

The code provided is for a Python function get\_pdf\_probability that calculates the probability of values falling within a specific range under a normal distribution, given a dataset.

**Code Explanation**

1. **Imports**:
   * pyplot from matplotlib and seaborn are used for plotting.
   * norm from scipy.stats is used to create a normal distribution.
2. **Distribution Plot**:
   * A distribution plot (sns.distplot) is created for the dataset, showing a Kernel Density Estimate (KDE) curve with the color blue and the underlying histogram with the color green.
   * Red vertical lines (pyplot.axvline) are added to mark the start and end of the range where the probability is being calculated.
3. **Sample Parameters**:
   * sample\_mean and sample\_std are calculated using the mean and standard deviation of the dataset, respectively.
   * These parameters are used to define a normal distribution (dist = norm(sample\_mean, sample\_std)).
4. **Probability Calculation**:
   * The one-liner for loop is used to calculate the probabilities for values in the specified range.

python

Copy code

probabilities = [dist.pdf(value) for value in values]

* + This is a list comprehension that iterates over each value in values (which is the range from startrange to endrange) and applies the probability density function (pdf) to each value.

1. **Sum of Probabilities**:
   * The probabilities are summed to calculate the total probability of values falling within the specified range.
   * The calculated probability is printed and returned by the function.

**One-Liner For Loop Explanation**

The one-liner for loop probabilities = [dist.pdf(value) for value in values] is a list comprehension that does the following:

* **dist.pdf(value):** Computes the probability density function (PDF) value for a specific value, given the normal distribution defined by sample\_mean and sample\_std.
* **for value in values:** Iterates over all values in the list values, which contains numbers from startrange to endrange.
* **[...]:** Collects all these PDF values into a list called probabilities.

**Alternate Code Without One-Liner For Loop**

Here is how the code would look without using the one-liner for loop:

**probabilities = [] # Initialize an empty list to store probabilities**

**for value in values:**

**prob = dist.pdf(value) # Calculate the PDF for each value**

**probabilities.append(prob) # Append the result to the list**

**How It's Executed**

1. The loop iterates over each value in the list values.
2. For each value, it calculates the PDF using dist.pdf(value).
3. It then appends the resulting probability to the probabilities list.
4. After the loop has processed all values, probabilities contains all the PDF values for the specified range.

Summary: The 1 liner for loop combines the loop and the list-building steps into a single line, making the code shorter and, once you're familiar with it, easier to read.