Module 2 R Activity

Zahlen Zbinden

1. Create the vector [1 4 9 12] using the c() function, and name it newVec1

```
newVec1 <- c(1, 4, 9, 12)
```

2. Create another object:

```
newVec2 <- (1:4)^2

newVec1 == newVec2
```

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

3. Use the cbind() command to creat the following matrix and name it newMat1

```
newMat1 <- matrix(c(1, 4, 2, 0, 7, -3, 5, 2, 1), ncol = 3)
```

4. Use the rbind command to create the following matrix

```
newMat2 <- rbind(c(1, 1, 2), c(4, 7, 2), c(5, -3, -1))
```

5. Companre newMat1 and newMat2 to identify which positions are not equal

```
newMat1 == newMat2
```

```
      [,1]  [,2]  [,3]
[1,]  TRUE FALSE FALSE
[2,]  TRUE  TRUE  TRUE
[3,] FALSE  TRUE FALSE
```

6. Use the matrix() command to create the following matrix and name it newMat3

```r
newMat3 <- matrix(c(1, 4, 2, 5, 3, 6), byrow = T, nrow = 3)
```

7. Compare newMat1 to the transpose of newMat2 to identify which positions are not equal

```r
newMat1 == newMat2
```

```
      [,1]  [,2]  [,3]
[1,]  TRUE FALSE FALSE
[2,]  TRUE  TRUE  TRUE
[3,] FALSE  TRUE FALSE
```

8. How could you use R to check whether a matrix is symmetric? Give code for checking whether newMat2 is symetric

```r
symetric <- function(matrix) {
    return (all(t(matrix) == matrix))
}
```

9. Create a diagonal matrix named diagMat1 with diagonal entries equal to the diagonal enteries of newMat1

```r
diagMat1 <- diag(newMat1)
```

10. Create the 5x5 identity matrix and name it ident5

```r
ident5 <- diag(5)
```

11. Calculate the sum of newMat1 and newMat2

```r
newMat1 + newMat2
```

```
     [,1] [,2] [,3]
[1,]    2    1    7
[2,]    8   14    4
[3,]    7   -6    0
```

12. Multiply newMat3 by 7

```r
newMat3 * 7
```

```
     [,1] [,2]
[1,]    7   28
[2,]   14   35
[3,]   21   42
```

13. Create a 3x3 identity matrix, and name it ident3. Perform matrix multiplication to find the product of ident3 and newmat3

```r
ident3 <- diag(3)

ident3 %*% newMat3
```

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

14. What happens when you try to perform the matrix multiplication to find the produt of newMat3 and ident3. "Error in …: non-comformble arguments". This is because newMat3 is a 3X2 matrix and, ident3 is a 3X3 matrix, the dim of each matrix makes it impossible for the multiplication to happen in this form newMat3 * ident3.

15. Find the matrix inverse of newMat1, name this matrix newMat1.inv and report its value

```r
newMat1.inv <- solve(newMat1)
newMat1
```

```
     [,1] [,2] [,3]
[1,]    1    0    5
[2,]    4    7    2
[3,]    2   -3    1
```

16. Demonstrate that the matrix prodcut of newMat1 and newMat1.inv is the 3X3 identity matrix, no matter which order you perfrom the multiplication. Because of "computer" there multiplcations don't end up perfect, therefor it doesn't recognize new inverse matrix acquired from solve() to be a true inverse, although the matrix that it presents are very close. if you round the result to the nearest integer than you get the true identity matrix

```r
all(ident3 == round(newMat1 %*% newMat1.inv))
```

```
[1] TRUE
```

```r
all(ident3 == round(newMat1.inv %*% newMat1))
```

```
[1] TRUE
```

17. What happens if you try to find the determinant of a matrix that is not square? Demonstrate an example. “Error in determinant.matrix…:‘x’ must be a square matrix

```r
tryCatch(
    {
        det(newMat3)
    },
    error = function(e) {
        print(e)
    }
)
```

```
<simpleError in determinant.matrix(x, logarithm = TRUE, ...): 'x' must be a square matrix>
```

18. What happens if you try to find the trace of a matrix that is not square? Demonstrate an example. It returns a number.

```r
sum(diag(newMat3))
```

```
[1] 6
```

19. Calculate the detminant of newMat2

```r
det(newMat2)
```

```
[1] -81
```

20. Calculate the trace of newMat2

```r
sum(diag(newMat2))
```

```
[1] 7
```

21. Create the following matrix and name it newMat2. Find the eigen decomposition of this matrix. What is the eigenvector corresponding to the smallest eigenvalue? The smalled eigenvalue is 0.5 and the vector corresponding to it is:

$$\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

```r
newMat2 <- matrix(c(6.8, 2.4, 0, 2.4, 8.2, 0, 0, 0, .5), ncol = 3)
eigen(newMat2)
```

```
eigen() decomposition
$values
[1] 10.0  5.0  0.5

$vectors
     [,1] [,2] [,3]
[1,]  0.6 -0.8    0
[2,]  0.8  0.6    0
[3,]  0.0  0.0    1
```