3장. 데이터 다루기

박영식(youngsik.park@bsl-lausanne.ch)

데이터 다루기

◎ R로 데이터를 넣기: 2장

◎ 입력된 데이터를 대상으로 바로 통계분석을 실시할 수 있는 경우는 거의 없음

◎ 데이터 전처리(Preprocessing): 입력된 데이터를 분석이 가능하도록
다듬고 변형시키는 작업

◎ 시간이 많이 걸리는 지루한 작업이나 반드시 필요한 작업

◎ 다룰 데이터: 벡터, 행렬, 데이터 프레임

벡터 다루기

1) 새 벡터 변수 만들기

2) 숫자형 벡터와 문자형 벡터 다루기

3) 벡터끼리의 비교

4) 조건에 의한 벡터의 인덱싱

5) 변수 변환하여 활용하기

6) NA값 다루기

1) 새 벡터 변수의 생성

◎ 벡터에 데이터 추가 및 벡터의 결합

◎ 일정한 구조를 갖는 벡터 생성하기

◎ 벡터의 연산

◎ 연산의 순환법칙

벡터에 데이터 추가(스칼라) 및 벡터들의 결합

1) Combine 함수 c ( )

```
> x <- c(1,2,3,4)
> c(x, 5)
[1] 1 2 3 4 5
> y <- c(6,7,8)
> c(x, y)
[1] 1 2 3 4 6 7 8
```

2) 함수 append ( ): 추가되는 스칼라 혹은 벡터의 위치를 조절하여 삽입이 가능

```
> append(x, 5)
[1] 1 2 3 4 5
> append(x, 5, after=2)
[1] 1 2 5 3 4
> append(x, y)
[1] 1 2 3 4 6 7 8
> append(x, y, after=2)
[1] 1 2 6 7 8 3 4
```

일정한 패턴을 갖는 벡터 만들기

1) 콜론(:)으로 생성하기

```
> 1:5
[1] 1 2 3 4 5
> -5:5
[1] -5 -4 -3 -2 -1  0  1  2  3  4  5

> 1.5:6.4
[1] 1.5 2.5 3.5 4.5 5.5
> 7:0
[1] 7 6 5 4 3 2 1 0
```

a:b
- a를 시작점으로 b를 초과하지 않을 때까지 1씩 증가하는 수열
- a > b이면 1씩 감소하는 수열

일정한 패턴을 갖는 벡터 만들기

2) 함수 seq ( )로 수열자료 생성

```
> seq(from=0,to=5)                  # seq(0,5)
[1] 0 1 2 3 4 5

> seq(from=0,to=5,by=2)             # seq(0,5,by=2)
[1] 0 2 4

> seq(from=0,to=5,length=3)         # seq(0,5,len=3)
[1] 0.0 2.5 5.0

> seq(from=0,by=2,length=3)         # seq(0,by=2,len=3)
[1] 0 2 4
```

◎ 한 숫자로만 구성할 경우: 1을 기준으로 1씩 증가 혹은 감소

```
> seq(3)
[1] 1 2 3
> seq(-3)
[1]  1  0 -1 -2 -3
```

일정한 패턴을 갖는 벡터 만들기

◎ 예제: 다음의 수열을 만들어 보자

① 2, 5, 8, 11

② 9, 18, 27, 36 ,45

③ 1, 3, 5, 7, 2, 4, 6, 8

HiNT: ①번과 ②번- seq ( ) 함수와 length를 활용해보자!

③번- seq ( ) 함수와 c ( ) 함수를 활용해보자!

일정한 패턴을 갖는 벡터 만들기

◎ 예제: 다음의 수열을 만들어 보자: 답!

① 2, 5, 8, 11

```
seq(from=2, to=11, by=3)
seq(from=2, to=11, length=4)
```

② 9, 18, 27, 36 ,45

```
seq(from=9, to=45, by=9)
seq(from=9, to=45,length=5)
```

③ 1, 3, 5, 7, 2, 4, 6, 8

```
c (seq(from=1,to=7,by=2),    seq(from=2,to=8,by=2))
```

함수 seq_along ( ) 또는 seq_len( )

1) 함수에 들어가는 벡터와 길이가 같으며 시작을 1로 하며 1씩 증가하는 수열 생성

```
> x <- c(5,15,20,25,30)

> seq(along=x)
[1] 1 2 3 4 5

> seq_along(x)
[1] 1 2 3 4 5

> seq(length=length(x))
[1] 1 2 3 4 5

> seq_len(length(x))
[1] 1 2 3 4 5
```

날짜에 함수 seq ( )

- 동일한 간격의 날짜 생성: 옵션 by에 숫자 지정

```
> s1 <- as.Date(“2019-04-01”)
> e1 <- as.Date(“2019-04-30”)
> seq(from=s1, to=e1, by= 7)
[1] "2019-04-01" "2019-04-08" "2019-04-15"
[4] "2019-04-22" "2019-04-29"
```

- 간격을 주간격 혹은 월간격/ 연 단위로 조절이 가능

```
> seq(from=s1, by= “week”, length=5)
[1] "2019-04-01" "2019-04-08" "2019-04-15"
[4] "2019-04-22" "2019-04-29“
> seq(from=s1, by="month", length=5)
[1] "2019-04-01" "2019-05-01" "2019-06-01"
[4] "2019-07-01" "2019-08-01“
seq(from=s1, by="year", length=5)
[1] "2019-04-01" "2020-04-01" "2021-04-01"
[4] "2022-04-01" "2023-04-01"
```

일정한 패턴을 갖고 있는 함수 rep ( )

- 옵션 times의 활용

```
> rep(1, times=5)
[1] 1 1 1 1 1

> rep(1:3, times=3)
[1] 1 2 3 1 2 3 1 2 3

> rep(c("A","B"), times=c(3,2))
[1] "A" "A" "A" "B" "B"
```

- times에 하나의 숫자 지정: 데이터 전체를 숫자만큼 반복

- times에 벡터 지정: 반복 대상 데이터와 일대일 대응시켜서 반복

일정한 패턴을 갖고 있는 함수 rep ( )

- 옵션 each와 times의 활용

```
> rep(1:3, each=3)    - 데이터 요소가 each번 반복
[1] 1 2 3 1 2 3 1 2 3

> rep(1:3, each=3, times=2)    - 데이터 요소가 each번 반복 전체로 times번 반복
[1] 1 1 1 2 2 2 3 3 3 1 1 1 2 2 2 3 3 3
```

- 옵션 each와 length의 활용

```
> rep(1:3, length =7 )    - 길이가 length가 될 때까지 데이터 전체가 반복
[1] 1 2 3 1 2 3 1
> rep(1:3, each=2, length =7)
[1] 1 1 2 2 3 3 1
```

벡터의 연산

- 벡터와 벡터의 연산은 대응되는 각 구성요소끼리의 연산으로 이루어짐

```
> x <- c(1,2,3,4)
> y <- c(5,6,9,16)

> x+y
[1]  6  8 12 20

> y-x
[1] 4 4 6 12

> y/x
[1] 5 3 3 4

>y^x
[1]     5    36   729 65536
```

벡터의 연산

- 벡터와 스칼라의 연산도 대응되도록 연산됨

```
> x <- c(1,2,3,4)
> y <- c(5,6,9,16)

> x
[1]  1  2 3 4

> x+3
[1] 4 5 6 7

> y/4
[1] 1.25 1.50 2.25 4.00

>2^x
[1]  2  4  8 16
```

벡터의 연산시 나올 수 있는 문자: -Inf, Inf, NaN

```
> c(-1,0,1)/0
[1] -Inf NaN Inf

> sqrt(-1)
[1] NaN Warning message:
In sqrt(-1) : NaNs produced

> Inf-Inf
[1] NaN

> Inf/Inf
[1] NaN
```

벡터의 순환법칙

> c(1,2,3,4,5,6) + c(1,2,3) 이라면?

▣길이가 짧은 c(1,2,3)을 순환 반복시켜 c(1,2,3,1,2,3)을 만들어 길이를 같게 만든 후 연산 수행

▣벡터와 스칼라의 연산도 동일하게 수행됨

▣다양한 함수에서 순환법칙이 적용

벡터의 순환법칙

```
> 1 :4 + 1:3 이라면?
```

▣길이가 짧은 벡터의 배수로 긴 벡터가 나오지 않는 경우???

```
[1] 2 4 6 5
Warning message:
In 1:4 + 1:3 :
longer object length is not a multiple of shorter object
length
```

주요 이력
現) ㈜RTMC 전략기획실장
前) ㈜B사 웹로그분석 및 DP사업 完
前) ㈜H금속사 회계팀
前) ㈜B건설사 회계팀
前) K문고 CRM VIP 군집전략 CRM프로젝트 보조연구원
前) L백화점 CRM Alert 전략 CRM프로젝트 보조연구원

BSL(스위스 로잔 비즈니스 스쿨) MBA
ASSIST 빅데이터경영통계 MBA

국가공인 ADSP(빅데이터 준전문가)
現 **코리아IT아카데미 빅데이터 R 강사**
現 **코리아IT아카데미 빅데이터 기초 파이썬 강사**
現 **코리아IT아카데미 빅데이터 기초통계 전담강사**

[박영식] 완성에 이르기까지