



서울대학교
SEOUL NATIONAL UNIVERSITY

HW2

- For loop 함수와 apply 함수 비교



과목명 : 대학원 신입생 세미나

학 과 : 통계학과

학 번 : 2023-24403

성 명 : 박혜림

제출일 : 23.03.25

For loop 함수와 lapply 함수, sapply 함수의 계산 속도를 비교하기 위해, 19의 거듭제곱을 1부터 10000까지 계산하였다.

1. for loop 함수를 이용하여 계산 시

```
ftn <- function(i){
  res<-NULL
  for (j in c(1:i)){
    res[j] <- 19^j
  }
  return (res)
}
```

위와 같이 반복문을 작성하여 numeric 형태의 결과값을 얻을 수 있었고, 계산 속도는 다음과 같다.

```
> start_time <- Sys.time()
> ftn <- function(i){
+   res<-NULL
+   for (j in c(1:i)){
+     res[j] <- 19^j
+   }
+   return (res)
+ }
> a=ftn(10000)
> end_time <- Sys.time()
> elapsed_time <- as.numeric(difftime(time1 = end_time,
+                                     time2 = start_time,
+                                     units = "secs"))
> cat("elapsed time : ",sprintf("%.3f",elapsed_time),"sec",sep="")
elapsed time : 0.011sec>
```

2. lapply 함수를 이용하여 계산 시

```
#lapply
start_time <- Sys.time()
b=lapply(c(1:10000),function(i){return(19^i)})
end_time <- Sys.time()
elapsed_time <- as.numeric(difftime(time1 = end_time,
                                     time2 = start_time,
                                     units = "secs"))
cat("elapsed time : ",sprintf("%.3f",elapsed_time),"sec",sep="")
```

lapply 함수를 이용했을 때 list 형태의 결과값을 얻을 수 있었고, 계산 속도는 다음과 같다.

```
> #lapply
> start_time <- Sys.time()
> b=lapply(c(1:10000),function(i){return(19^i)})
> end_time <- Sys.time()
> elapsed_time <- as.numeric(difftime(time1 = end_time,
+                                     time2 = start_time,
+                                     units = "secs"))
> cat("elapsed time : ",sprintf("%.3f",elapsed_time),"sec",sep="")
elapsed time : 0.009sec>
```

3. sapply 함수를 이용하여 계산 시

2번 lapply 함수를 sapply로 바꿔 동일하게 계산했을 때 numeric 형태의 결과값을 얻을 수 있었고, 계산 속도는 다음과 같다.

```
> #saapply
> start_time <- Sys.time()
> c=sapply(c(1:10000),function(i){return(19^i)})
> end_time <- Sys.time()
> elapsed_time <- as.numeric(difftime(time1 = end_time,
+                                     time2 = start_time,
+                                     units = "secs"))
> cat("elapsed time : ",sprintf("%.3f",elapsed_time),"sec",sep="")
elapsed time : 0.007sec
```

4. 결론

계산 효율은 sapply - lapply - for loop 순으로 좋다고 할 수 있다.

다만, 비교한 계산 식이 엄청 복잡하고 오래 걸리는 연산을 다루지 않아서 계산 속도의 차이가 크리티컬하지 않는다.

비교를 위해 작성한 코드를 여러번 반복했을 때 for문에 비해 sapply나 lapply가 항상 효율적이었던 것을 감안하면, 복잡한 데이터나 연산을 다룰 때 for loop를 사용하는 것보다 sapply나 lapply 함수를 사용하는 것이 합리적일 것이라 생각한다.