FE670 Homework Assignment #4

Due Date: Dec 7th (Thursday).

Problem 1. The Relative Strength Index is a momentum oscillator that is used by traders to identify whether the market is in the state of overbought or oversold. A market is considered to be in the state of overbought when an asset is constantly bought by traders moving it to an extremely bullish trend and bound to consolidate. Similarly, a market is considered to be in the state of oversold when an asset is constantly sold by traders moving it to a bearish trend and tends to bounce back.
As an oscillation market indicator, the values of RSI bound betyouen 0 to 100. The traditional way to evaluate a market state using the Relative Strength Index is that an RSI reading of 70 or above reveals a state of overbought, and similarly, an RSI reading of 30 or below represents the market is in the state of oversold. The standard setting of RSI is 14 as the lookback period.

RSI is determined with directional price moves during a given time period N (usually N=14 days).

$$RSI = 100.0 - (100.0/(1.0 + RS)), RS = n_{\text{up}}/n_{\text{down}} \tag{1}$$

n_{up} and n_{down} are the numbers of upward moves and downward moves of closing price, respectively.

Usually, these numbers are exponentially smoothed:

$$
\begin{aligned}
n_{\text{up}}(t) &= (1-\beta) * n_{\text{up}}(t-1) + \beta U(t), \\
n_{\text{down}}(t) &= (1-\beta) * n_{\text{down}}(t-1) + \beta D(t), \\
&\text{where} \\
U(t) &= 1, P_t > P_{t-1}; U(t) = 0, P_t \leq P_{t-1} \\
D(t) &= 1, P_{t-1} > P_t; D(t) = 0, P_{t-1} \leq P_t \qquad (2)
\end{aligned}
$$

There are three steps involved in the calculation of RSI.

1) Calculating the Exponential Moving Average (EMA) of the gain and loss of an asset: A word on Exponential Moving Average. EMA is a type of Moving Average (MA) that automatically allocates greater weighting (nothing but importance) to the most recent data point and lesser weighting to data points in the distant past. In this step, we will first calculate the returns of the asset and separate the gains from losses. Using these separated values, the two EMAs for a specified number of periods are calculated.

2) Calculating the Relative Strength of an asset: The Relative Strength of an asset is determined by dividing the Exponential Moving Average of the gain of an asset from the Exponential Moving Average of the loss of an asset for a specified number of periods. It can be mathematically represented in equation (2).

3) Calculating the RSI values: In this step, we will calculate the RSI itself by making use of the Relative Strength values we calculated in the previous step. To calculate the values of RSI of a given asset for a specified number of periods, there is a formula that we need to follow equation (1).

Our strategy reveals a buy signal whenever the previous RSI value is above the oversold level and the current RSI value crosses below the oversold level. Likewise, the strategy reveals a sell signal whenever the previous RSI value is below the oversold level and the current RSI value crosses above the oversold level. Our trading strategy can be represented as follows:

$$\begin{aligned} \text{If RSI}(t-1) > 30\&\ \text{RSI}(t) < 30, \Rightarrow \text{BUY} \\ \text{If RSI}(t-1) < 70\&\ \text{RSI}(t) > 70, \Rightarrow \text{SELL} \end{aligned} \tag{3}$$

Please use the dataset "SP20-2017-2021.csv" and select the stock with the highest volatility, and design a RSI based trading strategy and following the following steps:

(1) You are going to calculate the values of RSI with 14 as the lookback period using the RSI formula we discussed before. Note: you are defining a function named get_rsi that takes the closing price of a stock *ticker* and the lookback period *lookback* as parameters. Inside the function, you are first calculating the returns of the stock using the $diff$ function provided by the Pandas package and stored it into the *ret*

variable. This function basically subtracts the current value from the previous value. Next, you are passing a for-loop on the *ret* variable to distinguish gains from losses and append those values to the concerning variable (*up* or *down*).

(2) You are going to plot the calculated Relative Strength Index values of the selected stock to make more sense out of it. The main aim of this part is not on the coding section but instead to observe the plot to gain a solid understanding of RSI. Please plot the price and RSI on the same graph with different colors.

(3) You are going to implement the discussed Relative Strength Index trading strategy in python. Note: First, you are defining a function named *implement_rsi_strategy* which takes the stock prices (*prices*), and the RSI values (*rsi*) as parameters. Inside the function, you are creating three empty lists (*buy_price*, *sell_price*, and *rsi_signal*) in which the values will be appended while creating the trading strategy. Finally, you are returning the lists appended with values. Then, you are calling the created function and stored the values into their respective variables.

(4) You are going to plot the created trading lists to make sense out of them. You are plotting the Relative Strenght Index values along with the buy and sell signals generated by the

trading strategy. You can observe that whenever the RSI line crosses from above to below the lower band or the oversold level, a green-colored buy signal is plotted in the chart. Similarly, the RSI line crosses from below to above the upper band or the overbought level, a red-colored sell signal is plotted in the chart.

(5) You are going to create a list that indicates 1 if we hold the stock or 0 if we dont own or hold the stock. First, you are creating an empty list named position. You are passing two for-loops, one is to generate values for the position list to just match the length of the signal list. The other for-loop is the one we are using to generate actual position values. Finally, you are doing some data manipulations to combine all the created lists into one dataframe.

Problem 2. Price action trading is a systematic trading practice, aided by technical analysis tools and recent price history, where traders are free to take their own decisions within a given scenario to take trading positions, as per their subjective, behavioral and psychological state. Since price action trading is an approach to price predictions and speculation, it is used by retail traders, speculators, arbitrageurs and even trading firms who employ traders. It can be used on a wide range of securities including equities, bonds, forex, commodities, derivatives, etc.

Most scenarios involve a two-step process:

1. Identifying a scenario: Like a stock price getting into a bull/bear phase, channel range, breakout, etc.
2. Within the scenario, identifying trading opportunities: Like once a stock is in bull run, is it likely to (a) overshoot or (b) retreat. This is a completely subjective choice and can vary from one trader to the other, even given the same identical scenario.

The random forest algorithm expands on the randomization introduced by the bootstrap samples generated by bagging to reduce variance further and improve predictive performance. In addition to training each ensemble member on bootstrapped training data, random forests also randomly sample from the features used in the model (without replacement). Depending on the implementation, the random samples can be drawn for each tree or each split. As a result, the algorithm faces different options when learning new rules, either at the level of a tree or for each split.

You are given a dataset for the 10 stock which consists of the adjusted closing price of each of the 10 stocks. These stock tickers are . 'GOOGL', 'TSLA', 'FB', 'AMZN', 'AAPL', 'MSFT', 'VOD', 'ADBE', 'NVDA', 'CRM'. This dataset includes all data from January 01, 2015 to December 31, 2020 in "all_stocks_10_2015_2020.csv". You are asked to design a trading strategy based on the Random

Forest classififation algorithm to trade on all these ten stocks and use Sharpe ratio as your trading performance. You need to traing your model using the first 4 years of data and then evaluate the performance of your trading strategy using 2020 data. You are to use Python pandas.ta package to use generate set of following features:

1. EMA14: The Exponential Moving Average is more responsive moving average compared to the Simple Moving Average (SMA). Length = 14

2. EMA22: The Exponential Moving Average is more responsive moving average compared to the Simple Moving Average (SMA). Length =22

3. RSI: The Relative Strength Index is popular momentum oscillator used to measure the velocity as well as the magnitude of directional price movements.

4. PPO: The Percentage Price Oscillator is similar to MACD in measuring momentum.

5. MACD: The MACD is a popular indicator to that is used to identify a security's trend.

6. CMO: Attempts to capture the momentum of an asset with overbought at 50 and oversold at -50.

7. RETURN: The current return defined as log price difference.

(1) Design a simple trading rule:when $p_t > p_{t-1}$ and then buy; otherwise sell. Use the Random Forest algorihtm to build a classifier on the 7 features to forecast the next day price trend. Use the same rule to trade on the 10 stocks, and select a highest performing stock. Note: when compare the performance, use Sharpe ratio (mean/std) for the testing period.

(2) Now assume an updated trading rule: when $\frac{p_t - p_{t-1}}{p_{t-1}} \geq \delta$, and $\delta = 0.10$ and then buy; otherwise sell. From the last step find the highest performing stock, and change δ in different values such as 2%, 5%, 12% and 15%, and re-evaluate the strategy to recommemd a better trading strategy. Again, please use out-of-sample Sharpe ratio to measure strategy performance.

Homework Honor Policy: You are allowed to discuss the problems between yourselves, but once you begin writing up your solution, you must do so independently, and cannot show one another any parts of your written solutions.