

This function calculates and visualizes the **Probability Density Function (PDF)** of a given dataset, and computes the total probability (area under the curve of the PDF) between a specified range of values.

**Inputs:**

* dataset: The input data for which the PDF is calculated.
* startrange: The start of the range for which the probability is calculated.
* endrange: The end of the range for which the probability is calculated.

**sns.distplot:** Plots the data's distribution with a kernel density estimate (KDE), overlaying a smooth PDF curve.

* kde=True: Enables KDE plotting.
* kde\_kws={'color': 'blue'}: Specifies the KDE curve color.
* color='Green': Sets the histogram bar color.

**pyplot.axvline:** Draws vertical lines at startrange and endrange for visualizing the range.

**Dist**: Creates a **normal distribution object** based on the dataset's mean and standard deviation. This is used to calculate the PDF values.

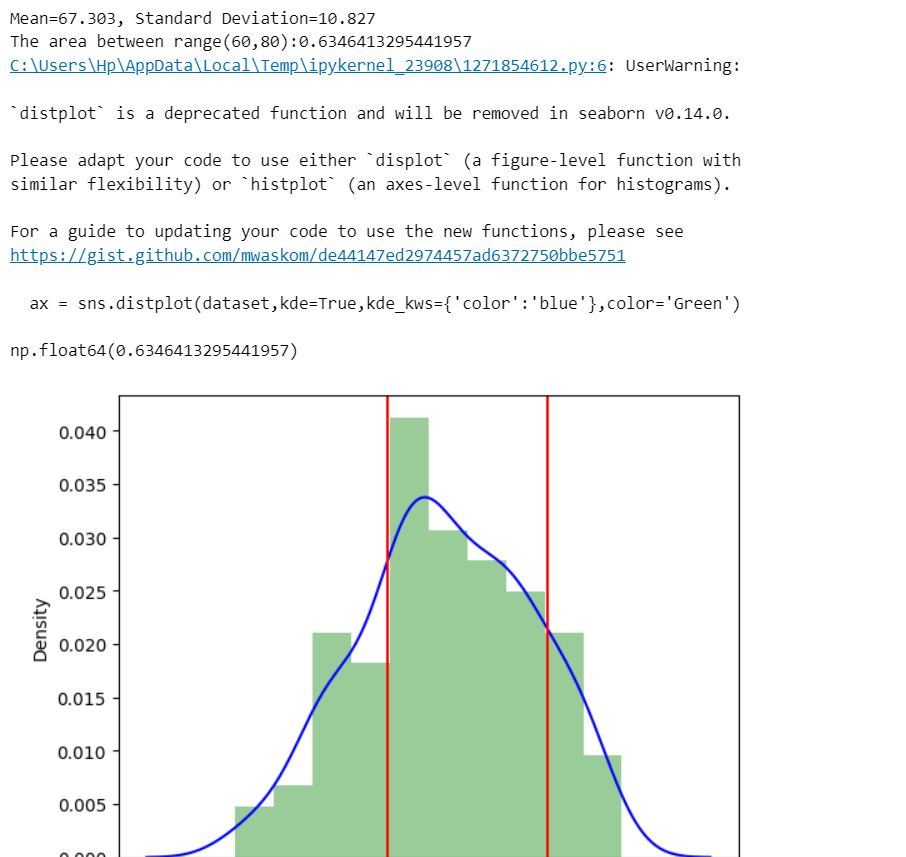
**values:** Generates all integer values between startrange and endrange.

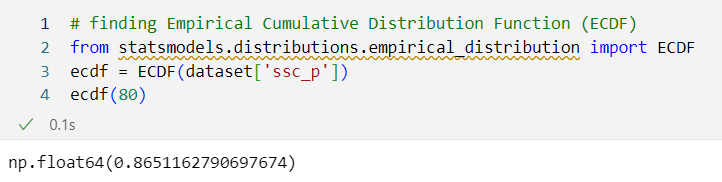
**dist.pdf(value):** Computes the PDF value (height of the curve) for each value in the range.

**sum(probabilities):** Adds up the PDF values to approximate the area under the curve, which represents the total probability in the range [startrange, endrange].

**get\_pdf\_probability(dataset["ssc\_p"],60,80)**

**Output:**

* The mean and standard deviation of the dataset.
* A plot showing the data's PDF and the specified range (with vertical red lines).
* The computed probability (area under the curve) for the range [70, 90].
* 

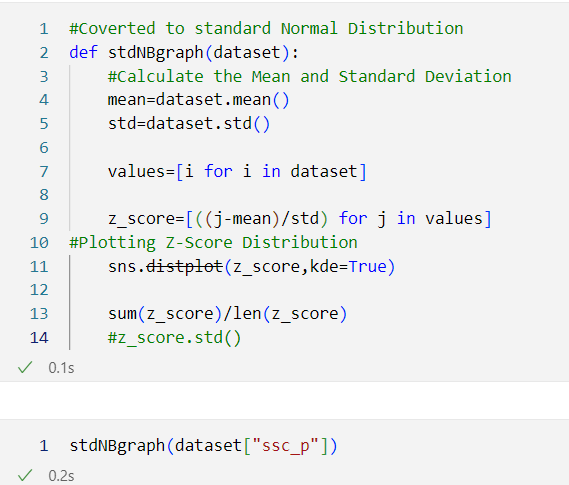


This code snippet is calculating the **Empirical Cumulative Distribution Function (ECDF)** for a specific value (80) using the data from the ssc\_p column of the dataset.

**ECDF**: A function from statsmodels that calculates the Empirical Cumulative Distribution Function (ECDF) of a dataset.

* The ECDF represents the proportion of data points in the dataset that are less than or equal to a given value.
* It’s a non-parametric method and doesn’t assume the data follows a particular distribution.

**dataset['ssc\_p']**: The column of the dataset used to compute the ECDF (in this case, the SSC percentages).



**values**: Extracts all data points from the dataset.

**Standardization**: Each data point is converted to its **z-score**:

* Formula: Z=X−μσZ = \frac{X - \mu}{\sigma}Z=σX−μ​
  + XXX: The value in the dataset.
  + μ\muμ: Mean of the dataset.
  + σ\sigmaσ: Standard deviation of the dataset.
* **Purpose**: Z-scores represent how many standard deviations a data point is from the mean.
* Example:
  + If X=70X = 70X=70, μ=60\mu = 60μ=60, and σ=10\sigma = 10σ=10:  
    Z=70−6010=1Z = \frac{70 - 60}{10} = 1Z=1070−60​=1
  + This means the value 707070 is **1 standard deviation above the mean**.

**sns.distplot**: Uses Seaborn to plot the distribution of z-scores.

* **kde=True**: Adds a Kernel Density Estimate (smooth curve) over the histogram.

Calculates the mean of all **z-scores**.