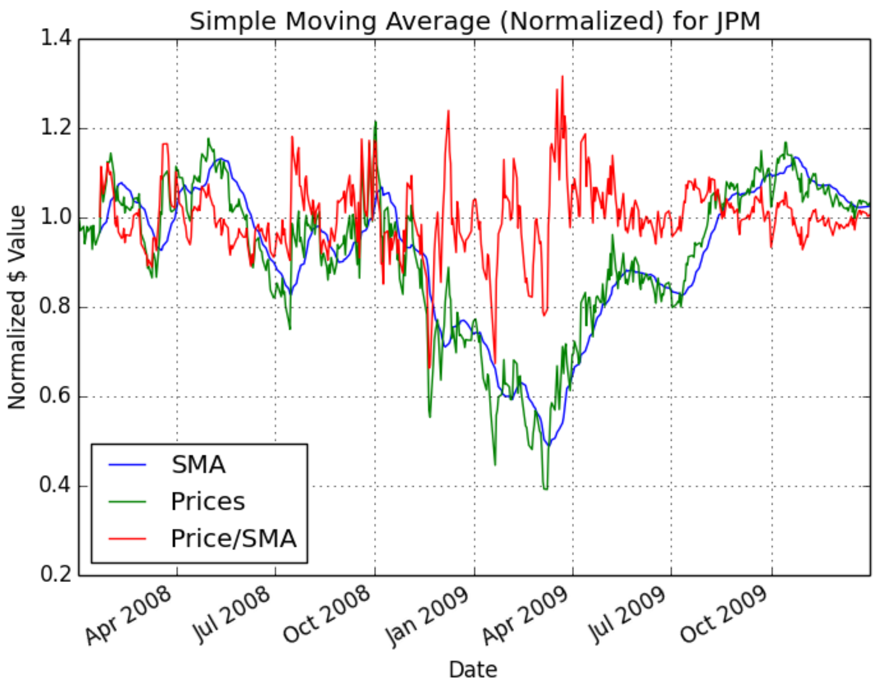
Corey Caskey
CS 4646/7646: Machine Learning for Trading
Homework 6: Manual Strategy

Part 1

For my report, I focused on the financial indicators of Simple Moving Average (SMA), Bollinger Band Percentage (%), and Moving Average Convergence Divergence (MACD). Both the SMA and Bollinger Band % indicators used a 14-day lookback period, while the MACD relied on both a 26-day and 12-day lookback for its calculation. This assignment focused on JP Morgan (JPM) stock data – all subsequent charts and data are related to this company.



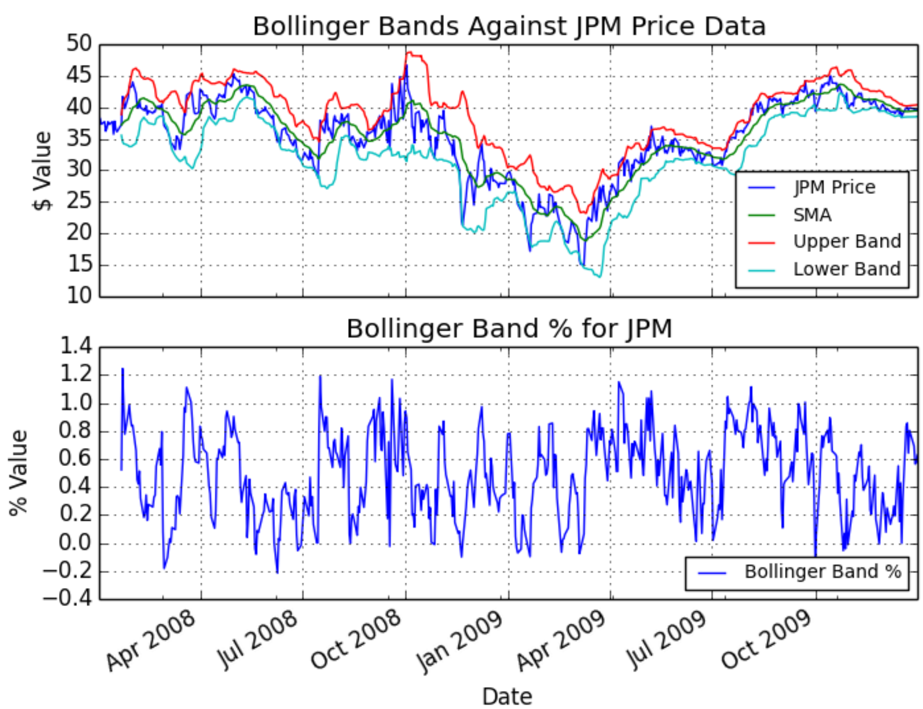
The first indicator I worked with was the Simple Moving Average (SMA). SMA is useful for identifying potential trends (e.g. uptrend or downtrend) in certain stocks by averaging out a window of adjusted close prices and smoothing out volatility. It is calculated by capturing prices of the previous *n* days for a stock, where *n* is the lookback period, and finding the unweighted mean of that summation.

The mathematical formula is as follows:

$$P_n = \frac{1}{n}\sum_{i=0}^{n-1} P_i$$

Here, P_n represents the average calculated for the n^{th} day, and P_i is the adjusted close price for some i^{th} day in the lookback period. The SMA computation applies this formula repeatedly for all trading dates that come before the end date but after the n^{th} day from the start date. When converting this to code, we can rely on pandas' *rolling* function to capture the repeated calculation over all applicable dates in the price dataframe (i.e. to create the line) using a window and minimum period size equal to the lookback value and then apply the *mean* function to that result to calculate the average value.

As seen in the graph above, the SMA line is smoother than the price data because it averages out the volatile fluctuations over the lookback period. The ratio between the price and SMA lines is important in identifying signals to buy or sell shares in a stock – this is depicted by the Price/SMA line in the chart above. Generally, if the ratio is above one (i.e. the price line is above the SMA line), it's a good indication that this stock is overbought and that it would be wise to sell. Conversely, if the ratio is below one (i.e. the SMA line is above the price line), it's a good indication that this stock is oversold and that it would be wise to buy.

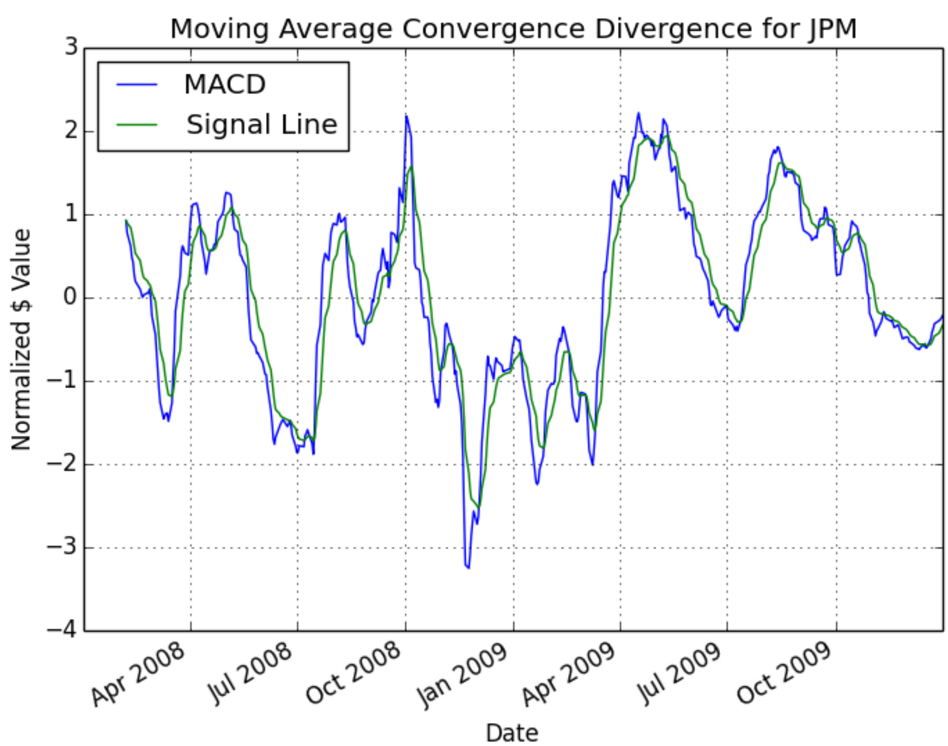


The next financial indicator I used was the Bollinger Band %. Bollinger Bands are useful indicators of stock volatility – relying on an upper band that is two standard deviations above the SMA line and a lower band that is two standard deviations below the SMA line. This is depicted in the top subplot of the above chart by the red, green, and light blue lines. An important feature of Bollinger Bands is the distance between the upper and lower bands. When the bands are close together, it's called a "squeeze" and is marked by a period of low volatility in the stock price, which could be an indication of increased volatility in the future and more trading opportunities. On the other hand, when the bands are far apart, the stock price line is marked by more fluctuation, leading to a potential decrease in volatility in the future and a possibility for trade exiting.

Using this knowledge, we can capture the Bollinger Band % value – a quantification of where the stock price is relative to the surrounding Bollinger Bands. This computation relies on the SMA methodology we previously implemented. After capturing the SMA data for the same lookback period as the Bollinger Band % – in my case, 14 days – we can proceed to finding the standard deviation over the prices dataframe between the start and end dates. This process is similar to the SMA process where we rely on the *rolling* function, except now we use the *standard deviation* function on the result. Once this is computed, we can calculate the Bollinger Bands by simply multiplying the above calculation by two and then adding to and subtracting from the SMA dataframe to get the upper and lower bands, respectively. To calculate the actual Bollinger Band % value, we subtract the bottom band values from the price values and then divide by the difference between the upper and lower bands. The equation for this would be:

(prices – bottom band) / (top band – bottom band)

Generally, if the Bollinger Band % is above one (i.e. the stock price is above the upper band), it's a good indication that the stock has been overbought and that you should sell. Conversely, if the percentage is below one (i.e. the stock price is below the lower band), the stock has likely been oversold, indicating a good time to buy shares.



The last financial indicator I used was Moving Average Convergence Divergence (MACD). It is useful for indicating trends in as well as the momentum of stock price data by showing the relationship between two moving averages. The MACD line is calculated by taking a 12-day exponential moving average (EMA) and subtracting it by a 26-day EMA for that stock. EMA is similar to SMA except that it assigns different weights to lookback prices, with more weight associated with more recent prices, as described here:

$$[\textbf{closing price} - \textbf{EMA (previous day)}] \textbf{ x weight} + \textbf{EMA (previous day)}$$

where

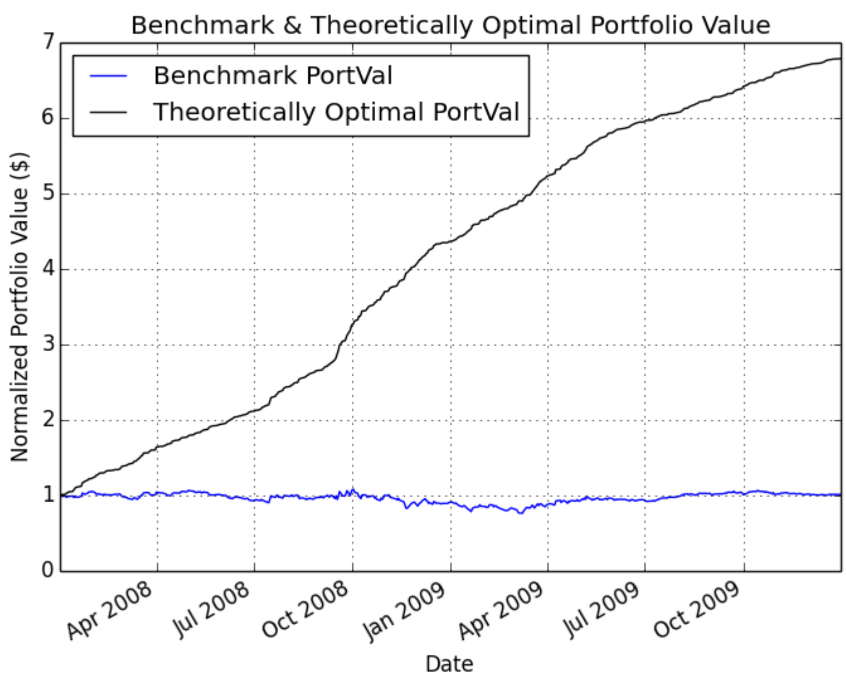
$$\mathbf{weight} = \frac{\mathbf{2}}{(\mathbf{lookback} + \mathbf{1})}$$

This same process can be done in code by taking advantage of pandas' *ewm* function, which accounts for the rolling nature of stock price data (i.e. calculating over all prices), and then applying the *mean* function. In this function, the span and minimum periods are equivalent to the size of the lookback.

The MACD line moves above and below the zero line. A positive MACD value means the 12-day EMA is larger than the 26-day EMA and this stock is experiencing upside momentum. Conversely, a negative MACD value means the opposite – the 26-day EMA is larger than the 12-day EMA, and this stock is experiencing downside momentum. To supplement this data, a 9-day EMA of the MACD line, called the signal line, is added to provide trading signals. This EMA is calculated similarly to the other EMAs except it is done so over the MACD data and uses a minimum period of one. When the MACD falls below this signal line, it's an indication that it may be time to sell. When the MACD rises above this signal line, it's an indication that the price is experiencing upward momentum and that it's time to buy.

Part 2

In this section, we performed a simulation of two different trading strategies: a benchmark approach and a theoretically optimal approach. For the benchmark approach, we bought and held 1000 shares of JPM stock for the entire duration of the date range. The shape of this line, while difficult to see because of the relative scale of the other line, mimics the same shape as the actual JPM price line because we maintain a constant holding position throughout. With the theoretically optimal approach, we assumed an ability to see into the future and know what the price will be for the next day when considering a buying/selling action for the current day. Furthermore, we assumed both commission and impact were 0.0. As a result, we made optimal decisions to maximize our cumulative return.



Due to restrictions, we could have holding positions equal to only +1000 (long position), 0 (no position), or -1000 (short position). To account for this while performing the theoretically optimal strategy, I applied several conditions on the adjusted closing prices of both the current day and the next day. The first condition pertained to the price of the current day being lower than the price of the next day (i.e. the price went up), informing us to buy now. Here, I applied two sub-conditions:

```
if net holdings is 0:
    add order of buying 1000 shares
    net holdings += 1000
else if net holdings is -1000:
    add order of buying 2000 shares
    net holdings += 2000
```

If our current net holdings are zero, we have no position in the stock. Therefore, we are able to buy at most 1000 shares. However, if our net holdings are -1000, where we're currently in a shorting position, we can buy at most 2000 shares of the stock. The order is added to the running list and the net holdings value gets updated.

The second condition pertained to the price of the current day being higher than the price of the next day (i.e. the price went down), informing us to sell now. Here, I also applied two sub-conditions:

```
if net holdings is 0:
    add order of selling 1000 shares
    net holdings -= 1000
else if net holdings is 1000:
    add order of selling 2000 shares
    net holdings -= 2000
```

If our current net holdings are zero, we have no position in the stock. Therefore, we are able to short at most 1000 shares. However, if our net holdings are +1000, where we're currently going long on a stock, we can sell those 1000 shares and then short another 1000 for a total of 2000 shares sold. Again, each order is added to the running list and the net holdings value gets updated.

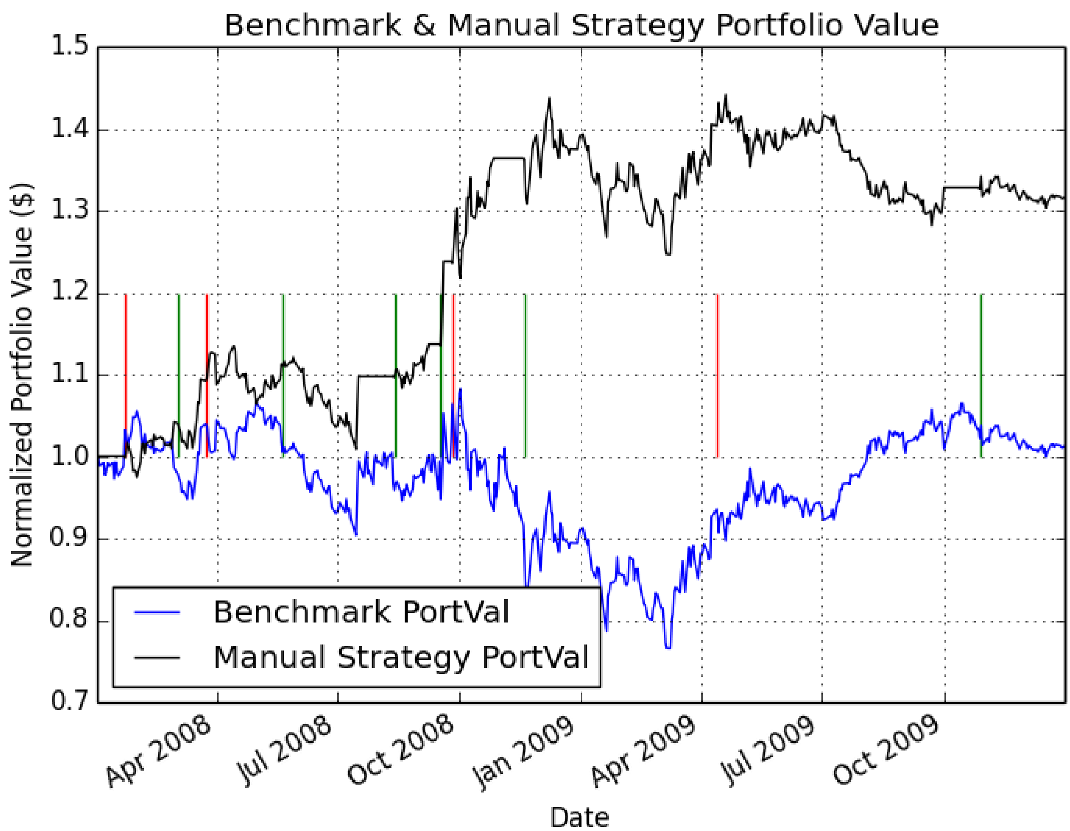
When applying this logic, I returned the following values:

	Benchmark Strategy	**Theoretically Optimal Strategy**
Cumulative Return	0.0123	5.7861
Average Daily Return	0.000168086978191	0.00381678615086
Std. Dev. of Daily Return	0.0170043662712	0.00454782319791
Sharpe Ratio	0.156918406424	13.3227698482

The values above have been normalized for easier comparison. As indicated by this table, the theoretically optimal approach performs significantly better than the benchmark approach, but it is, of course, impossible because we cannot actually peek into the future.

Part 3

In this section, I brought together my three indicators of SMA, Bollinger Band %, and MACD to make a more informed, plausible trading strategy. A combination of thresholds for each indicator were used to identify potential trading signals for buying or selling/shorting shares of JPM stock.



The three thresholds are as follows:

1) If the SMA value is greater than 1.05, the stock is overbought, and we will sell. If the SMA value is less than 0.95, the stock is oversold, and we will buy.
2) If the Bollinger Band % value is greater than 1.0, the stock is overbought, and we will sell. If the Bollinger Band % value is less than 0.0, the stock is oversold, and we will buy.
3) If the MACD line falls below the signal line, the stock is experiencing downward momentum, and we will sell. If the MACD line rises above the signal line, the stock is experiencing upward momentum, and we will buy.

Since the MACD incorporates a 26-day EMA, there is a gap of the same length preceding the starting point of the MACD line. Therefore, I began iterating from the starting point of lookback that both the SMA and Bollinger Band % use because they would still have data for this 12-day gap. In this case, I combined only the SMA and Bollinger Band % thresholds using an **and** connection to ensure that not just one indicator was being used to potentially make a hasty buy or sell decision. In terms of pseudocode implementation, it looked something like this:

```
if sma ratio > 1.05 and bollinger band % > 1:

    if net holdings is 0:
        add order of selling 1000 shares
        net holdings -= 1000
        add date of short order

    else if net holdings is 1000:
        add order of selling 1000 shares
        net holdings -= 1000

else if sma ratio < 0.95 and bollinger band % < 0:

    if net holdings is 0:
        add order of buying 1000 shares
        net holdings += 1000
        add date of long order

    else if net holdings is -1000:
        add order of buying 1000 shares
        net holdings += 1000
```

Having the two sub-conditions within the original two conditions account for the holding restrictions and ensure that I don't leave the positions of -1000, 0, or +1000 shares. Depending on the condition, if I currently have zero shares (no position in the stock), I could either short or long that stock. In these cases, I will capture the date of that short or long order, as indicated by the red and green lines, respectively, in the chart above. Otherwise, I would simply leave my current position by adding or subtracting the complement of my current holdings.

After this initial 12-day period, the MACD line no longer returns NaN values, so I applied the same thresholds and sub-conditions as before with the addition of the MACD threshold mentioned above. This is shown in the pseudocode below. One interesting discovery from my implementation of the manual strategy was that, while MACD < signal and MACD > signal should indicate selling and buying, respectively, applying these conditions with their SMA and Bollinger Band % counterparts actually resulted in a larger cumulative return. That is, applying MACD > signal to the selling threshold condition and MACD < signal to the buying threshold condition resulted in a better return differential from the benchmark than if the MACD conditions were reversed, like I would have expected them to be. This was particularly confusing because when I matched up the indicator charts visually, it appeared that the MACD line worked in unison with the other two indicators relative to the price. So, this anomaly in my indicator rules was intriguing.

```
if sma ratio > 1.05 and bollinger band % > 1 and macd > signal:

    if net holdings is 0:
        add order of selling 1000 shares
        net holdings -= 1000
        add date of short order

    else if net holdings is 1000:
        add order of selling 1000 shares
        net holdings -= 1000

else if sma ratio < 0.95 and bollinger band % < 0 and macd < signal:

    if net holdings is 0:
        add order of buying 1000 shares
        net holdings += 1000
        add date of long order

    else if net holdings is -1000:
        add order of buying 1000 shares
        net holdings += 1000
```

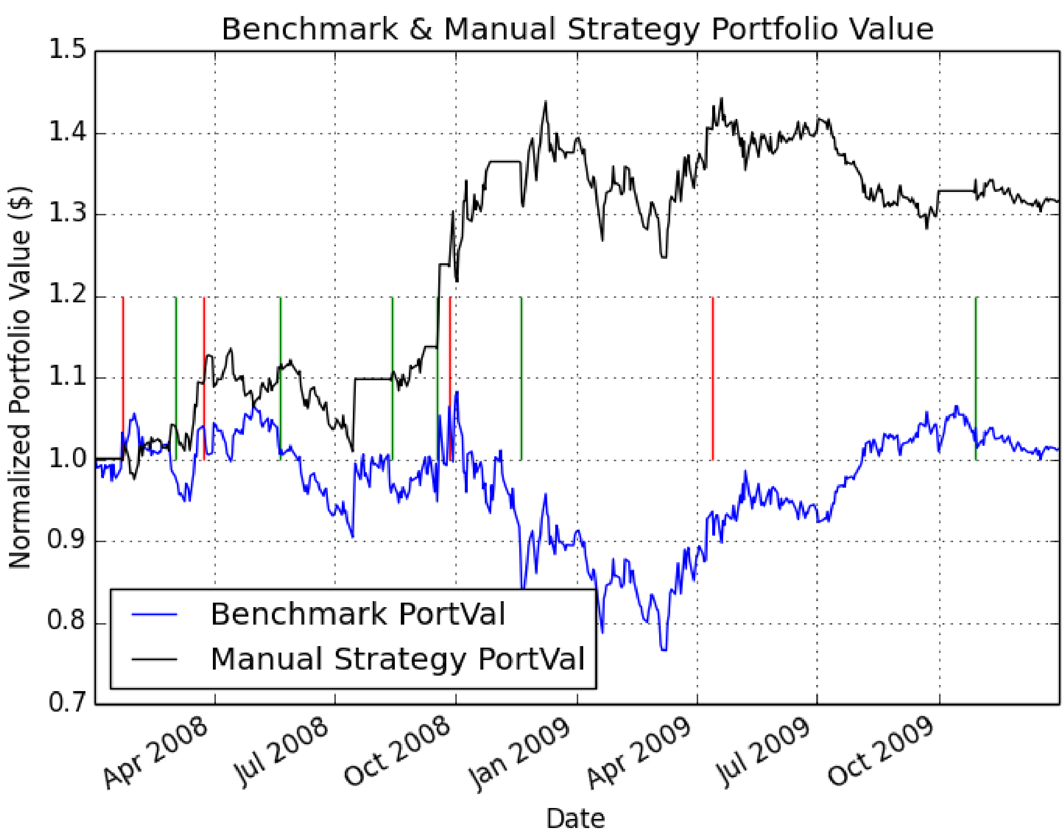
Overall, the manual strategy that I implemented was fairly effective because it yielded a cumulative return of over 0.3, which equates to over $30,000 (as depicted in the chart above).

Part 4

In this section, I expanded my manual strategy beyond just the in-sample data by applying it to the out-of-sample data as well. For reference, the table and chart below display the numeric and graphical representations of how the manual strategy performed compared to the benchmark strategy for the in-sample data.

In-Sample

	Benchmark Strategy	Manual Strategy
Cumulative Return	0.0123	0.316151
Average Daily Return	0.000168086978191	0.000608005716035
Std. Dev. of Daily Return	0.0170043662712	0.0112470342891
Sharpe Ratio	0.156918406424	0.858163252101



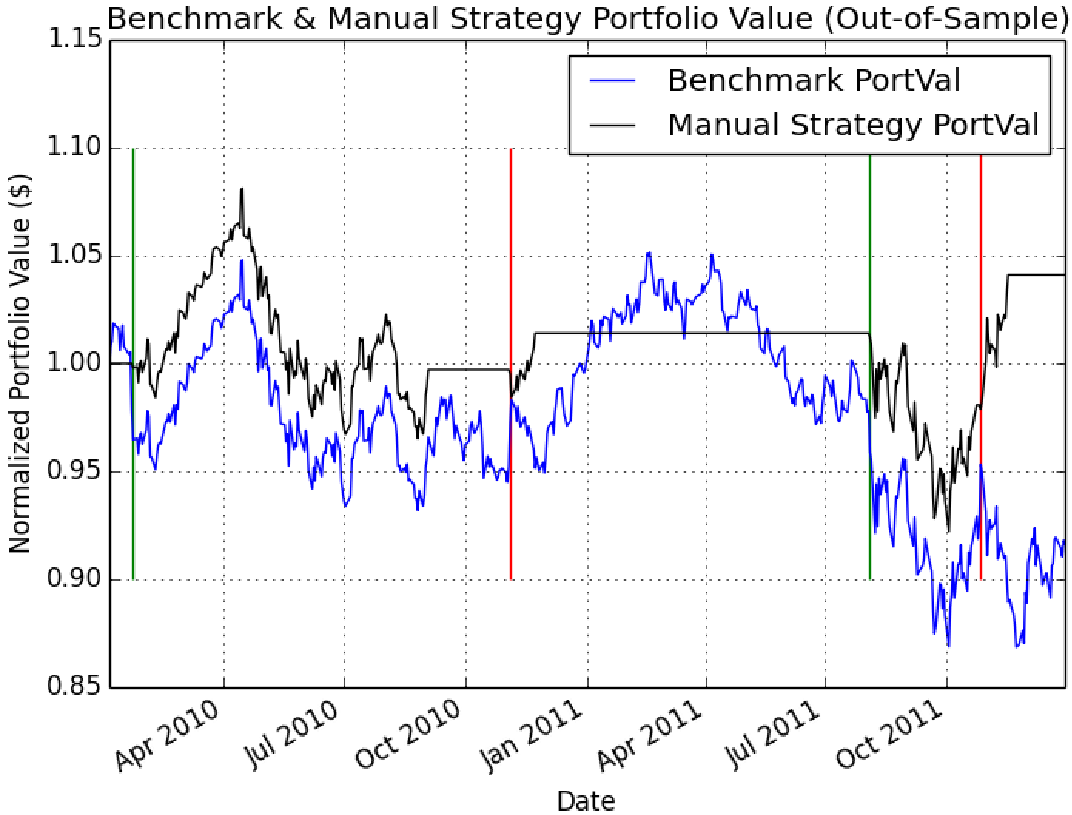
Naturally, it makes sense that the manual strategy did better because it relies on the financial indicators to create a practical approach to buying and selling and, theoretically, that additional information should give better results than just holding on to a set of shares for the entire time duration. Overall, the manual strategy had a larger cumulative return, larger average daily return, larger Sharpe ratio, and smaller standard deviation of daily returns, which all indicate that, again, the manual strategy performed more optimally than the benchmark strategy.

Below are the table and chart associated with the benchmark and manual strategies for the out-of-sample data. The in-sample backtests were expected to perform very well, and the same was expected for the out-of-sample backtests because there was no training involved and no rules were tweaked between the two backtest types. Having said that, based on the results given below, it does appear that the strategies for the in-sample data performed better than for the out-of-sample data. In this case, the benchmark strategy lost money due to the stock price dipping while the manual strategy still gave a positive cumulative return, though not as high. The numbers in the table below also support the notion that the manual strategy, once again, was the more optimal strategy. Despite these results, as mentioned, it performed worse than for the in-sample data.

Since we expected both in-sample and out-of-sample to perform generally about the same and there was no training involved that could impose bias, some other form of bias must've crept into the implementation. The obvious flaw that could've provided these performance differences is the indicator rules established for creating trading signals. During Part 3, I tweaked the rules to get a larger cumulative return when running the in-sample data. For example, I purposefully applied the MACD thresholds to their opposite actions to induce a larger cumulative return. Naturally, this could've been the downfall of the out-of-sample performance due to implicit bias in maximizing the in-sample return and going against the actual action the signals are supposed to indicate.

Out-of-Sample

	Benchmark Strategy	**Manual Strategy**
Cumulative Return	-0.0835791100328	0.041125
Average Daily Return	-0.000137429230389	0.00010002167718
Std. Dev. of Daily Return	0.00850015832233	0.00631459315346
Sharpe Ratio	-0.25665656052	0.251448488068



Though the manual strategy didn't perform as well with the out-of-sample data, it is important to note again that it still did outperform the benchmark and even led to a positive return. By tweaking the rules to be beneficial for both the in-sample and out-of-sample data, I can ensure more substantial returns for both backtests and not just one of them.