

How do you calculate Manhattan Distance in Python?

Authored by
stats writer

April 29, 2024

RECOMMENDED CITATION

stats writer (2024). *How do you calculate Manhattan Distance in Python?*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=140768>

Manhattan Distance is a mathematical term used to measure the distance between two points on a grid-like structure. In Python, it can be calculated by finding the absolute differences between the x-coordinates and y-coordinates of the two points and adding them together. This can be done using the `abs()` function in Python and then summing up the results. The result is a numerical value representing the distance between the two points. This method is commonly used in various fields such as mathematics, computer science, and engineering to measure the distance between two points in a grid-like structure.

Calculate Manhattan Distance in Python (With Examples)

The Manhattan distance between two vectors, A and B , is calculated as:

$$\sum |A_i - B_i|$$

where i is the i th element in each vector.

This distance is used to measure the dissimilarity between two vectors and is commonly used in many machine learning algorithms.

This tutorial shows two ways to calculate the Manhattan distance between two vectors in Python.

Method 1: Write a Custom Function

The following code shows how to create a custom function to calculate the Manhattan distance between

two vectors in Python:

```
from math import sqrt
```

```
#create function to calculate Manhattan distance  
def manhattan(a, b):
```

```
    return sum(abs(val1-val2) for val1, val2 in zip(a,b))
```

```
#define vectors
```

```
A =
```

```
B =
```

```
#calculate Manhattan distance between vectors
```

```
manhattan(A, B)
```

```
9
```

The Manhattan distance between these two vectors turns out to be 9.

We can confirm this is correct by quickly calculating the Manhattan distance by hand:

$$\sum |A_i - B_i| = |2-5| + |4-5| + |4-7| + |6-8| = 3 + 1 + 3 + 2 = 9.$$

Method 2: Use the `cityblock()` function

Another way to calculate the Manhattan distance between two vectors is to use the function from the SciPy package:

```
from scipy.spatial.distance import cityblock
```

```
#define vectors
```

```
A =
```

```
B =
```

```
#calculate Manhattan distance between vectors
```

```
cityblock(A, B)
```

```
9
```

Once again the Manhattan distance between these two vectors turns out to be 9.

Note that we can also use this function to find the Manhattan distance between two columns in a pandas DataFrame:

```
from scipy.spatial.distance import cityblock
```

```
import pandas as pd
```

```
#define DataFrame
```

```
df = pd.DataFrame({'A': ,  
'B': ,  
'C': })
```

```
#calculate Manhattan distance between columns A and  
B  
cityblock(df.A, df.B)
```

9

ARABPSYCHOLOGY.COM