

How can I plot a distribution in Seaborn?

Authored by
stats writer

June 25, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I plot a distribution in Seaborn?*. PSYCHOLOGICAL SCALES.
Retrieved from <https://scales.arabpsychology.com/?p=152393>

Seaborn is a popular Python library used for data visualization. It offers various functions and tools to create visually appealing and informative plots. To plot a distribution in Seaborn, one can follow the steps below:

1. Import the necessary libraries, including Seaborn.
2. Load or generate the data to be plotted.
3. Choose the appropriate plot type for the desired distribution (e.g. histogram, boxplot, etc.).
4. Use the relevant Seaborn function, such as "distplot" or "boxplot", to create the plot.
5. Customize the plot by adjusting parameters such as colors, labels, and axes limits.
6. Display and save the plot as needed.

In summary, Seaborn provides a user-friendly and efficient way to visualize distributions of data, making it a valuable tool for data analysis and exploration.

Plot a Distribution in Seaborn (With Examples)

You can use the following methods to plot a distribution of values in Python using the data visualization library:

Method 1: Plot Distribution Using Histogram

```
sns.displot(data)
```

Method 2: Plot Distribution Using Density Curve

```
sns.displot(data, kind='kde')
```

Method 3: Plot Distribution Using Histogram & Density Curve

```
sns.displot(data, kde=True)
```

The following examples show how to use each method in practice.

Example 1: Plot Distribution Using Histogram

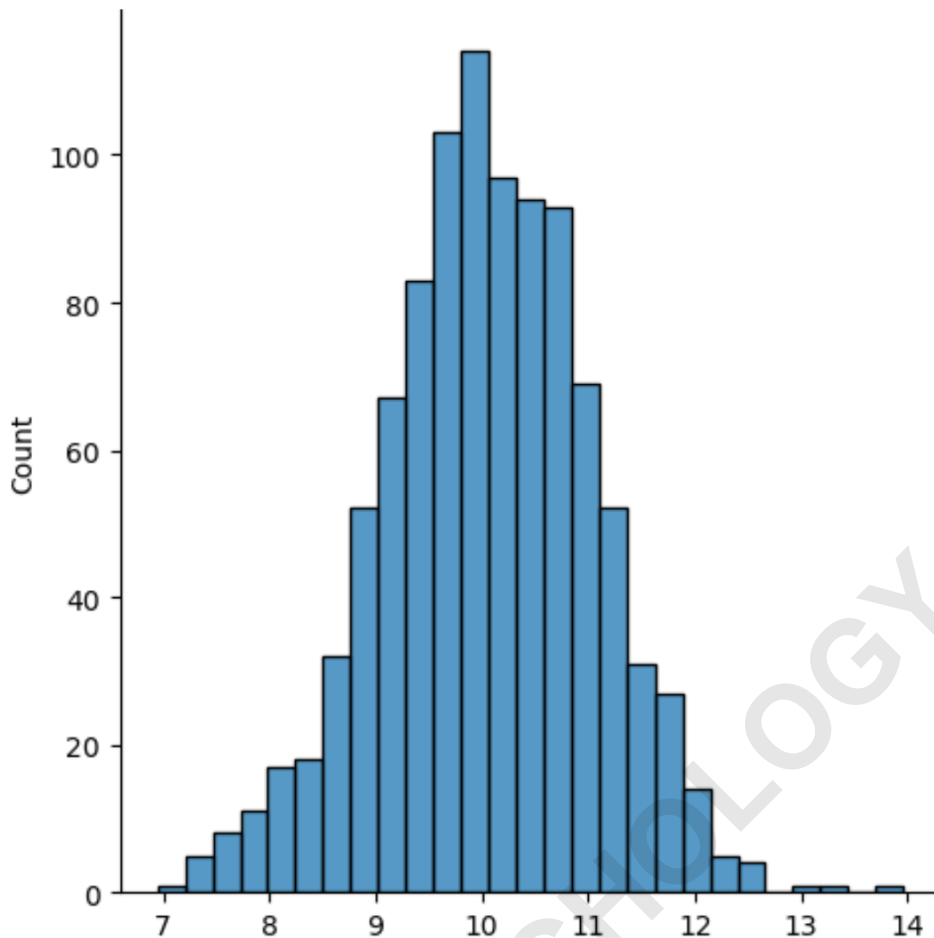
The following code shows how to plot the distribution of values in a NumPy array using the `displot()` function in seaborn:

```
import seaborn as sns
import numpy as np

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

#create array of 1000 values that follow a normal
distribution with mean of 10
data = np.random.normal(size=1000, loc=10)

#create histogram to visualize distribution of values
sns.displot(data)
```



The x-axis displays the values in the distribution and the y-axis displays the count of each value.

To change the number of bins used in the histogram, you can specify a number using the bins argument:

```
import seaborn as sns
```

```
import numpy as np
```

```
#make this example reproducible
```

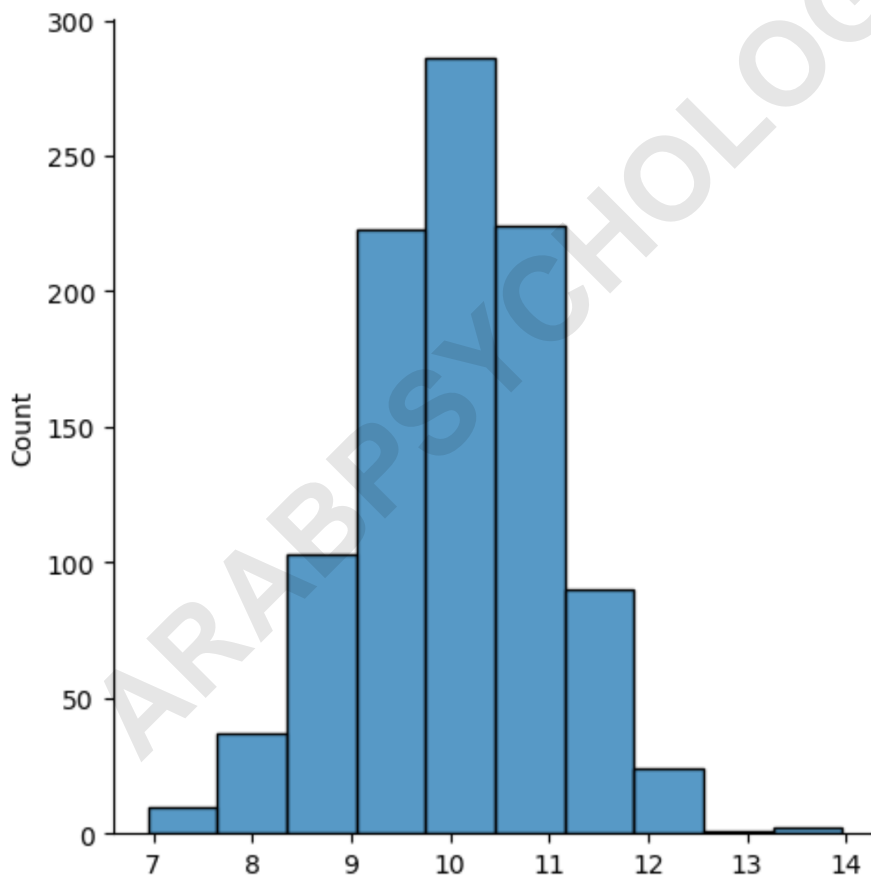
```
np.random.seed(1)
```

```
#create array of 1000 values that follow a normal  
distribution with mean of 10
```

```
data = np.random.normal(size=1000, loc=10)
```

```
#create histogram using 10 bins
```

```
sns.displot(data, bins=10)
```



Example 2: Plot Distribution Using Density Curve

```
import seaborn as sns
```

```
import numpy as np
```

```
#make this example reproducible
```

```
np.random.seed(1)
```

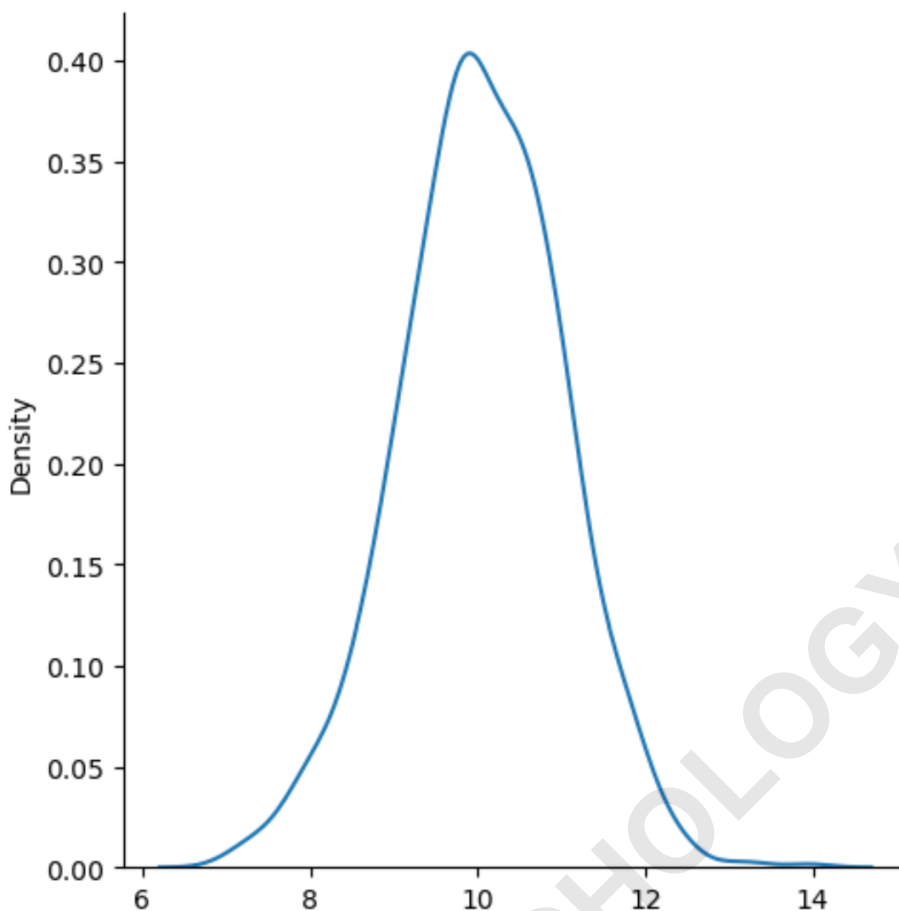
```
#create array of 1000 values that follow a normal  
distribution with mean of 10
```

```
data = np.random.normal(size=1000, loc=10)
```

```
#create density curve to visualize distribution of values
```

```
sns.displot(data, kind='kde')
```

ARABPSYCHOLOGY.COM



The x-axis displays the values in the distribution and the y-axis displays the relative frequency of each value.

Note that `kind='kde'` tells seaborn to use kernel density estimation, which produces a smooth curve that summarizes the distribution of values for a variable.

Example 3: Plot Distribution Using Histogram & Density Curve

The following code shows how to plot the distribution of values in a NumPy array using a histogram with a

density curve overlaid:

```
import seaborn as sns
```

```
import numpy as np
```

```
#make this example reproducible
```

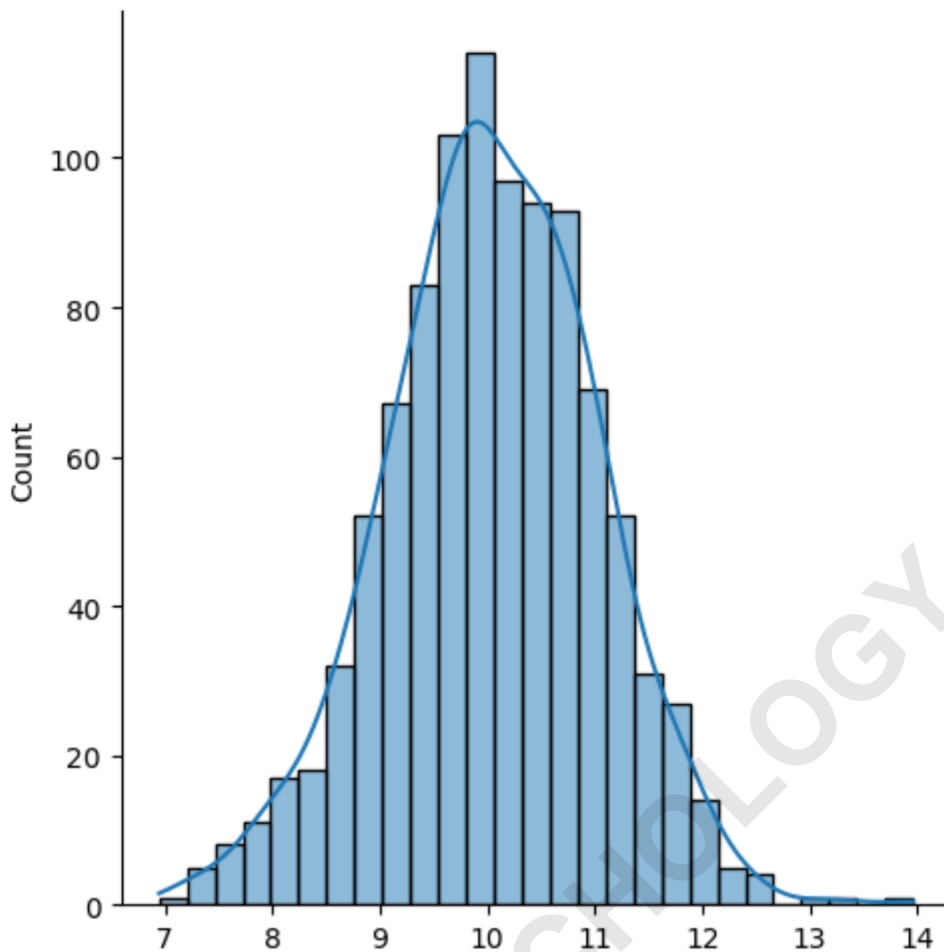
```
np.random.seed(1)
```

```
#create array of 1000 values that follow a normal  
distribution with mean of 10
```

```
data = np.random.normal(size=1000, loc=10)
```

```
#create histogram with density curve overlaid to  
visualize distribution of values
```

```
sns.displot(data, kde=True)
```



The result is a histogram with a density curve overlaid.

Note: You can find the complete documentation for the `seaborn displot()` function .

The following tutorials explain how to perform other common tasks using seaborn: