

- R 프로그램 기초 2 -

1. 발전된 자료형

(1) 리스트(List)

- 벡터와 비슷. But 서로 다른 데이터 타입의 원소를 포함할 수 있음. ※C언어의 구조체와 비슷.

① 리스트 만들기: list() 함수

```
x <- list(1,2,"3","good") #리스트 만들기
y <- list(name="chang", age=10, title="manager") #원소에 이름을 붙여서 리스트 만들기
z <- list(element.num=c(1,2,3), element.char=c("a", "b")) #각 원소의 크기가 다를 수 있음.
```

② 리스트 다루기

x <- list(name="chang", age=10, score=c(70,90), grade=c("B","A"))	
x[1]; x[[1]] #[] vs. [[]] indexing 비교 typeof(x[1]); typeof(x[[1]])	x\$name #원소 이름을 통한 indexing typeof(x\$name)
x[2]; x[[2]]	x\$age
x[3]; x[[3]]	x\$age[2] #list의 원소의 원소 접근

(2) 배열 (Array)

① 생성: array(), dim(), dimnames() 함수

(3) 데이터 프레임 (Data Frame)

① 생성: data.frame() 함수

② 원소 조작: attach(), detach() 함수

2. 추가적인 기본 사용법

(1) 제어문

- ① SWITCH() 함수: switch(변수, 변수값1={실행문1}, 변수값2={실행문2}, 변수값3={실행문3}, {실행문})
- 변수값에 해당하는 실행문 실행. 일치하는 변수값이 없는 경우 마지막 실행문을 실행.

#최빈값(mode) 계산 함수 getmode <- function(v) { uniqv <- unique(v) uniqv[which.max(tabulate(match(v, uniqv)))] } #표본 x<-c(seq(1:10),seq(20:40),seq(80:100),30)	#대표값 구하기 hist(x, breaks=30, prob=F) type<-'mean' #'mea' 또는 'med' 또는 'mod' switch(type, mea={mean(x)}, #mean med={median(x)}, #median mod={getmode(x)}, #median {mean(x)})
---	--

② while 문: while(Condition) {Condition이 True일 때의 실행문}

- Condition이 TRUE인 동안 실행문을 반복해서 실행.

<pre>#f(x) = k 풀기: $x_{n+1} = x_n - [f(x_n)-k]/f'(x_n)$ #문제 f <- function(x) x^2 #목적함수 k <- 5 #목적값 #알고리즘 수행을 위한 사전 세팅 f1 <- function(x) 2*x #목적함수의 1차 미분 tol <- 1e-10 #허용오차 x0 <- k/2 #초기값</pre>	<pre>#알고리즘 수행 x <- x0 #초기화 iter <- 0 while(abs(f(x)-k)>tol) { x <- x - (f(x)-k)/f1(x) iter <- iter + 1 } x; x^2; iter</pre>
--	--

③ break 문: 실행 중인 반복문을 중단

<pre>#f(x) = k 풀기: $x_{n+1} = x_n - [f(x_n)-k]/f'(x_n)$ #문제 f <- function(x) x^2 #목적함수 k <- 5 #목적값 #알고리즘 수행을 위한 사전 세팅 f1 <- function(x) 2*x #목적함수의 1차 미분 tol <- 1e-10 #허용오차 x0 <- k/2 #초기값</pre>	<pre>#알고리즘 수행 x <- x0 #초기화 iter <- 0 repeat { #repeat문을 사용 if(abs(f(x)-k)<tol) break x <- x - (f(x)-k)/f1(x) iter <- iter + 1 } x; x^2; iter</pre>
--	---

(2) 확률분포 관련 함수

① 분포별 R 내장함수:

분포	확률 or 확률밀도	누적확률	분위수	난수 생성
Normal 분포	$\text{dnorm}(x, \mu, \sigma)$	$\text{pnorm}(x, \mu, \sigma)$	$\text{qnorm}(p, \mu, \sigma)$	$\text{rnorm}(n, \mu, \sigma)$
Uniform 분포	$\text{dunif}(x, a, b)$	$\text{punif}(x, a, b)$	$\text{qunif}(p, a, b)$	$\text{runif}(n, a, b)$
χ^2 분포	$\text{dchisq}(x, v)$	$\text{pchisq}(x, v)$	$\text{qchisq}(p, v)$	$\text{rchisq}(n, v)$
t 분포	$\text{dt}(x, v)$	$\text{pt}(x, v)$	$\text{qt}(p, v)$	$\text{rt}(n, v)$
F 분포	$\text{df}(x, v_1, v_2)$	$\text{pf}(x, v_1, v_2)$	$\text{qf}(p, v_1, v_2)$	$\text{rf}(n, v_1, v_2)$
Log-Normal 분포	$\text{dlnorm}(x, \mu, \sigma)$	$\text{plnorm}(x, \mu, \sigma)$	$\text{qlnorm}(p, \mu, \sigma)$	$\text{rlnorm}(n, \mu, \sigma)$
Gamma 분포	$\text{dgamma}(x, \alpha, \beta)$	$\text{pgamma}(x, \alpha, \beta)$	$\text{qgamma}(p, \alpha, \beta)$	$\text{rgamma}(n, \alpha, \beta)$
Beta 분포	$\text{dbeta}(x, \alpha, \beta)$	$\text{pbeta}(x, \alpha, \beta)$	$\text{qbeta}(p, \alpha, \beta)$	$\text{rbeta}(n, \alpha, \beta)$

② (예) 몬테카를로 시뮬레이션으로 원주율 계산하기

<pre>set.seed(1) #매 시뮬레이션마다 동일한 난수가 생성되도록 seed number 고정.</pre>	
<pre>l <- 2 #정사각형 한 변의 길이 r <- l/2 #반지름 M <- 10000 #시뮬레이션 횟수 x <- runif(M, -r, r) #U(-r, r) y <- runif(M, -r, r) #U(-r, r)</pre>	<pre>in.prob <- sum(sqrt(x^2+y^2)<=r)/M app.pi <- in.prob*(l^2) app.pi pi #참값</pre>