



NumPy std()

If this Python Tutorial saves you hours of work, please **whitelist it in your ad blocker** 🙏 and

Donate Now

(<https://www.pythontutorial.net/donation/>)

to help us ❤️ pay for the web hosting fee and CDN to keep the

website running.

Summary: in this tutorial, you'll learn how to use the numpy `std()` function to calculate the standard deviation.

Standard deviation measures how spread out the elements of an array is. The more spread out elements is, the greater their standard deviation.

Standard deviation is the square root of the variance. To calculate the variance, check out the numpy [var\(\) function tutorial](#)

(<https://www.pythontutorial.net/python-numpy/numpy-var/>) .

To calculate standard deviation, you can use the numpy `std()` function as follows:

```
numpy.std(a, axis=None, dtype=None, out=None, ddof=0, keepdims=<no value>, *, where=<no value>)
```

The `std()` function has many parameters but we'll focus on only the first one in this tutorial.

NumPy std() function example

Suppose you have a list of trees with the broadest crown. The first column displays the tree name and the second column shows its corresponding diameter in feet:

Tree Name	Diameter (Feet)
Thimmamma Marrimanu	591
Monkira Monster	239
Oriental Plane Tree at Corsham Court	210
Saman de Guere	207
The Big Tree	201
Shugborough Yew	182
Moreton Bay Fig Tree	176
The Pechanga Great Oak	176

Tree Name	Diameter (Feet)
El Gigante	175
Benaroon	170
The E. O. Hunt Oak	170
The Lansdowne Sycamore	169
The Glencoe Tree	168

The following example uses the `std()` function to calculate the standard deviation of the diameters of the above trees:

```
import numpy as np

diameters = np.array([591, 239, 210, 207, 201, 182,
                      176, 176, 175, 170, 170, 169, 168, ])

result = np.std(diameters)
print(round(result, 1))
```

Output:

```
109.6
```

How it works.

First, create an array that holds the diameters of trees:

```
diameters = np.array([591, 239, 210, 207, 201, 182,  
                      176, 176, 175, 170, 170, 169, 168, ])
```

Second, calculate the standard deviation of diameters using the `std()` function:

```
result = np.std(diameters)
```

Third, round the standard deviation and display it:

```
print(round(result, 1))
```

By using the standard deviation, we have a “standard” way of knowing which trees have a normal diameter, and which trees have large or small diameters.

Summary

- Use the numpy `std()` function to calculate the standard deviation.