Assignment2

LingzhouAo

2/24/2018

Basic_R_Exercise_2

1

```
A <- matrix(c(1,1,3,5,2,6,-2,-1,-3), nrow = 3, byrow = TRUE)
A
```

```
##      [,1] [,2] [,3]
## [1,]    1    1    3
## [2,]    5    2    6
## [3,]   -2   -1   -3
```

```
A^3
```

```
##      [,1] [,2] [,3]
## [1,]    1    1   27
## [2,]  125    8  216
## [3,]   -8   -1  -27
```

```
A[,3] <- A[,2] + A[,3]
A
```

```
##      [,1] [,2] [,3]
## [1,]    1    1    4
## [2,]    5    2    8
## [3,]   -2   -1   -4
```

2

```
B <- matrix(c(10,-10,10),nrow = 15, ncol = 3 ,byrow=TRUE)
B
```

```
##       [,1] [,2] [,3]
##  [1,]   10  -10   10
##  [2,]   10  -10   10
##  [3,]   10  -10   10
##  [4,]   10  -10   10
##  [5,]   10  -10   10
##  [6,]   10  -10   10
##  [7,]   10  -10   10
##  [8,]   10  -10   10
##  [9,]   10  -10   10
## [10,]   10  -10   10
## [11,]   10  -10   10
## [12,]   10  -10   10
## [13,]   10  -10   10
## [14,]   10  -10   10
## [15,]   10  -10   10
```

```
crossprod(B,B)
```

```
##       [,1]  [,2]  [,3]
## [1,]  1500 -1500  1500
## [2,] -1500  1500 -1500
## [3,]  1500 -1500  1500
```

3

```
matE <- matrix(rep(0,36), nrow = 6, byrow = TRUE)
row(matE)
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    1    1    1    1    1    1
## [2,]    2    2    2    2    2    2
## [3,]    3    3    3    3    3    3
## [4,]    4    4    4    4    4    4
## [5,]    5    5    5    5    5    5
## [6,]    6    6    6    6    6    6
```

```
col(matE)
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    1    2    3    4    5    6
## [2,]    1    2    3    4    5    6
## [3,]    1    2    3    4    5    6
## [4,]    1    2    3    4    5    6
## [5,]    1    2    3    4    5    6
## [6,]    1    2    3    4    5    6
```

```
row(matE)-col(matE)
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    0   -1   -2   -3   -4   -5
## [2,]    1    0   -1   -2   -3   -4
## [3,]    2    1    0   -1   -2   -3
## [4,]    3    2    1    0   -1   -2
## [5,]    4    3    2    1    0   -1
## [6,]    5    4    3    2    1    0
```

```
matE[abs(row(matE)-col(matE))==1] <- 1
matE
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    0    1    0    0    0    0
## [2,]    1    0    1    0    0    0
## [3,]    0    1    0    1    0    0
## [4,]    0    0    1    0    1    0
## [5,]    0    0    0    1    0    1
## [6,]    0    0    0    0    1    0
```

4

```r
a <- 0:4
A <- outer(a,a,"+")
A
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    0    1    2    3    4
## [2,]    1    2    3    4    5
## [3,]    2    3    4    5    6
## [4,]    3    4    5    6    7
## [5,]    4    5    6    7    8
```

```r
B <- outer(a,a, "*")
B
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    0    0    0    0    0
## [2,]    0    1    2    3    4
## [3,]    0    2    4    6    8
## [4,]    0    3    6    9   12
## [5,]    0    4    8   12   16
```

```r
b <- 5:10
C <- outer(a,b,"+")
C
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    5    6    7    8    9   10
## [2,]    6    7    8    9   10   11
## [3,]    7    8    9   10   11   12
## [4,]    8    9   10   11   12   13
## [5,]    9   10   11   12   13   14
```

```r
D <- outer(b,a, "%%")
D
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]   NA    0    1    2    1
## [2,]   NA    0    0    0    2
## [3,]   NA    0    1    1    3
## [4,]   NA    0    0    2    0
## [5,]   NA    0    1    0    1
## [6,]   NA    0    0    1    2
```

5

```r
a <- 0:4
A <- outer(a,a,"+")%%5
A
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    0    1    2    3    4
## [2,]    1    2    3    4    0
## [3,]    2    3    4    0    1
## [4,]    3    4    0    1    2
## [5,]    4    0    1    2    3
```

```r
b <- 0:9
B <- outer(b,b,"+")%%10
B
```

```
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
##  [1,]    0    1    2    3    4    5    6    7    8     9
##  [2,]    1    2    3    4    5    6    7    8    9     0
##  [3,]    2    3    4    5    6    7    8    9    0     1
##  [4,]    3    4    5    6    7    8    9    0    1     2
##  [5,]    4    5    6    7    8    9    0    1    2     3
##  [6,]    5    6    7    8    9    0    1    2    3     4
##  [7,]    6    7    8    9    0    1    2    3    4     5
##  [8,]    7    8    9    0    1    2    3    4    5     6
##  [9,]    8    9    0    1    2    3    4    5    6     7
## [10,]    9    0    1    2    3    4    5    6    7     8
```

```r
c <- 0:8
C <- outer(c, c, "-")%%9
C
```

```
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
##  [1,]    0    8    7    6    5    4    3    2    1
##  [2,]    1    0    8    7    6    5    4    3    2
##  [3,]    2    1    0    8    7    6    5    4    3
##  [4,]    3    2    1    0    8    7    6    5    4
##  [5,]    4    3    2    1    0    8    7    6    5
##  [6,]    5    4    3    2    1    0    8    7    6
##  [7,]    6    5    4    3    2    1    0    8    7
##  [8,]    7    6    5    4    3    2    1    0    8
##  [9,]    8    7    6    5    4    3    2    1    0
```

6

```r
A <- matrix(c(1,2,3,4,5,2,1,2,3,4,3,2,1,2,3,4,3,2,1,2,5,4,3,2,1),nrow = 5, ncol = 5,byrow=TRUE)
A
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    2    3    4    5
## [2,]    2    1    2    3    4
## [3,]    3    2    1    2    3
## [4,]    4    3    2    1    2
## [5,]    5    4    3    2    1
```

```r
y <- matrix(c(7,-1,-3,5,17),nrow = 5, ncol = 1, byrow=TRUE)
y
```

```
##      [,1]
## [1,]    7
## [2,]   -1
## [3,]   -3
## [4,]    5
## [5,]   17
```

```r
x <- solve(A) %*% y
x
```

```
##      [,1]
## [1,]   -2
## [2,]    3
## [3,]    5
## [4,]    2
## [5,]   -4
```

7

```
set.seed(75)
aMat <- matrix(sample(10, size=60, replace=TRUE), nr=6)
aMat
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
## [1,]    3    6    7    7    2    4    3    7    1     4
## [2,]    1    9    8    7    2    6   10    9    5     2
## [3,]    7   10    8    4   10    5    4    8    4     4
## [4,]    4    3    1    1    3    3    9    7    4     2
## [5,]    1    8    1    9    9    8    1    3    7     7
## [6,]    2    6    7    5    6   10    4    6   10     1
```

```
apply(aMat, 1, function(x){sum(x > 4)})
```

```
## [1] 4 7 6 2 6 7
```

```
apply(aMat, 1, function(x){sum(x>6 & x <8)} == 2)
```

```
## [1] FALSE FALSE FALSE FALSE  TRUE FALSE
```

```
cSums <- colSums(aMat)
which( outer(cSums, cSums, "+") > 75, arr.ind = TRUE)
```

```
##      row col
## [1,]   2   2
## [2,]   6   2
## [3,]   8   2
## [4,]   2   6
## [5,]   8   6
## [6,]   2   8
## [7,]   6   8
## [8,]   8   8
```

8

```
sum((1:20)^4) * sum(1/(3+(1:5)))
```

```
## [1] 639215.3
```

R_Function

1a

```
tmpFn1 <- function(xVec){
  return(xVec^(1:length(xVec)))
}
tmpFn2 <- function(xVec){
  return(xVec^(1:length(xVec))/(1:length(xVec)))
}
```

1b

```
tmpFn3 <- function(x, n){
  return(1+sum(x^(1:n)/(1:n)))
}
```

2

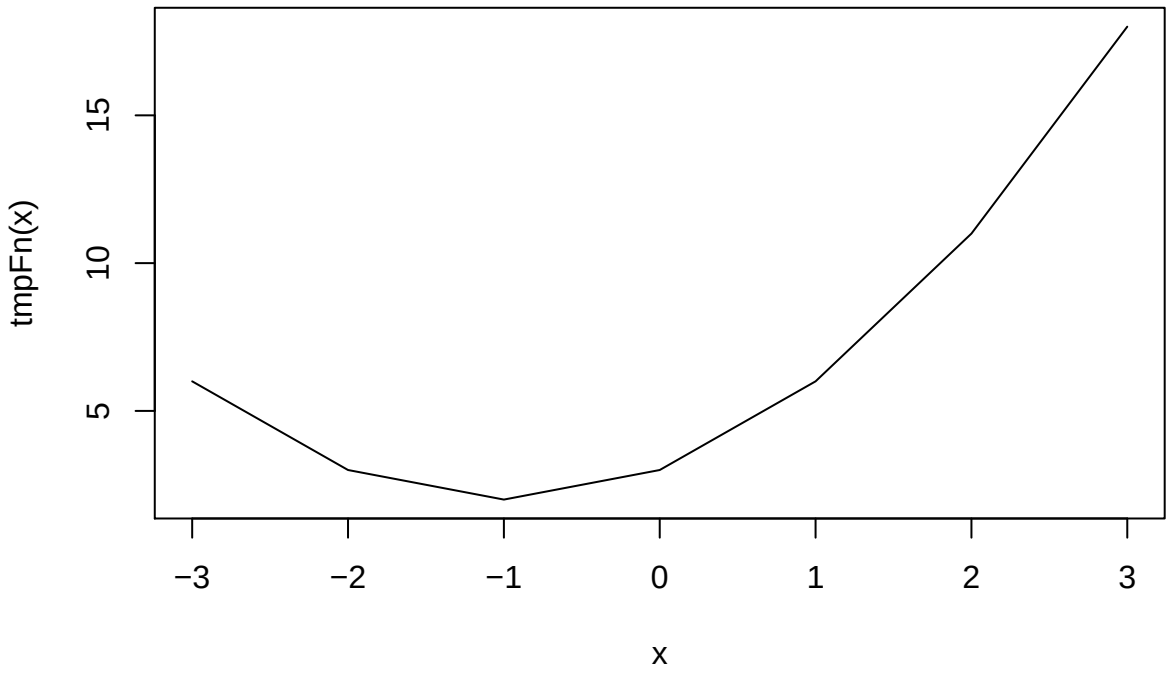
```
tmpFn <- function(xVec){
  return((xVec[-c(length(xVec)-1,length(xVec))] + xVec[-c(1,length(xVec))] + xVec[-c(1,2)])/3)
}

tmpFn(c(1:5,6:1))
```

```
## [1] 2.000000 3.000000 4.000000 5.000000 5.333333 5.000000 4.000000 3.000000
## [9] 2.000000
```

3

```
tmpFn <- function(x){
  if (x < 0){
    return (x^2+2*x+3)
  }
  else if (x<2){
    return (x+3)
  }
  else{
    return (x^2+4*x-7)
  }
}
x <- seq(-3,3)
plot(x, tmpFn(x), type="l")
```

```
## Warning in if (x < 0) {: the condition has length > 1 and only the first
## element will be used
```



###4

```
tmpFn <- function(mat){
  mat[mat%%2 == 1] <- 2 * mat[mat%%2 == 1]
  return(mat)
}
```

5

```
tmpFn <- function(n,k){
  tmp <- diag(k, nr = n)
  tmp[abs(row(tmp) - col(tmp)) == 1] <- 1
  return(tmp)
}
```

6

```
quadrant <- function(alpha){
  return(1 + (alpha%%360)%/%90)
}
```

7

```
weekday <- function(day,month,year){
  if ((month - 2) > 0){
    month <- month - 2
  }
  else{
    month <- month + 10
    year <- year - 1
```

```
  }
  cc <- year %/% 100
  year <- year %% 100
  tmp <- floor(2.6*month - 0.2) + day + year + year %/% 4 + cc %/% 4 - 2 * cc
  return(c("Sunday","Monday","Tuesday","Wednesday","Thursday","Friday","Saturday")[1+tmp%%7])
}
```

8a

```
testLoop <- function(n){
  xVec <- rep(NA,n-1)
  for (i in 1:(n-1)){
    if (i == 1){
      xVec[i] = 1
    }
    else if (i == 2){
      xVec[i] == 2
    }
    else{
      xVec[i] = xVec[i-1] + (2 / xVec[i])
    }
  }
  return(xVec)
}
```

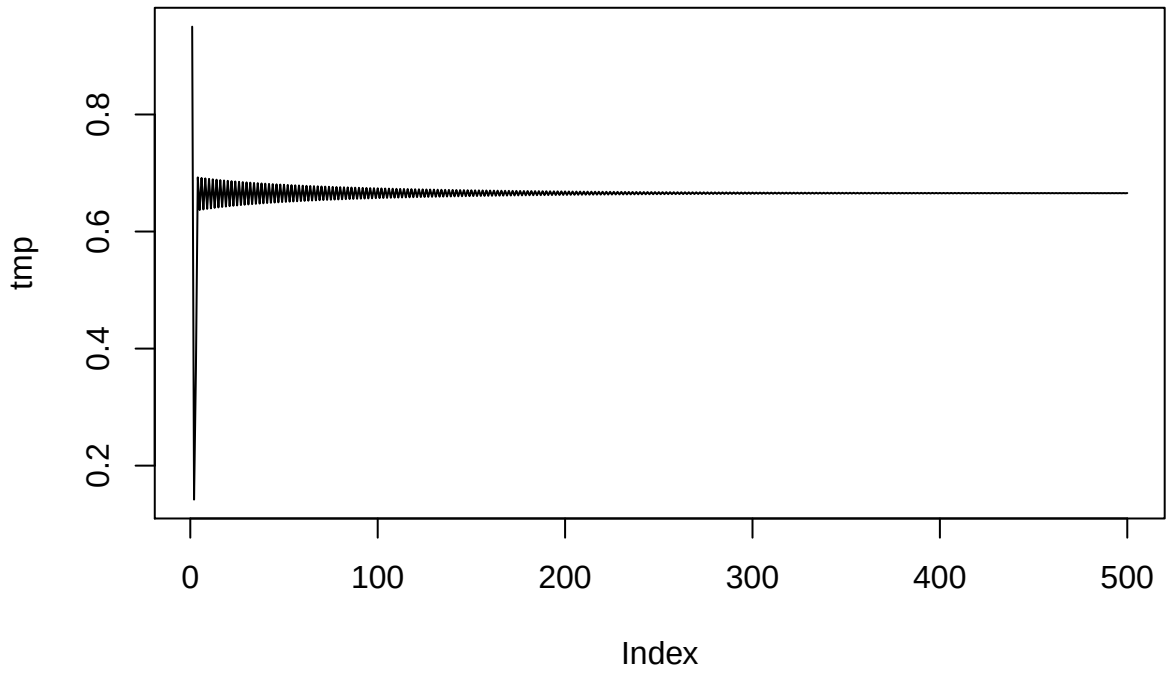
8b

```
testLoop2 <- function(yVec){
  return(sum(exp(seq(along=yVec))))
}
testLoop2(1:3)
```

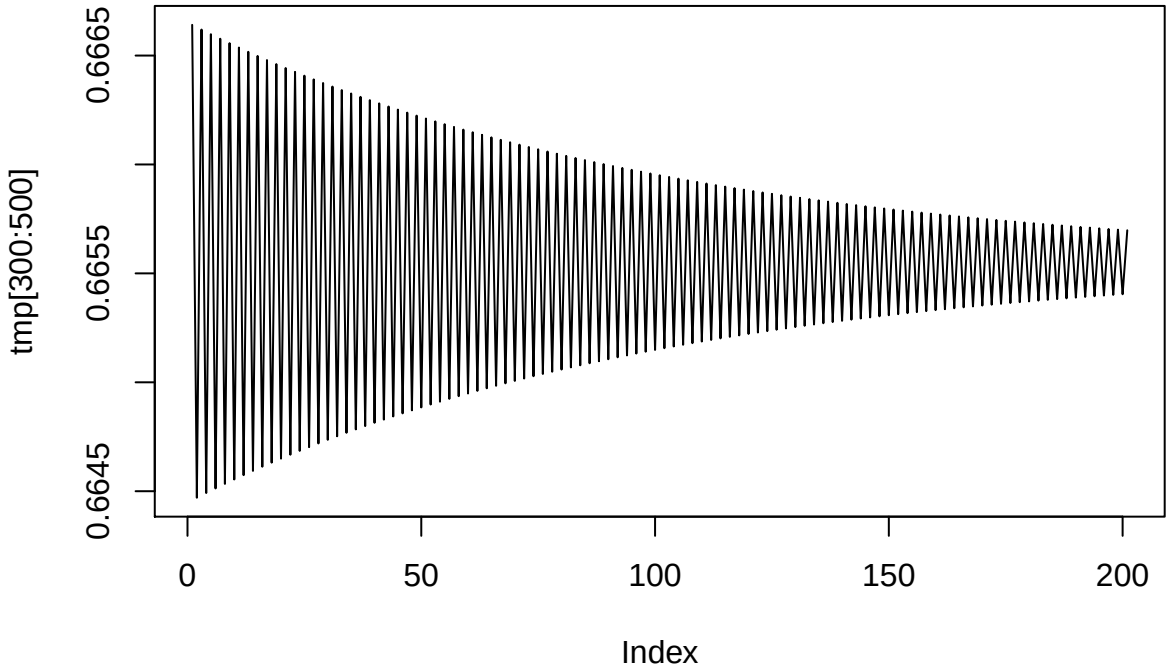
```
## [1] 30.19287
```

9a

```
quadmap <- function(start, rho, niter){
  xVec <- rep(NA,niter)
  xVec[1] <- start
  for (i in 1:(niter-1)){
    xVec[i+1] <- rho*xVec[i]*(1 - xVec[i])
  }
  return(xVec)
}

tmp <- quadmap(start=0.95, rho=2.99, niter=500)
plot(tmp, type="l")
```




```
plot(tmp[300:500], type="l")
```



###9b

```
quadmap2 <- function(start, rho){
  n <- 0
  pre <- start
  while(TRUE){
    now <- rho*pre*(1 - pre)
    n = n + 1
    if (abs(now - pre) < 0.02){
      break
```

```
    }
    pre <- now
  }
  return(n)
}
tmp2 <- quadmap2(start=0.95, rho=2.99)
tmp2
```

```
## [1] 84
```

10a

```
tmpFn <- function(xVec){
  new_xVec <- xVec - mean(xVec)
  d <- sum(new_xVec^2)
  n <- length(xVec)
  r1 <- sum(new_xVec[2:n] * new_xVec[1:(n-1)])/d
  r2 <- sum(new_xVec[3:n] * new_xVec[1:(n-2)])/d
  return(c(r1,r2))
}
tmpFn(seq(2,56,by=3))
```

```
## [1] 0.8421053 0.6859649
```

10b

```
tmpFn2 <- function(xVec, k){
  new_xVec <- xVec - mean(xVec)
  d <- sum(new_xVec^2)
  n <- length(xVec)
  temp <- function(j){
    sum(new_xVec[(j+1):n] * new_xVec[1:(n-j)] )/d
  }
  return(c(1, sapply(1:k, temp)))
}
```