How can the Y-axis of a Pandas histogram be displayed as a percentage?

Authored by

stats writer

June 25, 2024

The Y-axis of a Pandas histogram can be displayed as a percentage by using the "density" parameter in the "plot.hist" function. This will normalize the values on the Y-axis to represent the percentage of total occurrences in each bin. Additionally, the "normed" parameter can also be used to achieve the same result. This allows for a more accurate comparison of the distribution of data across different bins, as it takes into consideration any differences in the number of data points in each bin. By displaying the Y-axis as a percentage, it provides a more intuitive representation of the data and allows for easier interpretation and analysis.

Display Percentage on Y-Axis of Pandas Histogram

You can use the following basic syntax to display percentages on the y-axis of a pandas histogram:

```
import pandas as pd
import numpy as np
import matplotlib.pyplotas plt
from matplotlib.tickerimport PercentFormatter

#create histogram, using percentages instead of counts
plt.hist(df, weights=np.ones(len(df)) / len(df))

#apply percentage format to y-axis
plt.gca().yaxis.set_major_formatter(PercentFormatter(1)
)
plt.show()
```

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

Example: Display Percentage on Y-Axis of Pandas Histogram

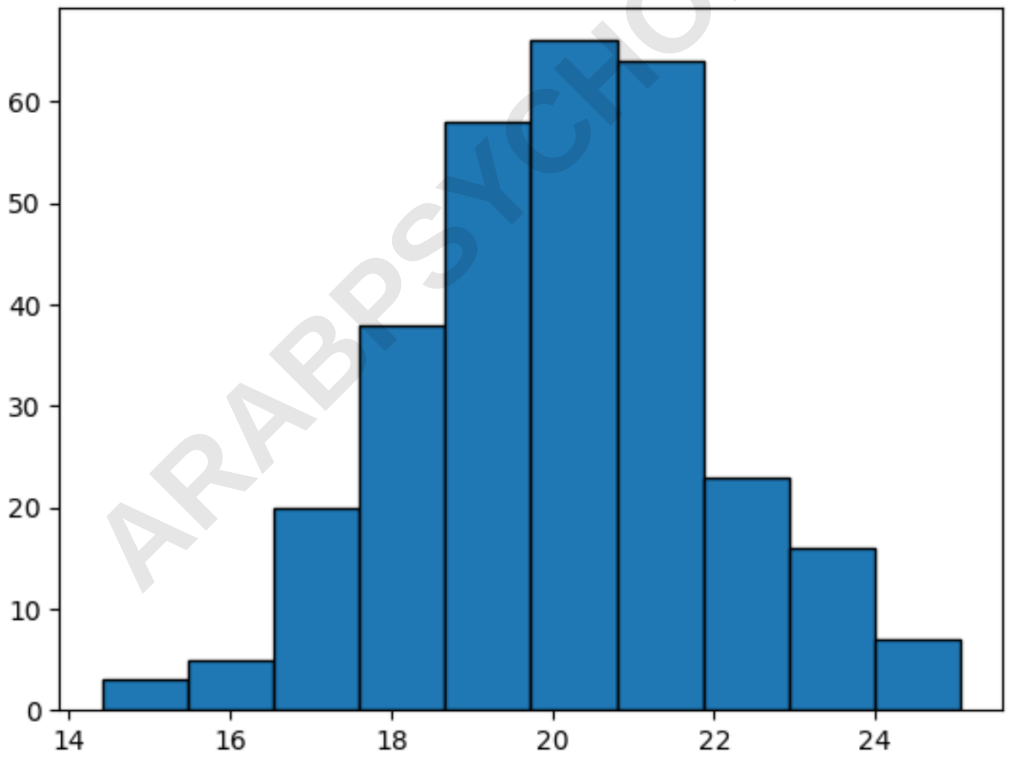
Suppose we have the following pandas DataFrame that contains information about various basketball players:

```
import pandas as pd
import numpy as np

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

#create DataFrame
df = pd.DataFrame({'points': np.random.normal(loc=20, scale=2, size=300),
'assists': np.random.normal(loc=14, scale=3, size=300),
'rebounds': np.random.normal(loc=12, scale=1, size=300)})

#view head of DataFrame
print(df.head())

points assists rebounds
0 23.248691 20.197350 10.927036
1 18.776487 9.586529 12.495159
2 18.943656 11.509484 11.047938
3 17.854063 11.358267 11.481854
```

4 21.730815 13.162707 10.538596

If we create a histogram to visualize the distribution of values in the points column, the y-axis will display counts by default:

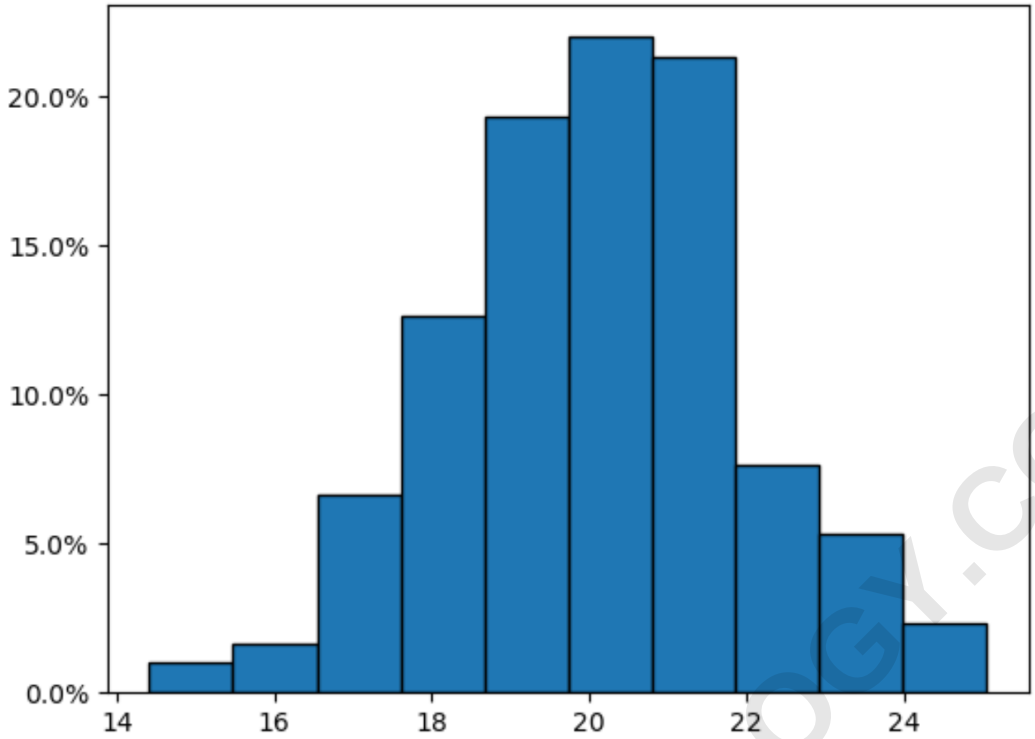
```
import matplotlib.pyplot as plt

#create histogram for points columb
plt.hist(df, edgecolor='black')
```



To instead display percentages on the y-axis, we can

use the PercentFormatter function:

```
import numpy as np
import matplotlib.pyplotas plt
from matplotlib.tickerimport PercentFormatter

#create histogram, using percentages instead of counts
plt.hist(df, weights=np.ones(len(df)) / len(df), edgecolor='black')

#apply percentage format to y-axis
plt.gca().yaxis.set_major_formatter(PercentFormatter(1))
plt.show()
```

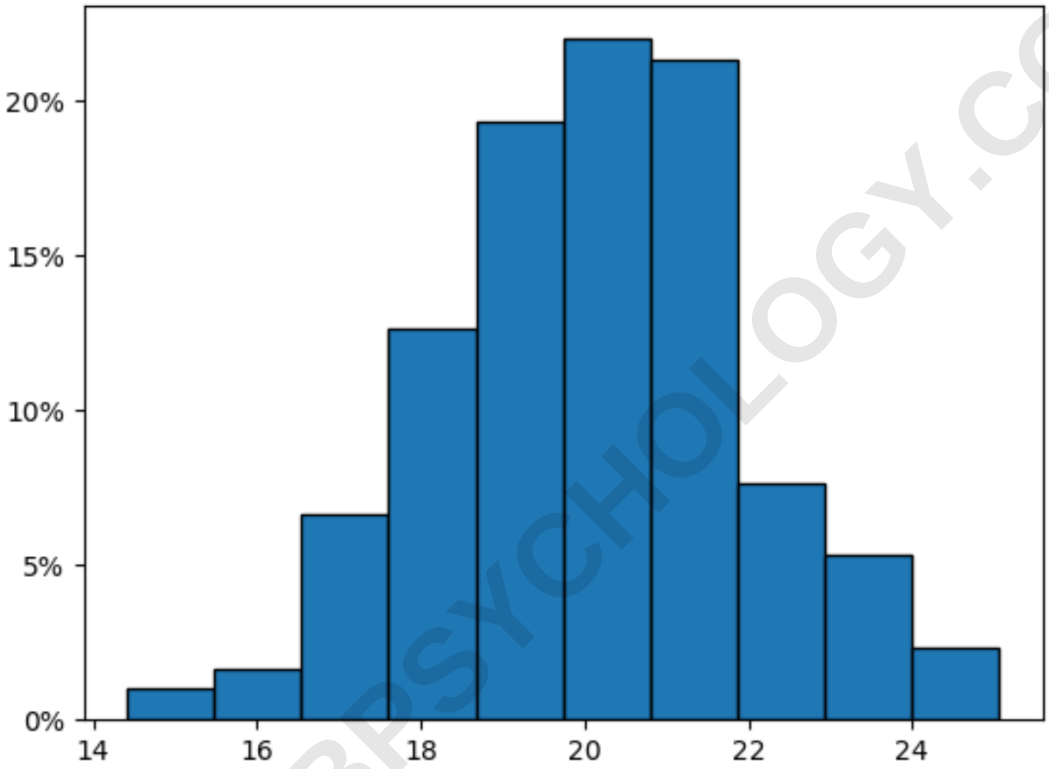


Notice that the y-axis now displays percentages.

If you'd like to remove the decimals from the percentages, simply use the argument decimals=0 within the PercentFormatter() function:

```
import numpy as np
import matplotlib.pyplotas plt
from matplotlib.tickerimport PercentFormatter

#create histogram, using percentages instead of counts
plt.hist(df, weights=np.ones(len(df)) / len(df), edgecolor='black')
```

#apply percentage format to y-axis
plt.gca().yaxis.set_major_formatter(PercentFormatter(1, decimals=0))
plt.show()



The y-axis now displays percentages without any decimals.

The following tutorials explain how to perform other common tasks in pandas: