      edu     석사      1242       240 0.07935 0.09174 0.8649 -0.14511 0.001798
5      edu     박사       564       120 0.03603 0.04587 0.7855 -0.24139 0.002375
```

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

```
> iv.mult(df=data.frame(health), y= "renewal", vars=c("customer_grade"))
Information Value 0.01
[[1]]
        variable class outcome_0 outcome_1   pct_0    pct_1   odds       woe       miv
1 customer_grade 개인1      2110       370 0.13481 0.141437 0.9531 -0.048012 3.183e-04
2 customer_grade 개인2      3634       610 0.23217 0.233180 0.9957 -0.004322 4.346e-06
3 customer_grade 개인3      5916       936 0.37797 0.357798 1.0564  0.054848 1.106e-03
4 customer_grade 법인1       622        96 0.03974 0.036697 1.0829  0.079640 2.423e-04
5 customer_grade 법인2      1014       176 0.06478 0.067278 0.9629 -0.037778 9.423e-05
6 customer_grade 법인3      1724       304 0.11015 0.116208 0.9478 -0.053578 3.248e-04
7 customer_grade 기타1       108        24 0.00690 0.009174 0.7521 -0.284875 6.479e-04
8 customer_grade 기타2       290        38 0.01853 0.014526 1.2755  0.243342 9.739e-04
9 customer_grade 기타3       234        62 0.01495 0.023700 0.6308 -0.460766 4.032e-03

> iv.mult(df=data.frame(health), y= "renewal", vars=c("house_type"))
Information Value 0
[[1]]
    variable class outcome_0 outcome_1   pct_0  pct_1   odds      woe       miv
1 house_type    상      1398       264 0.08932 0.1009 0.8851 -0.12210 0.0014164
2 house_type  일반     14254      2352 0.91068 0.8991 1.0129  0.01282 0.0001487
```

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

```
> iv.mult(df=data.frame(health), y= "renewal", vars=c("customer_type"))
Information Value 0
[[1]]
       variable class outcome_0 outcome_1   pct_0  pct_1   odds      woe       miv
1 customer_type  개인     11660       1916 0.74495 0.7324 1.0171  0.01697 0.0002128
2 customer_type  법인      3360        576 0.21467 0.2202 0.9750 -0.02536 0.0001399
3 customer_type  기타       632        124 0.04038 0.0474 0.8519 -0.16034 0.0011260

> iv.mult(df=data.frame(health), y= "renewal", vars=c("Provance"))
Information Value 0
[[1]]
  variable      class outcome_0 outcome_1   pct_0   pct_1   odds       woe       miv
1 Provance   서울경기      5388        912 0.34424 0.34862 0.9874 -0.012663 5.555e-05
2 Provance 경상남북도      1378        218 0.08804 0.08333 1.0565  0.054941 2.586e-04
3 Provance 전라남북도      2920        486 0.18656 0.18578 1.0042  0.004178 3.250e-06
4 Provance 충청남북도      4450        752 0.28431 0.28746 0.9890 -0.011029 3.478e-05
5 Provance     강원도      1516        248 0.09686 0.09480 1.0217  0.021450 4.409e-05
```

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

```
> iv.mult(df=data.frame(health), y= "renewal", vars=c("product_type"))
Information Value 0
[[1]]
      variable class outcome_0 outcome_1   pct_0   pct_1   odds        woe       miv
1 product_type  기본      9540      1596 0.60951 0.61009 0.9990 -0.0009593 5.612e-07
2 product_type  중급      4704       780 0.30054 0.29817 1.0080  0.0079223 1.879e-05
3 product_type  고급      1408       240 0.08996 0.09174 0.9805 -0.0196657 3.513e-05
```

3. 변수별 정보 가치(Information Value) 산출

■ 정보 가치(Information Value)

▪ WOE {InformationValue}

- 범주형 변수와 이항 반응변수와의 WOE를 계산해 줌

함수	의미	
WOE(X, Y, valueOfGood = 1)	X	• WOE를 계산할 범주형 변수, factor 유형이어야 함
	Y	• 이항 값으로 “GOOD”=1, “BAD”=0

▪ IV {InformationValue}

- 범주형 변수와 이항 반응변수와의 IV를 계산해 줌

함수	의미	
IV(X, Y, valueOfGood = 1)	X	• IV를 계산할 범주형 변수, factor 유형이어야 함
	Y	• 이항 값으로 “GOOD”=1, “BAD”=0

3장 피처 엔지니어링

1. 피처 중요도(Feature Importance) 산출
2. 파생 변수 생성
3. 입력 변수 사이의 상관관계 분석
4. 변수 축소

1. 피처 중요도(Feature Importance) 산출

■ 연속형 변수의 범주화

- **woebin {scorecard}: 최적 범주 선택**

함수	의미	
woebin(dt, y, x = NULL, var_skip = NULL, breaks_list = NULL, special_values = NULL, stop_limit = 0.1, count_distr_limit = 0.05, bin_num_limit = 8, positive = "bad|1", no_cores = 2, print_step = 0L, method = "tree", save_breaks_list = NULL, ignore_const_cols = TRUE, ignore_const_cols,..)	dt	• x, y를 갖는 데이터 프레임
	y	• 반응변수
	x	• x 변수들의 이름. 디폴트는 NULL
	var_skip	• binning에 제외되는 변수들
	breaks_list	• break point들의 리스트. 디폴트는 NULL
	special_values	• 분리된 bin에 특별한 값. 디폴트는 NULL
	bin_num_limit	• bin의 최대값(정수). 디폴트는 8
	positive	• positive 범주의 값. 디폴트는 “bad | 1”
	method	• 4개의 methods. "tree" and "chimerge“는 수치형, 범주형 모두 가능. 'width' and 'freq' 는 오직 수치형만 가능함
	save_breaks_list	• breaks_list 를 저장할 파일

1. 피처 중요도(Feature Importance) 산출

■ 연속형 변수의 범주화

- **woebin {scorecard}:** 최적 범주 선택

```
> fine_class_cont <- health %>%
                     woebin(y = "renewal",
                            x = c( "income", "claim_size", "retention", "period_keep", "period_claim"),
                            positive = 1, method = "tree", count_distr_limit = 0.05,
                            bin_num_limit = 6, save_breaks_list = 'health_bin_cont')

> fine_class_cont$income     # iv.mult() 결과 선택함
   variable                bin count count_distr  neg  pos posprob      woe   bin_iv
1:   income      [-Inf,2000000)  4634      0.2537 4238  396 0.08546 -0.58148 0.069422
2:   income  [2000000,26000000)  2192      0.1200 1628  564 0.25730  0.72890 0.081334
3:   income [26000000,38000000)  2428      0.1329 1996  432 0.17792  0.25848 0.009722
4:   income     [38000000, Inf)  9014      0.4934 7790 1224 0.13579 -0.06176 0.001841
   total_iv   breaks is_special_values
1:   0.1623  2000000             FALSE
2:   0.1623 26000000             FALSE
3:   0.1623 38000000             FALSE
4:   0.1623      Inf             FALSE
```

1. 피처 중요도(Feature Importance) 산출

■ 연속형 변수의 범주화

- **woebin {scorecard}:** 최적 범주 선택

```
> fine_class_cont$claim_size       # iv.mult() 결과 선택함
     variable        bin count count_distr  neg  pos posprob      woe    bin_iv total_iv
1: claim_size [-Inf,250)  4100      0.2244 3716  384 0.09366 -0.48081 0.0435731  0.07458
2: claim_size  [250,350)  3347      0.1832 2849  498 0.14879  0.04483 0.0003741  0.07458
3: claim_size  [350,800)  8887      0.4865 7369 1518 0.17081  0.20906 0.0228868  0.07458
4: claim_size [800, Inf)  1934      0.1059 1718  216 0.11169 -0.28469 0.0077416  0.07458
   breaks is_special_values
1:    250             FALSE
2:    350             FALSE
3:    800             FALSE
4:    Inf             FALSE
```

1. 피처 중요도(Feature Importance) 산출

■ 연속형 변수의 범주화

- **woebin {scorecard}:** 최적 범주 선택

```
> fine_class_cont$retention      # iv.mult() 결과와 비교하여 합리적 선택 (iv.mult() 결과 선택함)
    variable        bin count count_distr  neg pos posprob      woe   bin_iv total_iv
1: retention [-Inf,1.2)  2283     0.12497 1908 375 0.16426  0.16207 0.003476  0.08466
2: retention    [1.2,2)  1449     0.07932 1319 130 0.08972 -0.52814 0.018261  0.08466
3: retention      [2,3)  1867     0.10220 1498 369 0.19764  0.38786 0.017589  0.08466
4: retention    [3,5.6)  4541     0.24858 3875 666 0.14666  0.02794 0.000196  0.08466
5: retention  [5.6,6.8)  2173     0.11895 2004 169 0.07777 -0.68405 0.043391  0.08466
6: retention [6.8, Inf)  5955     0.32598 5048 907 0.15231  0.07235 0.001751  0.08466
   breaks is_special_values
1:    1.2            FALSE
2:      2            FALSE
3:      3            FALSE
4:    5.6            FALSE
5:    6.8            FALSE
6:    Inf            FALSE
```

1. 피처 중요도(Feature Importance) 산출

■ 연속형 변수의 범주화

- **woebin {scorecard}:** 최적 범주 선택

```
> fine_class_cont$period_keep      # iv.mult() 결과와 비교하여 합리적 선택 (woebin() 결과 선택함)
      variable       bin count count_distr  neg  pos posprob       woe    bin_iv total_iv
1: period_keep  [-Inf,6)  1098     0.06011  906  192 0.17486  0.237408 3.682e-03  0.05605
2: period_keep    [6,18)  2182     0.11944 1942  240 0.10999 -0.301882 9.760e-03  0.05605
3: period_keep   [18,32)  2678     0.14660 2294  384 0.14339  0.001543 3.490e-07  0.05605
4: period_keep   [32,38)  1168     0.06394 1072   96 0.08219 -0.623981 1.984e-02  0.05605
5: period_keep   [38,74)  6988     0.38253 5800 1188 0.17001  0.203366 1.700e-02  0.05605
6: period_keep [74, Inf)  4154     0.22739 3638  516 0.12422 -0.164130 5.775e-03  0.05605
   breaks is_special_values
1:      6             FALSE
2:     18             FALSE
3:     32             FALSE
4:     38             FALSE
5:     74             FALSE
6:    Inf             FALSE
```

1. 피처 중요도(Feature Importance) 산출

■ 연속형 변수의 범주화

- **woebin {scorecard}:** 최적 범주 선택

```
> fine_class_cont$period_claim    # iv.mult() 결과와 비교하여 합리적 선택 (woebin() 결과 선택함)
      variable      bin count count_distr  neg  pos posprob     woe   bin_iv total_iv
1: period_claim  [-Inf,3)  1976     0.10817 1724  252  0.1275 -0.1340 0.001852  0.02883
2: period_claim     [3,5)  1432     0.07839 1168  264  0.1844  0.3019 0.007937  0.02883
3: period_claim     [5,9)  2578     0.14112 2254  324  0.1257 -0.1508 0.003039  0.02883
4: period_claim    [9,25)  8285     0.45353 6989 1296  0.1564  0.1039 0.005079  0.02883
5: period_claim   [25,29)  1590     0.08704 1422  168  0.1057 -0.3469 0.009238  0.02883
6: period_claim [29, Inf)  2407     0.13176 2095  312  0.1296 -0.1154 0.001682  0.02883
   breaks is_special_values
1:      3             FALSE
2:      5             FALSE
3:      9             FALSE
4:     25             FALSE
5:     29             FALSE
6:    Inf             FALSE
```

1. 피처 중요도(Feature Importance) 산출

■ 범주형 변수의 범주 결합

- **woebin {scorecard}:** 최적 범주 선택

```
> fine_class_cat <- health %>%
                    woebin(y = "renewal",
                           x = c("incentive", "channel", "n_product", "edu",
                                 "customer_grade", "Provance"),
                           positive = 1, method = "tree", count_distr_limit = 0.05,
                           bin_num_limit = 8, save_breaks_list = 'health_bin_cat')

> fine_class_cat$incentive     #  "없음"과 "포인트" 결합은 의미가 없어 선택하지 않음
    variable         bin count count_distr  neg  pos posprob     woe  bin_iv total_iv      breaks
1: incentive      사은품  5852      0.3203 4484 1368 0.23377  0.6018 0.14230   0.4764      사은품
2: incentive      포인트  2864      0.1568 2804   60 0.02095 -2.0555 0.32109   0.4764      포인트
3: incentive 할인%,%없음  9552      0.5229 8364 1188 0.12437 -0.1627 0.01306   0.4764 할인%,%없음
   is_special_values
1:             FALSE
2:             FALSE
3:             FALSE
```

1. 피처 중요도(Feature Importance) 산출

■ 범주형 변수의 범주 결합

- **woebin {scorecard}:** 최적 범주 선택

```
> fine_class$channel    #  GA와 인터넷 결합은 의미가 없어 선택하지 않음
   variable         bin count count_distr  neg  pos posprob     woe   bin_iv total_iv      breaks
1:  channel    자사조직  6954      0.3807 5622 1332  0.1915  0.3489 0.052337  0.09264    자사조직
2:  channel GA%,%콜센터  8664      0.4743 7692  972  0.1122 -0.2796 0.033522  0.09264 GA%,%콜센터
3:  channel      인터넷  2650      0.1451 2338  312  0.1177 -0.2251 0.006777  0.09264      인터넷
   is_special_values
1:             FALSE
2:             FALSE
3:             FALSE

> fine_class_cat$n_product     # 변화 없음
    variable   bin count count_distr  neg  pos posprob      woe    bin_iv total_iv breaks
1: n_product     3  2336      0.1279 2072  264  0.1130 -0.27137 0.0085377  0.01712      3
2: n_product 4이상  4842      0.2651 4206  636  0.1314 -0.10012 0.0025630  0.01712  4이상
3: n_product     2  4588      0.2511 3904  684  0.1491  0.04715 0.0005679  0.01712      2
4: n_product     1  6502      0.3559 5470 1032  0.1587  0.12117 0.0054551  0.01712      1
   is_special_values
1:             FALSE
2:             FALSE
3:             FALSE
4:             FALSE
```

1. 피처 중요도(Feature Importance) 산출

■ 범주형 변수의 범주 결합

- **woebin {scorecard}:** 최적 범주 선택

```
> fine_class_cat$edu      # 석사,박사 결합
   variable         bin count count_distr  neg pos posprob      woe   bin_iv total_iv      breaks
1:      edu    고졸이하  5244      0.2871 4560 684  0.1304 -0.10817 0.003231 0.009329    고졸이하
2:      edu      전문대  5362      0.2935 4546 816  0.1522  0.07136 0.001533 0.009329      전문대
3:      edu        학사  5496      0.3009 4740 756  0.1376 -0.04680 0.000648 0.009329        학사
4:      edu 석사%,%박사  2166      0.1186 1806 360  0.1662  0.17619 0.003917 0.009329 석사%,%박사
   is_special_values
1:             FALSE
2:             FALSE
3:             FALSE
4:             FALSE
```

1. 피처 중요도(Feature Importance) 산출

■ 범주형 변수의 범주 결합

- **woebin {scorecard}:** 최적 범주 선택

```
> fine_class_cat$customer_grade
         variable                                    bin count count_distr  neg  pos posprob       woe
1: customer_grade                                  개인1  2480      0.1358 2110  370  0.1492  0.048012
2: customer_grade                                  개인2  4244      0.2323 3634  610  0.1437  0.004322
3: customer_grade                          개인3%,%법인1  7570      0.4144 6538 1032  0.1363 -0.057180
4: customer_grade 법인2%,%법인3%,%기타1%,%기타2%,%기타3  3974      0.2175 3370  604  0.1520  0.069858
      bin_iv total_iv                                  breaks is_special_values
1: 3.183e-04 0.002738                                  개인1             FALSE
2: 4.346e-06 0.002738                                  개인2             FALSE
3: 1.327e-03 0.002738                          개인3%,%법인1             FALSE
4: 1.088e-03 0.002738 법인2%,%법인3%,%기타1%,%기타2%,%기타3             FALSE
```

1. 피처 중요도(Feature Importance) 산출

■ 범주형 변수의 범주 결합

- **woebin {scorecard}:** 최적 범주 선택

```
> fine_class_cat$Provance
1: Provance   서울경기  6300     0.34487 5388 912  0.1448  0.012663 5.555e-05 0.0003962   서울경기
2: Provance 경상남북도  1596     0.08737 1378 218  0.1366 -0.054941 2.586e-04 0.0003962 경상남북도
3: Provance 전라남북도  3406     0.18645 2920 486  0.1427 -0.004178 3.250e-06 0.0003962 전라남북도
4: Provance 충청남북도  5202     0.28476 4450 752  0.1446  0.011029 3.478e-05 0.0003962 충청남북도
5: Provance     강원도  1764     0.09656 1516 248  0.1406 -0.021450 4.409e-05 0.0003962     강원도
   is_special_values
1:             FALSE
2:             FALSE
3:             FALSE
4:             FALSE
5:             FALSE
```

2. 파생 변수 생성

■ 범주의 그룹화에 대한 결정

- 범주형 변수 incentive는 기존 범주형 변수(4개 범주) 그대로 사용(IV=0.47)
- 연속형 변수 income은 woebin()에 의한 범주형 구분(4개 범주)을 사용(IV=0.1613)
- 범주형 변수 residence는 기존 범주형 변수(3개 범주) 그대로 사용(IV=0.12)
- 연속형 변수 claim_size는 iv.mult()에 의한 범주형 구분(5개 범주)을 사용(IV=0.10)
- 범주형 변수 channel은 기존 범주형 변수(4개 범주) 그대로 사용(IV=0.09)
- 범주형 변수 marrage는 기존 범주형 변수(3개 범주) 그대로 사용(IV=0.09)
- 연속형 변수 retention은 iv.mult()에 의한 범주형 구분(6개 범주)을 사용(IV=0.08)
- 연속형 변수 period_keep은 iv.mult()에 의한 범주형 구분(3개 범주)을 사용(IV=0.03)
- 범주형 변수 house_size는 기존 범주형 변수(3개 범주) 그대로 사용(IV=0.02)
- 범주형 변수 n_product은 기존 범주형 변수(4개 범주) 그대로 사용(IV=0.01712)
- 범주형 변수 work은 기존 범주형 변수(2개 범주) 그대로 사용(IV=0.01)
- 연속형 변수 period_claim은 iv.mult()에 의한 범주형 구분(6개 범주)을 사용(IV=0.01)
- 범주형 변수 edu는 woebin()에 의한 범주형 구분(4개 범주)을 사용(IV=0.0093)
- 범주형 변수 customer_grade는 {scorecard}에 의한 범주형 구분(5개 범주)을 사용(IV=0.0045)

2. 파생 변수 생성

■ 새로운 범주형 변수 생성

- **income는 woebin {scorecard}에 의한 범주형 구분(4개 범주)을 사용(IV=0.1613)**

```
>  health = health %>% mutate(income_g = ifelse(income < 2000000, 1,
                                          ifelse(income < 26000000, 2,
                                          ifelse(income < 38000000, 3, 4))))

> iv.mult(df=data.frame(health), y= "renewal", vars=c("income_g"))
Information Value 0.16
[[1]]
  variable     class outcome_0 outcome_1  pct_0  pct_1   odds      woe      miv
1 income_g    (;1.5)      4238       396 0.2708 0.1514 1.7887  0.58148 0.069422
2 income_g <1.5;2.5)      1628       564 0.1040 0.2156 0.4824 -0.72890 0.081334
3 income_g <2.5;3.5)      1996       432 0.1275 0.1651 0.7722 -0.25848 0.009722
4 income_g    <3.5;)      7790      1224 0.4977 0.4679 1.0637  0.06176 0.001841
                                                                    sql
1                      when income_g < 1.5 then 0.58148025682055
2 when income_g >= 1.5 AND income_g < 2.5 then -0.728898975688787
3 when income_g >= 2.5 AND income_g < 3.5 then -0.258477402125915
4                    when income_g >= 3.5 then 0.0617644050393087
```

2. 파생 변수 생성

■ 새로운 범주형 변수 생성

- **claim_size는 iv.mult()에 의한 범주형 구분(5개 범주)을 사용(IV=0.10)**

```
> health = health %>% mutate(claim_size_g = ifelse(claim_size < 255, 1,
                                            ifelse(claim_size < 415, 2,
                                            ifelse(claim_size < 495, 3,
                                            ifelse(claim_size < 695, 4, 5 )))))
> iv.mult(df=data.frame(health), y= "renewal", vars=c( "claim_size_g"))
Information Value 0.1
[[1]]
      variable     class outcome_0 outcome_1  pct_0  pct_1   odds      woe      miv
1 claim_size_g    (;1.5)      3875       384 0.2476 0.1468 1.6866  0.52271 0.052680
2 claim_size_g <1.5;2.5)      4719       870 0.3015 0.3326 0.9066 -0.09809 0.003048
3 claim_size_g <2.5;3.5)      1804       474 0.1153 0.1812 0.6361 -0.45240 0.029829
4 claim_size_g <3.5;4.5)      2767       546 0.1768 0.2087 0.8470 -0.16605 0.005303
5 claim_size_g    <4.5;)      2487       342 0.1589 0.1307 1.2154  0.19507 0.005493
                                                                              sql
1                          when claim_size_g < 1.5 then 0.522706118446507
2 when claim_size_g >= 1.5 AND claim_size_g < 2.5 then -0.0980932906754556
3  when claim_size_g >= 2.5 AND claim_size_g < 3.5 then -0.452397891825721
4  when claim_size_g >= 3.5 AND claim_size_g < 4.5 then -0.166052266711193
5                         when claim_size_g >= 4.5 then 0.195069435987586
```

2. 파생 변수 생성

■ 새로운 범주형 변수 생성

- **retention은 iv.mult()에 의한 범주형 구분(6개 범주)을 사용(IV=0.08)**

```
> health = health %>% mutate(retention_g = ifelse(retention < 1.15, 1,
                                          ifelse(retention < 2.25, 2,
                                          ifelse(retention < 3.35, 3,
                                          ifelse(retention < 5.55, 4,
                                          ifelse(retention < 6.85, 5, 6 ))))))

> iv.mult(df=data.frame(health), y= "renewal", vars=c("retention_g"))
Information Value 0.08
[[1]]
     variable     class outcome_0 outcome_1  pct_0   pct_1   odds      woe       miv
1 retention_g    (;1.5)      1738       360 0.1110 0.13761 0.8069 -0.21457 5.702e-03
2 retention_g <1.5;2.5)      1800       214 0.1150 0.08180 1.4058  0.34061 1.131e-02
3 retention_g <2.5;3.5)      1614       380 0.1031 0.14526 0.7099 -0.34265 1.444e-02
4 retention_g <3.5;4.5)      3448       586 0.2203 0.22401 0.9834 -0.01672 6.212e-05
5 retention_g <4.5;5.5)      2198       188 0.1404 0.07187 1.9541  0.66991 4.593e-02
6 retention_g    <5.5;)      4854       888 0.3101 0.33945 0.9136 -0.09037 2.650e-03
```

2. 파생 변수 생성

■ 새로운 범주형 변수 생성

- **period_keep은 iv.mult()에 의한 범주형 구분(3개 범주)을 사용(IV=0.03)**

```
> health = health %>% mutate(period_keep_g = ifelse(period_keep < 37.5, 1,
                                                    ifelse(period_keep < 72.5, 2, 3)))

> iv.mult(df=data.frame(health), y= "renewal", vars=c( "period_keep_g"))
Information Value 0.03
[[1]]
       variable     class outcome_0 outcome_1  pct_0  pct_1  odds     woe      miv
1 period_keep_g    (;1.5)      6214       912 0.3970 0.3486 1.139  0.1300 0.006289
2 period_keep_g <1.5;2.5)      5636      1176 0.3601 0.4495 0.801 -0.2219 0.019851
3 period_keep_g    <2.5;)      3802       528 0.2429 0.2018 1.203  0.1852 0.007608
                                                                          sql
1                           when period_keep_g < 1.5 then 0.129967829125448
2 when period_keep_g >= 1.5 AND period_keep_g < 2.5 then -0.221896526193215
3                          when period_keep_g >= 2.5 then 0.185233968588899
```

2. 파생 변수 생성

■ 새로운 범주형 변수 생성

- **period_claim은 iv.mult()에 의한 범주형 구분(2개 범주)을 사용(IV=0.01)**

```
> health = health %>% mutate(period_claim_g = ifelse(period_claim < 24.5, 1, 2))

> iv.mult(df=data.frame(health), y= "renewal", vars=c( "period_claim_g"))
Information Value 0.01
[[1]]
        variable  class outcome_0 outcome_1  pct_0  pct_1   odds      woe      miv
1 period_claim_g (;1.5)     12135       2136 0.7753 0.8165 0.9495 -0.05179 0.002135
2 period_claim_g <1.5;)      3517        480 0.2247 0.1835 1.2246  0.20263 0.008351
                                                 sql
1 when period_claim_g < 1.5 then -0.0517933526725974
2  when period_claim_g >= 1.5 then 0.202625257818832
```

2. 파생 변수 생성

■ 새로운 범주형 변수 생성

- **edu는 woebin()에 의한 범주형 구분(4개 범주)을 사용(IV=0.0093)**

```
> health = health %>% mutate(edu_g = fct_collapse(edu,
                                                  '1' = '고졸이하',
                                                  '2' = '전문대',
                                                  '3' = '학사',
                                                  '4' =c('석사', '박사')))

> iv.mult(df=data.frame(health), y= "renewal", vars=c("edu_g"))
Information Value 0.01
[[1]]
  variable class outcome_0 outcome_1  pct_0  pct_1   odds      woe      miv
1    edu_g     1      4560       684 0.2913 0.2615 1.1142  0.10817 0.003231
2    edu_g     2      4546       816 0.2904 0.3119 0.9311 -0.07136 0.001533
3    edu_g     3      4740       756 0.3028 0.2890 1.0479  0.04680 0.000648
4    edu_g     4      1806       360 0.1154 0.1376 0.8385 -0.17619 0.003917
```

2. 파생 변수 생성

■ 새로운 범주형 변수 생성

- **customer_grade는 woebin()에 의한 범주형 구분(4개 범주)을 사용(IV=0.0)**

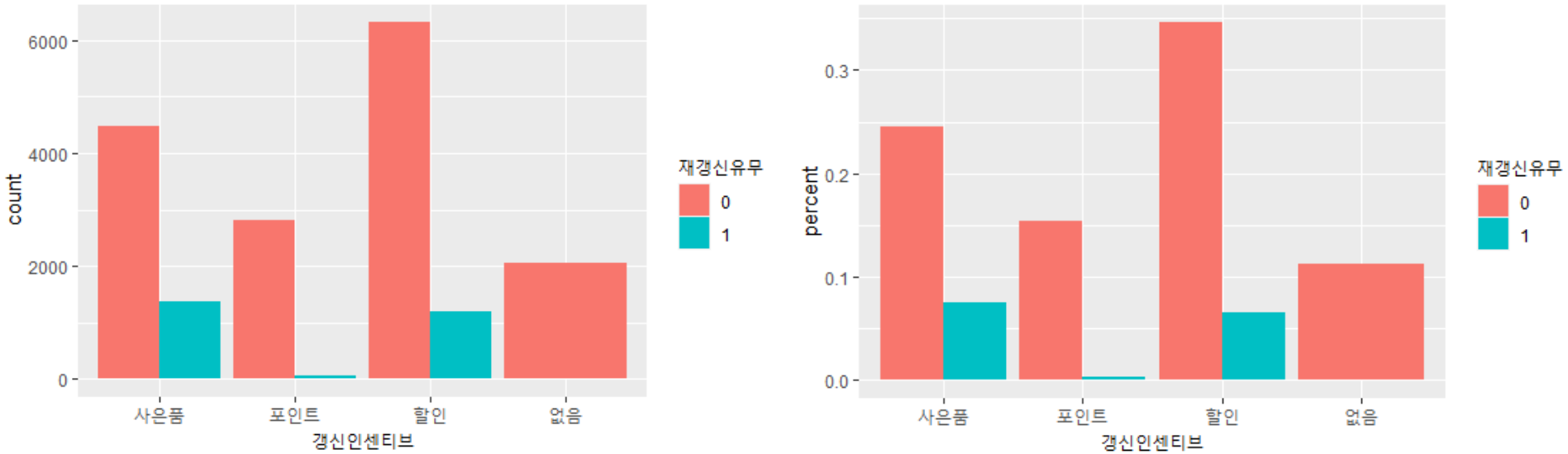
```
> health = health %>% mutate(customer_grade_g =
                               fct_collapse(customer_grade,
                                            '1' = '개인1',
                                            '2' = '개인2',
                                            '3' = c('개인3', '법인1'),
                                            '4' =c('법인2', '법인3', '기타1', '기타2', '기타3')))
> iv.mult(df=data.frame(health), y= "renewal", vars=c( "customer_grade_g"))
Information Value 0
[[1]]
          variable class outcome_0 outcome_1     pct_0     pct_1      odds          woe
1 customer_grade_g     1      2110       370 0.1348071 0.1414373 0.9531223 -0.048012050
2 customer_grade_g     2      3634       610 0.2321748 0.2331804 0.9956873 -0.004321979
3 customer_grade_g     3      6538      1032 0.4177102 0.3944954 1.0588468  0.057180370
4 customer_grade_g     4      3370       604 0.2153079 0.2308869 0.9325258 -0.069858445
           miv
1 3.183322e-04
2 4.346295e-06
3 1.327430e-03
4 1.088318e-03
```

3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 갱신인센티브(incentive)

```
> health %>% ggplot(aes(x=incentive, fill=renewal)) +
             geom_bar(position = "dodge") +
             labs(x= "갱신인센티브", y= "count", fill = " 재갱신유무", )
> health %>% ggplot(aes(x=incentive, fill=renewal)) +
             geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
             labs(x= " 갱신인센티브", y= "percent", fill = " 재갱신유무")
```

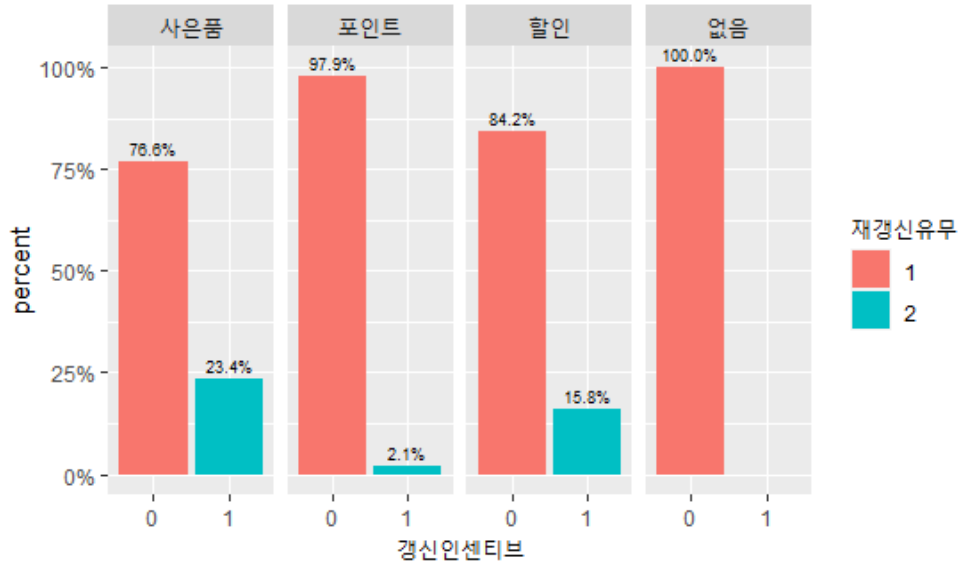


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 갱신인센티브(incentive)

```
> health %>% ggplot(aes(x= renewal, group=incentive)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "갱신인센티브", y= "percent", fill = " 재갱신유무") +
             facet_grid(~incentive) +
             scale_y_continuous(labels = scales::percent)
```

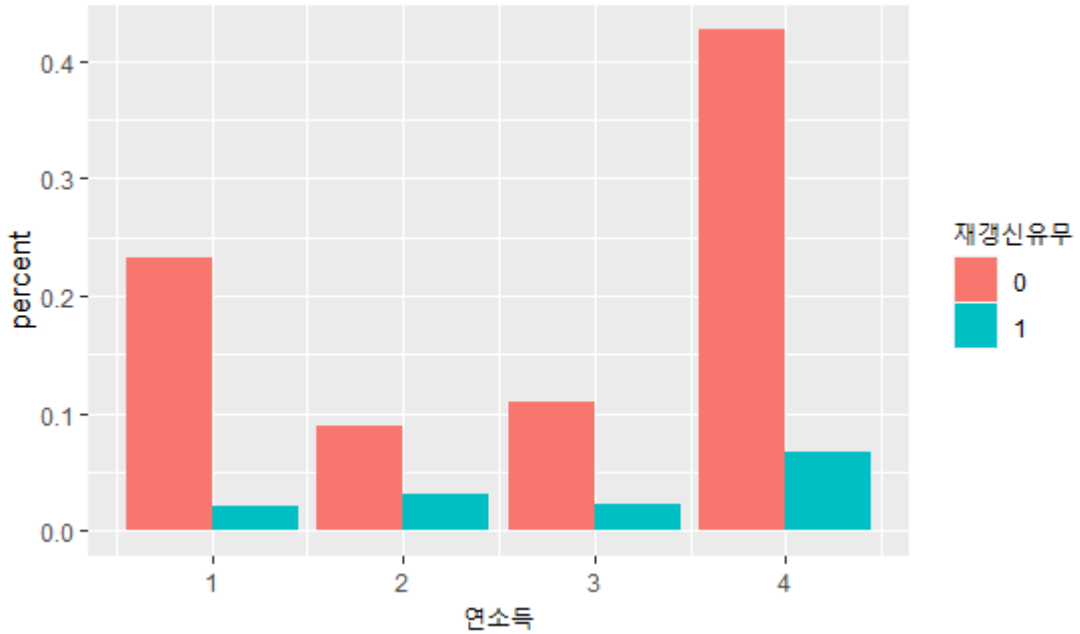


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- 연소득(income_g)

```
> health %>% ggplot(aes(x=income_g, fill=renewal)) +
             geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
             labs(x= "연소득", y= "percent", fill = "재갱신유무")
```

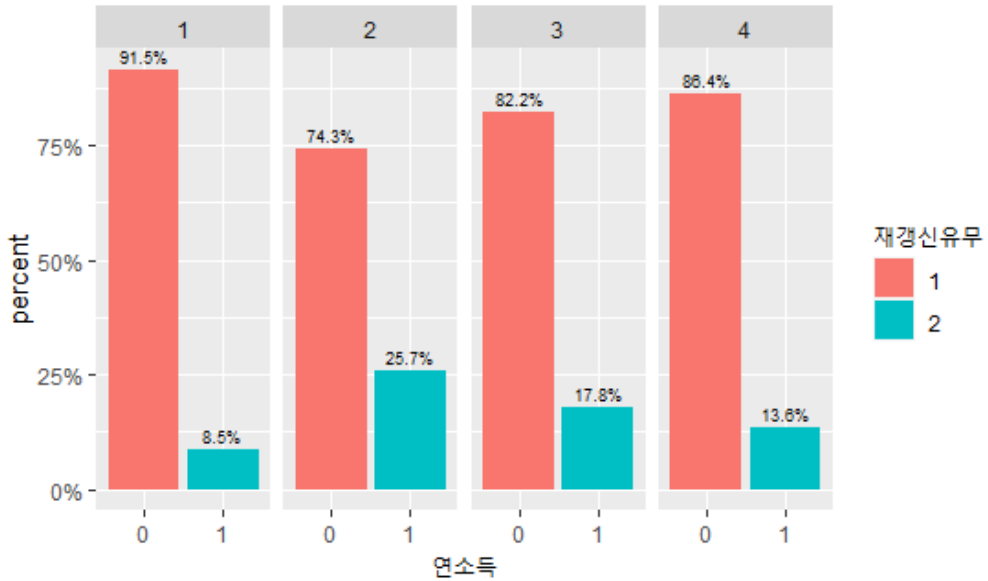


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- 연소득(income_g)

```
> health %>% ggplot(aes(x= renewal, group=income_g)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "연소득", y= "percent", fill = " 재갱신유무") +
             facet_grid(~ income_g) +
             scale_y_continuous(labels = scales::percent)
```

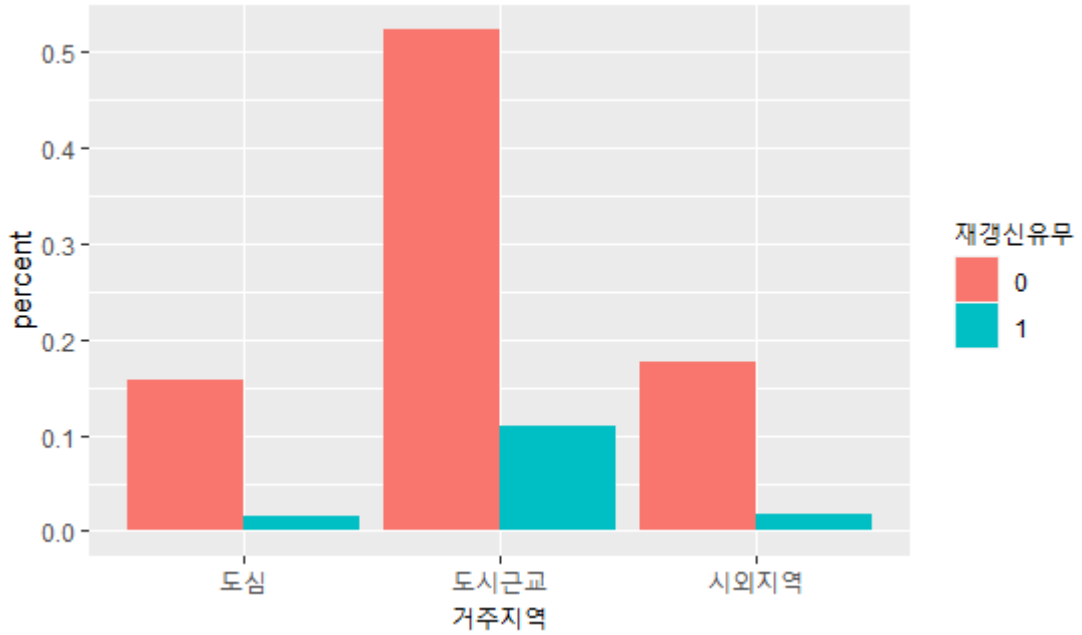


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 거주지역(residence)

```
> health %>% ggplot(aes(x=residence, fill=renewal)) +
             geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
             labs(x= "거주지역", y= "percent", fill = "재갱신유무")
```

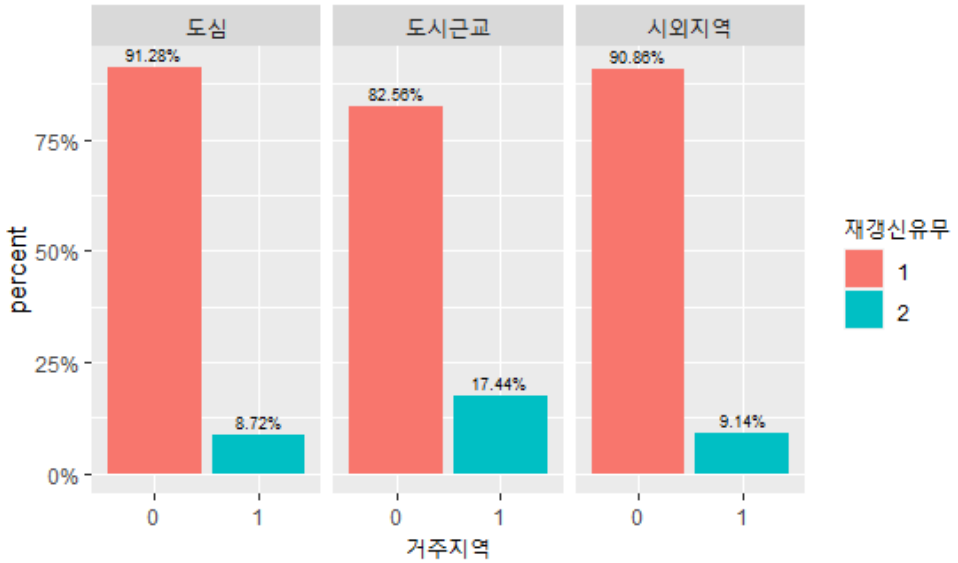


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- 거주지역(residence)

```
> health %>% ggplot(aes(x= renewal, group=residence)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "거주지역", y= "percent", fill = " 재갱신유무") +
             facet_grid(~ residence) +
             scale_y_continuous(labels = scales::percent)
```

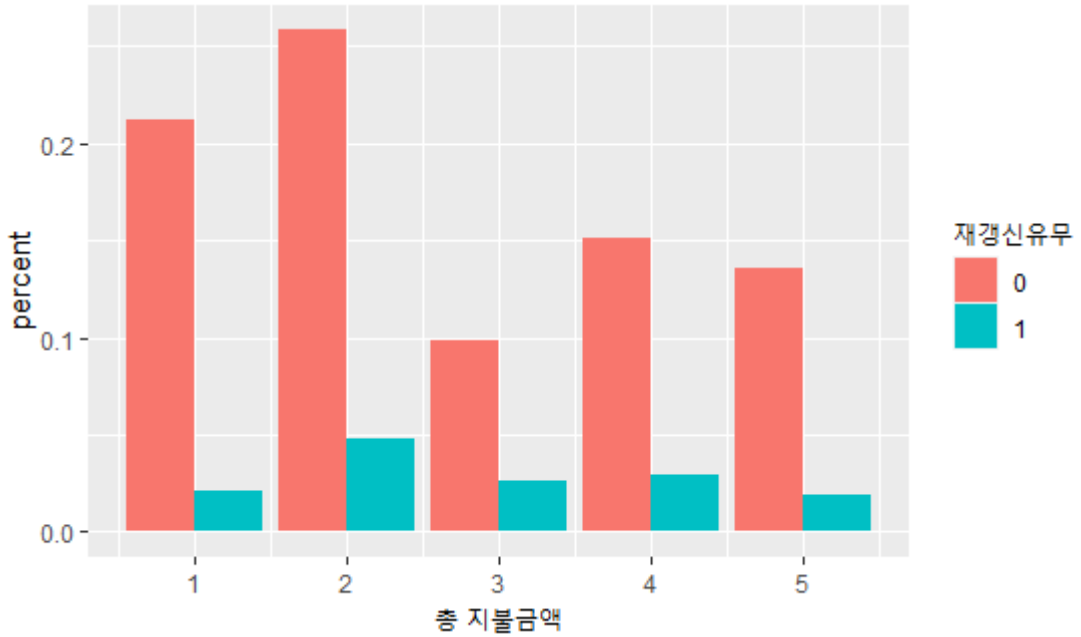


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- **총 지불금액(claim_size_g)**

```
> health %>% ggplot(aes(x=claim_size_g, fill=renewal)) +
             geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
             labs(x= "총 지불금액", y= "percent", fill = "재갱신유무")
```

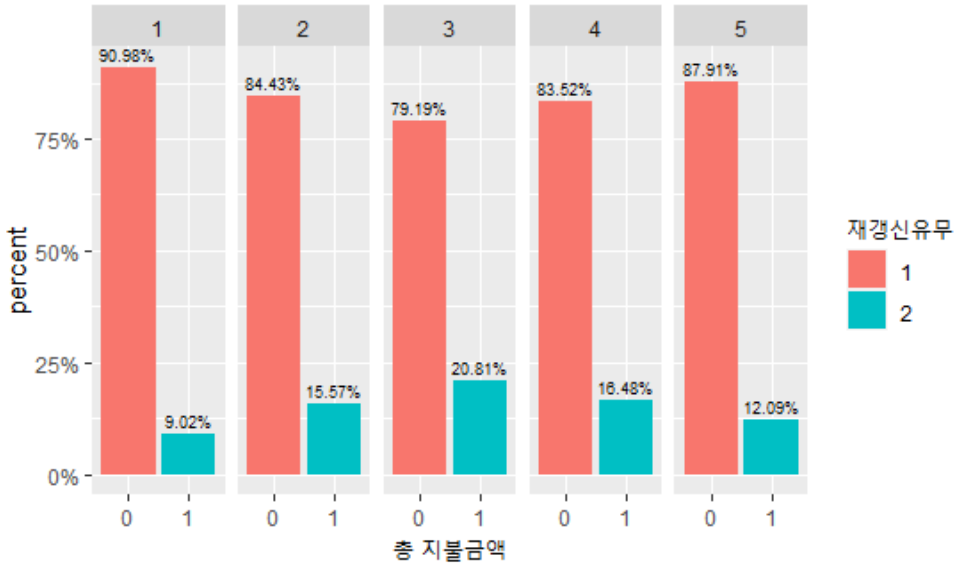


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- **총 지불금액(claim_size_g)**

```
> health %>% ggplot(aes(x= renewal, group=claim_size_g)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "총 지불금액", y= "percent", fill = " 재갱신유무") +
             facet_grid(~ claim_size_g) +
             scale_y_continuous(labels = scales::percent)
```

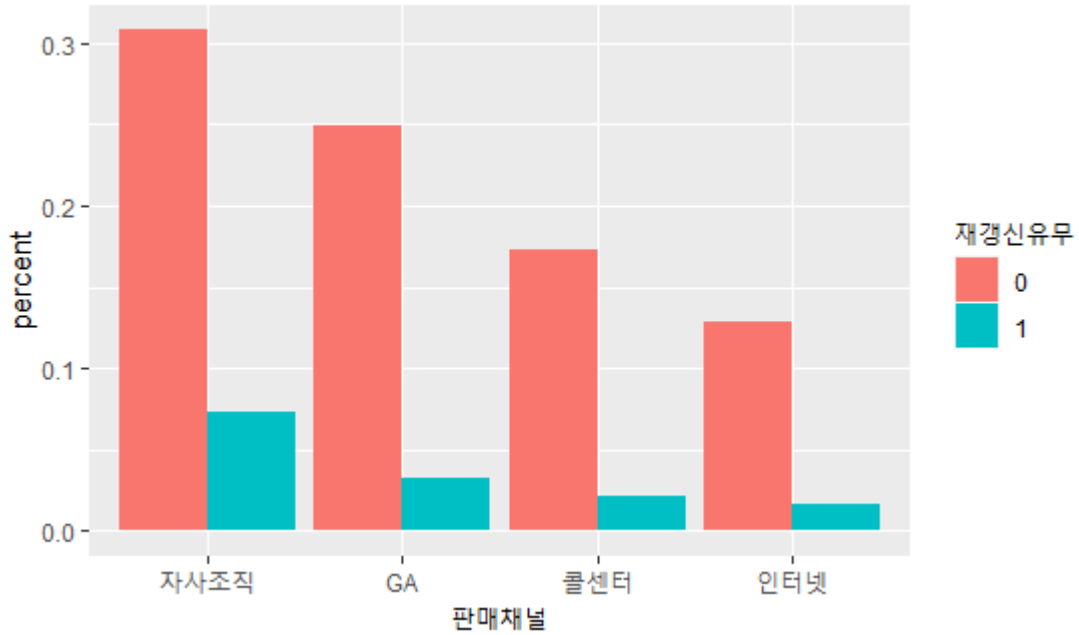


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- **판매채널(channel)**

```
> health %>% ggplot(aes(x= channel, fill=renewal)) +
                geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
                labs(x= "판매채널", y= "percent", fill = "재갱신유무")
```

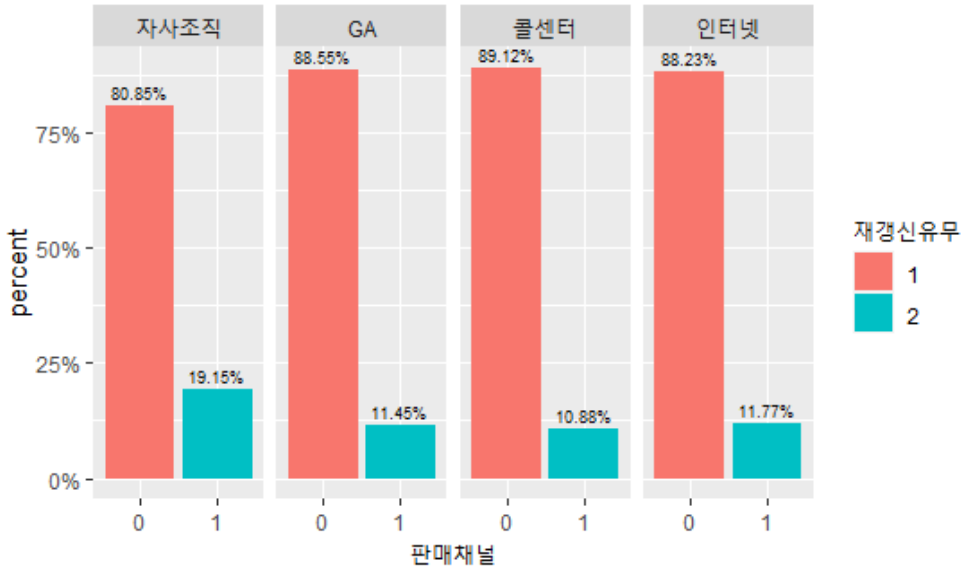


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 판매채널(channel)

```
> health %>% ggplot(aes(x= renewal, group= channel)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "판매채널", y= "percent", fill = " 재갱신유무") +
             facet_grid(~ channel) +
             scale_y_continuous(labels = scales::percent)
```

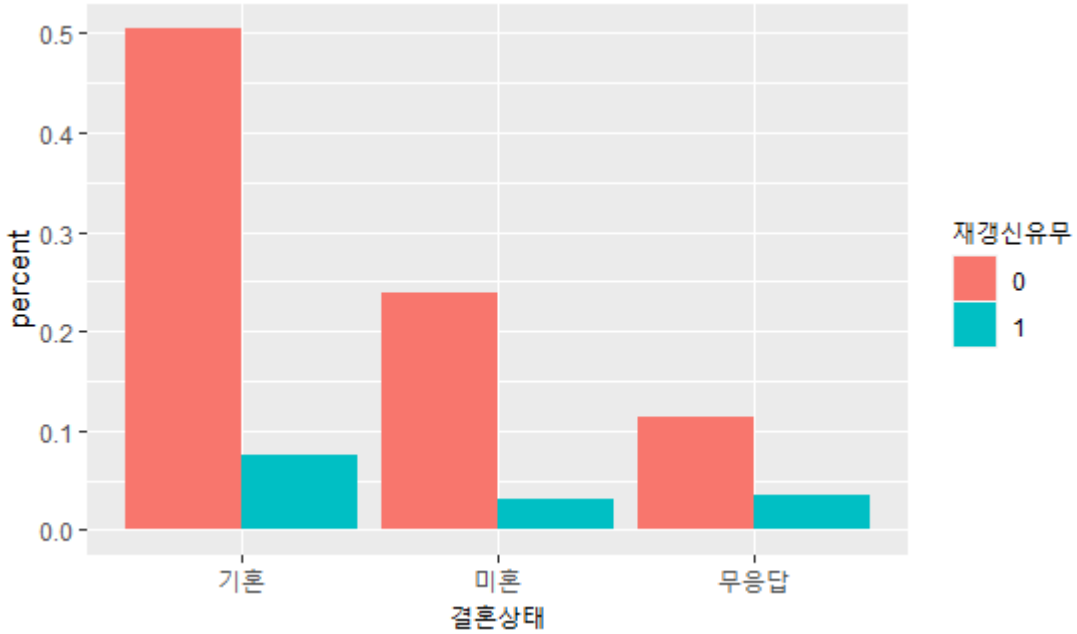


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 결혼상태(marrage)

```
> health %>% ggplot(aes(x= marrage, fill=renewal)) +
             geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
             labs(x= "결혼상태", y= "percent", fill = "재갱신유무")
```

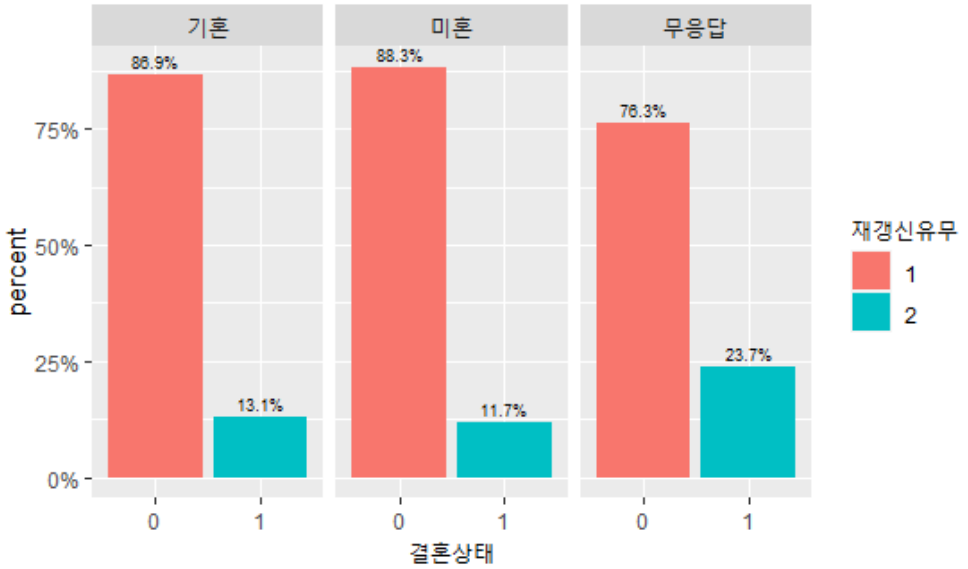


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 결혼상태(marrage)

```
> health %>% ggplot(aes(x= renewal, group= marrage)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "결혼상태", y= "percent", fill = " 재갱신유무") +
             facet_grid(~ marrage) +
             scale_y_continuous(labels = scales::percent)
```

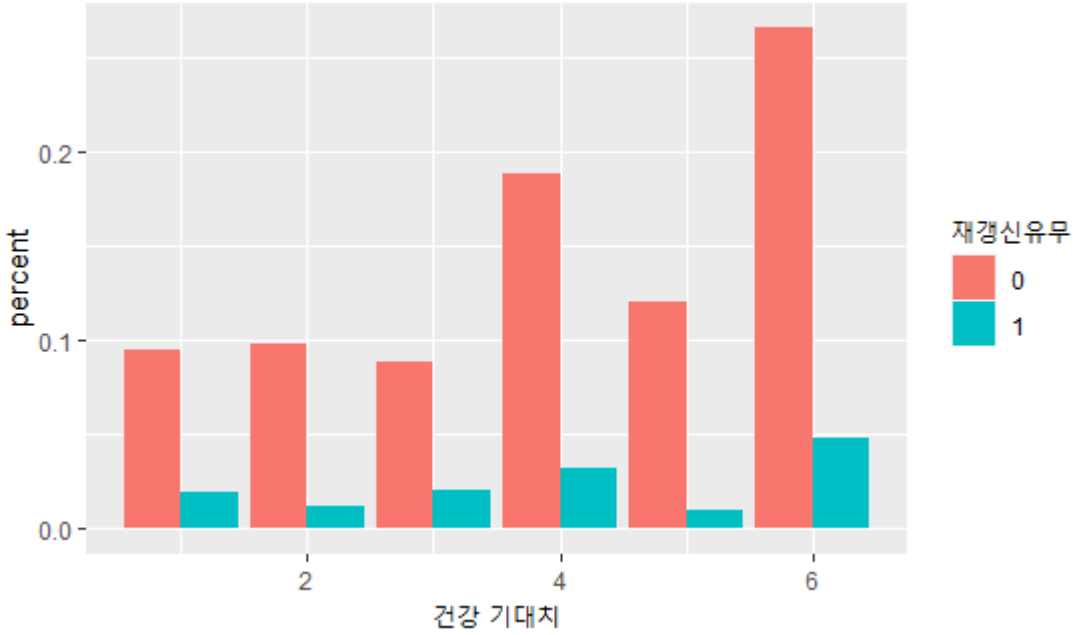


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- 건강 기대치**(retention_g)**

```
> health %>% ggplot(aes(x= retention_g, fill=renewal)) +
             geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
             labs(x= "건강 기대치", y= "percent", fill = "재갱신유무")
```

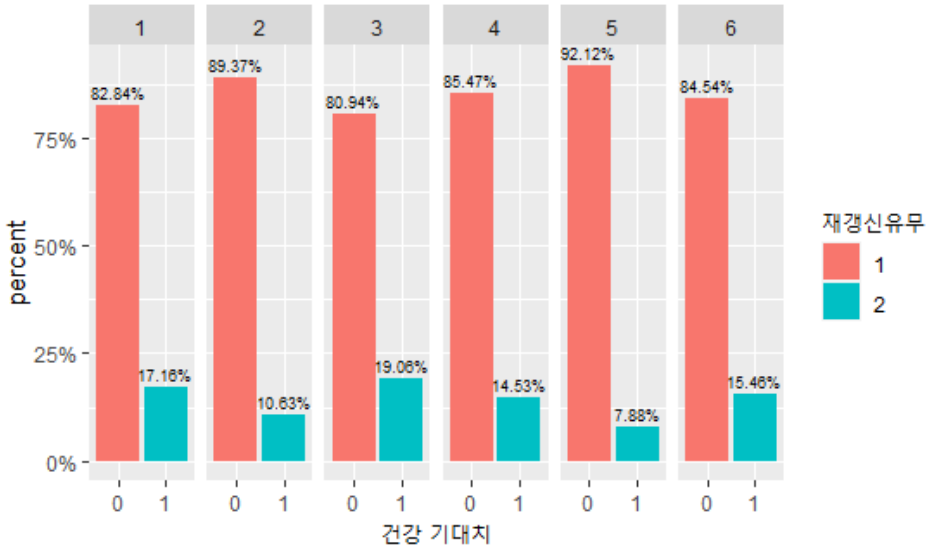


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- 건강 기대치**(retention_g)**

```
> health %>% ggplot(aes(x= renewal, group= retention_g)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "건강 기대치", y= "percent", fill = " 재갱신유무") +
             facet_grid(~ retention_g) +
             scale_y_continuous(labels = scales::percent)
```

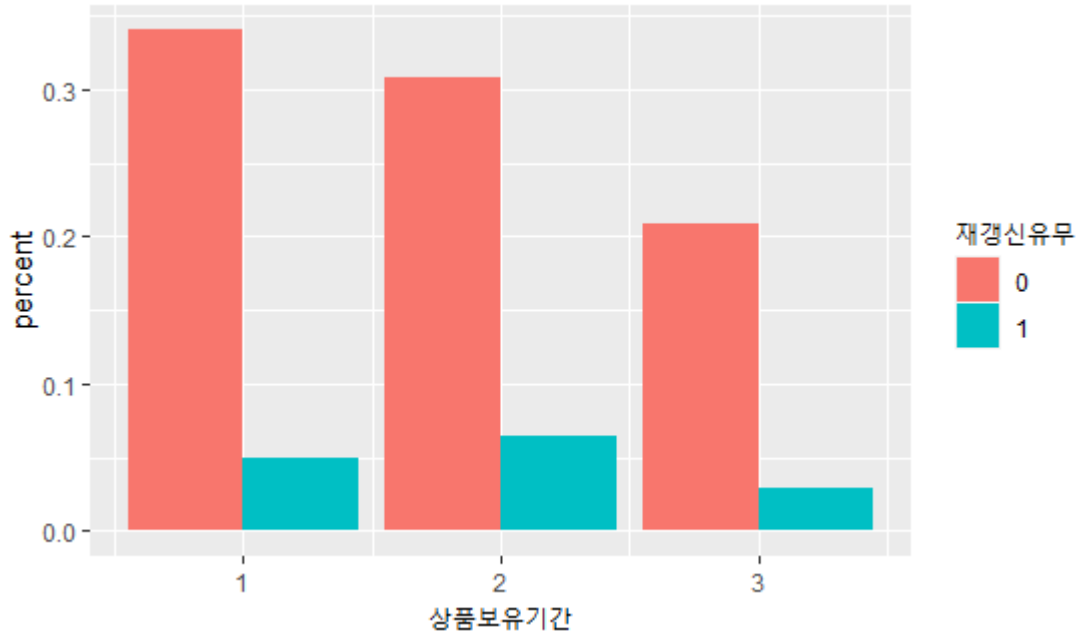


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- 상품보유기간**(retention_g)**

```
> health %>% ggplot(aes(x= period_keep_g, fill=renewal)) +
             geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
             labs(x= "상품보유기간", y= "percent", fill = "재갱신유무")
```

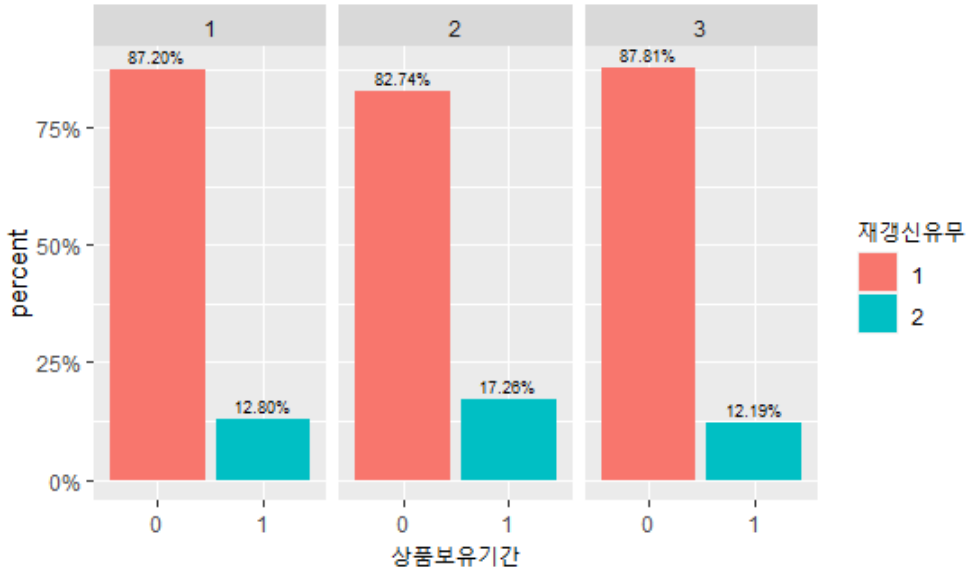


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- 상품보유기간**(retention_g)**

```
> health %>% ggplot(aes(x= renewal, group= period_keep_g)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "상품보유기간", y= "percent", fill = " 재갱신유무") +
             facet_grid(~ period_keep_g) +
             scale_y_continuous(labels = scales::percent)
```

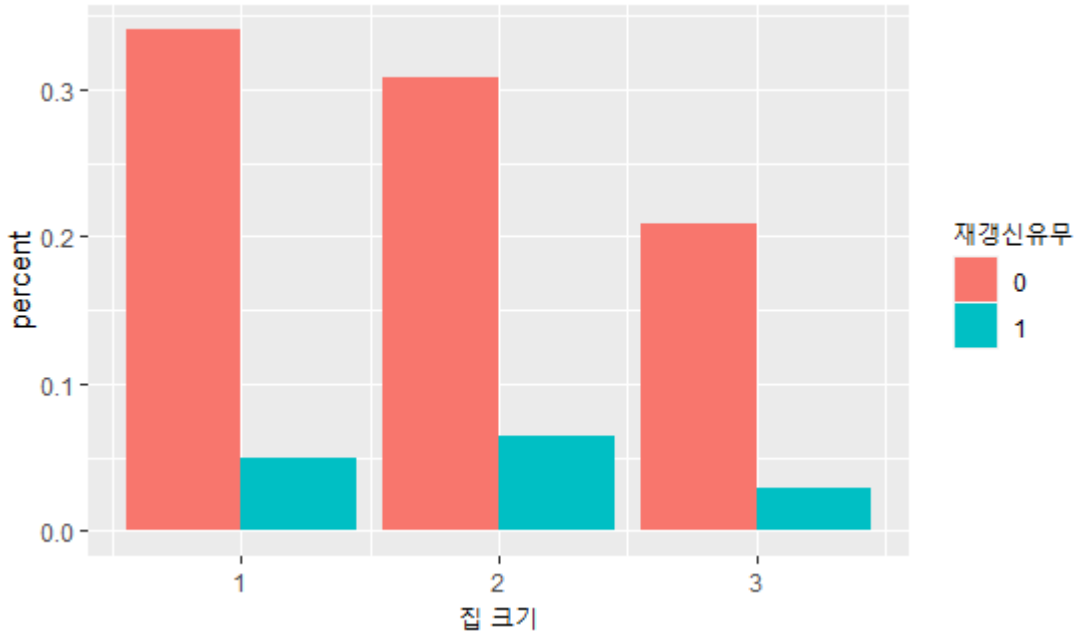


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- 집 크기(house_size)

```
> health %>% ggplot(aes(x= house_size, fill=renewal)) +
             geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
             labs(x= "집 크기", y= "percent", fill = "재갱신유무")
```

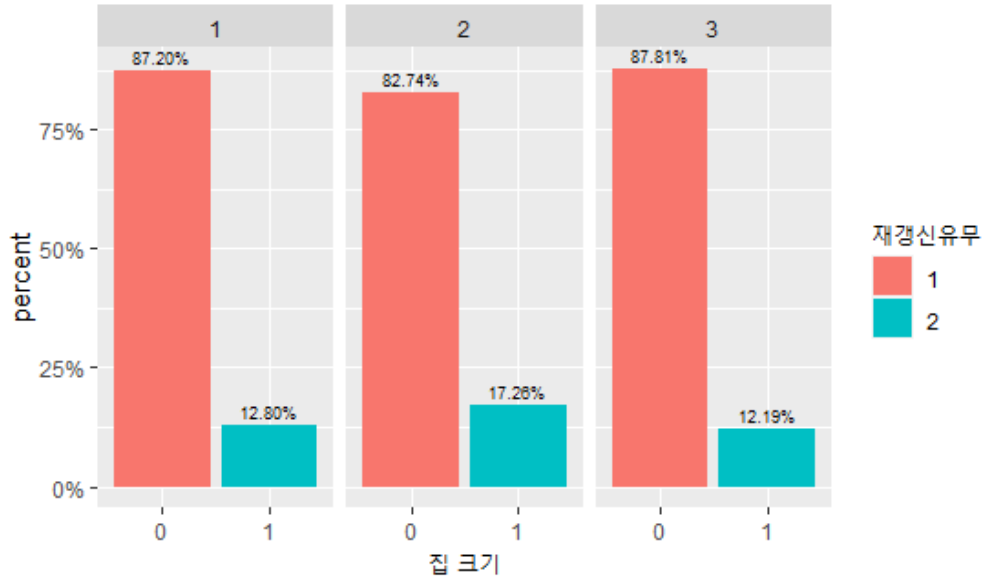


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 집 크기(house_size)

```
> health %>% ggplot(aes(x= renewal, group= house_size)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "집 크기", y= "percent", fill = " 재갱신유무") +
             facet_grid(~ house_size) +
             scale_y_continuous(labels = scales::percent)
```

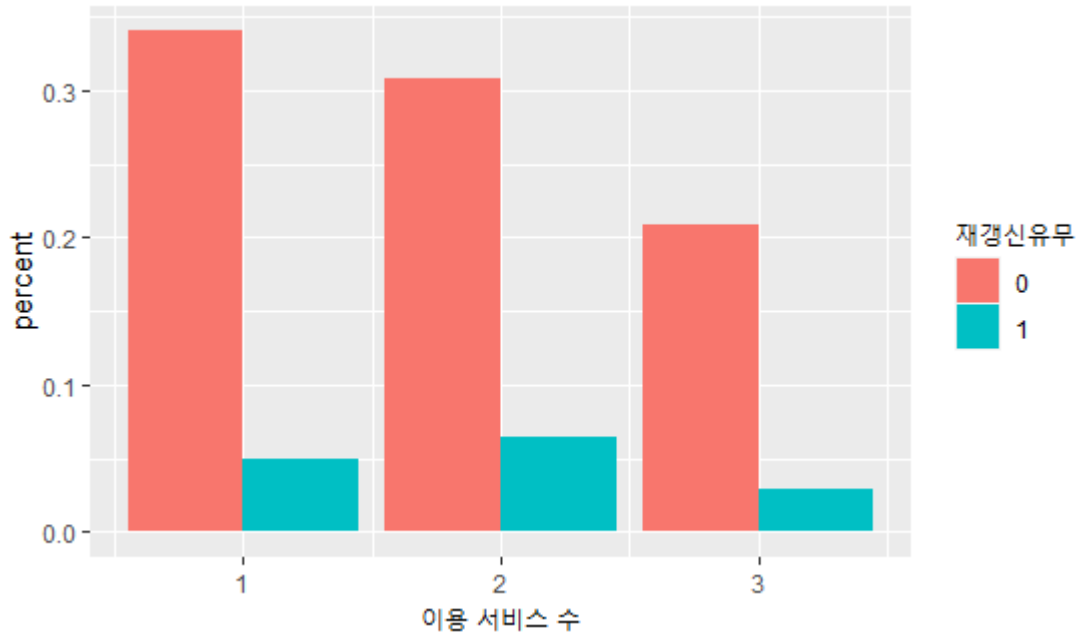


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 이용 서비스 수(n_product)

```
> health %>% ggplot(aes(x= n_product, fill=renewal)) +
             geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
             labs(x= "이용 서비스 수", y= "percent", fill = "재갱신유무")
```

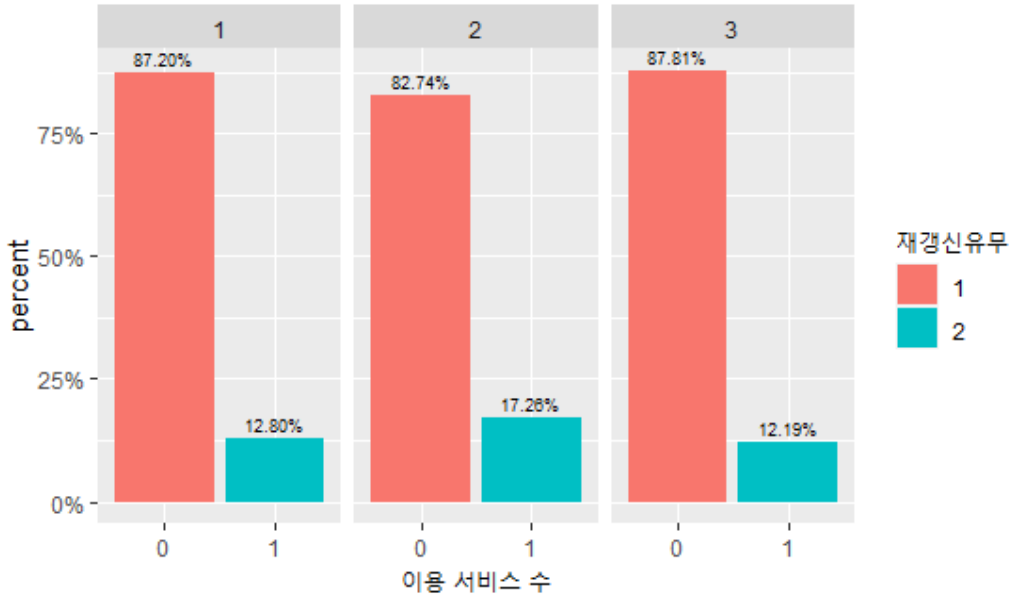


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 이용 서비스 수(n_product)

```
> health %>% ggplot(aes(x= renewal, group= n_product)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "이용 서비스 수", y= "percent", fill = " 재갱신유무") +
             facet_grid(~ n_product) +
             scale_y_continuous(labels = scales::percent)
```

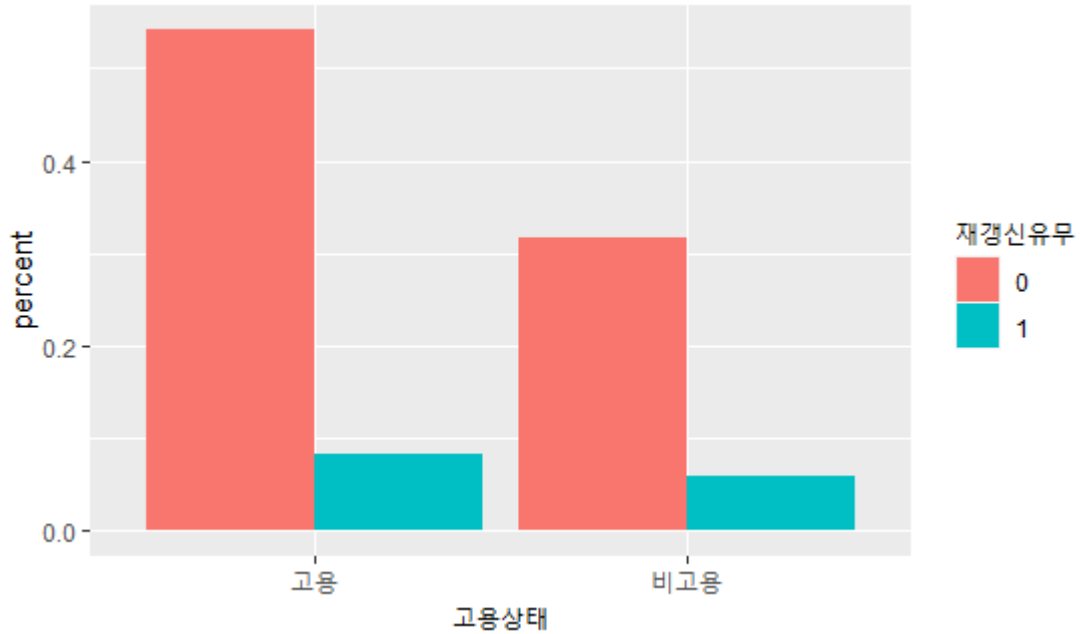


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 고용상태(work)

```
> health %>% ggplot(aes(x= work, fill=renewal)) +
             geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
             labs(x= "고용상태", y= "percent", fill = "재갱신유무")
```

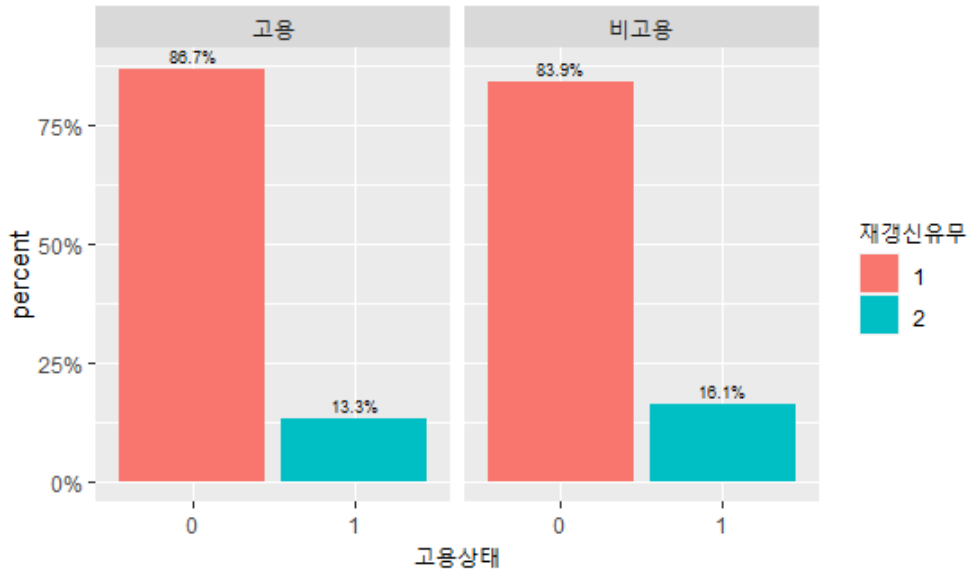


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 고용상태(work)

```
> health %>% ggplot(aes(x= renewal, group= period_claim_g)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "고용상태", y= "percent", fill = " 재갱신유무") +
             facet_grid(~ work) +
             scale_y_continuous(labels = scales::percent)
```

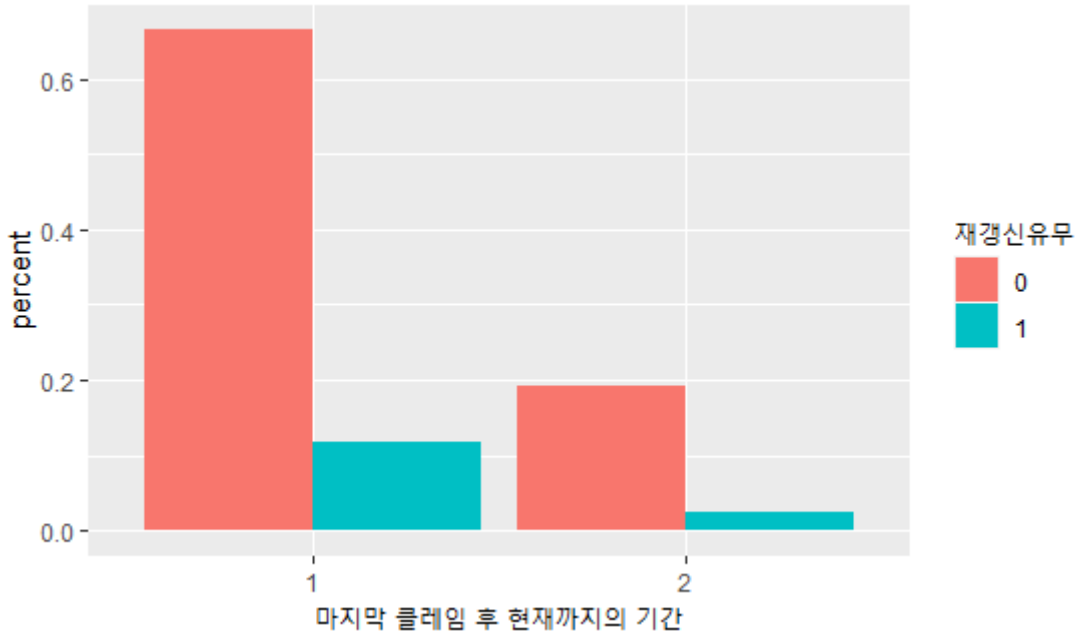


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- 마지막 클레임 후 현재까지의 기간**(period_claim_g)**

```
> health %>% ggplot(aes(x= period_claim_g, fill=renewal)) +
             geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
             labs(x= "마지막 클레임 후 현재까지의 기간", y= "percent", fill = "재갱신유무")
```

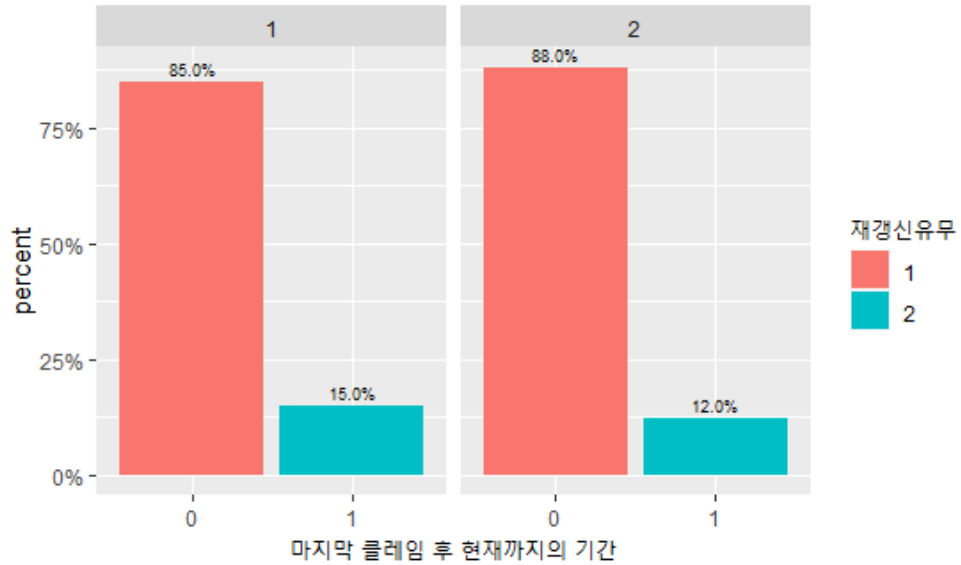


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- 마지막 클레임 후 현재까지의 기간**(period_claim_g)**

```
> health %>% ggplot(aes(x= renewal, group= period_claim_g)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "마지막 클레임 후 현재까지의 기간", y= "percent", fill = " 재갱신유무") +
             facet_grid(~ period_claim_g) +
             scale_y_continuous(labels = scales::percent)
```

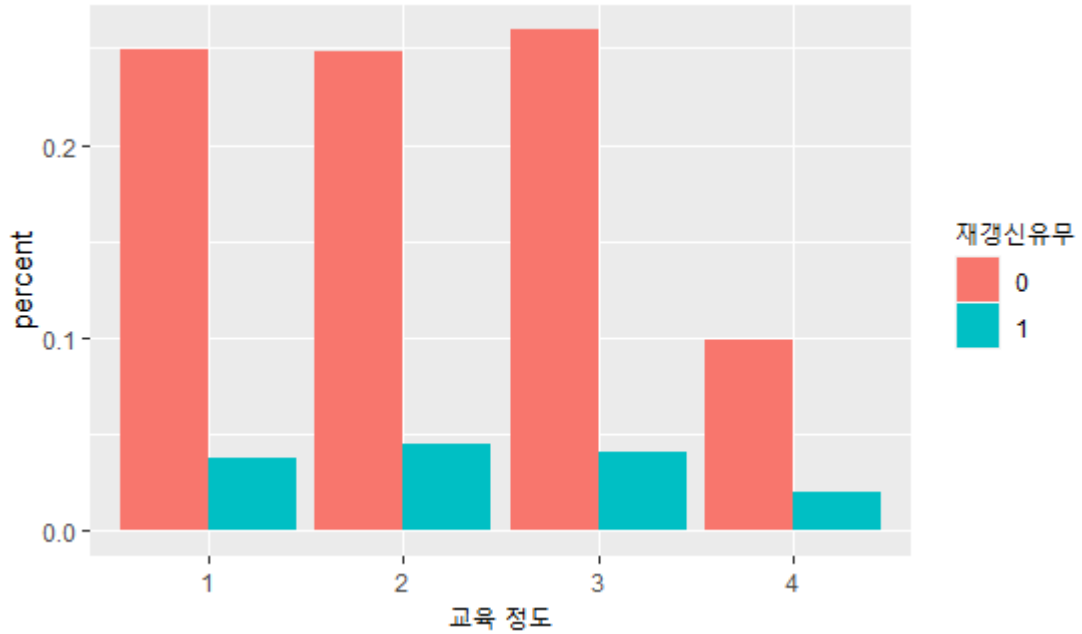


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- 교육 정도(edu_g)

```
> health %>% ggplot(aes(x= edu_g, fill=renewal)) +
             geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
             labs(x= "교육 정도", y= "percent", fill = "재갱신유무")
```

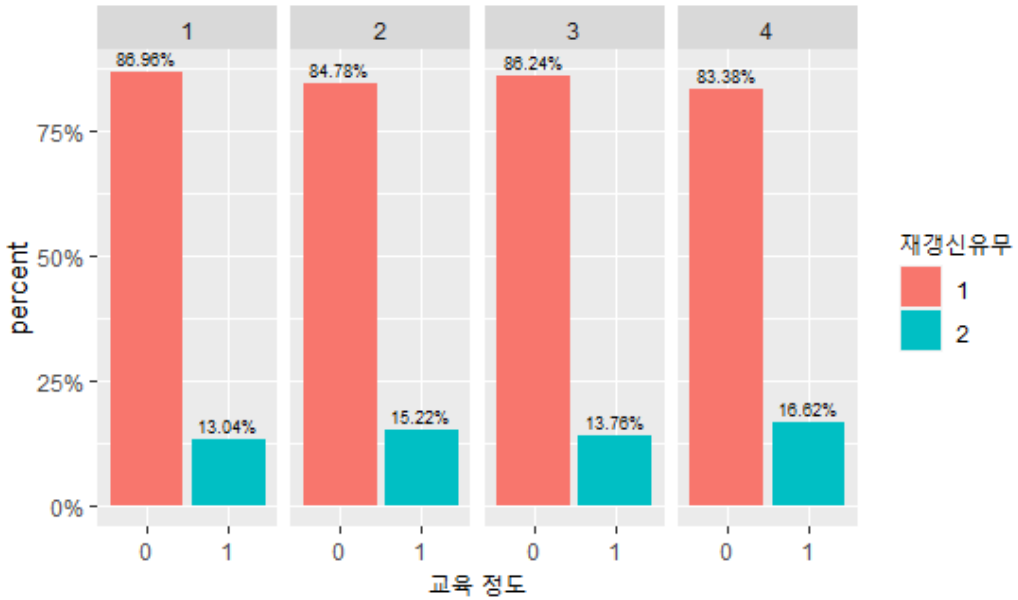


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- 교육 정도(edu_g)

```
> health %>% ggplot(aes(x= renewal, group= edu_g)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "교육 정도", y= "percent", fill = " 재갱신유무") +
             facet_grid(~ work) +
             scale_y_continuous(labels = scales::percent)
```

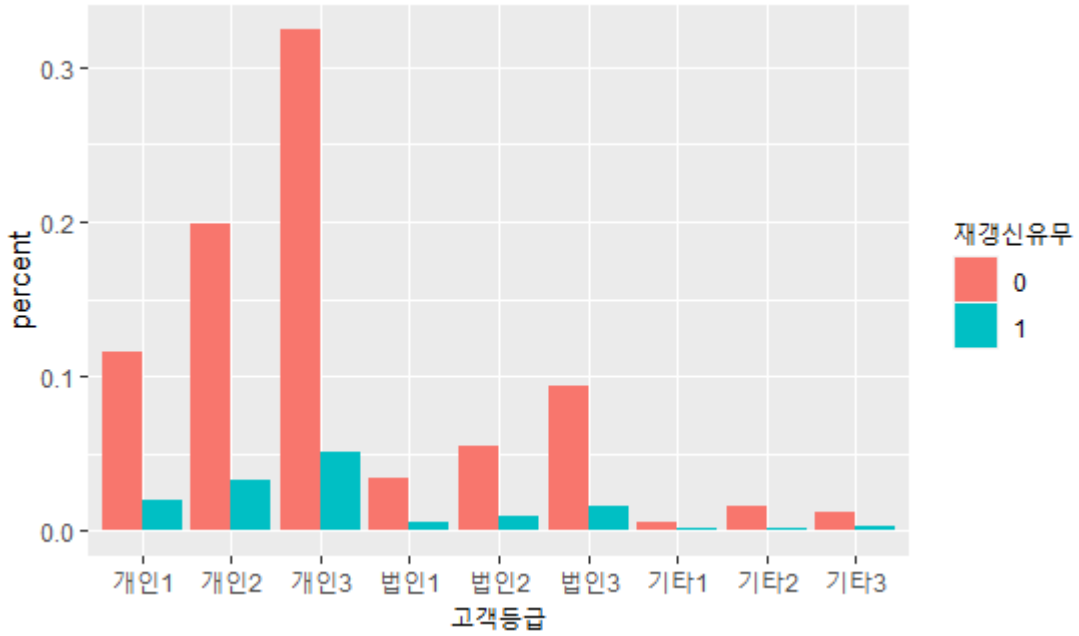


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 고객등급(customer_grade)

```
> health %>% ggplot(aes(x= customer_grade, fill=renewal)) +
                geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
                labs(x= "고객등급", y= "percent", fill = "재갱신유무")
```

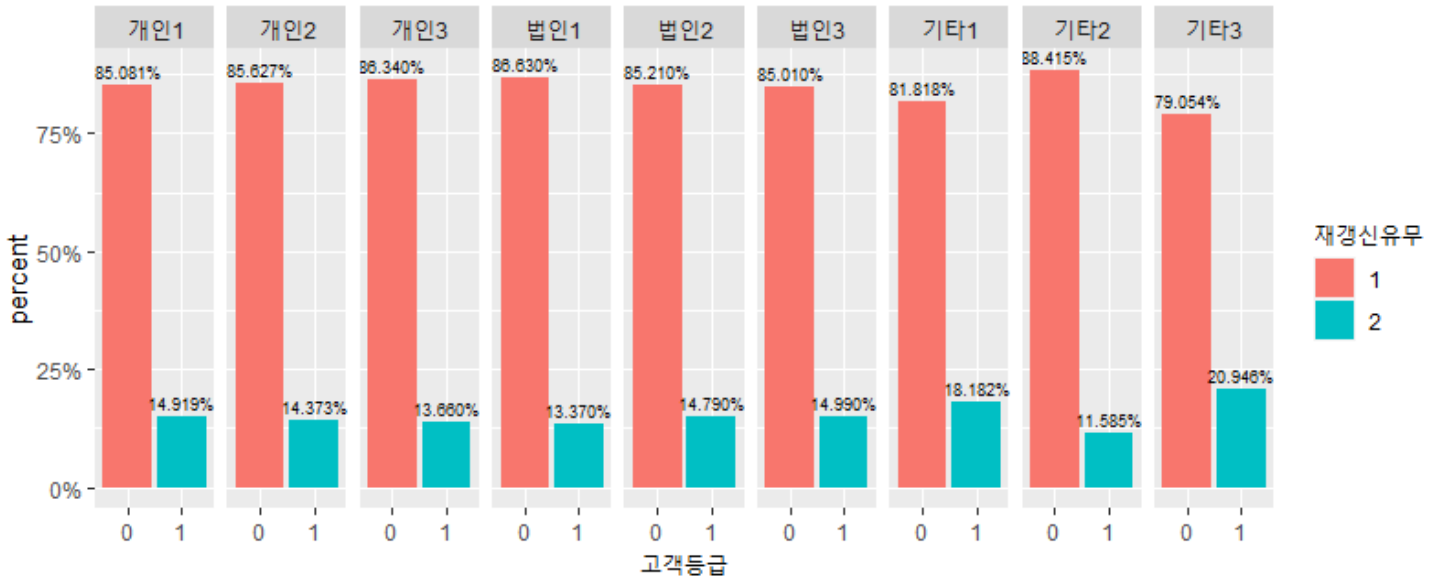


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

- **고객등급(customer_grade)**

```
> health %>% ggplot(aes(x= renewal, group= customer_grade)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "고객등급", y= "percent", fill = " 재갱신유무") +
             facet_grid(~ customer_grade) +
             scale_y_continuous(labels = scales::percent)
```

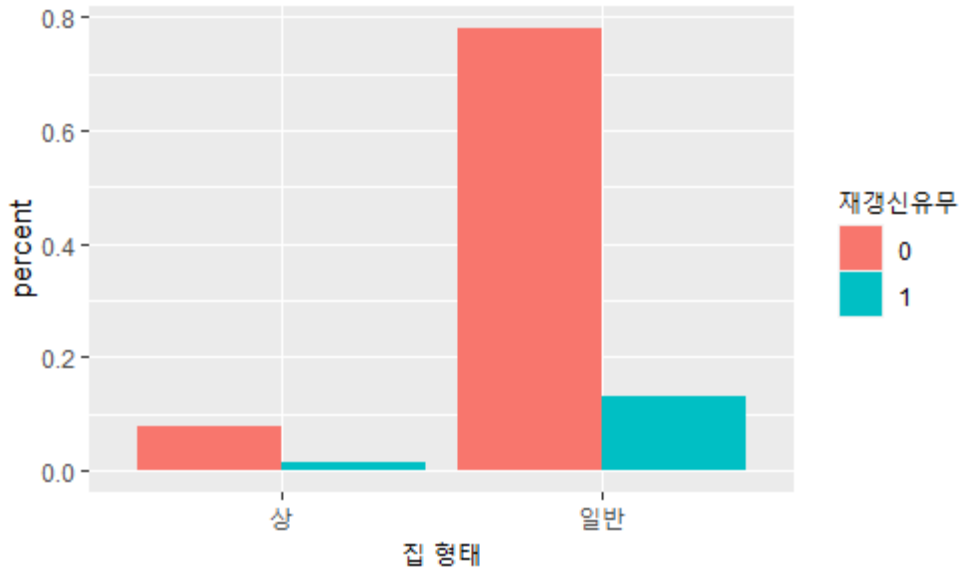


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 집 형태(house_type)

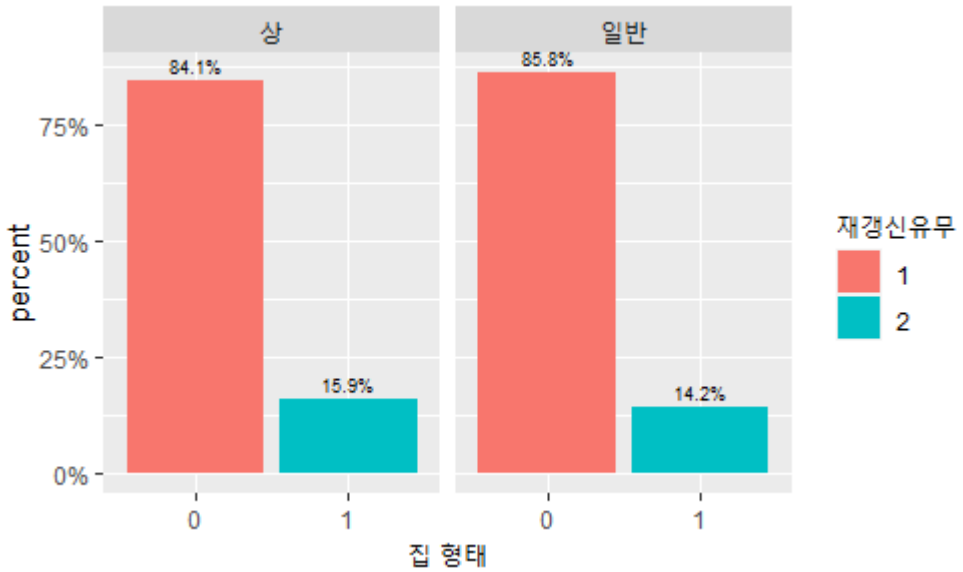
```
> health %>% ggplot(aes(x= house_type, fill=renewal)) +
                geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
                labs(x= "집 형태", y= "percent", fill = "재갱신유무")
```



■ 갱신유무에 대한 기초통계

▪ 집 형태(house_type)

```
> health %>% ggplot(aes(x= renewal, group= house_type)) +
              geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
              geom_text(aes( label = scales::percent(..prop..),
                             y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
              labs(x= "집 형태", y= "percent", fill = " 재갱신유무") +
              facet_grid(~ house_type) +
              scale_y_continuous(labels = scales::percent)
```

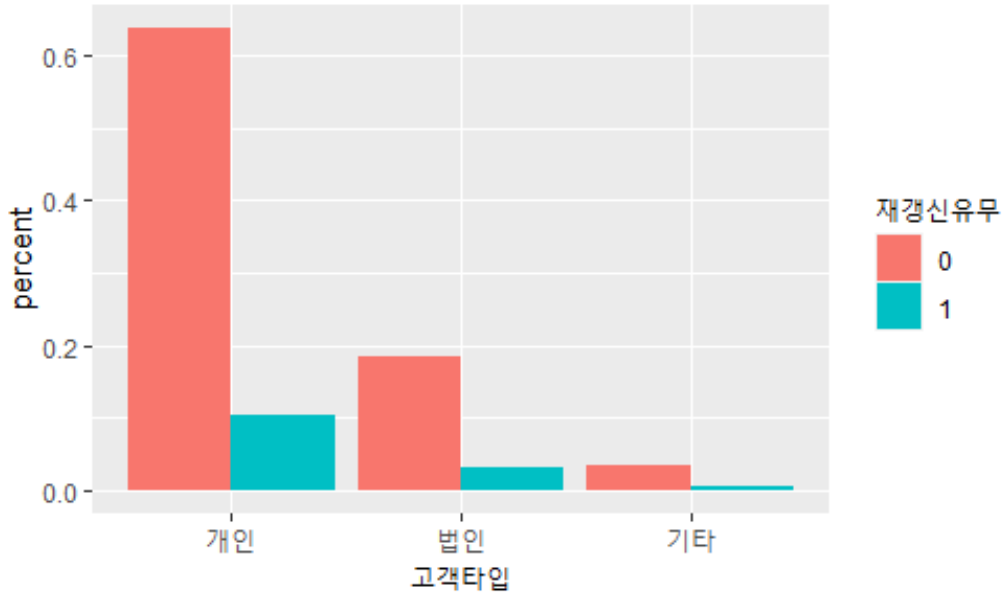


3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 고객타입(customer_type)

```
> health %>% ggplot(aes(x= customer_type, fill=renewal)) +
                geom_bar(aes(y = (..count..)/sum(..count..)), position = "dodge") +
                labs(x= "고객타입", y= "percent", fill = "재갱신유무")
```



3. 입력 변수 사이의 상관관계 분석

■ 갱신유무에 대한 기초통계

▪ 고객타입(customer_type)

```
> health %>% ggplot(aes(x= renewal, group= customer_type)) +
             geom_bar(aes(y=..prop.., fill = factor(..x..)), stat= "count") +
             geom_text(aes( label = scales::percent(..prop..),
                            y= ..prop.. ), stat= "count", vjust = -.5, size=2.5) +
             labs(x= "고객타입", y= "percent", fill = " 재갱신유무") +
             facet_grid(~ customer_type) +
             scale_y_continuous(labels = scales::percent)
```