젊은 임상 연구자들을 위한 필수 R

Jonggi Choi

March 14, 2021

본 문서는 배포는 가능하지만 출처는 밝혀주시길 바랍니다.

**작성자**  
울산의대 서울아산병원  
소화기내과 조교수 최종기  
작성 일시: March 14, 2021  
버전: Version 1.0

## 1. Basic function for data exploration

### 1.1 실습에 필요한 package loading

library(tidyverse)

### 1.2 Data 불러오기

* **현재 사용하고 있는 working directory 확인**

getwd()

[1] "/Users/ChoiJongGi/Dropbox/Data science study/R\_JonggiChoi"

Mac과 Windows는 폴더 경로 표시 방식이 다르기때문에 주의!

* **실습을 위해 사용할 data 불러오기 (chb\_example.csv)**

# Data (csv파일)를 불러오는 방식에는 2가지가 있다.  
# 일반 data.frame으로 불러오는 경우  
dat <- read.csv("chb\_example.csv") # read.csv 이용  
class(dat) # data 형태가 data.frame

[1] "data.frame"

# Tibble형태로 불러오는 경우  
dat <- read\_csv("chb\_example.csv") # read\_csv 이용  
class(dat) # data 형태가 tibble (tbl\_df)

[1] "spec\_tbl\_df" "tbl\_df" "tbl" "data.frame"

일반적인 data.frame형태로 불러와도 되지만 tibble형태가 조금 더 깔끔하고 tidyverse 명령어에 적합하다.  
하지만 기본적으로 tibble은 data.frame과 같은 속성에서 비롯되었다.

* **Data는 어떻게 생겼는지 구조 파악**

# Data의 구조를 보여주는 기본 명령어  
str(dat)

spec\_tbl\_df [1,000 × 23] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
 $ index\_date : Date[1:1000], format: "2007-01-05" "2007-01-10" ...  
 $ gender : chr [1:1000] "M" "F" "M" "M" ...  
 $ age : num [1:1000] 54 45 49 26 50 33 49 50 49 50 ...  
 $ last\_date : Date[1:1000], format: "2014-07-18" "2016-08-25" ...  
 $ treat\_gr : chr [1:1000] "ETV" "ETV" "ETV" "ETV" ...  
 $ lc : num [1:1000] 1 0 1 0 1 1 1 0 1 0 ...  
 $ dietpl : num [1:1000] 0 0 0 0 0 0 0 0 0 0 ...  
 $ dietpl\_date: Date[1:1000], format: "2017-09-13" "2016-08-25" ...  
 $ dietpl\_yr : num [1:1000] 10.84 9.76 10.59 6.02 2.36 ...  
 $ hcc : num [1:1000] 1 0 0 0 0 0 1 0 0 0 ...  
 $ hcc\_date : Date[1:1000], format: "2014-07-18" "2016-08-25" ...  
 $ hcc\_yr : num [1:1000] 7.64 9.76 10.59 6.02 2.36 ...  
 $ b\_lab\_date : Date[1:1000], format: "2006-12-28" "2006-12-28" ...  
 $ b\_alt : num [1:1000] 67 32 106 159 94 32 104 143 31 239 ...  
 $ b\_bil : num [1:1000] 1.2 1.1 1.8 1.4 0.8 0.7 1.1 1.5 1.3 1.3 ...  
 $ b\_inr : num [1:1000] 1.17 0.9 1.03 1.04 1.12 0.88 1.1 NA 1.04 1.03 ...  
 $ b\_cr : num [1:1000] 1 0.6 1 0.93 0.9 1.2 0.9 0.7 0.7 1.2 ...  
 $ b\_plt : num [1:1000] 110 187 133 164 153 170 NA 165 157 166 ...  
 $ b\_alb : num [1:1000] 3.8 3.8 3.9 4.2 4.2 3.4 3.7 3.5 3.9 3.2 ...  
 $ b\_eag : num [1:1000] 1 0 0 0 1 1 1 1 1 1 ...  
 $ b\_eab : num [1:1000] 0 1 1 1 0 0 0 NA 0 0 ...  
 $ b\_dna : chr [1:1000] "31934936.8" "1100" "35968299.4" "9145608.2" ...  
 $ b\_dna\_log : num [1:1000] 7.5 3.04 7.56 6.96 6.04 ...  
 - attr(\*, "spec")=  
 .. cols(  
 .. index\_date = col\_date(format = ""),  
 .. gender = col\_character(),  
 .. age = col\_double(),  
 .. last\_date = col\_date(format = ""),  
 .. treat\_gr = col\_character(),  
 .. lc = col\_double(),  
 .. dietpl = col\_double(),  
 .. dietpl\_date = col\_date(format = ""),  
 .. dietpl\_yr = col\_double(),  
 .. hcc = col\_double(),  
 .. hcc\_date = col\_date(format = ""),  
 .. hcc\_yr = col\_double(),  
 .. b\_lab\_date = col\_date(format = ""),  
 .. b\_alt = col\_double(),  
 .. b\_bil = col\_double(),  
 .. b\_inr = col\_double(),  
 .. b\_cr = col\_double(),  
 .. b\_plt = col\_double(),  
 .. b\_alb = col\_double(),  
 .. b\_eag = col\_double(),  
 .. b\_eab = col\_double(),  
 .. b\_dna = col\_character(),  
 .. b\_dna\_log = col\_double()  
 .. )

# 제일 위 6줄만 보여줌. 많이 쓰임  
head(dat)

# A tibble: 6 x 23  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date dietpl\_yr  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date> <dbl>  
1 2007-01-05 M 54 2014-07-18 ETV 1 0 2017-09-13 10.8   
2 2007-01-10 F 45 2016-08-25 ETV 0 0 2016-08-25 9.76  
3 2007-01-11 M 49 2017-06-21 ETV 1 0 2017-06-21 10.6   
4 2007-01-12 M 26 2012-12-17 ETV 0 0 2012-12-17 6.02  
5 2007-01-18 M 50 2009-05-15 ETV 1 0 2009-05-15 2.36  
6 2007-01-18 M 33 2013-01-11 ETV 1 0 2013-01-11 6.07  
# … with 14 more variables: hcc <dbl>, hcc\_date <date>, hcc\_yr <dbl>,  
# b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>, b\_inr <dbl>, b\_cr <dbl>,  
# b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>, b\_eab <dbl>, b\_dna <chr>,  
# b\_dna\_log <dbl>

# 이것도 가능, 3줄만 보여줌  
head(dat, 3)

# A tibble: 3 x 23  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date dietpl\_yr  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date> <dbl>  
1 2007-01-05 M 54 2014-07-18 ETV 1 0 2017-09-13 10.8   
2 2007-01-10 F 45 2016-08-25 ETV 0 0 2016-08-25 9.76  
3 2007-01-11 M 49 2017-06-21 ETV 1 0 2017-06-21 10.6   
# … with 14 more variables: hcc <dbl>, hcc\_date <date>, hcc\_yr <dbl>,  
# b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>, b\_inr <dbl>, b\_cr <dbl>,  
# b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>, b\_eab <dbl>, b\_dna <chr>,  
# b\_dna\_log <dbl>

# 제일 마지막 6줄 보여줌  
tail(dat)

# A tibble: 6 x 23  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date dietpl\_yr  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date> <dbl>  
1 2011-01-04 M 58 2017-10-19 ETV 1 0 2017-10-19 6.89  
2 2011-01-05 F 49 2017-10-31 ETV 1 0 2017-10-31 6.92  
3 2011-01-07 M 44 2017-05-08 ETV 1 0 2017-05-08 6.42  
4 2011-01-07 M 48 2017-04-06 ETV 0 0 2017-04-06 6.34  
5 2011-01-12 M 64 2013-03-31 ETV 1 0 2017-10-24 6.88  
6 2011-01-12 F 51 2017-10-31 ETV 1 0 2017-10-31 6.9   
# … with 14 more variables: hcc <dbl>, hcc\_date <date>, hcc\_yr <dbl>,  
# b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>, b\_inr <dbl>, b\_cr <dbl>,  
# b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>, b\_eab <dbl>, b\_dna <chr>,  
# b\_dna\_log <dbl>

# Tibble 형태의 data는 glimpse 형태 명령어를 쓰면 깔끔하게 요약되어 보인다  
glimpse(dat)

Rows: 1,000  
Columns: 23  
$ index\_date <date> 2007-01-05, 2007-01-10, 2007-01-11, 2007-01-12, 2007-01-…  
$ gender <chr> "M", "F", "M", "M", "M", "M", "M", "F", "F", "M", "M", "M…  
$ age <dbl> 54, 45, 49, 26, 50, 33, 49, 50, 49, 50, 32, 48, 51, 41, 5…  
$ last\_date <date> 2014-07-18, 2016-08-25, 2017-06-21, 2012-12-17, 2009-05-…  
$ treat\_gr <chr> "ETV", "ETV", "ETV", "ETV", "ETV", "ETV", "ETV", "ETV", "…  
$ lc <dbl> 1, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, …  
$ dietpl <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …  
$ dietpl\_date <date> 2017-09-13, 2016-08-25, 2017-06-21, 2012-12-17, 2009-05-…  
$ dietpl\_yr <dbl> 10.844444, 9.763889, 10.594444, 6.016667, 2.355556, 6.069…  
$ hcc <dbl> 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, …  
$ hcc\_date <date> 2014-07-18, 2016-08-25, 2017-06-21, 2012-12-17, 2009-05-…  
$ hcc\_yr <dbl> 7.641667, 9.763889, 10.594444, 6.016667, 2.355556, 6.0694…  
$ b\_lab\_date <date> 2006-12-28, 2006-12-28, 2006-12-28, 2007-01-06, 2007-01-…  
$ b\_alt <dbl> 67, 32, 106, 159, 94, 32, 104, 143, 31, 239, 359, 486, 56…  
$ b\_bil <dbl> 1.2, 1.1, 1.8, 1.4, 0.8, 0.7, 1.1, 1.5, 1.3, 1.3, 0.9, 0.…  
$ b\_inr <dbl> 1.17, 0.90, 1.03, 1.04, 1.12, 0.88, 1.10, NA, 1.04, 1.03,…  
$ b\_cr <dbl> 1.00, 0.60, 1.00, 0.93, 0.90, 1.20, 0.90, 0.70, 0.70, 1.2…  
$ b\_plt <dbl> 110, 187, 133, 164, 153, 170, NA, 165, 157, 166, 202, 159…  
$ b\_alb <dbl> 3.8, 3.8, 3.9, 4.2, 4.2, 3.4, 3.7, 3.5, 3.9, 3.2, 4.1, 2.…  
$ b\_eag <dbl> 1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0, 1, 0, 1, 1, 1, 0, 0, …  
$ b\_eab <dbl> 0, 1, 1, 1, 0, 0, 0, NA, 0, 0, 1, 1, 0, 1, 0, 0, 0, 1, 1,…  
$ b\_dna <chr> "31934936.8", "1100", "35968299.4", "9145608.2", "1095481…  
$ b\_dna\_log <dbl> 7.504266, 3.041393, 7.555920, 6.961213, 6.039605, 2.95424…

### 1.3 Data 특징 파악

* **Data의 구조 및 크기 파악**

# 전체 data dimension  
dim(dat)

[1] 1000 23

1000행(케이스 혹은 환자), 23열(컬럼 혹은 variable)

# 전체 케이스 숫자만 궁금할때  
nrow(dat)

[1] 1000

# column name (=variable name)  
colnames(dat)

[1] "index\_date" "gender" "age" "last\_date" "treat\_gr"   
 [6] "lc" "dietpl" "dietpl\_date" "dietpl\_yr" "hcc"   
[11] "hcc\_date" "hcc\_yr" "b\_lab\_date" "b\_alt" "b\_bil"   
[16] "b\_inr" "b\_cr" "b\_plt" "b\_alb" "b\_eag"   
[21] "b\_eab" "b\_dna" "b\_dna\_log"

* Data variable summary
  + index\_date: 투약 시작 시점
  + gender: 성별
  + age: 나이
  + last\_date: 마지막 추적관찰
  + treat\_gr: 항바이러스제 종류 (ETV: entecavir, TDF: tenofovir)
  + lc: liver cirrhosis
  + dietpl: Death or liver transplantation
  + dietpl\_date: Date of death or liver transplantation
  + dietpl\_yr: Time interval from index date to death/transplantation
  + hcc: HCC
  + hcc\_date: Date of HCC diagnosis
  + hcc\_yr: Time interval from index date to HCC diagnosis
  + b\_lab\_date: Baseline lab 시행 일자
  + b\_alt: Baseline ALT
  + b\_bil: Baseline total bilirubin
  + b\_inr: Baseline PT(INR)
  + b\_cr: Baseline creatinine
  + b\_plt: Baseline platelet count
  + b\_alb: Baseline albumin
  + b\_eag: Baseline HBeAg status
  + b\_eab: Baseline HBeAb status
  + b\_dna: Baseline HBV DNA level
  + b\_dna\_log: Baseline HBV DNA level (log)
* **각 variable들의 특징 파악**

# Age 변수 특성 파악  
# 평균  
mean(dat$age)

[1] 46.062

# 중앙값  
median(dat$age)

[1] 47

# 표준편차  
sd(dat$age)

[1] 10.20825

# 분산  
var(dat$age)

[1] 104.2084

# 최소값  
min(dat$age)

[1] 19

# 최대값  
max(dat$age)

[1] 74

# 범위  
range(dat$age)

[1] 19 74

# 1사분위수-3사분위수 범위  
IQR(dat$age)

[1] 13

# 한꺼번에 보여주기  
summary(dat$age)

Min. 1st Qu. Median Mean 3rd Qu. Max.   
 19.00 40.00 47.00 46.06 53.00 74.00

# Hmisc package의 describe 명령어가 유용  
# Age 변수의 특징 파악  
Hmisc::describe(dat$age)

dat$age   
 n missing distinct Info Mean Gmd .05 .10   
 1000 0 55 0.999 46.06 11.54 27 32   
 .25 .50 .75 .90 .95   
 40 47 53 59 62   
  
lowest : 19 20 21 22 23, highest: 70 71 72 73 74

* **결측값이 존재하는 경우 데이터 특성 파악**

# 결측값이 존재하는 variable은 평균 계산 불가  
mean(dat$b\_alb) # NA로 표시됨

[1] NA

# 결측값이 존재하는 경우 'na.rm=TRUE' option 추가  
mean(dat$b\_alb, na.rm=T) # na.rm의미 = NA remove

[1] 3.734274

NA를 제외하고 mean을 계산을 해준다

* **결측값(NA)이 존재하는지 확인하기**

# NA 존재 명령어 'is.na'  
# is.na 결과값이 True = 결측값  
# is.na 결과값이 False = 결측값 아님  
is.na(dat$b\_inr[1:20])

[1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE FALSE FALSE FALSE FALSE  
[13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

’b\_inr’변수의 1-20행까지 결측값이 존재하는지 확인가능

# 전체 1,000개 행중에 b\_inr 변수의 총 NA는 몇개?  
sum(is.na(dat$b\_inr))

[1] 16

b\_inr이 NA일경우 true (1로 게산됨) 이경우를 모두 sum하게 되니깐 총 16개의 NA가 b\_inr에 존재함

# summary로도 확인가능  
summary(dat$b\_inr)

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
 0.840 1.030 1.090 1.134 1.180 3.130 16

* **Data 형태 (class) 확인하기**

# is.\*() 함수를 이용한다.  
  
is.data.frame(dat) # data.frame?

[1] TRUE

is.matrix(dat) # matrix?

[1] FALSE

is.character(dat$age) # 문자형 데이터인 범주형 변수?

[1] FALSE

is.character(dat$gender) # 문자형 데이터인 범주형 변수?

[1] TRUE

is.numeric(dat$age) # 수치형 데이터인 연속형 변수?

[1] TRUE

is.numeric(dat$gender) # 수치형 데이터인 연속형 변수?

[1] FALSE

is.integer(dat$age) # 정수형 데이터인 연속형 변수?

[1] FALSE

is.double(dat$age) # 더블형 데이터인 연속형 변수?

[1] TRUE

<br?>

### 1.4 Baseline characteristics Table 만들기

# moonBook package를 이용하여 descriptive analysis  
# treat\_gr (ETV:entecavir, TDF:tenofovir)에 따른 두군의 # 비교  
library(moonBook)  
mytable(treat\_gr~age+gender,data=dat)

Descriptive Statistics by 'treat\_gr'  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_   
 ETV TDF p   
 (N=506) (N=494)   
-------------------------------------   
 age 46.0 ± 10.2 46.1 ± 10.2 0.822  
 gender 0.178  
 - F 150 (29.6%) 167 (33.8%)   
 - M 356 (70.4%) 327 (66.2%)   
-------------------------------------

# 전체 그룹도 표시하기  
mytable(treat\_gr~age+gender, data=dat, show.total=TRUE)

Descriptive Statistics by 'treat\_gr'   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_   
 ETV TDF Total p   
 (N=506) (N=494) (N=1000)   
-------------------------------------------------   
 age 46.0 ± 10.2 46.1 ± 10.2 46.1 ± 10.2 0.822  
 gender 0.178  
 - F 150 (29.6%) 167 (33.8%) 317 (31.7%)   
 - M 356 (70.4%) 327 (66.2%) 683 (68.3%)   
-------------------------------------------------

# mean +/- SD 말고 median [IQR]로 표시하기  
mytable(treat\_gr~age+gender,data=dat, method=2)

Descriptive Statistics by 'treat\_gr'   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_   
 ETV TDF p   
 (N=506) (N=494)   
-----------------------------------------------   
 age 47.0 [40.0;53.0] 46.0 [40.0;53.0] 0.979  
 gender 0.178  
 - F 150 (29.6%) 167 (33.8%)   
 - M 356 (70.4%) 327 (66.2%)   
-----------------------------------------------

# treat\_gr에 따른 모든 변수 비교  
mytable(treat\_gr~., data=dat) # ~. 의 의미는 'treat\_gr'제외하고 나머지 모든 변수를 넣어달라는 의미

Descriptive Statistics by 'treat\_gr'   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_   
 ETV TDF p   
 (N=506) (N=494)   
--------------------------------------------------------------   
 index\_date 2007-01-05-2011-01-12 2007-01-05-2011-01-12   
 gender 0.178  
 - F 150 (29.6%) 167 (33.8%)   
 - M 356 (70.4%) 327 (66.2%)   
 age 46.0 ± 10.2 46.1 ± 10.2 0.822  
 last\_date 2007-12-18-2017-11-01 2007-12-18-2017-11-01   
 lc 0.498  
 - 0 234 (46.2%) 240 (48.6%)   
 - 1 272 (53.8%) 254 (51.4%)   
 dietpl 0.072  
 - 0 492 (97.2%) 489 (99.0%)   
 - 1 14 ( 2.8%) 5 ( 1.0%)   
 dietpl\_date 2009-01-28-2017-11-01 2009-01-28-2017-11-01   
 dietpl\_yr 7.4 ± 2.8 6.7 ± 2.2 0.000  
 hcc 0.050  
 - 0 418 (82.6%) 431 (87.2%)   
 - 1 88 (17.4%) 63 (12.8%)   
 hcc\_date 2007-12-18-2017-11-01 2007-12-18-2017-11-01   
 hcc\_yr 6.8 ± 3.1 6.2 ± 2.4 0.003  
 b\_lab\_date 2006-12-28-2011-01-17 2006-12-28-2011-01-17   
 b\_alt 195.9 ± 257.7 203.2 ± 303.5 0.682  
 b\_bil 1.7 ± 2.1 1.9 ± 3.4 0.158  
 b\_inr 1.1 ± 0.2 1.2 ± 0.2 0.003  
 b\_cr 0.9 ± 0.5 1.0 ± 0.8 0.152  
 b\_plt 148.5 ± 58.8 149.0 ± 58.5 0.894  
 b\_alb 3.7 ± 0.5 3.8 ± 0.6 0.239  
 b\_eag 0.009  
 - 0 159 (31.4%) 195 (39.5%)   
 - 1 347 (68.6%) 299 (60.5%)   
 b\_eab 0.024  
 - 0 300 (68.6%) 257 (61.0%)   
 - 1 137 (31.4%) 164 (39.0%)   
 b\_dna unique values:466 unique values:466   
 b\_dna\_log 6.9 ± 1.9 6.4 ± 2.0 0.000  
--------------------------------------------------------------

# 만든 table 저장하기  
table1<-mytable(treat\_gr~age+gender, data=dat)  
table1

Descriptive Statistics by 'treat\_gr'  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_   
 ETV TDF p   
 (N=506) (N=494)   
-------------------------------------   
 age 46.0 ± 10.2 46.1 ± 10.2 0.822  
 gender 0.178  
 - F 150 (29.6%) 167 (33.8%)   
 - M 356 (70.4%) 327 (66.2%)   
-------------------------------------

mycsv(table1, file="table1.csv")   
# mycsv(table객체, file="파일이름")

특별히 경로를 지정해주지 않으면현재 working directory에 저장이 된다.

## 2. Data manipulation using dplyr

### 2.1 Key five functions in Tidyverse

1. filter
2. select
3. arrange
4. mutate
5. summarise

#### 2.1.1 Filter function

# 원본 data는 손대지말고 연습을 위해 복사본 (dat1)을 만들자  
dat1 <- dat

**R은 되돌아가기 (Ctrl+Z)기능이 없기에, data의 변형이 우려되면 복사본을 만들어두고 가능하면 original data는 그대로 두는것이 좋다**

* filter는 말그대로 필터다. 엑셀의 그 깔대기모양의 ‘필터’
* 즉 필요한 조건에 만족하는 값만을 출력

# Gender (Male: M or Female: F)를 이용  
dat1 %>%   
 count(gender)

# A tibble: 2 x 2  
 gender n  
\* <chr> <int>  
1 F 317  
2 M 683

* dat1에서 male값만을 선택

dat1 %>%   
 filter(gender == "M") # 같다는= 한개가 아니고 == 2개!

# A tibble: 683 x 23  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-01-05 M 54 2014-07-18 ETV 1 0 2017-09-13   
 2 2007-01-11 M 49 2017-06-21 ETV 1 0 2017-06-21   
 3 2007-01-12 M 26 2012-12-17 ETV 0 0 2012-12-17   
 4 2007-01-18 M 50 2009-05-15 ETV 1 0 2009-05-15   
 5 2007-01-18 M 33 2013-01-11 ETV 1 0 2013-01-11   
 6 2007-01-26 M 49 2013-03-31 ETV 1 0 2017-08-07   
 7 2007-02-01 M 50 2017-09-12 ETV 0 0 2017-09-12   
 8 2007-02-01 M 32 2010-09-30 ETV 0 0 2010-09-30   
 9 2007-02-01 M 48 2010-04-23 ETV 0 0 2010-04-23   
10 2007-02-01 M 51 2017-08-03 ETV 0 0 2017-08-03   
# … with 673 more rows, and 15 more variables: dietpl\_yr <dbl>, hcc <dbl>,  
# hcc\_date <date>, hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>,  
# b\_inr <dbl>, b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>,  
# b\_eab <dbl>, b\_dna <chr>, b\_dna\_log <dbl>

dat1 %>%   
 filter(gender != "F") # 같지 않다는 != 로 표현

# A tibble: 683 x 23  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-01-05 M 54 2014-07-18 ETV 1 0 2017-09-13   
 2 2007-01-11 M 49 2017-06-21 ETV 1 0 2017-06-21   
 3 2007-01-12 M 26 2012-12-17 ETV 0 0 2012-12-17   
 4 2007-01-18 M 50 2009-05-15 ETV 1 0 2009-05-15   
 5 2007-01-18 M 33 2013-01-11 ETV 1 0 2013-01-11   
 6 2007-01-26 M 49 2013-03-31 ETV 1 0 2017-08-07   
 7 2007-02-01 M 50 2017-09-12 ETV 0 0 2017-09-12   
 8 2007-02-01 M 32 2010-09-30 ETV 0 0 2010-09-30   
 9 2007-02-01 M 48 2010-04-23 ETV 0 0 2010-04-23   
10 2007-02-01 M 51 2017-08-03 ETV 0 0 2017-08-03   
# … with 673 more rows, and 15 more variables: dietpl\_yr <dbl>, hcc <dbl>,  
# hcc\_date <date>, hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>,  
# b\_inr <dbl>, b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>,  
# b\_eab <dbl>, b\_dna <chr>, b\_dna\_log <dbl>

즉 위의 2개 코드는 같은 값을 출력 해준다.

* **복합 조건 (AND)**

# older than 50 years old  
dat1 %>%   
 filter(age>=50) # 385명

# A tibble: 385 x 23  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-01-05 M 54 2014-07-18 ETV 1 0 2017-09-13   
 2 2007-01-18 M 50 2009-05-15 ETV 1 0 2009-05-15   
 3 2007-01-31 F 50 2017-08-24 ETV 0 0 2017-08-24   
 4 2007-02-01 M 50 2017-09-12 ETV 0 0 2017-09-12   
 5 2007-02-01 M 51 2017-08-03 ETV 0 0 2017-08-03   
 6 2007-02-08 M 50 2012-11-21 ETV 1 0 2012-11-21   
 7 2007-02-08 M 51 2011-01-21 ETV 1 0 2011-01-21   
 8 2007-02-09 F 62 2010-07-28 ETV 1 0 2010-07-28   
 9 2007-02-14 F 51 2009-06-17 ETV 1 0 2009-06-17   
10 2007-02-15 F 50 2012-01-31 ETV 1 1 2012-01-31   
# … with 375 more rows, and 15 more variables: dietpl\_yr <dbl>, hcc <dbl>,  
# hcc\_date <date>, hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>,  
# b\_inr <dbl>, b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>,  
# b\_eab <dbl>, b\_dna <chr>, b\_dna\_log <dbl>

# older than 50 years old + Male  
dat1 %>%   
 filter(gender == "M", age >=50) # 220명

# A tibble: 220 x 23  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-01-05 M 54 2014-07-18 ETV 1 0 2017-09-13   
 2 2007-01-18 M 50 2009-05-15 ETV 1 0 2009-05-15   
 3 2007-02-01 M 50 2017-09-12 ETV 0 0 2017-09-12   
 4 2007-02-01 M 51 2017-08-03 ETV 0 0 2017-08-03   
 5 2007-02-08 M 50 2012-11-21 ETV 1 0 2012-11-21   
 6 2007-02-08 M 51 2011-01-21 ETV 1 0 2011-01-21   
 7 2007-02-15 M 52 2011-10-13 ETV 1 0 2011-10-13   
 8 2007-02-28 M 53 2008-12-25 ETV 1 1 2009-01-28   
 9 2007-03-09 M 51 2012-03-05 ETV 1 0 2012-03-05   
10 2007-03-15 M 52 2010-12-16 ETV 0 0 2010-12-16   
# … with 210 more rows, and 15 more variables: dietpl\_yr <dbl>, hcc <dbl>,  
# hcc\_date <date>, hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>,  
# b\_inr <dbl>, b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>,  
# b\_eab <dbl>, b\_dna <chr>, b\_dna\_log <dbl>

# older than 50 years old + Male + Cirrhosis  
dat1 %>%   
 filter(gender == "M", age >=50, lc==1) # 152명

# A tibble: 152 x 23  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-01-05 M 54 2014-07-18 ETV 1 0 2017-09-13   
 2 2007-01-18 M 50 2009-05-15 ETV 1 0 2009-05-15   
 3 2007-02-08 M 50 2012-11-21 ETV 1 0 2012-11-21   
 4 2007-02-08 M 51 2011-01-21 ETV 1 0 2011-01-21   
 5 2007-02-15 M 52 2011-10-13 ETV 1 0 2011-10-13   
 6 2007-02-28 M 53 2008-12-25 ETV 1 1 2009-01-28   
 7 2007-03-09 M 51 2012-03-05 ETV 1 0 2012-03-05   
 8 2007-03-22 M 54 2009-12-03 ETV 1 0 2009-12-03   
 9 2007-03-29 M 57 2014-08-22 ETV 1 0 2015-05-26   
10 2007-04-17 M 61 2017-08-21 ETV 1 0 2017-08-21   
# … with 142 more rows, and 15 more variables: dietpl\_yr <dbl>, hcc <dbl>,  
# hcc\_date <date>, hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>,  
# b\_inr <dbl>, b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>,  
# b\_eab <dbl>, b\_dna <chr>, b\_dna\_log <dbl>

각 조건들간의 AND 조합은 , & 어느것을 써도 된다.

* **복합 조건 (OR)**

# age 30세 미만이거나 80세 이상인 경우  
dat1 %>%   
 filter(age <30 | age >80) # 74명

# A tibble: 74 x 23  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-01-12 M 26 2012-12-17 ETV 0 0 2012-12-17   
 2 2007-02-16 M 29 2017-07-14 ETV 0 0 2017-07-14   
 3 2007-03-02 M 25 2012-04-30 ETV 0 0 2012-04-30   
 4 2007-03-08 M 24 2017-04-12 ETV 0 0 2017-04-12   
 5 2007-04-02 F 22 2010-05-25 ETV 0 0 2010-05-25   
 6 2007-04-13 M 24 2015-09-22 ETV 0 0 2015-09-22   
 7 2007-04-16 M 29 2017-07-20 ETV 0 0 2017-07-20   
 8 2007-04-19 M 27 2009-02-27 ETV 0 0 2009-02-27   
 9 2007-05-03 M 26 2009-04-03 ETV 0 0 2009-04-03   
10 2007-07-03 F 24 2016-09-13 ETV 0 0 2016-09-13   
# … with 64 more rows, and 15 more variables: dietpl\_yr <dbl>, hcc <dbl>,  
# hcc\_date <date>, hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>,  
# b\_inr <dbl>, b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>,  
# b\_eab <dbl>, b\_dna <chr>, b\_dna\_log <dbl>

# age 70세 이상이거나 cirrhosis 있는 경우  
dat1 %>%   
 filter(age >70 | lc==1) # 526명

# A tibble: 526 x 23  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-01-05 M 54 2014-07-18 ETV 1 0 2017-09-13   
 2 2007-01-11 M 49 2017-06-21 ETV 1 0 2017-06-21   
 3 2007-01-18 M 50 2009-05-15 ETV 1 0 2009-05-15   
 4 2007-01-18 M 33 2013-01-11 ETV 1 0 2013-01-11   
 5 2007-01-26 M 49 2013-03-31 ETV 1 0 2017-08-07   
 6 2007-01-31 F 49 2009-03-20 ETV 1 0 2009-03-20   
 7 2007-02-02 M 41 2011-07-07 ETV 1 0 2011-07-07   
 8 2007-02-08 M 50 2012-11-21 ETV 1 0 2012-11-21   
 9 2007-02-08 M 51 2011-01-21 ETV 1 0 2011-01-21   
10 2007-02-09 F 62 2010-07-28 ETV 1 0 2010-07-28   
# … with 516 more rows, and 15 more variables: dietpl\_yr <dbl>, hcc <dbl>,  
# hcc\_date <date>, hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>,  
# b\_inr <dbl>, b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>,  
# b\_eab <dbl>, b\_dna <chr>, b\_dna\_log <dbl>

* **복합 여러 조건 (AND + OR, not)**

# 80세 이상의 남성이나(OR) 30세 미만의 여자  
dat1 %>%   
 filter( (age>80 & gender =="M") | (age<30 & gender =="F"))

# A tibble: 20 x 23  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-04-02 F 22 2010-05-25 ETV 0 0 2010-05-25   
 2 2007-07-03 F 24 2016-09-13 ETV 0 0 2016-09-13   
 3 2007-11-12 F 22 2009-08-17 ETV 0 0 2009-08-17   
 4 2008-02-04 F 29 2011-07-21 ETV 0 0 2011-07-21   
 5 2008-06-05 F 23 2017-09-29 ETV 1 0 2017-09-29   
 6 2008-07-11 F 25 2011-07-13 ETV 0 0 2011-07-13   
 7 2008-07-26 F 28 2017-02-20 ETV 1 0 2017-02-20   
 8 2008-12-23 F 28 2017-07-20 TDF 0 0 2017-07-20   
 9 2008-12-29 F 23 2017-09-11 TDF 0 0 2017-09-11   
10 2009-01-13 F 26 2017-06-02 TDF 0 0 2017-06-02   
11 2009-02-02 F 25 2013-05-23 TDF 0 0 2013-05-23   
12 2009-02-09 F 27 2014-06-09 TDF 0 0 2014-06-09   
13 2009-04-16 F 22 2013-01-09 TDF 0 0 2013-01-09   
14 2009-04-17 F 26 2015-01-12 TDF 0 0 2015-01-12   
15 2009-05-26 F 25 2013-02-18 TDF 0 0 2013-02-18   
16 2009-07-31 F 25 2012-12-10 TDF 0 0 2012-12-10   
17 2009-08-28 F 25 2017-03-24 TDF 1 0 2017-03-24   
18 2009-09-23 F 29 2017-06-21 TDF 0 0 2017-06-21   
19 2009-09-24 F 25 2013-01-10 TDF 0 0 2013-01-10   
20 2010-01-28 F 26 2012-03-27 TDF 1 0 2012-03-27   
# … with 15 more variables: dietpl\_yr <dbl>, hcc <dbl>, hcc\_date <date>,  
# hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>, b\_inr <dbl>,  
# b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>, b\_eab <dbl>,  
# b\_dna <chr>, b\_dna\_log <dbl>

# 80세 이상이면서 간경화가 없는 환자  
dat1 %>%   
 filter(age >=80 & lc!=1)

# A tibble: 0 x 23  
# … with 23 variables: index\_date <date>, gender <chr>, age <dbl>,  
# last\_date <date>, treat\_gr <chr>, lc <dbl>, dietpl <dbl>,  
# dietpl\_date <date>, dietpl\_yr <dbl>, hcc <dbl>, hcc\_date <date>,  
# hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>, b\_inr <dbl>,  
# b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>, b\_eab <dbl>,  
# b\_dna <chr>, b\_dna\_log <dbl>

수학에서 먼저 해야되는 연산은 ()로 묶어 주듯이 조건별로 적절히 ()로 묶어준다.

* **유용한 명령어**

[index\_date]로 부터 연도(year)만 분리해서 variable을 새로 만들자

dat1$year<-year(dat1$index\_date)

Error가 날겁니다.

class(dat1$index\_date)

[1] "Date"

dat1$index\_date의 형태는 “Date” 입니다.

“Date” 형태의 data에서 자동으로 Year만 분리해줄 수 있는 함수가 필요합니다. (lubridate package)

library(lubridate)  
dat1$year<-year(dat1$index\_date)  
  
# year 분포를 봅시다.  
dat1 %>%   
 count(year)

# A tibble: 5 x 2  
 year n  
\* <dbl> <int>  
1 2007 199  
2 2008 315  
3 2009 287  
4 2010 193  
5 2011 6

만약 filter를 이용해서 2007, 2008, 2009년 자료만 골라낸다면?

dat1 %>%   
 filter(year <= 2009) # 801명

# A tibble: 801 x 24  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-01-05 M 54 2014-07-18 ETV 1 0 2017-09-13   
 2 2007-01-10 F 45 2016-08-25 ETV 0 0 2016-08-25   
 3 2007-01-11 M 49 2017-06-21 ETV 1 0 2017-06-21   
 4 2007-01-12 M 26 2012-12-17 ETV 0 0 2012-12-17   
 5 2007-01-18 M 50 2009-05-15 ETV 1 0 2009-05-15   
 6 2007-01-18 M 33 2013-01-11 ETV 1 0 2013-01-11   
 7 2007-01-26 M 49 2013-03-31 ETV 1 0 2017-08-07   
 8 2007-01-31 F 50 2017-08-24 ETV 0 0 2017-08-24   
 9 2007-01-31 F 49 2009-03-20 ETV 1 0 2009-03-20   
10 2007-02-01 M 50 2017-09-12 ETV 0 0 2017-09-12   
# … with 791 more rows, and 16 more variables: dietpl\_yr <dbl>, hcc <dbl>,  
# hcc\_date <date>, hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>,  
# b\_inr <dbl>, b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>,  
# b\_eab <dbl>, b\_dna <chr>, b\_dna\_log <dbl>, year <dbl>

만약 2007, 2008, 2010년을 골라내야 한다면

dat1 %>%   
 filter(year==2007 | year==2008 | year==2010) # 707명

# A tibble: 707 x 24  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-01-05 M 54 2014-07-18 ETV 1 0 2017-09-13   
 2 2007-01-10 F 45 2016-08-25 ETV 0 0 2016-08-25   
 3 2007-01-11 M 49 2017-06-21 ETV 1 0 2017-06-21   
 4 2007-01-12 M 26 2012-12-17 ETV 0 0 2012-12-17   
 5 2007-01-18 M 50 2009-05-15 ETV 1 0 2009-05-15   
 6 2007-01-18 M 33 2013-01-11 ETV 1 0 2013-01-11   
 7 2007-01-26 M 49 2013-03-31 ETV 1 0 2017-08-07   
 8 2007-01-31 F 50 2017-08-24 ETV 0 0 2017-08-24   
 9 2007-01-31 F 49 2009-03-20 ETV 1 0 2009-03-20   
10 2007-02-01 M 50 2017-09-12 ETV 0 0 2017-09-12   
# … with 697 more rows, and 16 more variables: dietpl\_yr <dbl>, hcc <dbl>,  
# hcc\_date <date>, hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>,  
# b\_inr <dbl>, b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>,  
# b\_eab <dbl>, b\_dna <chr>, b\_dna\_log <dbl>, year <dbl>

하지만 골라내야될 조건이 많아지거나 불규칙적이다면?

%in%을 사용해보자

dat1 %>%   
 filter(year %in% c(2007, 2008, 2010)) # 같은 결과

# A tibble: 707 x 24  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-01-05 M 54 2014-07-18 ETV 1 0 2017-09-13   
 2 2007-01-10 F 45 2016-08-25 ETV 0 0 2016-08-25   
 3 2007-01-11 M 49 2017-06-21 ETV 1 0 2017-06-21   
 4 2007-01-12 M 26 2012-12-17 ETV 0 0 2012-12-17   
 5 2007-01-18 M 50 2009-05-15 ETV 1 0 2009-05-15   
 6 2007-01-18 M 33 2013-01-11 ETV 1 0 2013-01-11   
 7 2007-01-26 M 49 2013-03-31 ETV 1 0 2017-08-07   
 8 2007-01-31 F 50 2017-08-24 ETV 0 0 2017-08-24   
 9 2007-01-31 F 49 2009-03-20 ETV 1 0 2009-03-20   
10 2007-02-01 M 50 2017-09-12 ETV 0 0 2017-09-12   
# … with 697 more rows, and 16 more variables: dietpl\_yr <dbl>, hcc <dbl>,  
# hcc\_date <date>, hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>,  
# b\_inr <dbl>, b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>,  
# b\_eab <dbl>, b\_dna <chr>, b\_dna\_log <dbl>, year <dbl>

between을 이용해도 된다

# 30세이상 80세 이하  
dat1 %>%   
 filter(age >=30 & age <80) # 926명

# A tibble: 926 x 24  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-01-05 M 54 2014-07-18 ETV 1 0 2017-09-13   
 2 2007-01-10 F 45 2016-08-25 ETV 0 0 2016-08-25   
 3 2007-01-11 M 49 2017-06-21 ETV 1 0 2017-06-21   
 4 2007-01-18 M 50 2009-05-15 ETV 1 0 2009-05-15   
 5 2007-01-18 M 33 2013-01-11 ETV 1 0 2013-01-11   
 6 2007-01-26 M 49 2013-03-31 ETV 1 0 2017-08-07   
 7 2007-01-31 F 50 2017-08-24 ETV 0 0 2017-08-24   
 8 2007-01-31 F 49 2009-03-20 ETV 1 0 2009-03-20   
 9 2007-02-01 M 50 2017-09-12 ETV 0 0 2017-09-12   
10 2007-02-01 M 32 2010-09-30 ETV 0 0 2010-09-30   
# … with 916 more rows, and 16 more variables: dietpl\_yr <dbl>, hcc <dbl>,  
# hcc\_date <date>, hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>,  
# b\_inr <dbl>, b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>,  
# b\_eab <dbl>, b\_dna <chr>, b\_dna\_log <dbl>, year <dbl>

# between 이용시  
dat1 %>%   
 filter(between(age, 30,80)) #이상, 이하의미로 30, 80은 포함됨!

# A tibble: 926 x 24  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-01-05 M 54 2014-07-18 ETV 1 0 2017-09-13   
 2 2007-01-10 F 45 2016-08-25 ETV 0 0 2016-08-25   
 3 2007-01-11 M 49 2017-06-21 ETV 1 0 2017-06-21   
 4 2007-01-18 M 50 2009-05-15 ETV 1 0 2009-05-15   
 5 2007-01-18 M 33 2013-01-11 ETV 1 0 2013-01-11   
 6 2007-01-26 M 49 2013-03-31 ETV 1 0 2017-08-07   
 7 2007-01-31 F 50 2017-08-24 ETV 0 0 2017-08-24   
 8 2007-01-31 F 49 2009-03-20 ETV 1 0 2009-03-20   
 9 2007-02-01 M 50 2017-09-12 ETV 0 0 2017-09-12   
10 2007-02-01 M 32 2010-09-30 ETV 0 0 2010-09-30   
# … with 916 more rows, and 16 more variables: dietpl\_yr <dbl>, hcc <dbl>,  
# hcc\_date <date>, hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>,  
# b\_inr <dbl>, b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>,  
# b\_eab <dbl>, b\_dna <chr>, b\_dna\_log <dbl>, year <dbl>

주의! between 이용시 지정값은 포함됨

* **결측값이 없는 자료만 filter**

# b\_inr (prothrombine time INR)의 결측값은 몇개?  
summary(dat1$b\_inr)

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
 0.840 1.030 1.090 1.134 1.180 3.130 16

# 혹은  
sum(is.na(dat1$b\_inr))

[1] 16

HBeAg 값이 있는 자료만 filter 해보자

dat1 %>%   
 filter( is.na(b\_inr)) # 이렇게 하면 b\_inr이 NA만 나온다

# A tibble: 16 x 24  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-01-31 F 50 2017-08-24 ETV 0 0 2017-08-24   
 2 2007-02-15 M 52 2011-10-13 ETV 1 0 2011-10-13   
 3 2007-08-01 M 58 2013-03-31 ETV 1 0 2017-09-20   
 4 2007-08-01 M 48 2009-09-30 ETV 1 0 2009-09-30   
 5 2007-08-02 M 48 2017-09-14 ETV 1 0 2017-09-14   
 6 2007-08-02 M 48 2017-07-04 ETV 1 0 2017-07-04   
 7 2007-08-07 M 25 2009-11-05 ETV 0 0 2009-11-05   
 8 2007-08-09 M 48 2009-08-27 ETV 0 0 2009-08-27   
 9 2008-06-24 M 46 2013-07-16 ETV 1 0 2013-07-16   
10 2008-06-24 M 41 2012-12-28 ETV 0 0 2012-12-28   
11 2008-06-26 F 53 2017-09-13 ETV 1 0 2017-09-13   
12 2008-06-26 F 48 2011-09-05 ETV 1 0 2014-07-15   
13 2008-06-26 M 45 2017-07-31 ETV 1 0 2017-07-31   
14 2008-06-26 F 52 2017-06-14 ETV 0 0 2017-06-14   
15 2008-06-27 F 30 2013-04-16 ETV 0 0 2013-04-16   
16 2008-06-27 M 44 2017-07-04 ETV 0 0 2017-07-04   
# … with 16 more variables: dietpl\_yr <dbl>, hcc <dbl>, hcc\_date <date>,  
# hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>, b\_inr <dbl>,  
# b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>, b\_eab <dbl>,  
# b\_dna <chr>, b\_dna\_log <dbl>, year <dbl>

# 따라서  
dat1 %>%   
 filter( !is.na(b\_inr)) #즉 NA가 !(not)값들만 나온다.

# A tibble: 984 x 24  
 index\_date gender age last\_date treat\_gr lc dietpl dietpl\_date  
 <date> <chr> <dbl> <date> <chr> <dbl> <dbl> <date>   
 1 2007-01-05 M 54 2014-07-18 ETV 1 0 2017-09-13   
 2 2007-01-10 F 45 2016-08-25 ETV 0 0 2016-08-25   
 3 2007-01-11 M 49 2017-06-21 ETV 1 0 2017-06-21   
 4 2007-01-12 M 26 2012-12-17 ETV 0 0 2012-12-17   
 5 2007-01-18 M 50 2009-05-15 ETV 1 0 2009-05-15   
 6 2007-01-18 M 33 2013-01-11 ETV 1 0 2013-01-11   
 7 2007-01-26 M 49 2013-03-31 ETV 1 0 2017-08-07   
 8 2007-01-31 F 49 2009-03-20 ETV 1 0 2009-03-20   
 9 2007-02-01 M 50 2017-09-12 ETV 0 0 2017-09-12   
10 2007-02-01 M 32 2010-09-30 ETV 0 0 2010-09-30   
# … with 974 more rows, and 16 more variables: dietpl\_yr <dbl>, hcc <dbl>,  
# hcc\_date <date>, hcc\_yr <dbl>, b\_lab\_date <date>, b\_alt <dbl>, b\_bil <dbl>,  
# b\_inr <dbl>, b\_cr <dbl>, b\_plt <dbl>, b\_alb <dbl>, b\_eag <dbl>,  
# b\_eab <dbl>, b\_dna <chr>, b\_dna\_log <dbl>, year <dbl>