How do I calculate the relative frequency in Python?

Authored by

stats writer

April 18, 2024

Calculating relative frequency in Python refers to the process of determining the proportion of times a specific event or value occurs in a given dataset. This can be achieved by dividing the number of occurrences of the event by the total number of observations in the dataset. The result is a percentage or decimal value that represents the relative frequency of the event. In Python, this can be done by using simple mathematical operations and functions such as division and count. By calculating the relative frequency, one can gain insights into the distribution and occurrence of data in a dataset, which can be useful in various statistical analyses and data visualizations.

Calculate Relative Frequency in Python

Relative frequency measures how frequently a certain value occurs in a dataset *relative* to the total number of values in a dataset.

You can use the following function in Python to calculate relative frequencies:

defrel_freq(x): freqs = return freqs

The following examples show how to use this function in practice.

Example 1: Relative Frequencies for a List of Numbers

The following code shows how to use this function to calculate relative frequencies for a list of numbers:

#define data

data =

#calculate relative frequencies for each value in list
rel_freq(data)

The way to interpret this output is as follows:

The value "1" has a relative frequency of 0.42857 in the dataset.The value "2" has a relative frequency of 0.142857 in the dataset.The value "3" has a relative frequency of 0.142857 in the dataset.The value "4" has a relative frequency of 0.28571 in the dataset.

You'll notice that all of the relative frequencies add up to 1.

Example 2: Relative Frequencies for a List of Characters

The following code shows how to use this function to calculate relative frequencies for a list of characters:

#define data
data =

#calculate relative frequencies for each value in list

rel_freq(data)

The way to interpret this output is as follows:

The value "a" has a relative frequency of 0.4 in the dataset.The value "b" has a relative frequency of 0.4 in the dataset.The value "c" has a relative frequency of 0.2 in the dataset.

Once again, all of the relative frequencies add up to 1.

Example 3: Relative Frequencies for a Column in a pandas DataFrame

```
import pandas as pd

#define data
data = pd.DataFrame({'A': ,
'B': ,
'C': })

#calculate relative frequencies of values in column 'A'
rel_freq(list(data))
```

The way to interpret this output is as follows:

The value "25" has a relative frequency of 0.2 in the column.The value "19" has a relative frequency of 0.2 in the column.The value "14" has a relative frequency of 0.2 in the column.The value "15" has a relative frequency of 0.4 in the column.

Once again, all of the relative frequencies add up to 1.

Relative Frequency Calculator

Relative Frequency Histogram: Definition + Example

How to Calculate Relative Frequency in Excel