데이터 정제

결측치 정제하기

결측치

- ❖ 누락된 값, 비어있는 값
- ❖ 함수 적용 불가, 분석 결과 왜곡
- ❖ 제거 후 분석 실시

원자료

id	class	english	science
1	1	98	50
2	1	97	60
3	1	86	78
4	1	98	58
5		80	65
6	2	89	
7	2	90	45
8	2		99999
9	3	98	15
10	3	98	45
11	3	99999	65
12	3	85	32

➡

정제하기

id	class	english	science
1	1	98	50
2	1	97	60
3	1	86	78
4	1	98	58
7	2	90	45
9	3	98	15
10	3	98	45
12	3	85	32

결측지 만들기

```
df <- data.frame(sex = c("M", "F", NA, "M", "F"),
                 score = c(5, 4, 3, 4, NA))
df
##    sex score
## 1    M     5
## 2    F     4
## 3 <NA>     3
## 4    M     4
## 5    F    NA
```

결측치 확인하기

```
is.na(df)        # 결측치 확인
##        sex score
## [1,] FALSE FALSE
## [2,] FALSE FALSE
## [3,]  TRUE FALSE
## [4,] FALSE FALSE
## [5,] FALSE  TRUE
table(is.na(df))  # 결측치 빈도 출력
## FALSE  TRUE
##     8     2
table(is.na(df$sex))    # sex 결측치 빈도 출력
## FALSE  TRUE
##     4     1
table(is.na(df$score))  # score 결측치 빈도 출력
## FALSE  TRUE
##     4     1
```

결측치 포함된 상태로 분석

```
mean(df$score)  # 평균 산출
## [1] NA
sum(df$score)   # 합계 산출
## [1] NA
```

결측치 제거하기

```
library(dplyr) # dplyr 패키지 로드
df %>% filter(is.na(score))    # score가 NA인 데이터만 출력
##   sex score
## 1   F    NA
df %>% filter(!is.na(score))  # score 결측치 제거
##    sex score
## 1    M     5
## 2    F     4
## 3 <NA>     3
## 4    M     4

df_nomiss <- df %>% filter(!is.na(score))  # score 결측치 제거
mean(df_nomiss$score)                      # score 평균 산출
## [1] 4
sum(df_nomiss$score)                       # score 합계 산출
## [1] 16
```

모든 결측치 제거

```
# score, sex 결측치 제외
df_nomiss <- df %>% filter(!is.na(score) & !is.na(sex))
df_nomiss
##   sex score
## 1   M     5
## 2   F     4
## 3   M     4

df_nomiss2 <- na.omit(df)  # 모든 변수에 결측치 없는 데이터 추출
df_nomiss2                 # 출력
##   sex score
## 1   M     5
## 2   F     4
## 4   M     4
```

함수를 이용한 결측치 제거

```
mean(df$score, na.rm = T)  # 결측치 제외하고 평균 산출
## [1] 4
sum(df$score, na.rm = T)   # 결측치 제외하고 합계 산출
## [1] 16
```

실습 - 평균구하기

```
# 데이터 불러오기
exam <- read.csv("csv_exam.csv")
# 3, 8, 15행의 math에 NA 할당
exam[c(3, 8, 15), "math"] <- NA
# 평균 산출
exam %>% summarise(mean_math = mean(math))
# 결측치 제외하고 평균 산출
exam %>% summarise(mean_math = mean(math, na.rm = T))
```

실습

```
exam %>% summarise(mean_math = mean(math, na.rm = T),      # 평균 산출
                   sum_math = sum(math, na.rm = T),        # 합계 산출
                   median_math = median(math, na.rm = T))  # 중앙값 산출
```

결측치 대체하기

결측치 대체하기

- ❖ 문제점
  - 결측치 많을 경우 모두 제외하면 데이터 손실 큼
- ❖ 대안
  - 다른 값 채워넣기
- ❖ 결측치 대체법(Imputation)
  - 대표값(평균, 최빈값 등)으로 일괄 대체
  - 통계분석 기법 적용, 예측값 추정해서 대체

실습 – 평균값 적용

```
# 결측치 제외하고 math 평균 산출
meanMath = mean(exam$math, na.rm = T)
#정수화
meanMath = as.integer(meanMath)
# math가 NA면 대표값 meanMath로 대체
exam$math <- ifelse(is.na(exam$math), meanMath, exam$math)
# 결측치 빈도표 생성
table(is.na(exam$math))
# math 평균 산출
mean(exam$math)
```

Quiz

- mpg 데이터를 이용해서 분석 문제를 해결해 보세요.
- mpg 데이터 원본에는 결측치가 없습니다. 우선 mpg 데이터를 불러와 몇 개의 값을 결측치로 만들겠습니다. 아래 코드를 실행하면 다섯 행의 hwy 변수에 NA가 할당됩니다.

```
# mpg 데이터 불러오기
mpg <- as.data.frame(ggplot2::mpg)
# NA 할당하기
mpg[c(65, 124, 131, 153, 212), "hwy"] <- NA
```

- 결측치가 들어있는 mpg 데이터를 활용해서 문제를 해결해보세요.
  - Q1. drv(구동방식)별로 hwy(고속도로 연비) 평균이 어떻게 다른지 알아보려고 합니다. 분석을 하기 전에 우선 두 변수에 결측치가 있는지 확인해야 합니다. drv 변수와 hwy 변수에 결측치가 몇 개 있는지 알아보세요.
  - Q2. filter()를 이용해 hwy 변수의 결측치를 제외하고, 어떤 구동방식의 hwy 평균이 높은지 알아보세요. 하나의 dplyr 구문으로 만들어야 합니다.

이상치 정제하기

이상치

❖ **정의**
- 정상범주에서 크게 벗어난 값

❖ **분석**
- 이상치 포함 시 분석 결과 왜곡
- 결측 처리 후 제외하고 분석

이상치 종류	예	해결 방법
존재할 수 없는 값	성별 변수에 5	결측 처리
극단적인 값	몸무게 변수에 200	정상범위 기준 정해서 결측 처리

이상치 제거 – 존재할 수 없는 값

❖ 논리적으로 존재할 수 없으므로 바로 결측 처리 후 분석 시 제외

```
#이상치 생성
#sex : 1 or 2
#score : 1~5
outlier <- data.frame(
  sex = c(1, 2, 1, 5, 2, 1),
  score = c(5, 4, 3, 4, 2, 6))
#이상치 확인
table(outlier$sex)
table(outlier$score)
```

결측치 제거 및 분석

```
#결측 처리
outlier$sex <- ifelse(outlier$sex == 5, NA, outlier$sex)
outlier

# score가 1~5 아니면 NA 할당
outlier$score <- ifelse(outlier$score > 5, NA, outlier$score)
outlier

#결측치 제거 및 분석
outlier %>%
  filter(!is.na(sex) & !is.na(score)) %>%
  group_by(sex) %>%
  summarise(mean_score = mean(score))
```

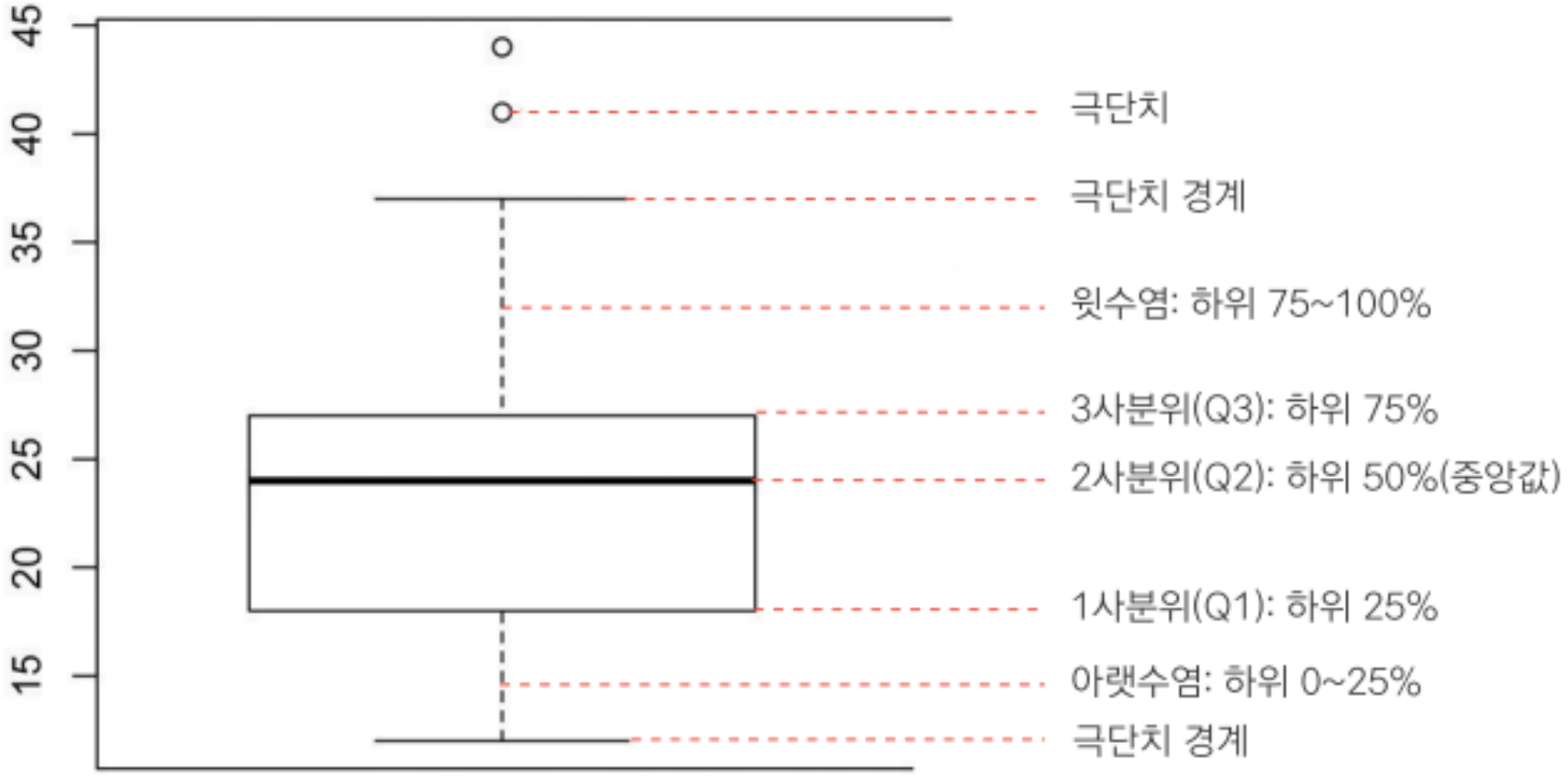
이상치 제거 – 극단적인 값

❖ 정상범위 기준을 벗어난 결측 처리

판단 기준	예
논리적 판단	성인 몸무게 40kg~150kg 벗어나면 극단치
통계적 판단	상하위 0.3% 극단치 또는 상자그림 1.5 IQR 벗어나면 극단치

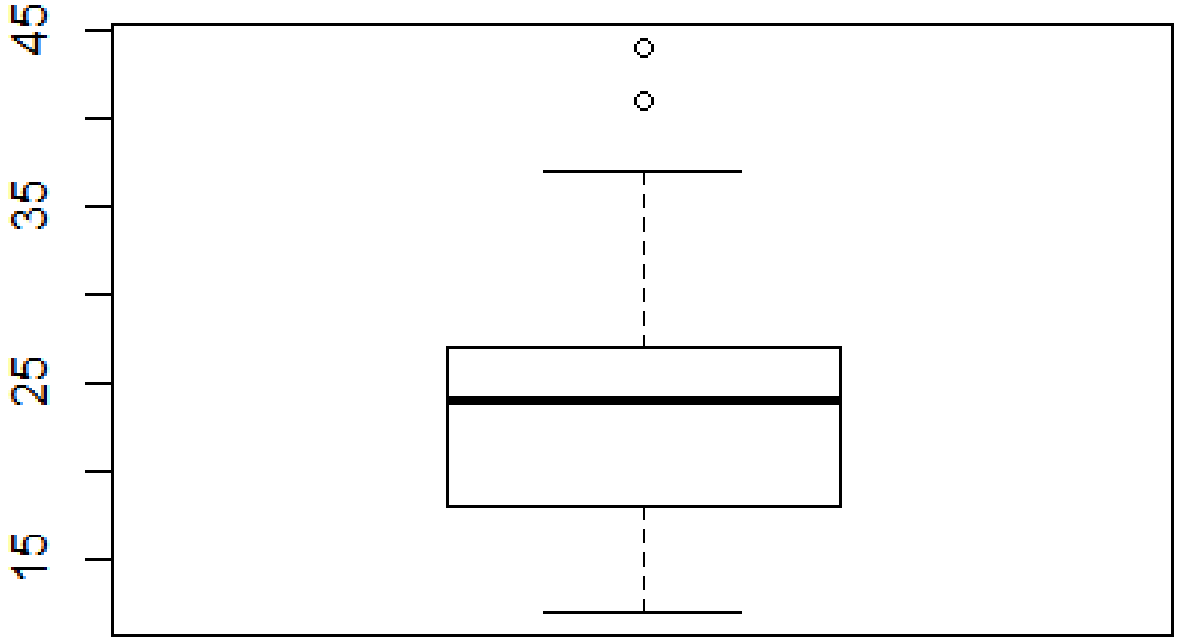
Boxplot을 이용한 극단치 기준 설정

```
mpg <- as.data.frame(ggplot2::mpg)
boxplot(mpg$hwy)
```



통계치 출력

```
boxplot(mpg$hwy)$stats  # 상자그림 통계치 출력
##      [,1]
## [1,]   12
## [2,]   18
## [3,]   24
## [4,]   27
## [5,]   37
## attr(,"class")
##         1
## "integer"
```



결측처리

```
# 12~37 벗어나면 NA 할당
mpg$hwy <- ifelse(mpg$hwy < 12 | mpg$hwy > 37, NA, mpg$hwy)
table(is.na(mpg$hwy))

#결측치 제외 및 분석
mpg %>%
  group_by(drv) %>%
  summarise(mean_hwy = mean(hwy, na.rm = T))
```

Quiz - 사전준비

- mpg 데이터를 이용해서 분석 문제를 해결해 보세요.
- 우선 mpg 데이터를 불러와서 일부러 이상치를 만들겠습니다. drv(구동방식) 변수의 값은 4(사륜구동), f(전륜구동), r(후륜구동) 세 종류로 되어있습니다. 몇 개의 행에 존재할 수 없는 값 k를 할당하겠습니다. cty(도시 연비) 변수도 몇 개의 행에 극단적으로 크거나 작은 값을 할당하겠습니다.
-
```
mpg <- as.data.frame(ggplot2::mpg)                  # mpg 데이터 불러오기
mpg[c(10, 14, 58, 93), "drv"] <- "k"                # drv 이상치 할당
mpg[c(29, 43, 129, 203), "cty"] <- c(3, 4, 39, 42)  # cty 이상치 할당
```

Quiz

- 이상치가 들어있는 mpg 데이터를 활용해서 문제를 해결해보세요.
- 구동방식별로 도시 연비가 다른지 알아보려고 합니다. 분석을 하려면 우선 두 변수에 이상치가 있는지 확인하려고 합니다.
  - Q1. drv에 이상치가 있는지 확인하세요. 이상치를 결측 처리한 다음 이상치가 사라졌는지 확인하세요. 결측 처리 할 때는 %in% 기호를 활용하세요.
  - Q2. 상자 그림을 이용해서 cty에 이상치가 있는지 확인하세요. 상자 그림의 통계치를 이용해 정상 범위를 벗어난 값을 결측 처리한 후 다시 상자 그림을 만들어 이상치가 사라졌는지 확인하세요.
  - Q3. 두 변수의 이상치를 결측처리 했으니 이제 분석할 차례입니다. 이상치를 제외한 다음 drv별로 cty 평균이 어떻게 다른지 알아보세요. 하나의 dplyr 구문으로 만들어야 합니다.

Boxplot 이해

이해하기

```
boxplot(
  x # 상자 그림을 그릴 데이터
)

boxplot(
  formula, # y ~ grp의 형식으로 y는 분포를 그릴 값, grp는 값들을 그룹 짓는 변수다.
           # 결국 y ~ grp는 y에 대한 상자 그림을 grp마다 그린다.
  data=NULL,        # 포뮬러가 적용될 데이터 프레임(또는 리스트)
  horizontal=FALSE, # TRUE면 세로로, FALSE면 가로로 상자 그림을 그린다.
  notch=FALSE,   # TRUE면 중앙값에 대한 일종의 신뢰 구간을 표시하는 움푹 들어간
                 # 구간을 그린다. 만약 두 개의 상자 그림에 notch=TRUE를 지정해
                 # 그린 다음 두 그래프의 움푹 들어간 구간이 서로 겹치지 않으면
                 # 이는 두 데이터의 중앙값이 서로 다르다는 강한 증거가 된다.
...
)
```

데이터 분석

❖ ?mtcars

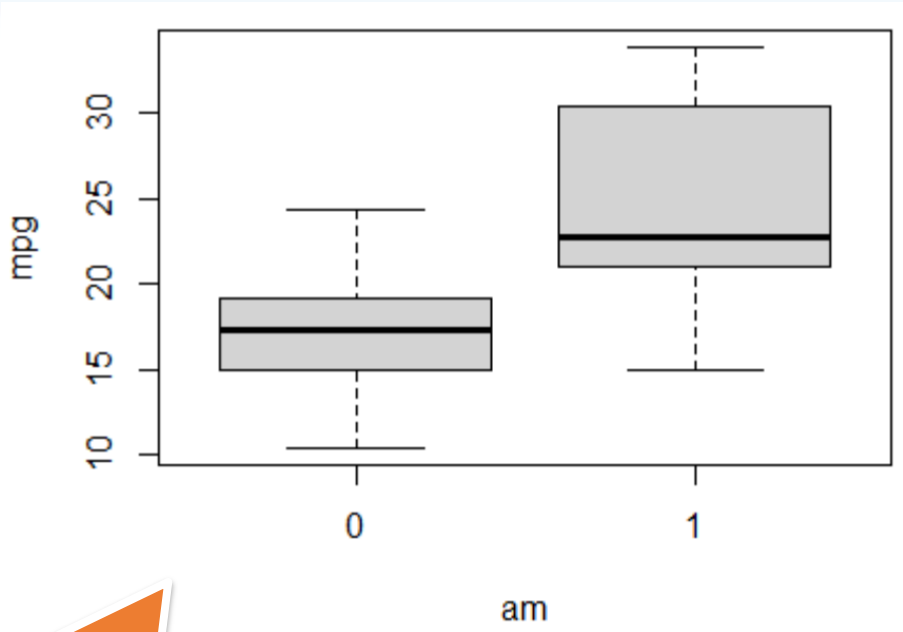
Format

A data frame with 32 observations on 11 (numeric) variables.

[, 1]	mpg	Miles/(US) gallon
[, 2]	cyl	Number of cylinders
[, 3]	disp	Displacement (cu.in.)
[, 4]	hp	Gross horsepower
[, 5]	drat	Rear axle ratio
[, 6]	wt	Weight (1000 lbs)
[, 7]	qsec	1/4 mile time
[, 8]	vs	Engine (0 = V-shaped, 1 = straight)
[, 9]	am	Transmission (0 = automatic, 1 = manual)
[,10]	gear	Number of forward gears
[,11]	carb	Number of carburetors

실습 – formula, data

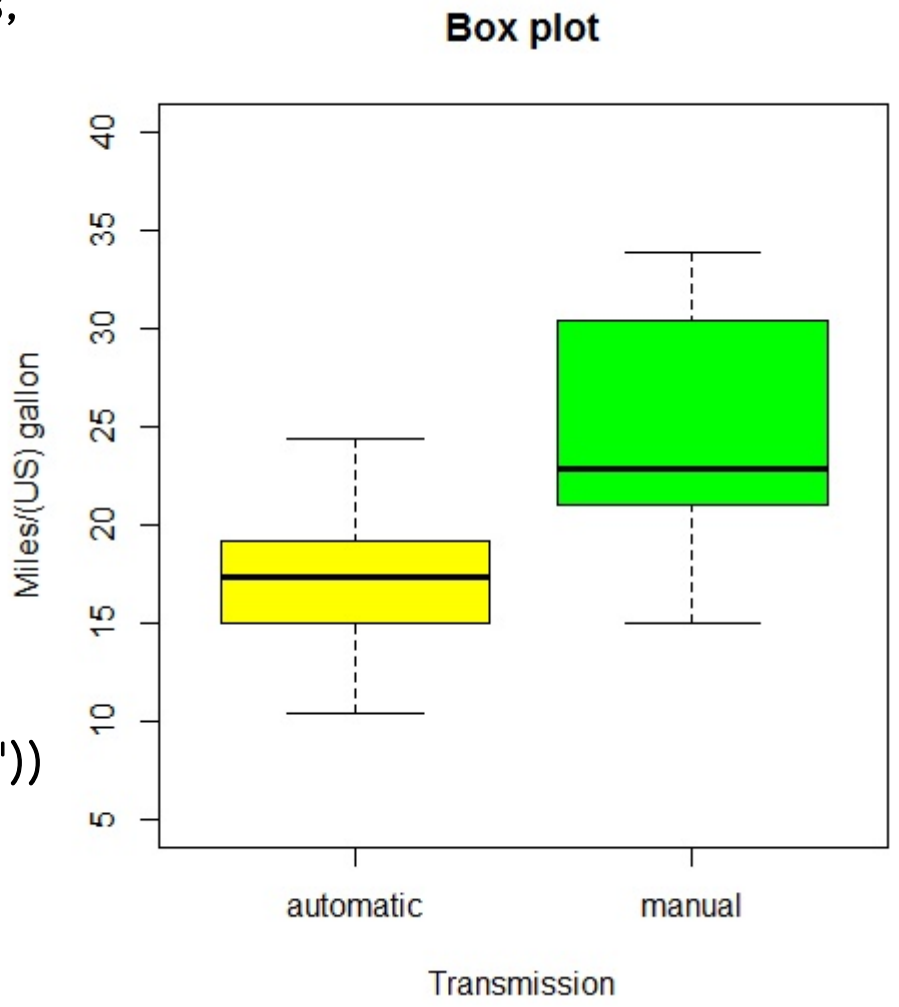
```
#R 내장 데이터
str(mtcars)
> #변수 설명
> #mpg: Miles/(US) gallon
> #am:  Transmission
> (0 = automatic, 1 = manual)
boxplot(formula=mpg~am, data=mtcars)
```



변속기에 따른
연비의 차이

실습

```
boxplot(formula=mpg~am,data=mtcars,
        #그룹별 색상 지정
        col=c("yellow","green"),
        #x축, y축 범위 지정
        xlim=c(0.5,2.5),
        ylim=c(5,40),
        #제목(라벨) 지정
        main="Box plot",
        xlab="Transmission",
        ylab="Miles/(US) gallon",
        #names= 범주 이름
        names=c("automatic","manual"))
```

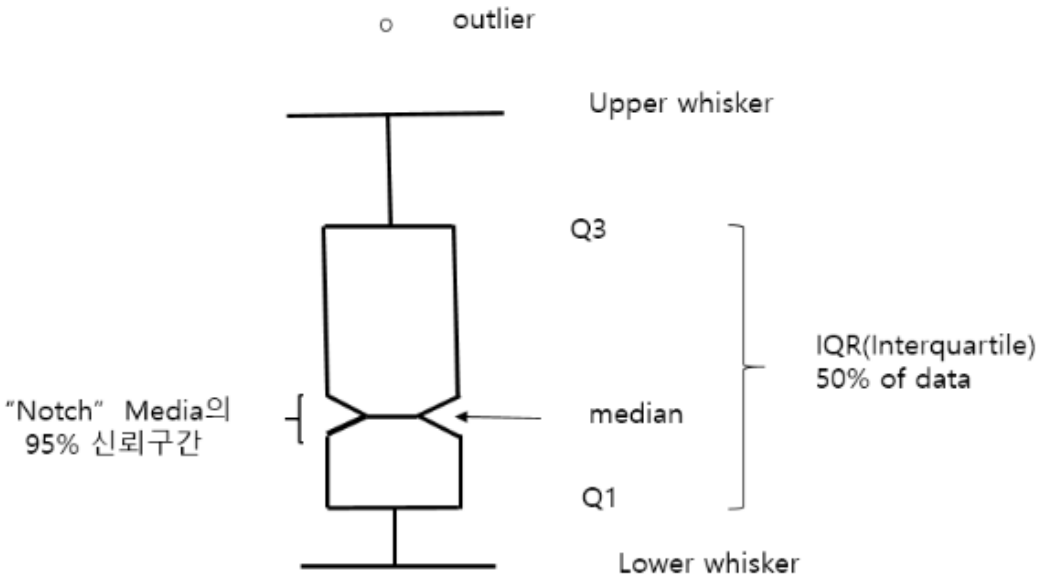


실습

```
#x, y축 제거
boxplot(formula=mpg~am,data=mtcars,axes=F)
axis(1)   #하단 축
axis(2)    #좌측 축
axis(3)    #상단 축
axis(4)    #우측 축
mtext(c("automatic", "manual"))     #축에 출력할 내용
mtext(c("automatic", "manual"), side=1) #상(3)하(1)좌(2)우(4)
mtext(c("automatic", "manual"), side=1, line=2.5)     #기본위치에서 간격
mtext(c("automatic", "manual"), side=1, line=2.5, at=c(0, 1)) #출력할 위치
mtext(c("automatic", "manual"), side=1, line=2.5, at=c(1, 2))
```

notch

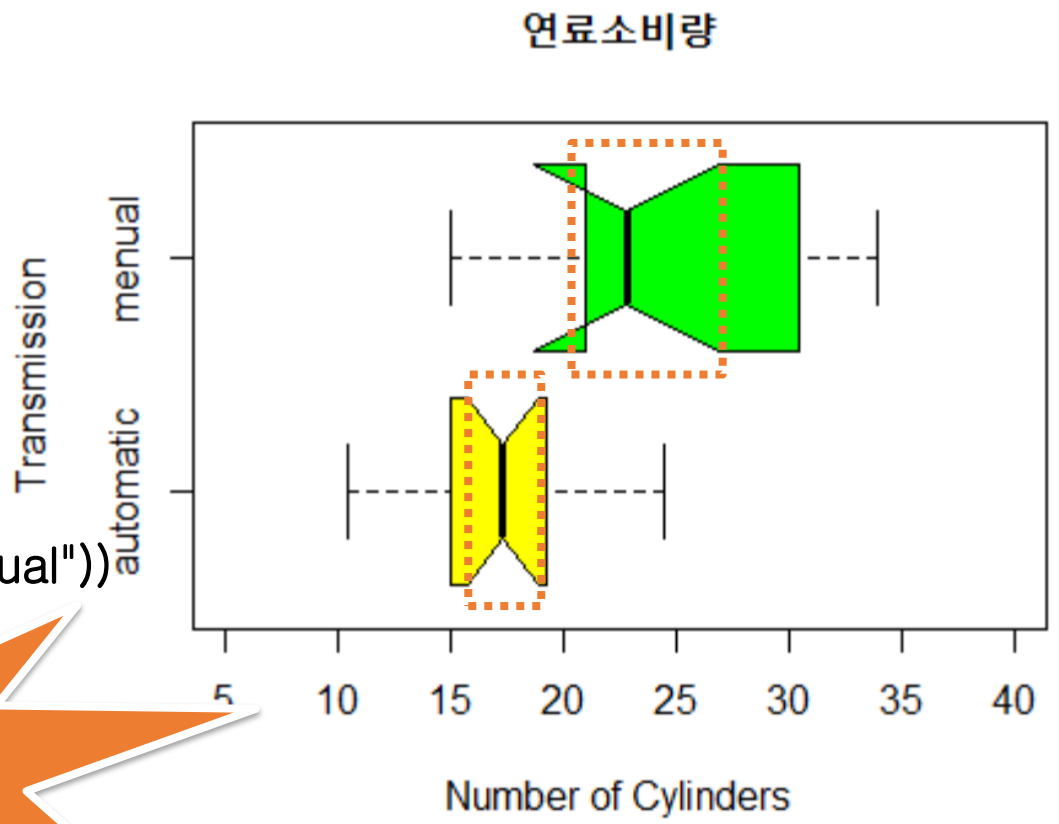
- notch 미국식 [nɑːtʃ] 영국식 [nɒtʃ] 중요
  - (질·성취 정도를 나타내는) 급수
  - (기록 등을 위해 새겨 놓은 V 자나 동그라미) 표시
  - (승리·높은 점수 등을) 올리다
- Notch=TRUE



Notched Box plot

실습

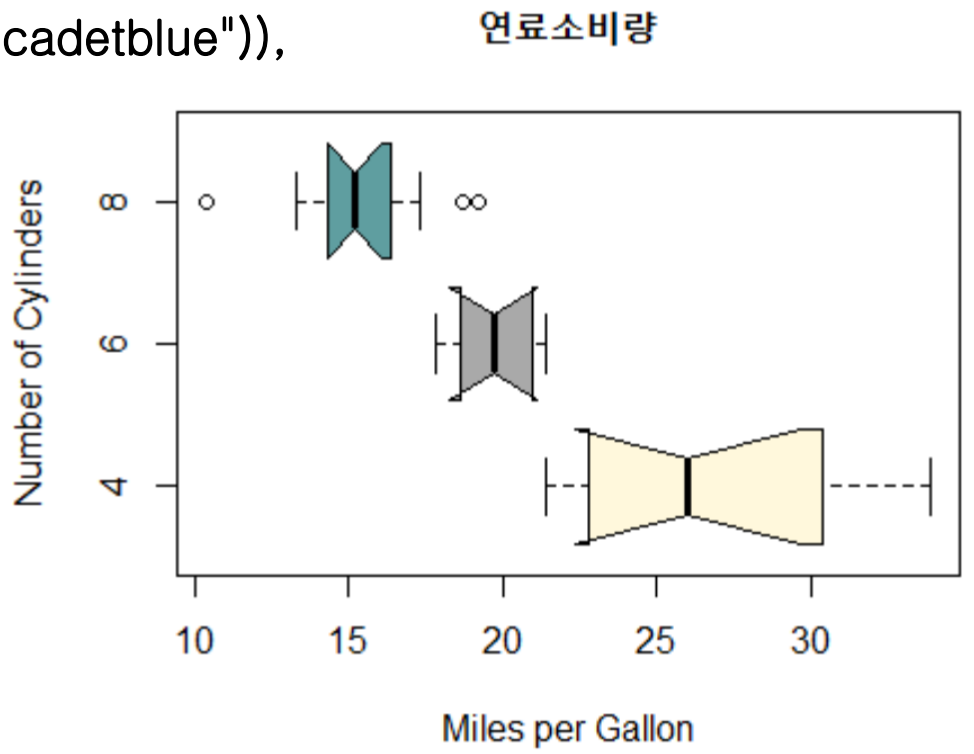
```
boxplot(mpg~am, data=mtcars,
        main="연료소비량",
        col=(c("yellow", "green")),
        ylim=c(5, 40),
        xlab="Number of Cylinders",
        ylab="Transmission",
        notch=TRUE,
        horizontal=T,
        names=c("automatic", "menual"))
```



중간값의 신뢰구간이
겹쳐지지 않음으로
변속기에 따른
연비의 차이는
유의미하다.

실린더에 따른 주행연비

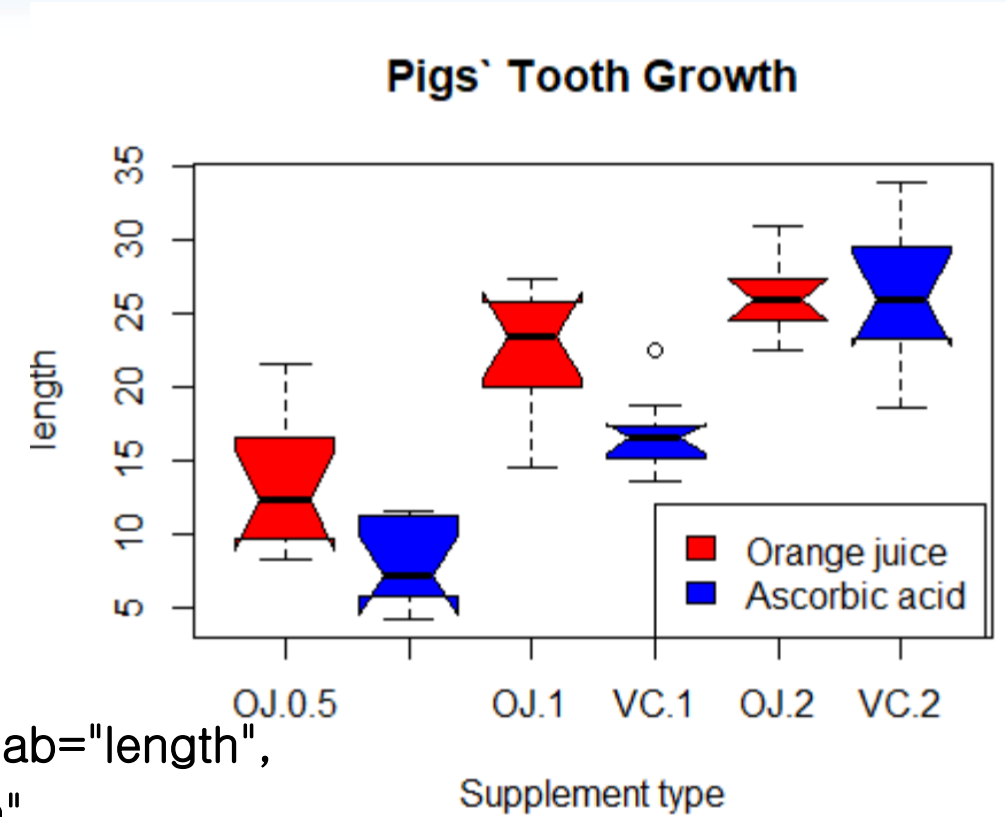
```
boxplot(mpg~cyl, data=mtcars,
        main="연료소비량",
        col=(c("cornsilk", "darkgray", "cadetblue")),
        ylab="Number of Cylinders",
        xlab="Miles per Gallon",
        notch=TRUE,
        horizontal=TRUE, range=1)
```



Range는 극단치
경계를 조절

실습

```
#정보 확인
?ToothGrowth
head(ToothGrowth)
summary(ToothGrowth)
#factor 생성
attach(ToothGrowth)
supp<-as.factor(supp)
dose<-as.factor(dose)
#boxplot 생성
boxplot(len~supp+dose,
        xlab="Supplement type",ylab="length",
        main="Pigs` Tooth Growth",
        notch=TRUE,col=c("red","blue"))
#legend(좌표, 출력글, 색상)
legend(4,12,c("Orange juice","Ascorbic acid"),fill=c("red","blue"))
```



실습 - text

```
boxstats <- boxplot(iris$Sepal.Width)
#출력좌표 x, y, 출력 데이터 labels
#NROW row의 개수
#rep(적용데이터, 적용횟수)
text(x=rep(1, NROW(boxstats$out)),
     y=boxstats$out,
     labels=boxstats$out,
     #하(1), 좌(2), 상(3), 우(4)
     pos=c(1, 2, 3, 4)
     )
```

실습 - text

```
#행으로 출력
boxstats <- boxplot(iris$Sepal.Width, horizontal=TRUE)
text(boxstats$out,
     rep(1, NROW(boxstats$out)),
     labels=boxstats$out,
     pos=c(1, 1, 3, 1))
```