



Sigma Internship Coding Challenge

Goal

This exercise will help familiarize you with the Quantrocket platform and also test your coding & algorithms knowledge while giving you a flavor of how to set up a minimalist trading workflow.

Getting Started

1. Install [Quantrocket](#) on your local or on cloud of your choice. Go through a few [quick tutorials](#) to understand how to use the basic capabilities of the platform.
2. Ensure you are able to pull price data (daily close prices only) for Apple stock (symbol='AAPL') for the year 2023 (01-01-2023 to 12-31-2023). This should be available as part of their freely available us-stock price data.

Daily Stock Trading Prompt

Objective is to build a simple model to make decisions on certain days using the below pre-specified logic and publish the output (along with the code).

Let for a day d in $[1, 2, \dots, N]$, $p(d)$ represent the close price of that day. We want to build a model to decide whether to place a buy order trade for the day $d+1$ to maximize the portfolio value. Let $r(d)$ be the % returns on day d . That is, $r(d) = (p(d) - p(d-1))/p(d-1)$. Conduct the following state classification:

if $r(d) \geq 0.01$, $s(d) = +1$
else if $r(d) > -0.01$, $s(d) = 0$
else, $s(d) = -1$

That is, depending on whether the returns on day d are high, medium or low, we classify the state as Bull (+1), Flat (0) or Bear (-1). Let's define a simplistic value function as below.

Assuming we decide to place a buy order trade for the day $d+1$,
if $s(d+1) = 1$ & $s(d) = 0$, then $V(d+1) = V(d) + 1$
else if $s(d+1) = -1$ & $s(d) = 0$, then $V(d+1) = V(d) - 1$
and $V(d+1) = V(d)$ in all other cases (including when we decide not to place a buy order trade for the day $d+1$). That is on day $d+1$, assuming we executed a buy, our portfolio value increases by 1 if the observed returns for $d+1$ is in the Bull state, decreases by 1 if it is in the Bear state, and stays unchanged for all other scenarios.

Now based on the previous observations $[1, \dots, d]$, you can calculate the probability distribution of going from the state $s(d)$ to different possible states.

Write the code to calculate the transition distribution in a streaming manner while making decisions on the optimal points where you'll place buy orders with the ultimate goal to maximize $V(N)$. Submit the value $V(N)$, the optimal buy indices & code in your github repository.

Feel free to join [Sigma Slack group](#) (it's an open group) and ping @Arpit Goel directly if you have any specific questions during the coding challenge duration.