

## 問題

R-tips P.53 の `matpow` 関数を用いて計算した正方行列  $A$  の  $n$  乗の値が、行列の  $n$  乗を  $A \% \% A \% \% \dots \% \% A$  と計算した結果と一致するかを確認せよ。一致しない場合には、その原因がどこにあるか考察せよ。

## matpow 関数

```
> matpow <- function(x, pow=2) {  
+   y <- eigen(x)  
+   y$vectors %%% diag( (y$values)^pow ) %%% t(y$vectors)  
+ }
```

## 実行結果

### 対角行列の場合

```
> A <- diag(1:2)  
> matpow(A, 3)  
      [,1] [,2]  
[1,]    1    0  
[2,]    0    8  
> A %%% A %%% A  
      [,1] [,2]  
[1,]    1    0  
[2,]    0    8
```

R-tips に掲載されている通り、 $A$  が対角行列の場合、結果は一致する。

### 対角行列でない場合

```
> A <- matrix(c(8,4,1,5), 2,2)  
> matpow(A, 3)  
      [,1] [,2]  
[1,] 368.2647 349.4412  
[2,] 349.4412 424.7353  
> A %%% A %%% A  
      [,1] [,2]  
[1,]  596  133  
[2,]  532  197
```

結果は一致しなかった。これは、 $A$  の固有値分解の式が正しくないからである。R-tips では、 $A$  の固有値分解の式は  $A = VD^tV$  とされている。これが正しいければ、`matpow(A, 1)` は  $A$  と一致するはずである。しかし、以下のように `matpow(A, 1)` は  $A$  と一致しない。したがって、固有値分解の式が原因であると考えられる。

```
> A <- matrix(c(8,4,1,5), 2,2)  
> matpow(A, 1)  
      [,1] [,2]  
[1,] 4.735294 3.558824  
[2,] 3.558824 8.264706
```

## 修正 matpow 関数

### 修正内容

正しい固有値分解の式は、 $A = VDV^{-1}$  である。したがって、matpow 関数を以下のように修正することで、結果が一致するようになる。

```
> matpow = function(x, pow=2) {  
+   y=eigen(x)  
+   y$vectors %*% diag( (y$values)^pow ) %*% solve(y$vectors)  
+ }
```

### 実行結果

```
> matpow(A,1)  
      [,1] [,2]  
[1,]    8    1  
[2,]    4    5  
> A  
      [,1] [,2]  
[1,]    8    1  
[2,]    4    5  
>  
> matpow(A, 3)  
      [,1] [,2]  
[1,]   596   133  
[2,]   532   197  
> A %*% A %*% A  
      [,1] [,2]  
[1,]   596   133  
[2,]   532   197
```