

Matematika 3

Minkowski, Hamming, and Chebyshev



Name

Dicha Zelianivan Arkana

NIM

2241720002

Class

1i

Department

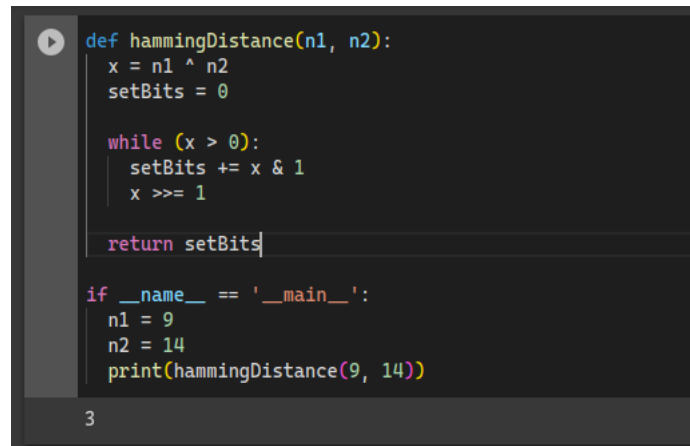
Information Technology

Study Program

D4 Informatics Engineering

1 Hamming Distance

1.1 Task 1



```
def hammingDistance(n1, n2):
    x = n1 ^ n2
    setBits = 0

    while (x > 0):
        setBits += x & 1
        x >>= 1

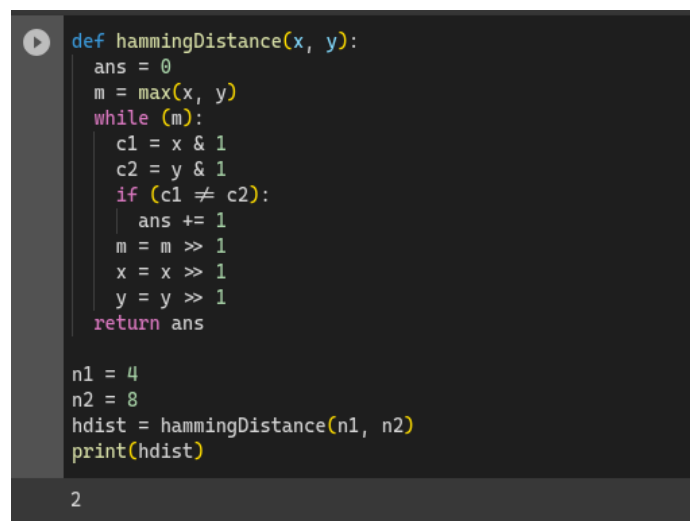
    return setBits

if __name__ == '__main__':
    n1 = 9
    n2 = 14
    print(hammingDistance(9, 14))
```

3

Figure 1: The Hamming Distance implementation in Python

1.2 Task 2



```
def hammingDistance(x, y):
    ans = 0
    m = max(x, y)
    while (m):
        c1 = x & 1
        c2 = y & 1
        if (c1 != c2):
            ans += 1
        m = m >> 1
        x = x >> 1
        y = y >> 1
    return ans

n1 = 4
n2 = 8
hdist = hammingDistance(n1, n2)
print(hdist)
```

2

Figure 2: The Hamming Distance alternative implementation in Python

1.3 Extra Task

1. • BEEN and BEAN

B	E	E	N
B	E	A	N
0	0	1	0

$$d = 0 + 0 + 1 + 0$$

$$d = 1$$

- CEREAL and SERIAL

C	E	R	E	A	L
S	E	R	I	A	L
1	0	0	1	0	0

$$d = 1 + 0 + 0 + 1 + 0 + 0$$

$$d = 2$$

- 10 and 15 in binary

1	0	1	0
1	1	1	1
0	1	0	1

$$d = 0 + 1 + 0 + 1$$

$$d = 2$$

- 6 and 11 in binary

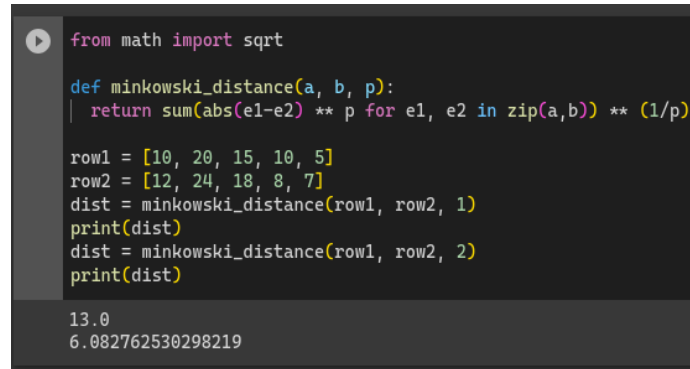
0	1	1	0
1	1	0	1
1	0	1	1

$$d = 1 + 0 + 1 + 1$$

$$d = 3$$

2 Minkowski

2.1 Task 3



```
from math import sqrt

def minkowski_distance(a, b, p):
    return sum(abs(e1-e2) ** p for e1, e2 in zip(a,b)) ** (1/p)

row1 = [10, 20, 15, 10, 5]
row2 = [12, 24, 18, 8, 7]
dist = minkowski_distance(row1, row2, 1)
print(dist)
dist = minkowski_distance(row1, row2, 2)
print(dist)

13.0
6.082762530298219
```

Figure 3: Minkowski implementation in Python

Manual calculation

$$\begin{aligned} \text{row1} &= [10 \ 20 \ 15 \ 10 \ 5] \\ \text{row2} &= [12 \ 24 \ 18 \ 8 \ 7] \end{aligned}$$

- Euclidian

$$\begin{aligned} d &= \sqrt[2]{|10 - 12|^2 + |20 - 24|^2 + |15 - 18|^2 + |10 - 8|^2 + |5 - 7|^2} \\ d &= \sqrt[2]{4 + 16 + 9 + 4 + 4} \\ d &= \sqrt[2]{37} \\ d &= 6.082762530298219 \end{aligned}$$

- Manhattan / Cityblock

$$\begin{aligned} d &= \sqrt[1]{|10 - 12| + |20 - 24| + |15 - 18| + |10 - 8| + |5 - 7|} \\ d &= \sqrt[1]{2 + 4 + 3 + 2 + 2} \\ d &= \sqrt[1]{13} \\ d &= 13 \end{aligned}$$

2.2 Task 4

Minkowski

Minkowski distance is a distance metric that is used to measure the distance between two points in a normed vector space. Minkowski can also be considered as a generalisation of Euclidean and Manhattan distance.

The Minkowski distance between two vectors p and q is defined as:

$$d = \sqrt[r]{\sum_{i=1}^n |x_i - y_i|^r}$$

where r is the order of the norm. When $r = 1$, this is equivalent to the Manhattan distance. When $r = 2$, this is equivalent to the Euclidean distance. When $r \rightarrow \infty$, this is equivalent to the Chebyshev distance.

An example of Minkowski Distance real world application is Brain Tumor Detection using Minkowski Distance and K-Nearest Neighbor (KNN) Algorithm. The Minkowski distance is used to build a modified version of KNN algorithm to detect brain tumor.

Chebyshev

Chebyshev distance is a distance metric that is used to measure the distance between two points in a normed vector space. Chebyshev distance is also known as chessboard distance, maximum metric, or L-infinity metric. This is because the distance between two points is the greatest of their differences along any coordinate dimension.

The Chebyshev distance between two vectors p and q is defined as:

$$d_{\infty}(p, q) = \lim_{r \rightarrow \infty} \sqrt[r]{\sum_{i=1}^n |x_i - y_i|^r}$$
$$d_{\infty}(p, q) = \max_{i=1}^n |x_i - y_i|$$

An example of Chebyshev Distance real world application is Energy-Efficient Gossiping protocol (EEGossip). This uses Chebyshev distance

algorithm to determine the distance between two nodes in a network. The goal of EEGossip is to minimize the energy consumption of the network.