QUANT Winter Camp
Assignment 1

December 9, 2023

Instructions

- Submission deadline is 11th EOD.
- You have to attempt a minimum of 2 problems.
- The submission would be on GitHub classroom as stated in the meet. Submit separate .ipynb files for the problems you have attempted and most importantly **RUN ALL THE CELLS BEFORE SUBMITTING**
- Plagiarism of any sort would not be tolerated and would have serious repercussions.
- We know what ChatGPT generates so keep that in mind before submitting
- Attach a document along with the submissions to explain how you tackled the questions.
- In case of any query please feel free to contact the mentors.

Question 1:

Use the csv file and import it into your Colab file or Jupyter notebook. This can be done using the following code:

```
import pandas as pd
data= pd.read_csv (" The file location")
```

The `data` variable is a Pandas DataFrame. Your job in this task is to add the below columns to this DataFrame:

- Add a column named `temp` which stores the value of % profit or loss, which one would gain if they buy the stock at open and sell at close.
- Add a column for daily returns.
- Add a column for the Sharpe ratio.

- Add columns for open/close, low/high.

Additionally, plot the `temp` column using Matplotlib.

Sort the `low/high` columns into 5 categories: $[0, 0.2)$, $[0.2, 0.4)$, [0.4, 0.6), [0.6, 0.8), and $[0.8, 1]$, and plot them on a histogram.

Question 2:

In this question, you will be implementing a trading strategy (a long-only strategy) based on stop loss and take profit. First of all, let's have some definitions:

- **Stop Loss:** If the price goes below this level, we square off the trade to limit the loss.
- **Take Profit:** If the price goes above this level, we square off the trade to take the profit before it turns into a loss.

Create two lists, one for storing the buy indices and the other for sell indices. To simplify the problem, you can only be in 1 trade at a time. The trades should be continuous. You have to append the 1st date in the buy list As soon as either the stop loss or take profit hits, you have to square off the trade and start the next trade from the next day. The stop loss and take profit are defined in ratio format applied to the buying price.

The problem requires you to create a function (All trades have to be taken on the closing price):

Input: dataframe from csv, take profit percent and stop loss percent
Output print buy and sell list

(Extra: Try to take trades based on these signals and compute the net returns).

Let's take an example:

Table 1: Stock Prices

Day	1	2	3	4	5	6	7
Close	1000	1020	1040	1050	1035	1030	1025

The closing price is as follows:

You entered the trade on day 1, with a stop-loss level set at 0.02, which means the stop loss for the 1st trade would be 980, and similarly, take profit is 1040.

$$\textit{Stop loss} = (1000 - 0.02 \times 1000) \quad \textit{and} \quad \textit{Take profit} = (1000 + 0.04 \times 1000)$$

The price reaches 1040 on day 3, so you exit the trade on day 3 with a profit of 40.

Similarly, you enter the trade on day 4, stop loss $= (1 - 0.02) \times 1050$ $= 1029$ and take profit $= (1 + 0.04) \times 1050 = 1092$.

The price fell below the stop-loss level on day 7, so you would exit the trade on that day with a loss of 25.

Hence, for these two trades, the buy list would be $[1, 4]$, while the sell list would be $[3, 7]$.

Question 3:

A number is called "awesome" if it can be split into 3 numbers (a, b, c)

such that $(a + b + c) = n$ and (sum of digits of a) + (
digits of b) + (sum of digits of c) = (sum of digits of n).
You are free to use any library to solve this problem.

E.g., for $26 = 4 + 12 + 10$ and $2 + 6 = (4) + (1 + 2) + (1 + 0)$, one such triple is $(4, 12, 10)$. This triple gives 3! triples as we can reorder these triples.

Implement a function:

The input to the function is an integer n. The output of the function is the number of triples possible.

Your function should output the following values for n:

Table 2: Results

Integer(n)	Output
11	9
0	1
1	3
2	6
3	10
4	15
5	21
3141	1350
999	166375
2718	29160
10000000	3

For 11, the 9 triples are:

$$\begin{pmatrix}0 & 0 & 11\end{pmatrix}$$
$$\begin{pmatrix}0 & 1 & 10\end{pmatrix}$$
$$\begin{pmatrix}0 & 10 & 1\end{pmatrix}$$
$$\begin{pmatrix}0 & 11 & 0\end{pmatrix}$$
$$\begin{pmatrix}1 & 0 & 10\end{pmatrix}$$
$$\begin{pmatrix}1 & 10 & 0\end{pmatrix}$$
$$\begin{pmatrix}10 & 0 & 1\end{pmatrix}$$
$$\begin{pmatrix}10 & 1 & 0\end{pmatrix}$$
$$\begin{pmatrix}11 & 0 & 0\end{pmatrix}$$