

Operace matic v programu R

R je open-source statistický programovací balíček, který obsahuje mnohé operátory vektorů a matic.

Matic

```
# maticová funkce
# R zadává data vložená po sloupcích začínajících sloupcem jedna
# 1. arg: c(2,3,-2,1,2,2) hodnoty prvků vyplňující sloupcy
# 2. arg: 3 počet řádků (rows)
# 3. arg: 2 počet sloupců (columns)

A <- matrix(c(2,3,-2,1,2,2),3,2)
A <- matrix(c(2,3,-2,1,2,2), nrow=3, ncol=2)
```

```
A
     [,1] [,2]
[1,]    2    1
[2,]    3    2
[3,]   -2    2
```

Je to matice?

```
is.matrix(A)
```

```
[1] TRUE
```

```
is.vector(A)
```

```
[1] FALSE
```

Násobení skalárem

```
c <- 3
c*A
     [,1] [,2]
[1,]    6    3
[2,]    9    6
[3,]   -6    6
```

Sčítání a odečítání matic

```
B <- matrix(c(1,4,-2,1,2,1),3,2)
```

```
B
     [,1] [,2]
[1,]    1    1
[2,]    4    2
[3,]   -2    1
```

```
C <- A + B
```

```
C
     [,1] [,2]
[1,]    3    2
[2,]    7    4
[3,]   -4    3
```

```
D <- A - B
```

```
D
      [,1] [,2]
[1,]    1    0
[2,]   -1    0
[3,]    0    1
```

Násobení matic %*%

```
D <- matrix(c(2,-2,1,2,3,1),2,3)
```

```
D
      [,1] [,2] [,3]
[1,]    2    1    3
[2,]   -2    2    1
```

```
C <- D %*% A
```

```
C
      [,1] [,2]
[1,]    1   10
[2,]    0    4
```

```
C <- A %*% D
```

```
C
      [,1] [,2] [,3]
[1,]    2    4    7
[2,]    2    7   11
[3,]   -8    2   -4
```

```
D <- matrix(c(2,1,3),1,3)
```

```
D
      [,1] [,2] [,3]
[1,]    2    1    3
```

```
C <- D %*% A
```

```
C
      [,1] [,2]
[1,]    1   10
```

```
C <- A %*% D
```

Chyba v A %*% D : nekonformní k násobení

Transpozice matic

```
AT <- t(A)
```

```
AT
      [,1] [,2] [,3]
[1,]    2    3   -2
[2,]    1    2    2
```

```
ATT <- t(AT)
```

```
>ATT
      [,1] [,2]
[1,]    2    1
[2,]    3    2
[3,]   -2    2
```

Jedničkový vektor

```
U <- matrix(1,3,1)
```

```
U
      [,1]
[1,]    1
[2,]    1
[3,]    1
```

Jedničková matice

```
U <- matrix(1,3,2)
U
      [,1] [,2]
[1,]    1    1
[2,]    1    1
[3,]    1    1
```

Diagonální matice

```
S <- matrix(c(2,3,-2,1,2,2,4,2,3),3,3)
S
      [,1] [,2] [,3]
[1,]    2    1    4
[2,]    3    2    2
[3,]   -2    2    3
```

```
D <- diag(S)
D
[1] 2 2 3
```

```
D <- diag(diag(S))
D
      [,1] [,2] [,3]
[1,]    2    0    0
[2,]    0    2    0
[3,]    0    0    3
```

Jednotková (identická) matice

```
I <- diag(c(1,1,1))
```

```
id = function(n)
diag(c(1),nrow=n,ncol=n)
I = id(3)
```

```
I
      [,1] [,2] [,3]
[1,]    1    0    0
[2,]    0    1    0
[3,]    0    0    1
```

```
id = function(n) diag(c(1),nrow=n,ncol=n)
I3 = id(3)
```

Symetrická matice

```
C <- matrix(c(2,1,5,1,3,4,5,4,-2),3,3)
C
      [,1] [,2] [,3]
[1,]    2    1    5
[2,]    1    3    4
[3,]    5    4   -2
```

```
CT <- t(C)
CT
      [,1] [,2] [,3]
[1,]    2    1    5
[2,]    1    3    4
[3,]    5    4   -2
```

Inverze matice

```
A <- matrix(c(4,4,-2,2,6,2,2,8,4),3,3)
```

```
A
      [,1] [,2] [,3]
[1,]    4    2    2
[2,]    4    6    8
[3,]   -2    2    4
```

```
AI <- solve(A)
```

```
AI
      [,1] [,2] [,3]
[1,]  1.0 -0.5  0.5
[2,] -4.0  2.5 -3.0
[3,]  2.5 -1.5  2.0
```

```
A %*% AI
```

```
      [,1] [,2] [,3]
[1,]    1    0    0
[2,]    0    1    0
[3,]    0    0    1
```

```
AI %*% A
```

```
      [,1] [,2] [,3]
[1,]    1    0    0
[2,]    0    1    0
[3,]    0    0    1
```

Inverze & determinant matice

```
C <- matrix(c(2,1,6,1,3,4,6,4,-2),3,3)
```

```
C
      [,1] [,2] [,3]
[1,]    2    1    6
[2,]    1    3    4
[3,]    6    4   -2
```

```
CI <- solve(C)
```

```
CI
      [,1] [,2] [,3]
[1,]  0.2156863 -0.25490196  0.13725490
[2,] -0.2549020  0.39215686  0.01960784
[3,]  0.1372549  0.01960784 -0.04901961
```

```
d <- det(C)
```

```
d
[1] -102
```

Počet řádků & sloupců

```
X <- matrix(c(3,2,4,3,2,-2,6,1),4,2)
```

```
X
      [,1] [,2]
[1,]    3    2
[2,]    2   -2
[3,]    4    6
[4,]    3    1
```

```
dim(X)
[1] 4 2
```

```
r <- nrow(X)
r
[1] 4
```

```
c <- ncol(X)
c
[1] 2
```

Stopa matice

```
trA = sum(diag(A))
```

Výběr řádku

```
H[1, ]      * výběr 1. řádku
[1] 5 2 1
```

Výběr sloupce

```
H[, 2]      * výběr 2. sloupce
[1] 1 1 2 2 6
```

Výpočet součtů sloupců a řádků

```
# pozor na velké písmeno S
```

```
A <- matrix(c(2,3,-2,1,2,2),3,2)
A
```

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

```
c <- colSums(A)
c
[1] 3 5
```

```
r <- rowSums(A)
r
[1] 3 5 0
```

```
a <- sum(A)
a
[1] 8
```

Výpočet průměrů sloupců a řádků

```
# pozor na velké písmeno M
```

```
cm <- colMeans(A)
cm
[1] 1.000000 1.666667
```

```
rm <- rowMeans(A)
rm
[1] 1.5 2.5 0.0
```

```
m <- mean(A)
m
[1] 1.333333
```

Výběr submatice z matice

A[, 1:3] * výběr 1. až 3. sloupce

A[2:3,] * výběr 2. až 3. řádku

Horizontální spojování matic

```
A
      [,1] [,2]
[1,]    2    1
[2,]    3    2
[3,]   -2    2
```

```
B <- matrix(c(1,3,2,1,4,2),3,2)
B
```

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

```
C <- cbind(A,B)
C
```

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

Vertikální spojování matic

```
C <- rbind(A,B)
C
```

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

2. odmocnina

```
sqrt(b)
```

Řešení soustavy rovnic

$$3x_1 + 4x_2 = 4$$

$$x_1 + 6x_2 = 2$$

V maticovém zápisu: $\mathbf{Ax} = \mathbf{y}$, kde

$$\mathbf{A} = \begin{bmatrix} 3 & 4 \\ 1 & 6 \end{bmatrix} \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} 4 \\ 2 \end{bmatrix}$$

Pak $\mathbf{x} = \mathbf{A}^{-1}\mathbf{y}$, nebo v R

```
A <- matrix(c(3, 1, 4, 6), nrow=2)
y <- matrix(c(4, 2), nrow=2)
x <- solve(A) %*% y
x
```

Maticové příkazy

V následujících příkladech jsou \mathbf{A} a \mathbf{B} matice a \mathbf{x} a \mathbf{b} jsou vektory.

Operátor nebo funkce Popis

$\mathbf{A} * \mathbf{B}$	Násobení prvku prvkem
$\mathbf{A} \%*\% \mathbf{B}$	Násobení matic
$\mathbf{A} \%o\% \mathbf{B}$	Outer product. \mathbf{AB}' (Kronekerův součin; $\mathbf{u} \otimes \mathbf{v}$)
<code>crossprod(A, B)</code>	$\mathbf{A}'\mathbf{B}$ a $\mathbf{A}'\mathbf{A}$
<code>crossprod(A)</code>	
<code>t(A)</code>	Transpozice
<code>diag(x)</code>	Vytvoření diagonální matice s prvky \mathbf{x} na hlavní diagonále.
<code>diag(A)</code>	Vrátí vektor obsahující prvky hlavní diagonály.
<code>diag(k)</code>	Jestliže k je skalár, vytvoří $k \times k$ matici identity.
<code>solve(A, b)</code>	Vrátí vektor \mathbf{x} z rovnice $\mathbf{b} = \mathbf{Ax}$ (tj. $\mathbf{A}^{-1}\mathbf{b}$)
<code>solve(A)</code>	Inverze matice \mathbf{A} (\mathbf{A} je čtvercová matice)
<code>ginv(A)</code>	Moore-Penrose zobecněná inverze matice \mathbf{A} . <code>ginv(A)</code> vyžaduje nahrání MASS balíčku.
<code>cbind(A, B, ...)</code>	Kombinuje matice (vektory) horizontálně.
<code>rbind(A, B, ...)</code>	Kombinuje matice (vektory) vertikálně.
<code>rowMeans(A)</code>	Vrátí vektor průměrů řádků.
<code>rowSums(A)</code>	Vrátí vektor součtů řádků.
<code>colMeans(A)</code>	Vrátí vektor průměrů sloupců.
<code>colSums(A)</code>	Vrátí vektor součtů sloupců.

y <- **eigen(A)** **y\$val** jsou vlastní hodnoty (eigenvalues) **A**
y\$vec jsou vlastní vektory (eigenvectors) **A**
Single value decomposition of **A**.

y <- **svd(A)** **y\$d** = vector containing the singular values of **A**
y\$u = matrix with columns contain the left singular vectors of **A**
y\$v = matrix with columns contain the right singular vectors of **A**

R <- **chol(A)** Choleski factorization of **A**. Returns the upper triangular factor, such that **R'R** = **A**.
QR decomposition of **A**.
y\$qr has an upper triangle that contains the decomposition and a lower triangle that contains information on the Q decomposition.

y <- **qr(A)** **y\$rank** is the rank of **A**.
y\$ql a vector which contains additional information on Q.
y\$pivot contains information on the pivoting strategy used.