**Assignment No. 4**

**Name : Aditya Lagad**

**PRN No : 122B1F075**

**Problem Statement:**

Implement normal distribution in python and visualize it for Mean =100, standard \_deviation =4, dataset size=100000

**Objective:**  
The goal of this challenge is to use Python to create and display a dataset with a normal distribution. This entails building a dataset according to a predetermined mean and standard deviation, then utilizing a histogram and a theoretical probability density curve to visualize the distribution. By doing this, we want to gain an understanding of a normal distribution's properties, behavior, and graphic representation.

**Prerequisites:**

You will need the following in order to finish this task:

• Your system has Python installed.

• Knowledge of the fundamentals of Python programming.

• Installed required libraries:

For creating the normal distribution, use NumPy.

* Matplotlib: For distribution visualization.
* If the required libraries aren't already installed, install them using:
* Installing numpy matplotlib with pip

**Theory:**

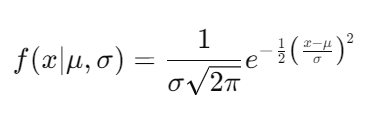
Data close to the mean occur more frequently than data distant from the mean, according to the normal distribution, a probability distribution that is symmetric about the mean. A normal distribution is characterized by a bell-shaped curve.

1. Mean (μ): The average or central value of the data, in this case 100.

2. Standard Deviation (σ): Indicates how widely distributed or dispersed the data is (in this case, 4).

3. Dataset Size: The quantity of data points produced (in this example, 100,000).

Since many natural occurrences are described by the normal distribution and because normalcy is assumed by many statistical tests, it is significant in statistics.

For a normal distribution, the probability density function (PDF) is: ****

**Script Overview:**

The script will:

1. Import Matplotlib and NumPy, two required libraries.

2. Create 100,000 data points with a mean of 100 and a standard deviation of 4 that exhibit a normal distribution.

3. Use a histogram to see the distribution of the data and superimpose a normal distribution curve for comparison.

**Algorithm:**

Step 1:

is to import the necessary libraries. Numpy is needed to generate the data for the normal distribution.

• To visualize the data using a curve and histogram, import matplotlib.pyplot.

Step 2:

Establish the Distribution's Parameters

• Determine the dataset size, mean, and standard deviation.

• The dataset size in this instance is 100,000, the mean is 100, and the standard deviation is 4.

Step 3 :

To create random data points that adhere to the normal distribution, use numpy.random.normal().

• A dataset of 100,000 points with the given mean and standard deviation will be produced by the function.

Step 4 :

• To construct a histogram of the generated data, use matplotlib.pyplot.hist().

• To get a smooth distribution, set the number of bins to 50.

• Apply density=True.

Step 5 :  
• To create a series of x-values that cover the histogram's range, use numpy.linspace().  
• Using the formula, determine the normal distribution's probability density function (PDF).  
• To compare the histogram with the theoretical distribution, overlay the curve using plt.plot().   
Step 6:   
• Give the plot a title that explains the distribution.  
•Put "Data points" on the x-axis and "Probability Density" on the y-axis.   
• To improve the plot's clarity, use plt.grid().   
Step 7:   
• To see the normal distribution curve and histogram, use plt.show().

**References:**

 **NumPy Documentation**: https://numpy.org/doc/stable/

 **Matplotlib Documentation**: https://matplotlib.org/stable/index.html

 **Normal Distribution**: [Wikipedia: Normal Distribution](https://en.wikipedia.org/wiki/Normal_distribution)

**Conclusion:**  
A normal distribution with a mean of 100, a standard deviation of 4, and a dataset size of 100,000 was successfully implemented and shown in this script. While the superimposed curve showed the theoretical normal distribution for the identical parameters, the histogram gave a clear picture of the empirical distribution. This graphic can be used for a variety of statistical analysis tasks and is helpful in comprehending the behavior of the normal distribution.