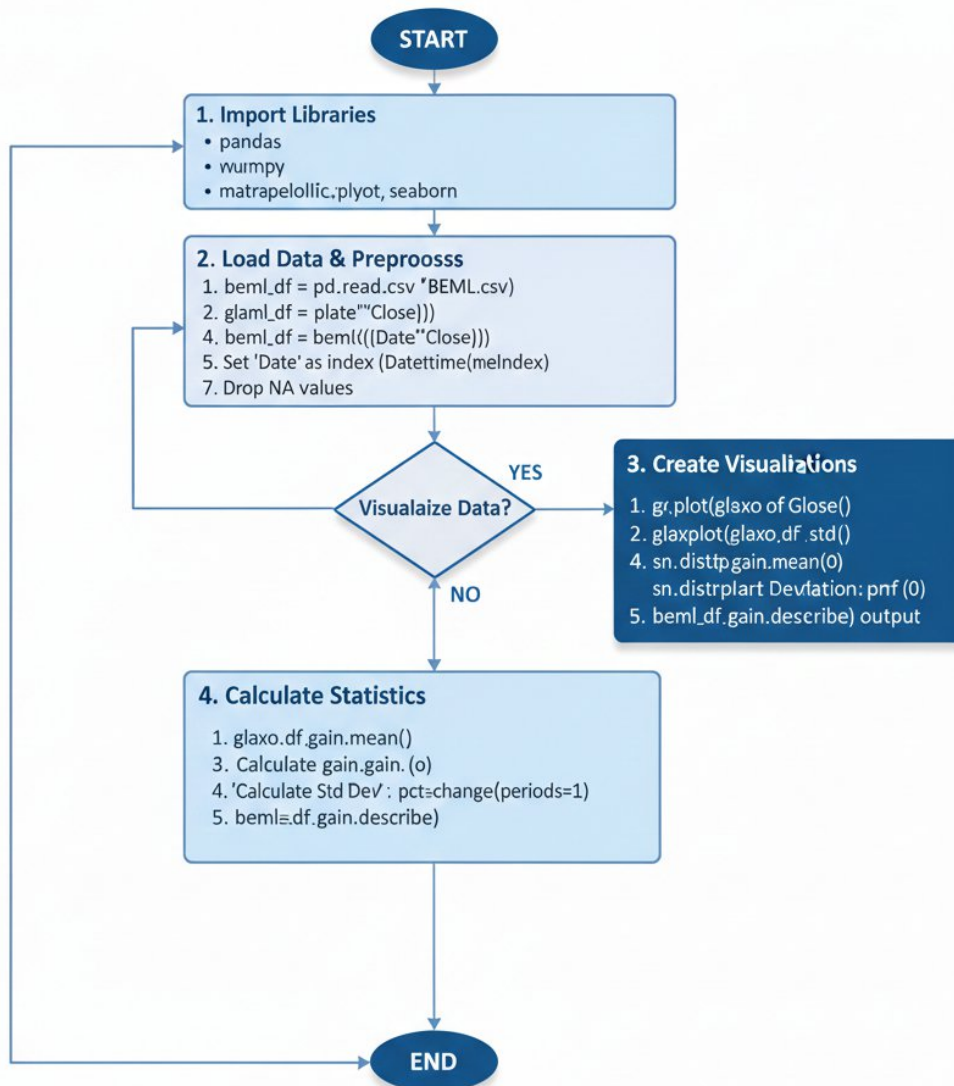# Experiment No - 1

Objective: To Study Normal Distribution function with different dataset.

Date of experiment: 08/01/2026

Date of Completion: 08/01/2026

Flow Chart:

START

1. Import Libraries
- pandas
- wurmpy
- matrapelollic.plyot, seaborn

2. Load Data & Preproosss
1. beml_df = pd.read.csv 'BEML.csv)
2. glaml_df = plate""Close)))
4. beml_df = beml((([Date"Close)))
5. Set 'Date' as index (Datettime(meIndex)
7. Drop NA values

Visualaize Data?

YES

NO

3. Create Visualiations
1. gr.plot(glaxo of Glose()
2. glaxplot(glaxo.df .std()
4. sn.disttp gain.mean(0)
   sn.distrplart Devlation: pnf (0)
5. beml_df.gain.describe) output

4. Calculate Statistics
1. glaxo.df.gain.mean()
3. Calculate gain.gain. (o)
4. 'Calculate Std Dev : pct≡change(periods=1)
5. beml≡.df.gain.describe)

END

Source Code:

import pandas as pd

import numpy as np

import warnings

beml_df = pd.read_csv('BEML.csv')

```python
beml_df[0:5]

glaxo_df = pd.read_csv('GLAXO.csv')

glaxo_df[0:5]

beml_df = beml_df[['Date','Close']]

glaxo_df = glaxo_df[['Date','Close']]

glaxo_df = glaxo_df.set_index(pd.DatetimeIndex(glaxo_df['Date']))

beml_df = beml_df.set_index(pd.DatetimeIndex(beml_df['Date']))

glaxo_df.head()

import matplotlib.pyplot as plt

import seaborn as sn

%matplotlib inline


plt.plot(glaxo_df.Close)

plt.xlabel('Time')

plt.ylabel('Close Price')

plt.plot(beml_df.Close)

plt.xlabel('Time')

plt.ylabel('Close Price')

glaxo_df['gain'] = glaxo_df.Close.pct_change(periods = 1)

beml_df['gain'] = beml_df.Close.pct_change(periods = 1)
```

```python
glaxo_df.head(5)

glaxo_df = glaxo_df.dropna()

beml_df = beml_df.dropna()

plt.figure(figsize = (8,6))

plt.plot(glaxo_df.index, glaxo_df.gain)

plt.xlabel('Time')

plt.ylabel('Gain')

sn.distplot(glaxo_df.gain, label = 'Glaxo')

sn.distplot(beml_df.gain, label = 'BEML')

plt.xlabel('gain')

plt.ylabel('Density')

plt.legend()

print("Daily gain of Glaxo")

print("----------------------")

print("Mean: ",round(glaxo_df.gain.mean(), 4))

print("Standard Deviation: ", round(glaxo_df.gain.std(),4))

print("Daily gain of BEML")

print("----------------------")

print("Mean: ",round(beml_df.gain.mean(), 4))

print("Standard Deviation: ", round(beml_df.gain.std(),4))

beml_df.gain.describe()
```

Dataset

The notebook uses daily trading data from the Bombay Stock Exchange (BSE) for the period between 2010 and 2016.
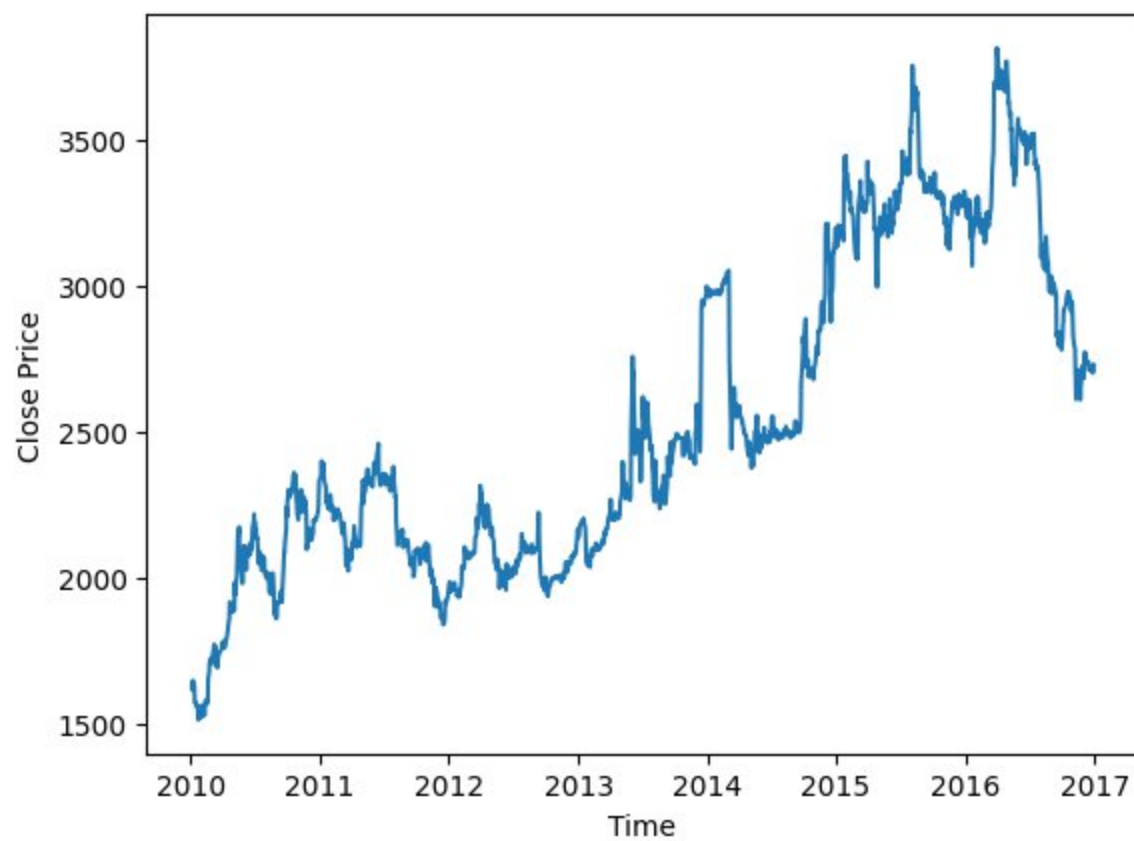
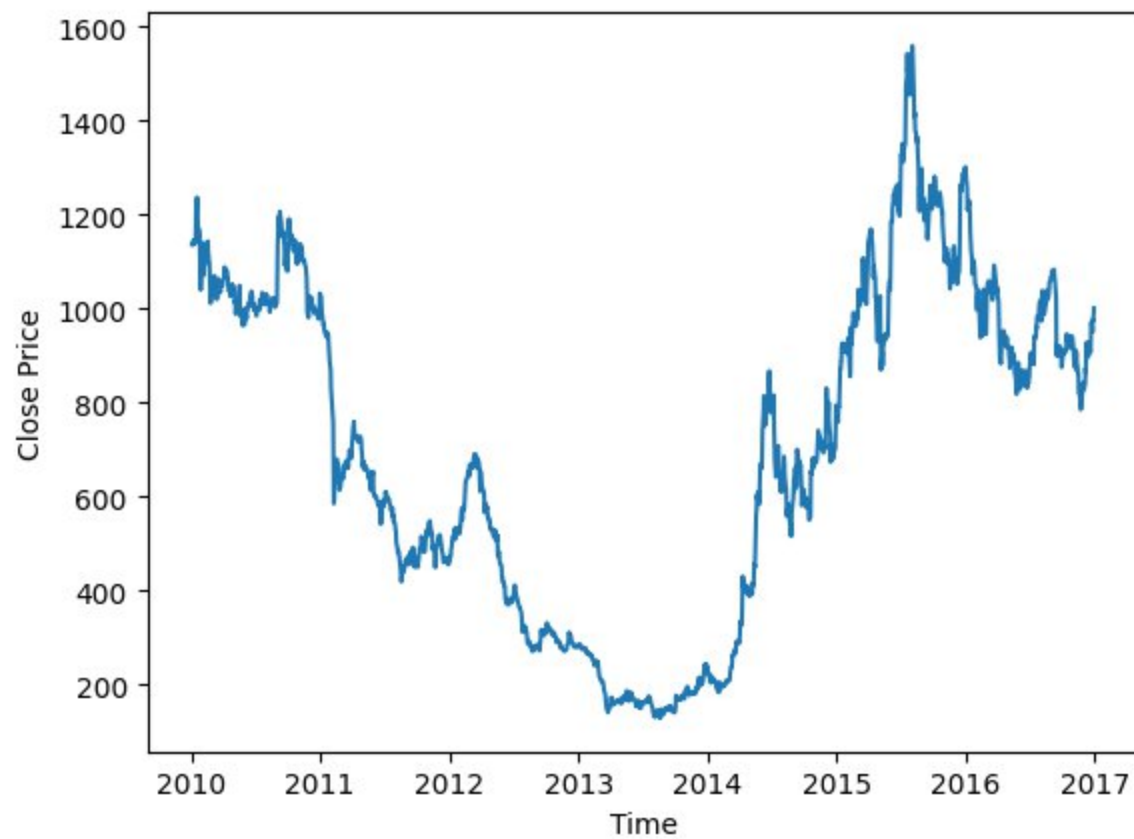The analysis specifically focuses on the daily returns of two stocks:

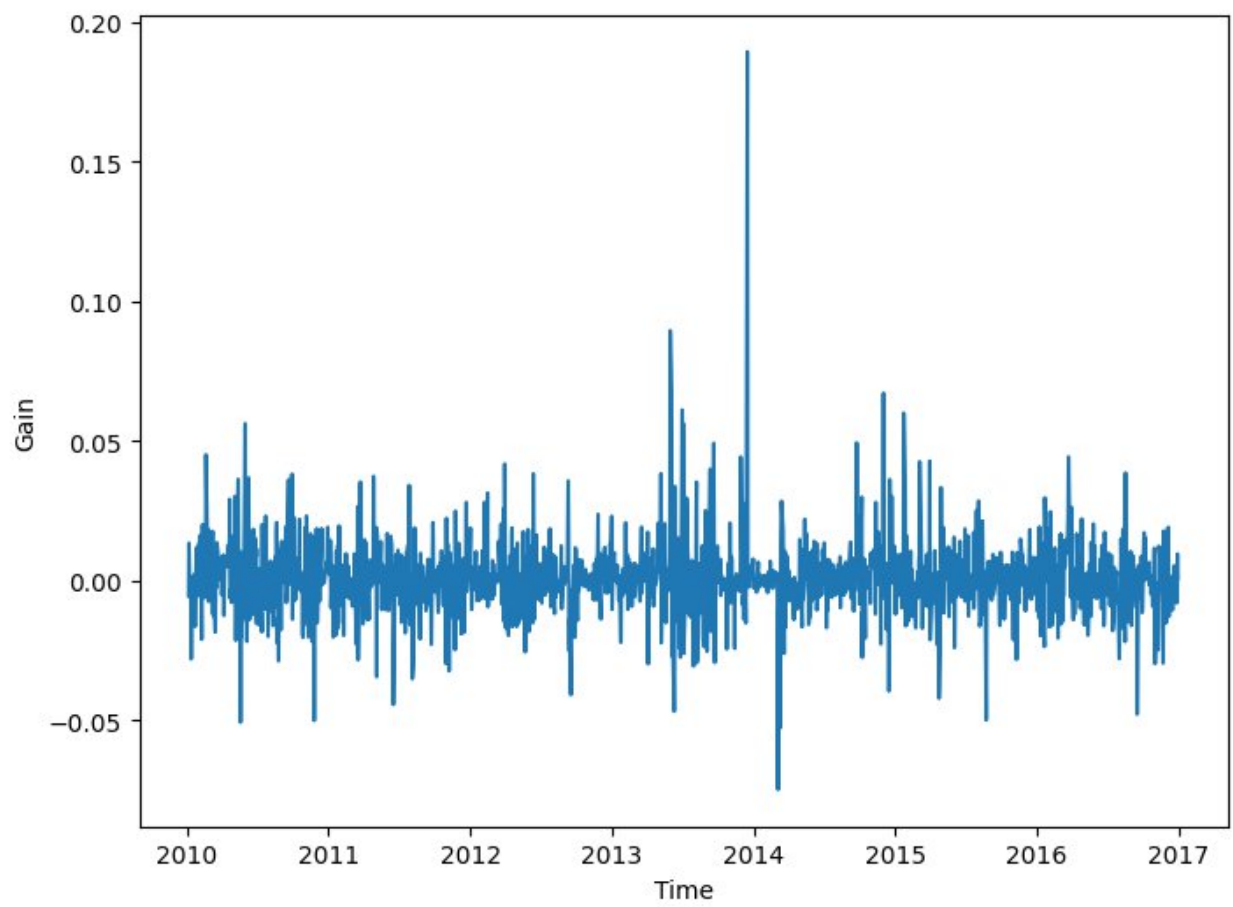- BEML (loaded from BEML.csv).
- GLAXO (loaded from GLAXO.csv).

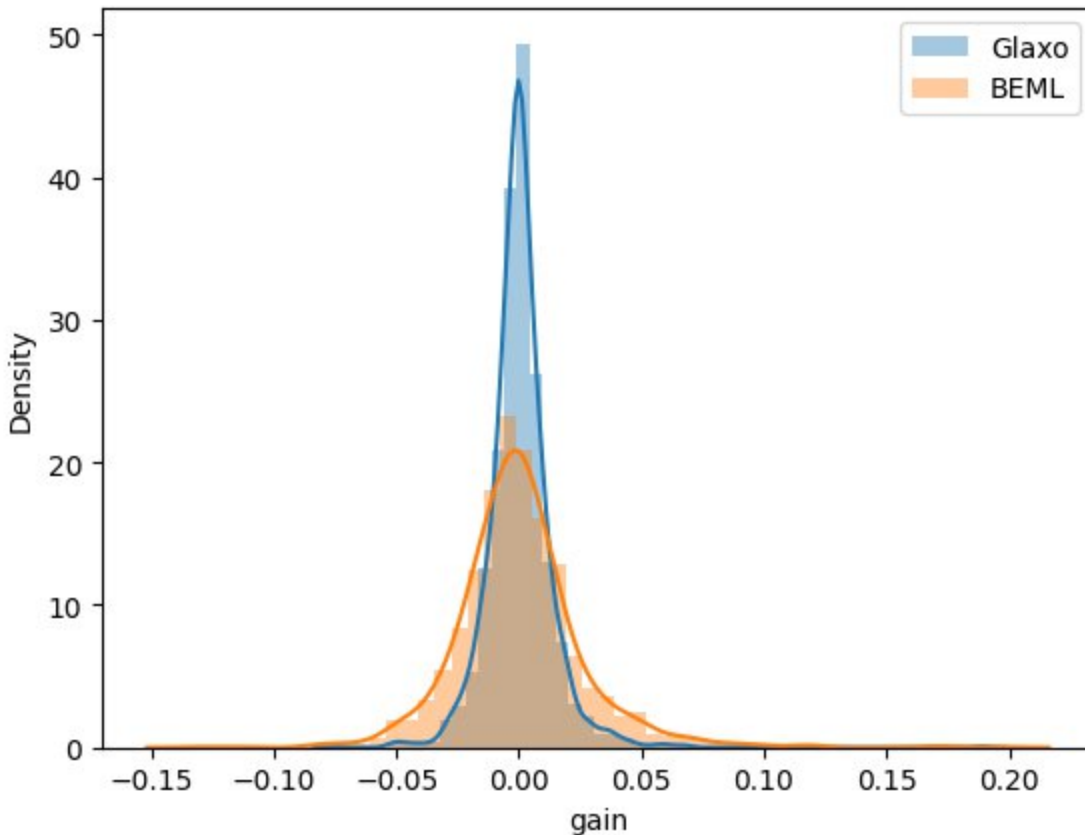The dataset includes the following variables for each stock:

- Date
- Open Price
- Close Price
- Daily High and Low prices
- Total Trade Quantity
- Turnover (Lacs)

Result

Conclusion

Based on the analysis performed in the notebook, here is a small conclusion:

The notebook demonstrates the application of the Normal Distribution (Gaussian distribution) in a financial context by evaluating the risks and returns of BEML and GLAXO stocks from 2010 to 2016. By calculating the daily rate of return (gain) from daily close prices, the analysis provides a way to quantify stock performance beyond simple price trends.

Key takeaways include:

- Risk and Volatility: The use of a bell curve allows for the visualization of naturally occurring measures in financial data,

helping investors understand the probability of various return outcomes.

- Comparative Performance: The expected daily rate of return for these stocks is approximately 0.000386 for GLAXO and 0.000271 for BEML, with BEML showing slightly higher volatility (standard deviation of 0.026431 vs. 0.013361 for GLAXO).
- Practical Utility: This statistical approach transforms raw trading data into actionable insights, enabling investors to assess stock stability and potential gains before making investment decisions.