Distribution

$$\sigma^2 = \frac{\sum_{i=1}^{N} (x_i - \mu)^2}{N}$$

$$\mu = \frac{\sum_{i=1}^{N} x_i}{N}$$

$$s^2 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n-1}$$

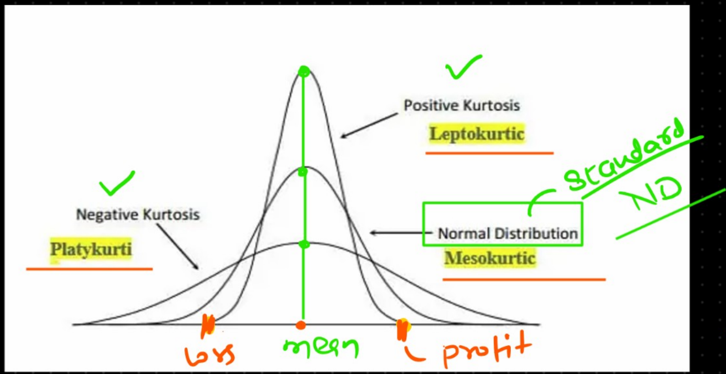
$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

→ Reduce the biasness → Bessel's correction

$$s^2 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n}$$

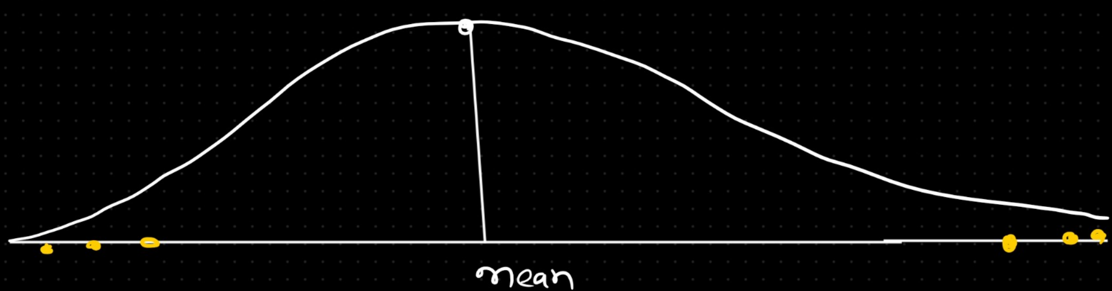
Sample data evaluated value (Mean, SD, Variance) $\approx$ Population data

Kurtosis → Measure the tailedness of the data distribution



Heavy tail & sharp peak → Lepto

Light tail & flatten peak → Platy



Lepto

High Kurtosis → High risk of experiencing significant gains or losses.

Risky

Platy

Low Kurtosis → lower likelihood of extreme returns

Stable

Random Variables

0.5 (1/2) — 36°C

Discrete (finite) → PMF

Continuous (specific range) → PDF

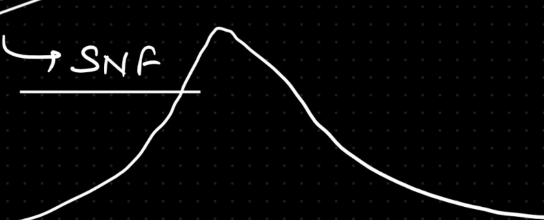
Probability Distribution Function (PDF)

- (PMF) Probability Mass function
  - Bernoulli Distribution → True/false
  - Binomial Distribution
  - Poisson Distribution
- (PDF) Probability Density function
  1. Gamma Distribution (Right skewed graph)
  2. Normal Distribution

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

$\mu = 0, \ \sigma = 1$

Standard → SNF



```
np.random.seed(42)

# standard normal distribution
portfolio_returns = np.random.normal(loc=0, scale=1, size=1000)

# adding the extreme values
portfolio_returns = np.random.normal(loc=0, scale=50, size=50)

# task1: can you add a few samples in such a way that it will become postive kurtosis


# calculate the kurtosis
from scipy.stats import kurtosis
kurt_value = kurtosis(portfolio_returns)
print(f"Kurtosis: {kurt_value:.2f}")

sns.histplot(portfolio_returns, bins=30, kde=True)
plt.title(f'Portfolio Returns Distribution (Kurtosis: {kurt_value:.2f})')
plt.xlabel('Returns')
plt.ylabel('Frequency')
plt.show()
```

Normal Distribution

samples

loc → mean

scale → standard deviation