

## Logiske matriser – Zero-one matrices

En logisk matrise er en matrise der elementene er 0 (usann) eller 1 (sann). Dette kalles også for en boolsk matrise.

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

2x3

3x2

3x3

Vi bruker de logiske operatorene  $\neg$  (ikke),  $\vee$  (eller) og  $\wedge$  (og):

1)  $\neg 0 = 1$

$\neg 1 = 0$

2)  $1 \wedge 1 = 1$

$1 \wedge 0 = 0$

$0 \wedge 1 = 0$

$0 \wedge 0 = 0$

3)  $1 \vee 1 = 1$

$1 \vee 0 = 1$

$0 \vee 1 = 1$

$0 \vee 0 = 0$

### Logisk $\vee$ (eller) mellom to logiske matriser (eng. join)

La A og B være to logiske  $m \times n$ -matriser (NB! begge har samme dimensjon.) Matrisen  $A \vee B$  er den logiske matrisen vi får ved å ta parvis  $\vee$  (eller) mellom elementene i A og B. Hvis  $a_{i,j}$  er element i A og  $b_{i,j}$  er element i B så vil elementet  $(a_{i,j} \vee b_{i,j})$  være element i matrisen  $A \vee B$ . Dvs. elementet på plass  $\underline{i,j}$  i  $A \vee B$  er  $(a_{i,j} \vee b_{i,j})$ .

#### Eksempel

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

$$A \vee B = \begin{bmatrix} 1 \vee 1 & 0 \vee 1 & 1 \vee 1 \\ 0 \vee 0 & 1 \vee 0 & 0 \vee 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

### Logisk $\wedge$ (og) mellom to logiske matriser (eng. meet)

La A og B være to logiske  $m \times n$ -matriser (dvs. begge har samme dimensjon.) Matrisen  $A \wedge B$  er den logiske matrisen vi får ved å ta parvis  $\wedge$  (og) mellom elementene i A og B.

Hvis  $a_{i,j}$  er element i A og  $b_{i,j}$  er element i B så vil elementet  $(a_{i,j} \wedge b_{i,j})$  være element i matrisen  $A \wedge B$ . Dvs. elementet på plass  $\underline{i,j}$  i  $A \wedge B$  er  $(a_{i,j} \wedge b_{i,j})$ .

Eksempel

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

$$A \wedge B = \begin{bmatrix} 1 \wedge 1 & 0 \wedge 1 & 1 \wedge 1 \\ 0 \wedge 0 & 1 \wedge 0 & 0 \wedge 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

**Logisk matrise-multiplikasjon**

La A og B være to logiske matriser. Hvis antallet kolonner i A er lik antallet rader i B kan vi danne det logiske produktet  $A \odot B$ . Det betyr at hvis A er en  $m \times n$ -matrise og B er en  $n \times k$  matriser blir det logiske produktet  $A \odot B$  en  $m \times k$ -matrise.

Hvis  $a_{i,j}$  er element i A og  $b_{i,j}$  er element i B så er elementet på plass  $\underline{i,j}$  i  $A \odot B$  gitt ved:

$$(a_{i,1} \wedge b_{1,j}) \vee (a_{i,2} \wedge b_{2,j}) \vee \dots \vee (a_{i,n} \wedge b_{n,j})$$

Elementet på plass  $\underline{i,j}$  i  $A \odot B$  er den «logiske multiplikasjonen» av rad i i A og kolonne j i B.

Figur: Rad i i A «ganges med» kolonne j i B:

$$A: m \times n \quad B_j \quad B: n \times k$$

$$A_i [a_{i,1}, a_{i,2}, \dots, a_{i,n}] \odot \begin{bmatrix} b_{1,j} \\ b_{2,j} \\ \vdots \\ b_{n,j} \end{bmatrix} =$$

$$(a_{i,1} \wedge b_{1,j}) \vee (a_{i,2} \wedge b_{2,j}) \vee \dots \vee (a_{i,n} \wedge b_{n,j})$$

Eksempel med bruk av multiplikasjonsskjema

$$A = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$$

				1	0
				0	1
				1	0
A	0	1	1	x	y
	1	0	1	z	u

$$\begin{aligned} x &= (0 \wedge 1) \vee (1 \wedge 0) \vee (1 \wedge 1) = 1 \\ y &= (0 \wedge 0) \vee (1 \wedge 1) \vee \dots = 1 \\ z &= (1 \wedge 1) \vee \dots = 1 \\ u &= (1 \wedge 0) \vee (0 \wedge 1) \vee (1 \wedge 0) = 0 \end{aligned}$$

$$A \odot B = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

Her er også  $B \odot A$  definert siden antall kolonner i B er lik antall rader i A:

				0	1	1
				1	0	1
B	1	0	x	y	z	
	0	1				
	1	0				

$$\begin{aligned} x &= (1 \wedge 0) \vee (0 \wedge 1) = 0 \\ y &= (1 \wedge 1) \vee \dots = 1 \\ z &= (1 \wedge 1) \vee \dots = 1 \end{aligned}$$

OSU.

$$B \odot A = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

Legg merke til at  $B \odot A \neq A \odot B$ .

### Kvadratiske logiske matriser

En kvadratisk logisk matrise kan (logisk) multipliseres med seg selv. Vi skriver da

$$A^{[2]} = A \odot A$$

$$A^{[3]} = A \odot A \odot A$$

osv. Vi har videre at

$$A^{[1]} = A$$

$$A^{[0]} = I \quad (\text{identitetsmatrisen med samme dimensjon som } A.)$$

Operasjonene implementert i Java.

```
public static int[][] matriseKonjunksjon( int[][] A, int[][] B)
{
    //undersøke om A har samme dimensjon
    if( A.length != B.length || A[0].length != B[0].length )
        return null;

    // A og B har samme dimensjon
    int[][] AogB = new int[A.length][A[0].length];

    for( int rad = 0; rad < A.length; rad++)
    {
        for( int kolonne = 0; kolonne < A[0].length; kolonne++)
        {
            AogB[rad][kolonne] = 0;
            if( A[rad][kolonne] == 1 && B[rad][kolonne]== 1)
                AogB[rad][kolonne] = 1;
        }
    }
    return AogB;
}
```

```
public class LogiskMatriseRegning
{
    public static int[][] matriseDisjunksjon( int[][] A, int[][] B)
    {
        //undersøke om A har samme dimensjon
        if( A.length != B.length || A[0].length != B[0].length )
            return null;

        // A og B har samme dimensjon
        int[][] AellerB = new int[A.length][A[0].length];

        for( int rad = 0; rad < A.length; rad++)
        {
            for( int kolonne = 0; kolonne < A[0].length; kolonne++)
            {
                AellerB[rad][kolonne] = 1;
                if( A[rad][kolonne] == 0 && B[rad][kolonne]== 0)
                    AellerB[rad][kolonne] = 0;
            }
        }
        return AellerB;
    }
}
```

```

public static int[][] logiskMatriseProdukt( int[][] A, int[][] B)
{
    // Undersøker om antall kolonner i A er lik antall rader B.
    if( A[0].length != B.length )
        return null;

    // Antall kolonner i A er lik antall rader B.
    int[][] AxB = new int[A.length][B[0].length];

    for( int rad = 0; rad < A.length; rad++ )
    {
        for( int kolonne = 0; kolonne < B[0].length; kolonne++ )
        {
            for(int i = 0; i < A[0].length && i < B.length; i++ )
            {
                if( A[rad][i] == 1 && B[i][kolonne] == 1)
                    AxB[rad][kolonne] = 1;
            }
        }
    }
    return AxB;
}

```

Se resten av koden:

<https://www.cs.oslomet.no/~evav/DM/Java17/LogiskMatriseRegning.java>

<https://www.cs.oslomet.no/~evav/DM/Java17/LogiskMatriseRegning.txt>