What is the equivalent function of rnorm() in Python?

Authored by

stats writer

June 26, 2024

The equivalent function of rnorm() in Python is the random.normal() function from the NumPy library. It generates random numbers from a normal distribution with a specified mean and standard deviation, similar to how rnorm() generates random numbers from a normal distribution in R. This function is commonly used in statistical analysis and simulations in Python.

Use the Equivalent of rnorm() in Python

In the R programming language, we can use the rnorm() function to generate a vector of random values that follow a with a specific mean and standard deviation.

For example, the following code shows how to use rnorm() to create a vector of 8 random values that follow a normal distribution with a mean of 5 and standard deviation of 2:

```
#make this example reproducible
set.seed(1)

#generate vector of 8 values that follow normal distribution with mean=5 and sd=2
rnorm(n=8, mean=5, sd=2)

3.747092 5.367287 3.328743 8.190562 5.659016 3.359063 5.974858 6.476649
```

The equivalent of the rnorm() function in Python is the

np.random.normal() function, which uses the following basic syntax:

np.random.normal(loc=0, scale=1, size=None)

where:

loc: Mean of the distributionscale: Standard deviation of the distributionsize: Sample size

The following example shows how to use this function in practice.

Example: Using the Equivalent of rnorm() in Python

The following code shows how to use the np.random.normal() function to generate an array of random values that follow a normal distribution with a specific mean and standard deviation.

```
import numpy as np

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

#generate array of 8 values that follow normal distribution with mean=5 and
```

sd=2np.random.normal(loc=5, scale=2, size=8)

array()

The result is a NumPy array that contains 8 values generated from a normal distribution with a mean of 5 and a standard deviation of 2.

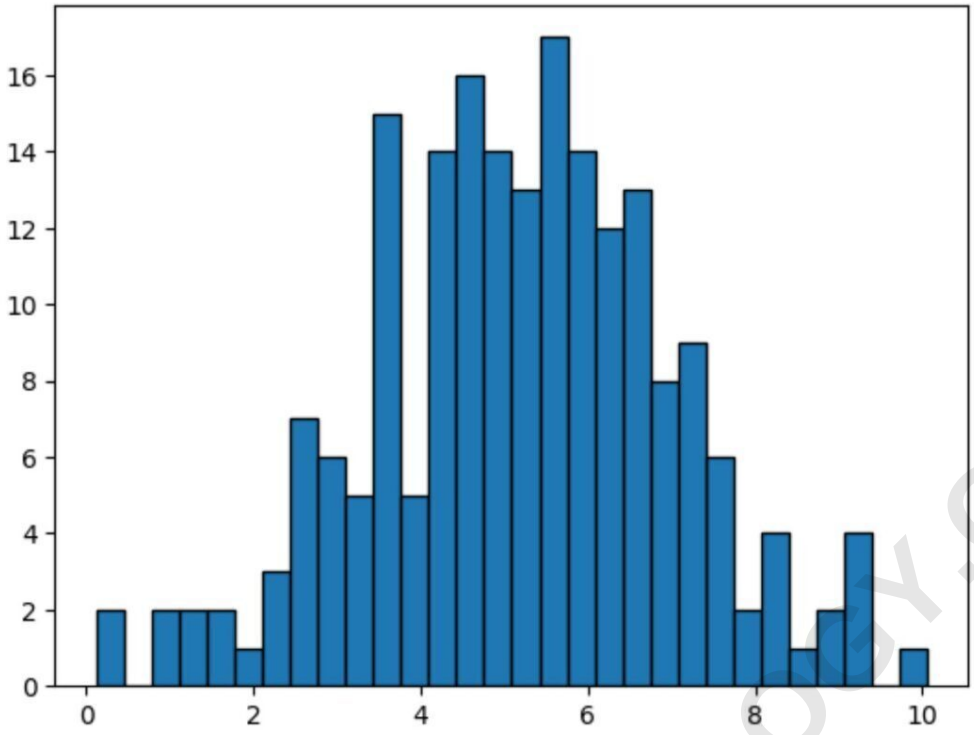
You can also create a histogram using Matplotlib to visualize a normal distribution generated by the np.random.normal() function:

```
import numpy as np
import matplotlib.pyplotas plt

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

#generate array of 200 values that follow normal distribution with mean=5 and sd=2data = np.random.normal(loc=5, scale=2, size=200)

#create histogram to visualize distribution of values
plt.hist(data, bins=30, edgecolor='black')
```



We can see that the distribution of values is roughly bell-shaped with a mean located at 5 and a standard deviation of 2.

Note: You can find the complete documentation for the np.random.normal() function .

The following tutorials explain how to perform other common operations in Python: