In this section, you'll apply the models to calculate the futures price, price a call option, and run Monte Carlo simulations. Here’s a step-by-step guide on how to approach each project, including where to find the data and what tools to use.

**Applying the Models**

**Cost of Carry Model**

**Project**: Calculate the futures price for a coffee contract maturing in six months using the current spot price, the risk-free interest rate, and the estimated storage costs.

**Steps**:

1. **Gather Data**:
   * **Spot Price**: Find the current spot price of coffee. You can obtain this from financial news websites like Bloomberg, Reuters, or commodity-specific sites like the International Coffee Organization (ICO).
   * **Risk-Free Rate**: Use the current yield on a six-month US Treasury bill as a proxy for the risk-free rate. This data can be found on financial websites like the US Department of the Treasury or FRED (Federal Reserve Economic Data).
   * **Storage Costs**: Estimate storage costs as a percentage of the spot price. This information might be available in commodity market reports or industry publications.
2. **Calculate Futures Price**:
   * Use the cost of carry formula:

A black text on a white background

Description automatically generated

* + Where:
    - Ft​: Futures price
    - St​: Spot price
    - r: Risk-free rate
    - d: Storage cost
    - T: Time to maturity (0.5 years for six months)

1. **Python Implementation**:

python

Copy code

import numpy as np

# Given values

S\_t = 1.20 # Spot price in dollars

r = 0.02 # Risk-free rate (2%)

d = 0.01 # Storage cost (1%)

T = 0.5 # Time to maturity in years

# Calculating futures price

F\_t = S\_t \* np.exp((r + d) \* T)

print(f"The fair price of the coffee futures contract is ${F\_t:.3f} per pound.")

**Black-Scholes Model**

**Project**: Price a call option on a coffee futures contract using the current spot price, strike price, risk-free rate, time to maturity, and volatility.

**Steps**:

1. **Gather Data**:
   * **Spot Price**: Use the same spot price data as for the futures contract.
   * **Strike Price**: Choose a strike price based on market conditions and option contracts available on exchanges like the ICE (Intercontinental Exchange).
   * **Risk-Free Rate**: Use the same risk-free rate as for the futures contract.
   * **Time to Maturity**: Specify the time to maturity (e.g., six months).
   * **Volatility**: Calculate or find the historical volatility of coffee prices. This data can be derived from historical price data available on financial websites or through databases like Yahoo Finance.
2. **Calculate Option Price**:
   * Use the Black-Scholes formula:



A math equation with a number and a plus

Description automatically generated with medium confidence

A black text with a white background

Description automatically generated

* + Where:
    - C: Call option price
    - S0​: Spot price
    - X: Strike price
    - r: Risk-free rate
    - T: Time to maturity
    - σ: Volatility

1. **Python Implementation**:

python

Copy code

from scipy.stats import norm

import numpy as np

# Given values

S\_0 = 1.20 # Spot price in dollars

X = 1.25 # Strike price in dollars

r = 0.02 # Risk-free rate (2%)

T = 0.5 # Time to maturity in years

sigma = 0.25 # Volatility (25%)

# Calculating d1 and d2

d1 = (np.log(S\_0 / X) + (r + 0.5 \* sigma \*\* 2) \* T) / (sigma \* np.sqrt(T))

d2 = d1 - sigma \* np.sqrt(T)

# Calculating call option price using Black-Scholes formula

C = S\_0 \* norm.cdf(d1) - X \* np.exp(-r \* T) \* norm.cdf(d2)

print(f"The price of the call option is ${C:.3f}.")

**Monte Carlo Simulation**

**Project**: Run simulations to forecast the price of coffee futures under different market conditions. Evaluate the impact of varying supply, demand, and weather scenarios.

**Steps**:

1. **Gather Data**:
   * **Initial Price**: Use the same spot price data.
   * **Risk-Free Rate**: Use the same risk-free rate.
   * **Volatility**: Use the same volatility data.
   * **Simulated Period**: Define the time period for the simulation (e.g., six months).
2. **Run Simulations**:
   * Use Monte Carlo simulations to generate numerous possible future price paths based on the inputs.
   * Model the stochastic process of coffee prices using Geometric Brownian Motion (GBM).
3. **Python Implementation**:

python

Copy code

import numpy as np

# Simulation parameters

S\_0 = 1.20 # Spot price in dollars

r = 0.02 # Risk-free rate (2%)

sigma = 0.25 # Volatility (25%)

T = 0.5 # Time to maturity in years

num\_simulations = 10000 # Number of simulations

num\_steps = 252 # Number of steps (daily)

# Time increment

dt = T / num\_steps

# Simulating price paths

np.random.seed(42) # For reproducibility

price\_paths = np.zeros((num\_steps, num\_simulations))

price\_paths[0] = S\_0

for t in range(1, num\_steps):

z = np.random.standard\_normal(num\_simulations)

price\_paths[t] = price\_paths[t-1] \* np.exp((r - 0.5 \* sigma \*\* 2) \* dt + sigma \* np.sqrt(dt) \* z)

# Calculating the average simulated price at maturity

average\_simulated\_price = np.mean(price\_paths[-1])

print(f"The average simulated price of the coffee futures contract at maturity is ${average\_simulated\_price:.3f}.")

**Data Sources and Tools**

1. **Data Sources**:
   * **Spot Prices**: Bloomberg, Reuters, International Coffee Organization (ICO)
   * **Risk-Free Rates**: US Department of the Treasury, FRED (Federal Reserve Economic Data)
   * **Storage Costs**: Commodity market reports, industry publications
   * **Volatility Data**: Yahoo Finance, historical price databases
2. **Tools**:
   * **Python**: Use Python for data analysis and simulations. Libraries like NumPy, SciPy, and pandas are essential for financial modeling.
   * **Spreadsheet**: Use Excel for initial data gathering and simple calculations.

By following these steps and utilizing the provided data and tools, you will be able to apply quantitative models to price coffee futures contracts, structure options, and run simulations effectively. This hands-on approach will help you develop practical skills in quantitative analysis and financial modeling.