How can the median absolute deviation be calculated in Python?

Authored by

stats writer

April 23, 2024

The median absolute deviation (MAD) is a measure of variability in a set of data, calculated by finding the median of the absolute deviations from the median of the data. In Python, the MAD can be calculated by first finding the median of the data using the median() function, and then finding the absolute deviations from this median using the abs() function. These absolute deviations can then be sorted and the median of this sorted list can be calculated to find the MAD. Alternatively, the median absolute deviation can also be calculated using the mad() function from the statistics module in Python. This function takes in a list of data and returns the MAD value, making it a simple and efficient method for calculating the MAD in Python.

Calculate Median Absolute Deviation in Python

The median absolute deviation measures the spread of in a dataset.

It's a particularly useful metric because it's less affected by outliers than other like standard deviation and variance.

The formula to calculate median absolute deviation, often abbreviated MAD, is as follows:

MAD = median(|xi - xm|)

where:

xi: The ith value in the datasetxm: The median value in the dataset

The following examples shows how to calculate the

median absolute deviation in Python by using the mad function from .

Example 1: Calculate MAD for an Array

The following code shows how to calculate the median absolute deviation for a single NumPy array in Python:

```
import numpy as np
from statsmodels import robust

#define data
data = np.array()

#calculate MAD
robust.mad(data)

11.1195
```

The median absolute deviation for the dataset turns out to be 11.1195.

It's important to note that the formula used to calculate MAD computes a robust estimate of the standard deviation assuming a by scaling the result by a factor of roughly 0.67.

To avoid using this scaling factor, simply set c = 1 as follows:

#calculate MAD without scaling factor
robust.mad(data, c=1)

7.5

Example 2: Calculate MAD for a DataFrame

The following code shows how to calculate MAD for a single column in a pandas DataFrame:

#make this example reproducible
np.random.seed(1)

#create pandas DataFrame
data = pd.DataFrame(np.random.randint(0, 10, size=(5, 3)), columns=)

#view DataFrame
data

A B C
0 5 8 9
1 5 0 0

2 1 7 6

3 9 2 4

4 5 2 4

#calculate MAD for column *B*data].apply(robust.mad)

B 2.965204

dtype: float64

We can use similar syntax to calculate MAD for multiple columns in the pandas DataFrame:

#calculate MAD for all columns

data].apply(robust.mad)

A 0.000000

B 2.965204

C 2.965204

dtype: float64

The median absolute deviation is 0 for column A, 2.965204 for column B, and 2.965204 for column C.