

데이터과학을 위한 **R**프로그래밍

3주차. R 데이터구조(생성, 추출)



이혜선 교수

포항공과대학교 산업경영공학과



목차

3주차. R 데이터구조(생성, 추출)

1차시

R 데이터 생성 (불러들이기)

2차시

R 데이터 활용 I (subset, 내보내기)

3차시

R 데이터 활용 II (dplyr 활용)



3주차

3차시

R 데이터 활용 II

(dplyr 활용)

데이터 핸들링

- ☑ dplyr : 데이터 핸들링을 편리하게 하는 라이브러리

dplyr 패키지의 주요 함수

- ❖ select : 일부변수를 선택
- ❖ filter : 필터링 기능 (조건에 맞는 데이터 추출)
- ❖ mutate : 새로운 변수 생성
- ❖ group_by : 그룹별 통계량을 얻을 때
- ❖ summarize : 요약통계량 (mean, min, max, sum)
- ❖ arrange : 행 정렬시 사용

- ☑ dplyr 설치 및 설정

```
# Data handling using "dplyr"  
install.packages("dplyr")  
library(dplyr)
```

앞에서의 subset과 같은 기능

데이터 핸들링

✓ dplyr 패키지 설치 및 기본 설정

프로그램 편집 창

```
# lec3_3.r
# Data handling
# Data analysis with autompg.csv

# data manipulation package
# select, filter, group by, summarise in dplyr package
install.packages("dplyr")
library(dplyr)

# set working directory
# change working directory
setwd("D:/tempstore/moocr")

# Read txt file with variable name
# http://archive.ics.uci.edu/ml/datasets/Auto+MPG
```

```
# Data reading in R
car<-read.csv(file="autompg.csv")
attach(car)
```

Step0 : 분석을 위한 설정
(install, library, setwd)

Step1 : 데이터핸들링
(csv파일 불러들이기, ...)

❖ tbl_df() : 빅데이터에 유용함. df는 데이터프레임을 의미

오픈 데이터 활용

✓ UCI repository data

UCI Machine Learning Repository
Center for Machine Learning and Intelligent Systems

Auto MPG Data Set
Download [Data Folder](#) [Data Set Description](#)

Abstract: Revised from CMU Statlib library, data concerns city-cycle fuel consumption.

Data Set Characteristics:	Multivariate	Number of instances:	398	Area:	NA
Attribute Characteristic:	Categorical: Real	Number of Attributes:	9	Date Derived:	1983.07.07
Associated Tasks:	Regression	Missing Values?	Yes	Number of Web Hits:	218927

Source:
This dataset was taken from the Statlib library which is maintained at Carnegie Mellon University. The dataset was used in the 1983 American...

Data Set Information:
This dataset is a slightly modified version of the dataset provided in the Statlib library. It has with the use by Ross Quinlan (1983) in predicting city fuel economy values for the "mpg" attribute. This original dataset is available in the file "auto-mpg.data-original".
The data concerns city-cycle fuel consumption in miles per gallon, to be predicted in terms of 3 multivalued discrete and 3 continuous attributes.

autompg.csv

	A	B	C	D	E	F	G	H	I
1	mpg	cyl	disp	hp	wt	accler	year	origin	carname
2	18	8	307	17	3504	12	70	1	chevrolet c
3	15	8	350	35	3693	11.5	70	1	buick skyla
4	18	8	318	29	3436	11	70	1	plymouth s
5	16	8	304	29	3433	12	70	1	amc rebel s
6	17	8	302	24	3449	10.5	70	1	ford torino
7	15	8	429	42	4341	10	70	1	ford galaxie
8	14	8	454	47	4354	9	70	1	chevrolet in
9	14	8	440	46	4312	8.5	70	1	plymouth f
10	14	8	455	48	4425	10	70	1	pontiac cat
11	15	8	390	40	3850	8.5	70	1	amc ambas
12	15	8	383	37	3563	10	70	1	dodge chal
13	14	8	340	34	3609	8	70	1	plymouth 'c

● 오픈 데이터 활용

☑ autmpg 데이터

- 1. mpg: continuous (연비 : 연속형변수)
- 2. cylinders: multi-valued discrete (실린더 : 정수값)
- 3. displacement: continuous (배기량 : 연속형변수)
- 4. horsepower: continuous (마력 : 연속형변수)
- 5. weight: continuous (무게 : 연속형변수)
- 6. acceleration: continuous (가속 : 연속형변수)
- 7. year: multi-valued discrete (모델연도 : 정수값)
- 8. origin: multi-valued discrete (정수값)
- 9. car name: string (unique for each instance) (차종류 이름)

☑ 변수특성 변경 (as.numeric, as.integer, factor)

데이터 활용 II

☑ 데이터 불러들이기


```
# Data reading in R  
car<-read.csv(file="autompg.csv")  
attach(car)
```

```
head(car)  
dim(car)  
str(car)
```

R 데이터 이름은? : car

car 데이터의 수는? : 398

car 는 몇개의 변수? : 9



```
> head(car)  
  mpg cyl  disp  hp   wt  accler  year origin      carname  
1  18   8  307  17 3504   12.0    70      1 chevrolet chevelle malibu  
2  15   8  350  35 3693   11.5    70      1      buick skylark 320  
3  18   8  318  29 3436   11.0    70      1    plymouth satellite  
4  16   8  304  29 3433   12.0    70      1          amc rebel sst  
5  17   8  302  24 3449   10.5    70      1          ford torino  
6  15   8  429  42 4341   10.0    70      1    ford galaxie 500  
  
> dim(car)  
[1] 398  9
```


데이터 활용 II

☑ 데이터의 전체 구조 파악하기 : str(데이터이름)

```
# Data reading in R
car<-read.csv(file="autompg.csv")
attach(car)

head(car)
dim(car)
str(car)
```

car 변수들 중 실수값으로 저장된 변수들은? 4개

car 변수들 중 정수값으로 저장된 변수들은? 4개

car 변수들 중 문자변수는 ? 1개

```
> str(car)
'data.frame':  398 obs. of  9 variables:
 $ mpg      : num  18 15 18 16 17 15 14 14 15 ...
 $ cyl      : int   8  8  8  8  8  8  8  8  8 ...
 $ disp     : num  307 350 318 304 302 429 454 440 455 390 ...
 $ hp       : num   17 35 29 29 24 42 47 46 48 40 ...
 $ wt       : int  3504 3693 3436 3433 3449 4341 4354 4312 4425 3850
 $ accler   : num   12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...
 $ year     : int   70 70 70 70 70 70 70 70 70 70 ...
 $ origin   : int    1  1  1  1  1  1  1  1  1 ...
 $ carname  : chr   "chevrolet chevelle malibu" "buick skylark 320" "p
th satellite" "amc rebel sst" ...
```

데이터 활용 II

☑ 데이터 요약하기 : summary(데이터이름)

```
# summary  
summary(car)
```

최소값
25%값
중위수
평균
75%값
최대값

```
> summary(car)  
      mpg      cyl      disp      hp  
Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 1.00  
1st Qu.:17.50   1st Qu.:4.000   1st Qu.:104.2   1st Qu.:26.00  
Median :23.00   Median :4.000   Median :148.5   Median :60.50  
Mean   :23.51   Mean   :5.455   Mean   :193.4   Mean   :51.39  
3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:262.0   3rd Qu.:79.00  
Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :94.00  
      wt      accler      year      origin  
Min.   :1613   Min.   : 8.00   Min.   :70.00   Min.   :1.000  
1st Qu.:2224   1st Qu.:13.82   1st Qu.:73.00   1st Qu.:1.000  
Median :2804   Median :15.50   Median :76.00   Median :1.000  
Mean   :2970   Mean   :15.57   Mean   :76.01   Mean   :1.573  
3rd Qu.:3608   3rd Qu.:17.18   3rd Qu.:79.00   3rd Qu.:2.000  
Max.   :5140   Max.   :24.80   Max.   :82.00   Max.   :3.000  
      carname  
Length:398  
Class :character
```

데이터 활용 II

- ☑ 데이터의 요약통계치 (빈도 구하기) : table(데이터이름)

```
table(origin)
table(year)
```



```
> table(origin)
origin
 1    2    3
249  70  79
```

attach(데이터이름) : 현재 세션에서 나오는 변수들은 그 '데이터'의 변수로 인식 => 데이터이름을 안써줘도 됨!

```
> table(year)
year
70 71 72 73 74 75 76 77 78 79 80 81 82
freq 29 28 28 40 27 30 34 28 36 29 29 29 31
```

1970년도부터 1982년까지의 차량

- ☑ 데이터의 요약통계치 (평균, 표준편차) 구하기

개별변수의 평균 구하기

```
# mean and standard deviation
mean(mpg)
mean(hp)
mean(wt)
```



```
> mean(mpg)
[1] 23.51457
> mean(hp)
[1] 51.38945
> mean(wt)
[1] 2970.425
```

데이터 핸들링 - 데이터 추출 (select)

☑ 변수 추출 : `select(데이터, 변수이름, ...)`

car 데이터에서 mpg, hp 변수만 추출

```
# 1. subset data : selecting a few variables  
set1<-select(car, mpg, hp)
```

car 데이터에서 mpg로 시작하는 변수를 제외하고 set2 라는 데이터를 생성

```
# 2. subset data : Drop variables with -  
set2<-select(car, -starts_with("mpg"))
```

starts_with(): 변수 시작

데이터핸들링 - 데이터 추출 (filter)

✓ 조건식에 맞는 데이터 추출 : filter(데이터, 변수조건, ...)

car 데이터에서 mpg가 30보다 큰 행 추출

```
# 3. subset data : filter mpg>30  
set3<-filter(car, mpg>30)  
head(set3)
```

```
> head(set3)  
# A tibble: 6 x 9  
  mpg   cyl  disp    hp  wt  accler  year origin carname  
  <dbl> <int> <dbl> <dbl> <int> <dbl> <int> <int> <fct>  
1    31     4    71     62  1773    19     71     3 toyota corolla 1200  
2    35     4    72     66  1613    18     71     3 datsun 1200  
3    31     4    79     64  1950    19     74     3 datsun b210  
4    32     4    71     62  1836    21     74     3 toyota corolla 1200  
5    31     4    76     53  1649   16.5    74     3 toyota corona  
6    32     4    83     58  2003    19     74     3 datsun 710
```

데이터 핸들링 - 변수 생성 (mutate)

☑ 변수 생성 : mutate(새로운 변수이름=기존변수 활용)

%>% (파이프 연산자) 연산자 사용하여 연결

```
# 4. create a derived variable
set4 <- car %>% ①
  filter(!is.na(mpg)) %>% ②
  mutate(mpg_km = mpg * 1.609) ③
```

파이프 연산자 : 앞에서부터 ①, ②, ③
순서대로 수행하여 데이터 전처리를 하고
set4라는 이름으로 저장

☑ filter car 데이터 mpg 열의 NA가 아닌 모든 데이터 추출

is.na() NA 여부 판단하는 함수
(! 기호는 부정하는 기호)

☑ mutate 기존의 mpg 사용하여 새로운 변수 mpg_km 생성

	mpg	cyl	disp	hp	wt	accler	year	origin	carname	mpg_km
1	16.0	6	337.0	17	3504	12.0	70	1	chevrolet chevelle malibu	25.9620
2	15.0	8	350.0	35	3693	11.5	70	1	buick skylark 320	24.1350
3	18.0	6	318.0	28	3436	11.0	70	1	plymouth satellite	28.9620
4	16.0	8	304.0	29	3433	12.0	70	1	amc rebel sst	25.7440
5	17.0	8	302.0	24	3449	10.5	70	1	ford torino	27.3530
6	15.0	8	429.0	42	4341	10.0	70	1	ford galaxie 500	24.1330
7	14.0	8	454.0	47	4356	9.0	70	1	chevrolet impala	22.6560

데이터 핸들링 - 데이터 요약통계치

☑ 데이터 요약통계치(평균 구하기) : summarize(mean(변수이름))

mpg, hp, wt의 평균값 구하기

```
# mean and standard deviation  
car %>%  
  summarize(mean(mpg), mean(hp), mean(wt))
```



```
> car %>%  
+   summarize(mean(mpg), mean(hp), mean(wt))  
# A tibble: 1 x 3  
  mean(mpg) mean(hp) mean(wt)  
    <dbl>    <dbl>    <dbl>  
1    23.5    51.4    2970.
```

몇 개 변수들의 평균값 한번에 구하기

```
# mean of some variables  
select(car, 1:6) %>%  
  colMeans()
```



```
> select(car, 1:6) %>%  
+   colMeans()  
      mpg      cyl      disp  
23.514573  5.454774 193.425879  
      hp      wt      accler  
51.389447 2970.424623 15.568090
```

1~6열 추출함

colMeans 데이터를 열로 재구성하여 평균값 구함

데이터 핸들링 - 데이터 요약통계치

✓ 벡터화 요약치 : summarize_all(FUN)

변수를 추출해 기술통계치 구하고 요약치 보여줌

```
# table with descriptive statistics
a1 <- select(car, 1:6) %>% summarize_all(mean)
a2 <- select(car, 1:6) %>% summarize_all(sd)
a3 <- select(car, 1:6) %>% summarize_all(min)
a4 <- select(car, 1:6) %>% summarize_all(max)
table1 <- rbind(a1,a2,a3,a4)
rownames(table1) <- c("mean", "sd", "min", "max")
table1
```

> table1

	mpg	cyl	disp	hp	wt	accler
mean	23.514573	5.454774	193.4259	51.38945	2970.4246	15.568090
sd	7.815984	1.701004	104.2698	29.93236	846.8418	2.757689
min	9.000000	3.000000	68.0000	1.00000	1613.0000	8.000000
max	46.600000	8.000000	455.0000	94.00000	5140.0000	24.800000

데이터핸들링 - 그룹별 기술통계치

☑ 그룹별 통계량 얻기 : `group_by(변수), summarize(__=FUN())`

그룹별 요약통계량 구하기

```
# summary statistics by group variable
car %>%
  group_by(cyl) %>%
  summarize(mean_mpg = mean(mpg, na.rm = TRUE))
```

```
# A tibble: 5 x 2
  cyl mean_mpg
<int> <dbl>
1     3    20.6
2     4    29.3
3     5    27.4
4     6    20.0
5     8    15.0
```

함수	요약통계량
mean	평균
min	최솟값
max	최댓값
sum	합계
var	분산
sd	표준편차
median	중앙값
n	빈도

➤ `group_by` car데이터의 cyl열을 그룹으로 묶음

➤ `summarize()` cyl그룹의 mpg 평균을 구함

➔ `na.rm = TRUE` 통계 분석 시 결측 값을 제외함
 ➔ remove

✧ 요약통계량을 구할 때 `group_by`와 `summarize` 함께 사용하는 경우 많음