



**Track to the Future!**

**FinTech**  
Lesson 5.3



# Class Objectives

---

By the end of today's class, you will be able to:



Define what a simulation is and why it's used.



Deconstruct the Monte Carlo simulation process: probability distributions and iterations.



Interpret probability distributions (normal, bell curve) and random number generators.



Comprehend the use of confidence intervals and what they suggest.



Implement a single Monte Carlo simulation on the price trajectory of a stock.



Execute multiple Monte Carlo simulations on the price trajectories of a stock.



Break down portfolio forecasting: Monte Carlo simulations on stock price & portfolio returns.

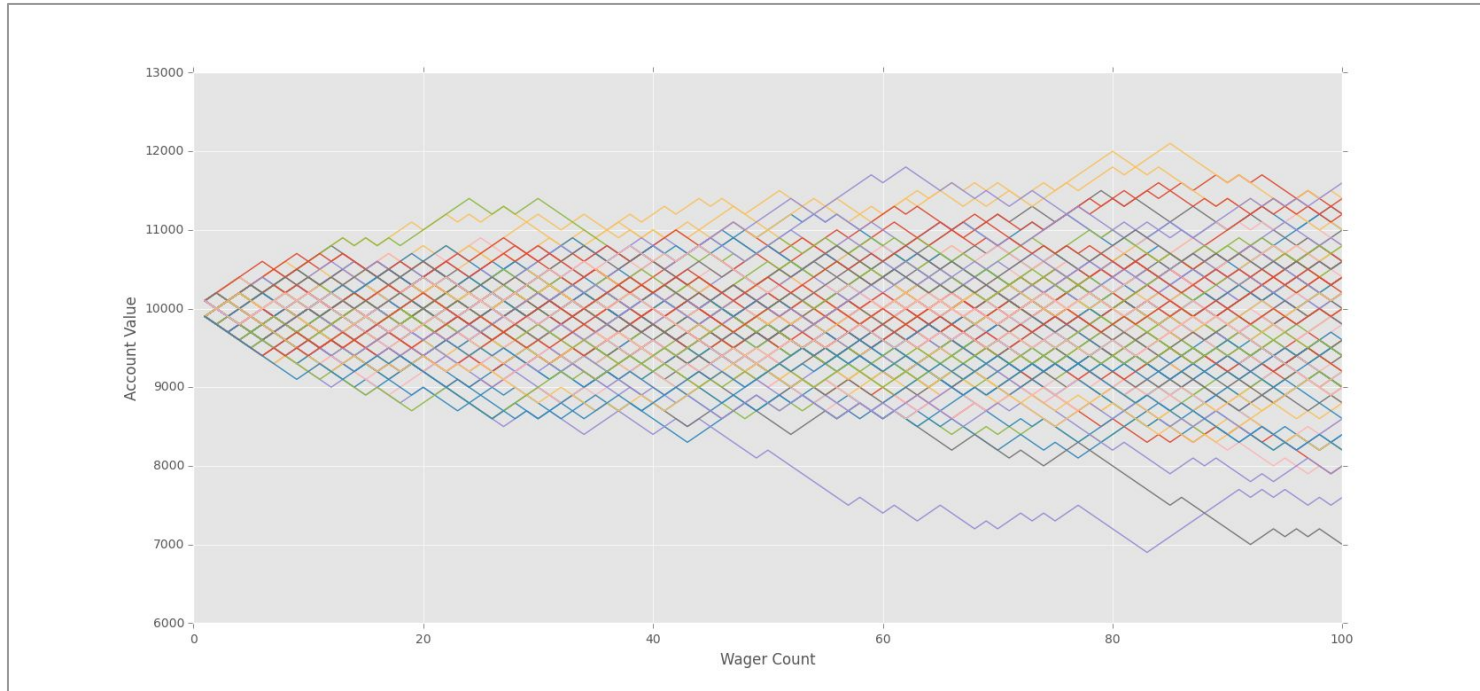


Implement multiple Monte Carlo simulations on the potential returns of a stock portfolio.

# Monte Carlo Simulations

# Monte Carlo Simulations

Today we will combine what we've learned so far on using APIs to pull in stock data and forecast single stock/portfolio returns using Monte Carlo simulations.



# Monte Carlo Simulations

Simulations will require a switch from historical analysis to predicting the future.

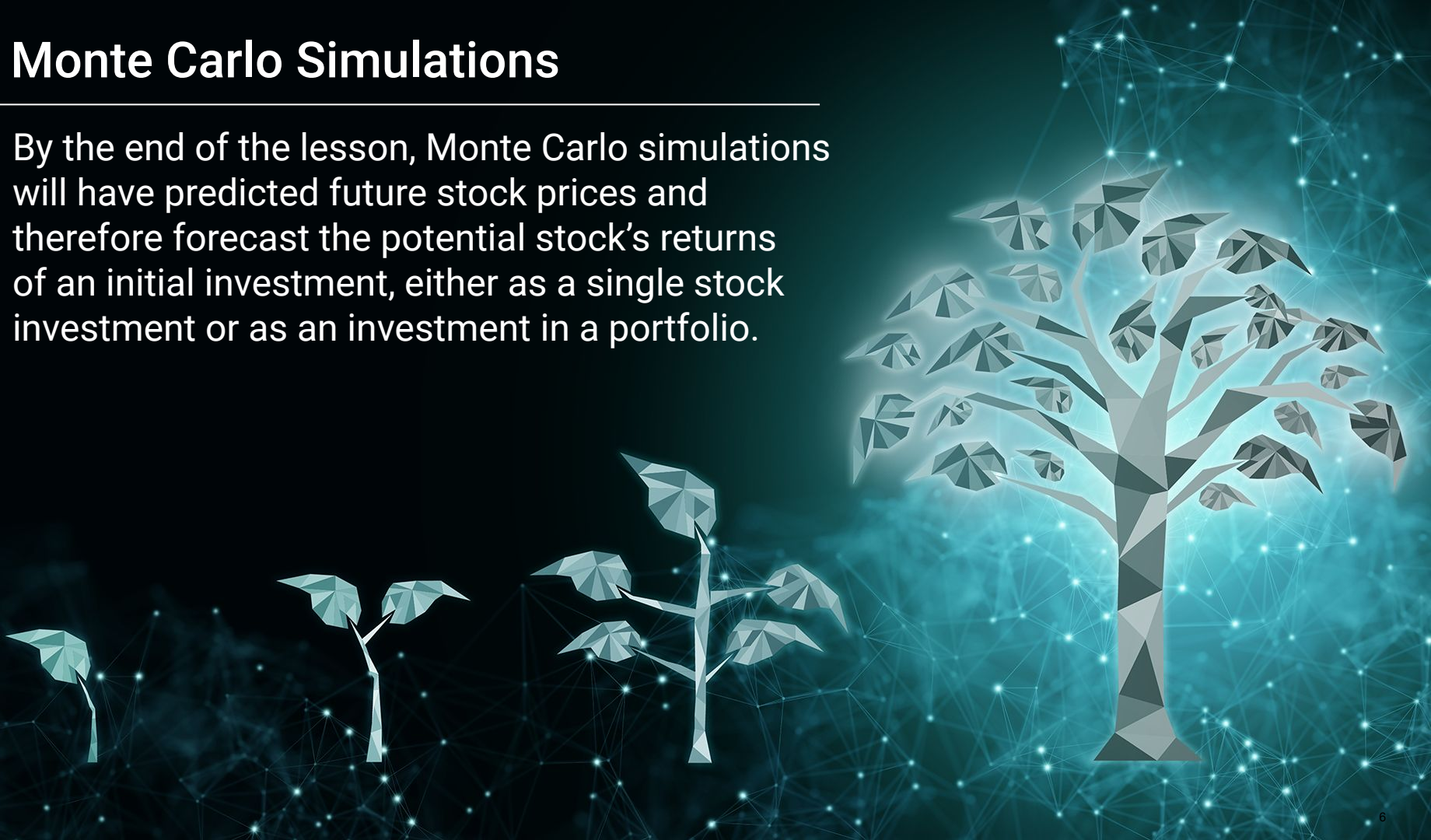




# Monte Carlo Simulations

---

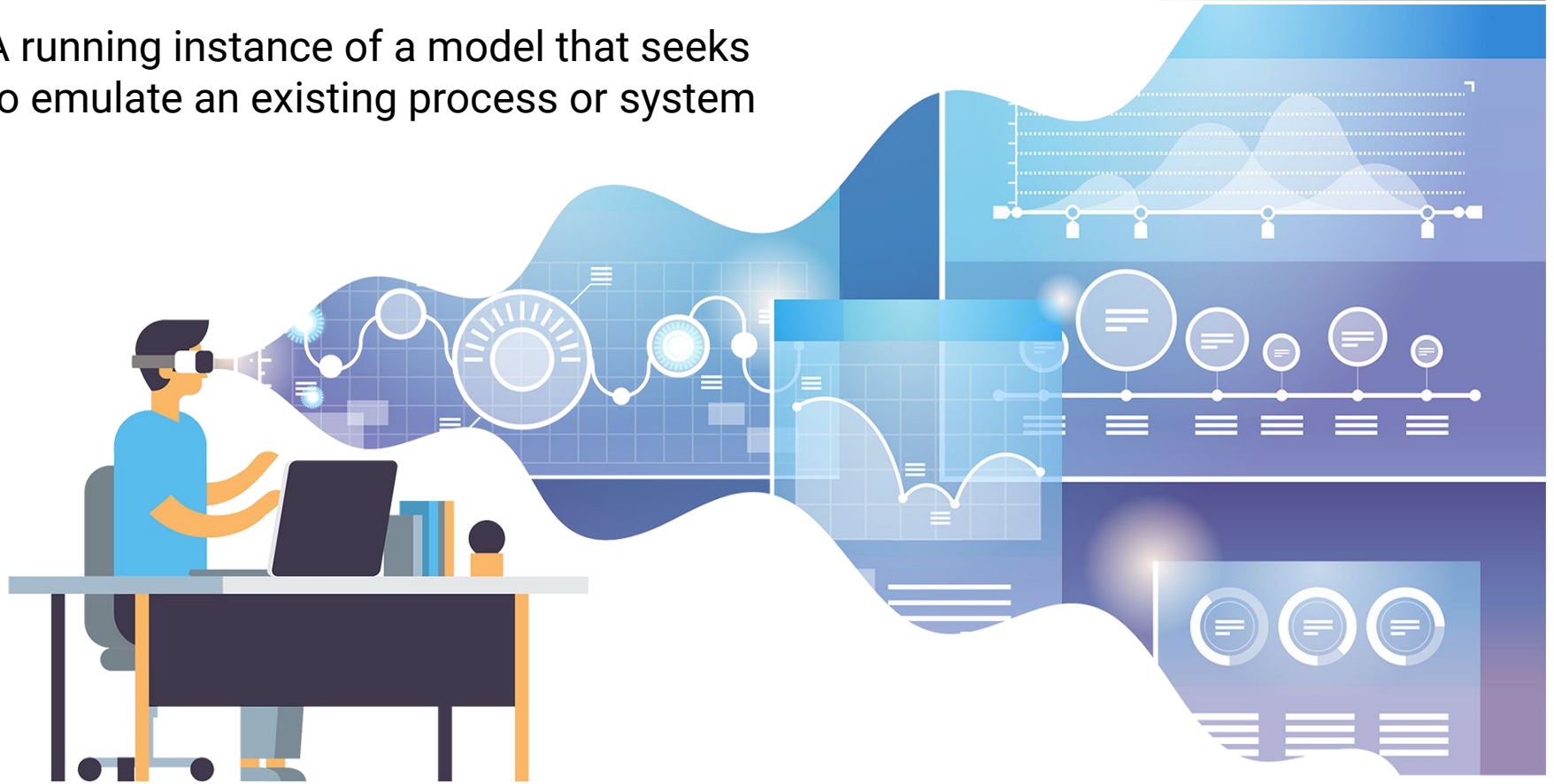
By the end of the lesson, Monte Carlo simulations will have predicted future stock prices and therefore forecast the potential stock's returns of an initial investment, either as a single stock investment or as an investment in a portfolio.



# Simulations

# What are simulations?

A running instance of a model that seeks to emulate an existing process or system

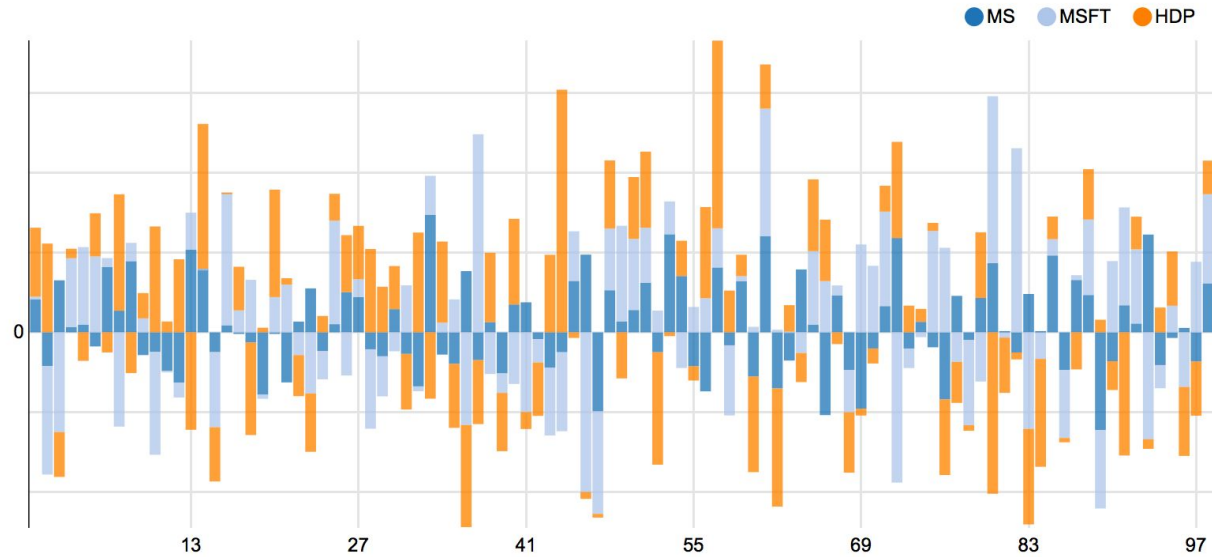




# What are Monte Carlo simulations?

Simulations that use probability and variables to predict the future potential outcomes of a randomly occurring process

Daily Value At Risk ( 1 Simulation for 100 future days)



VaR Percentiles

percentile	outcome
-1.87%	worst
0.25%	typical
2.15%	best

# Why use Monte Carlo simulations?

---

They help make sense of the risk of uncertainty in prediction and forecasting models, which are particularly helpful when dabbling in the domain of capital investments and stock price uncertainty



# Understanding Probability and Probability Distributions

# Grasping Probability

---

Imagine you are a scientist who wants to know how often a coin could land on heads for five trials of ten coin flips. Flipping a coin has a 50% chance of landing on heads and a 50% chance of landing on tails.



# Grasping Probability

---

Because of the randomly occurring nature of flipping a coin, results could vary: for example, a coin could produce 6 heads and 4 tails; 3 heads and 7 tails; 8 heads and 2 tails, 5 heads and 5 tails, or 4 heads and 6 tails.

Probability is the chance of an event happening, in this example, having head or tail.



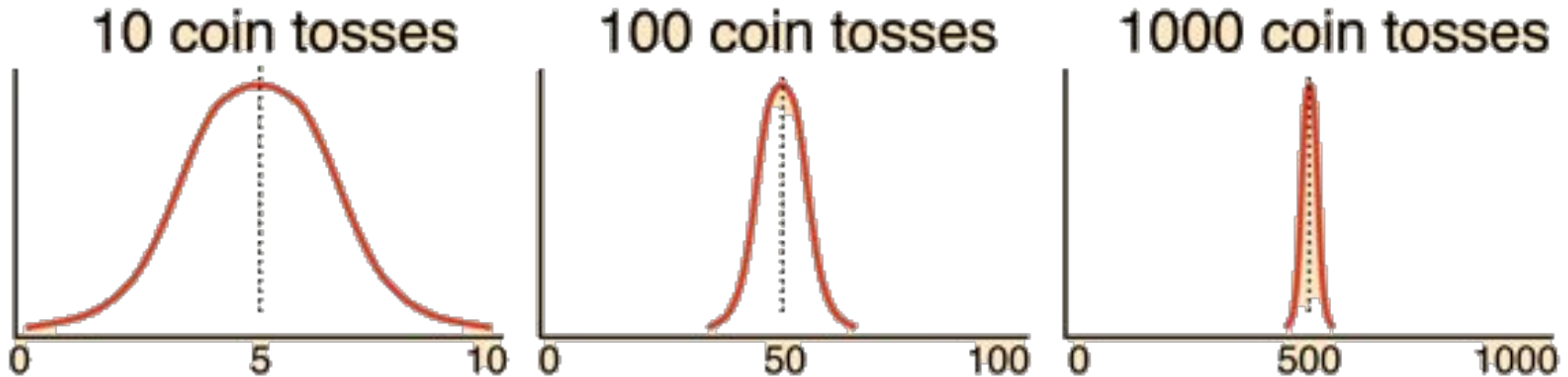


# Probability Distribution

---

A probability distribution is a mathematical function that describes the likelihood of possible outcomes for a given range of values.

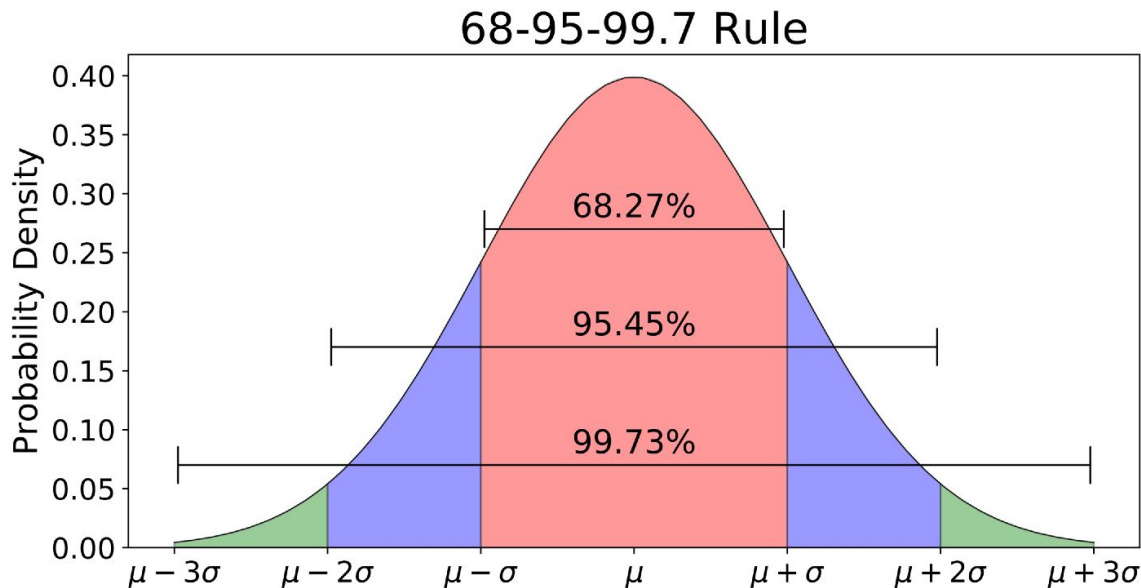
For example we can define a function to calculate the likelihood of getting 7 heads on 10, 100, or 1000 coin flips.





# Normal Probability Distributions

These distributions showcase the various probabilities of returning a value based on the number of standard deviations it is from the mean (how far the value may lie plus or minus from the average expected value).

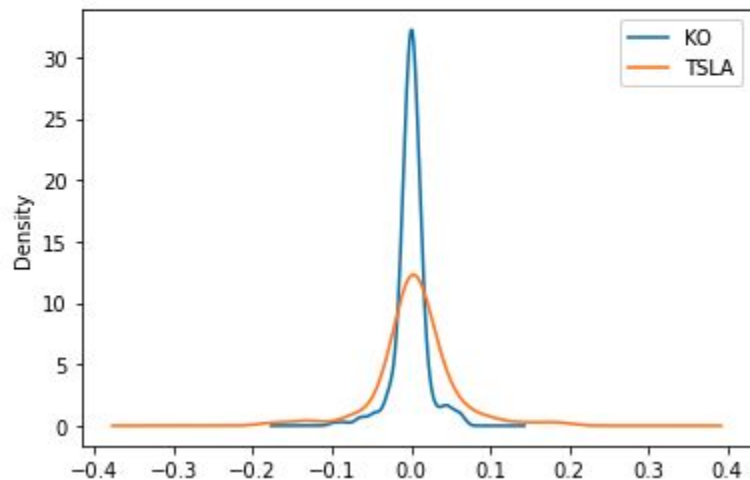


# Normal Distributions in Finance

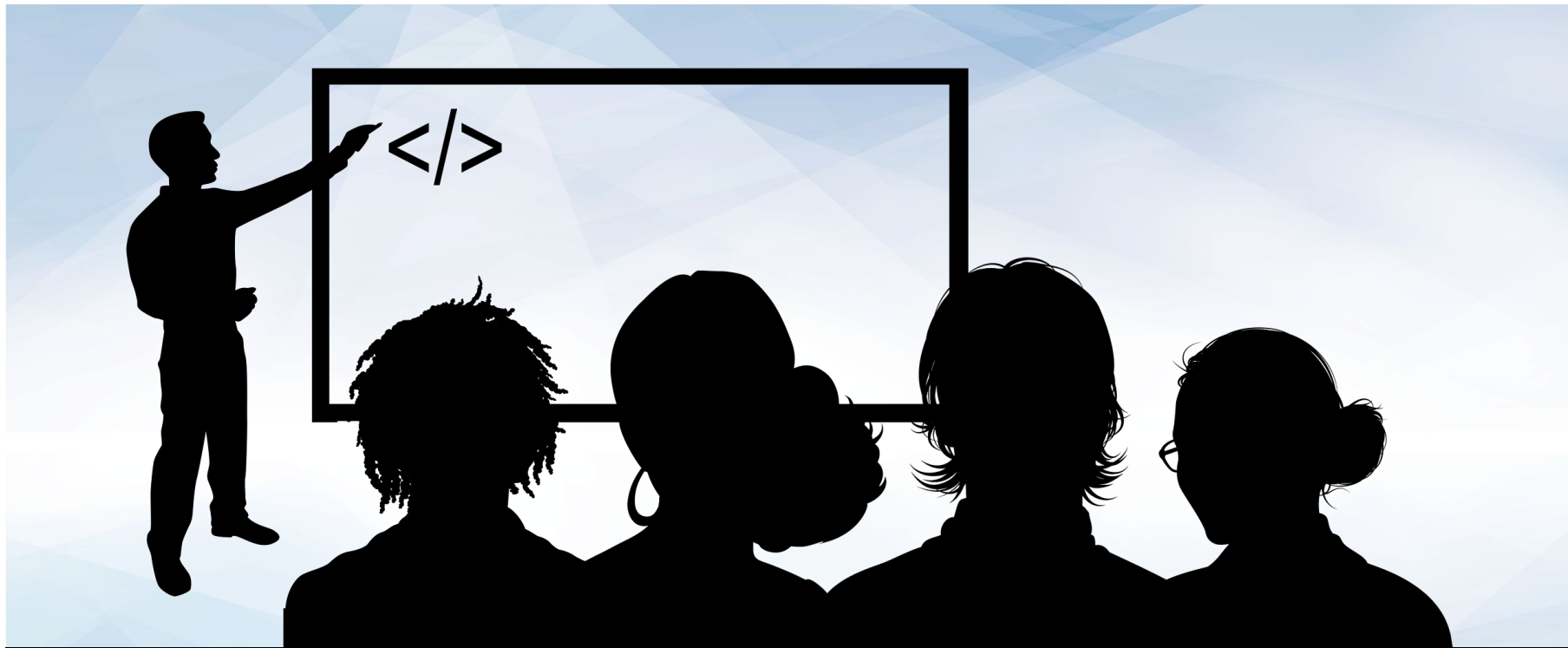
Normal distributions are particularly useful in finance because they adequately approximate the volatility of stock prices, forex rates and other commodities.

```
# Visualize the distribution of percent change in closing price for both stocks using a density plot  
df_daily_returns.plot.density()
```

<matplotlib.axes.\_subplots.AxesSubplot at 0x7fa475e35a50>



The daily price change (in percent) from a high volatility stock such as Tesla and a low volatility stock such as Coca-Cola can both demonstrate normal distributions.



# Instructor Demonstration

## Getting into Probability Distributions Using Python



## Activity: Decisive Distributions

In this activity you will gain hands-on experience fetching historical stock data and plotting distributions to make investment decisions.  
(Instructions sent via Slack.)

**Suggested Time:**  
20 minutes





**Time's Up!** Let's Review.

# Portfolio Forecasting Using Monte Carlo Simulations



# What is portfolio forecasting?

---

Portfolio forecasting is the process of projecting the future performance of a portfolio and attempting to analyze its most probable outcome.



# How is portfolio forecasting done?

Similar to the forecasting of a stock's price trajectory, Monte Carlo simulations are applied to forecast the potential price trajectories of the individual stocks that comprise the portfolio.



# Who is performing portfolio forecasting?

---

Portfolio managers, quantitative analysts, and retirement planners are just some of many who need to forecast the future performance of a portfolio to gauge the potential risk of investment.



# Implementing Monte Carlo Simulations in Python

We need two things, historical financial data from our portfolio to input into the simulation and a framework to run our Monte Carlo simulation.





# Instructor Demonstration

## Portfolio Forecasting









## Activity: Three Stock Monte

In this activity you will use the **MCForecastTools** toolkit to determine how much of each stock is worth to purchase in a portfolio in order to maximize your chances of profit.

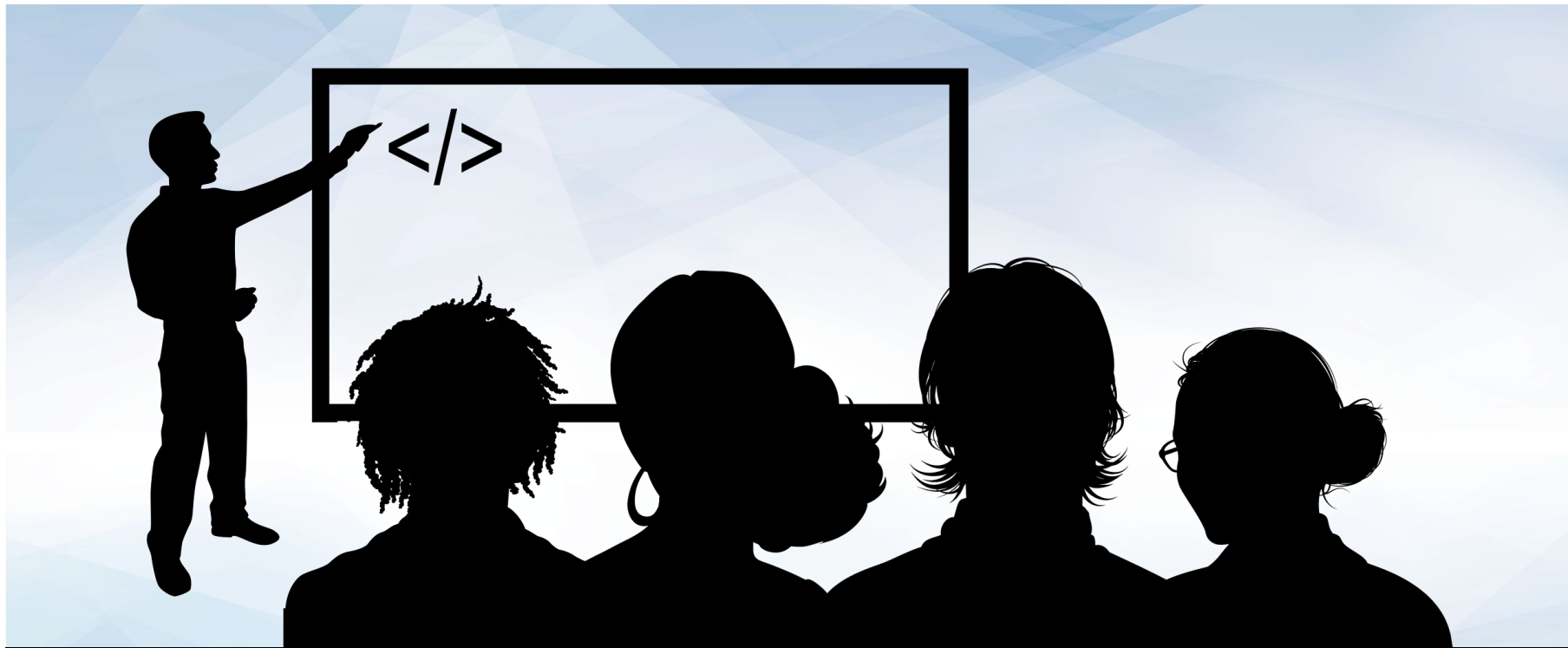
(Instructions sent via Slack.)

**Suggested Time:**  
30 min





**Time's Up!** Let's Review.



# Instructor Demonstration

## Simulation of Stock Price Trajectory



## Activity: Financial Forecasting

In this activity, you will execute a Monte Carlo simulation to forecast stock price behavior of historical `TSLA` daily returns.  
(Instructions sent via Slack.)

**Suggested Time:**  
15 minutes





**Time's Up!** Let's Review.



Questions?



*The  
End*