Manual Strategy Report

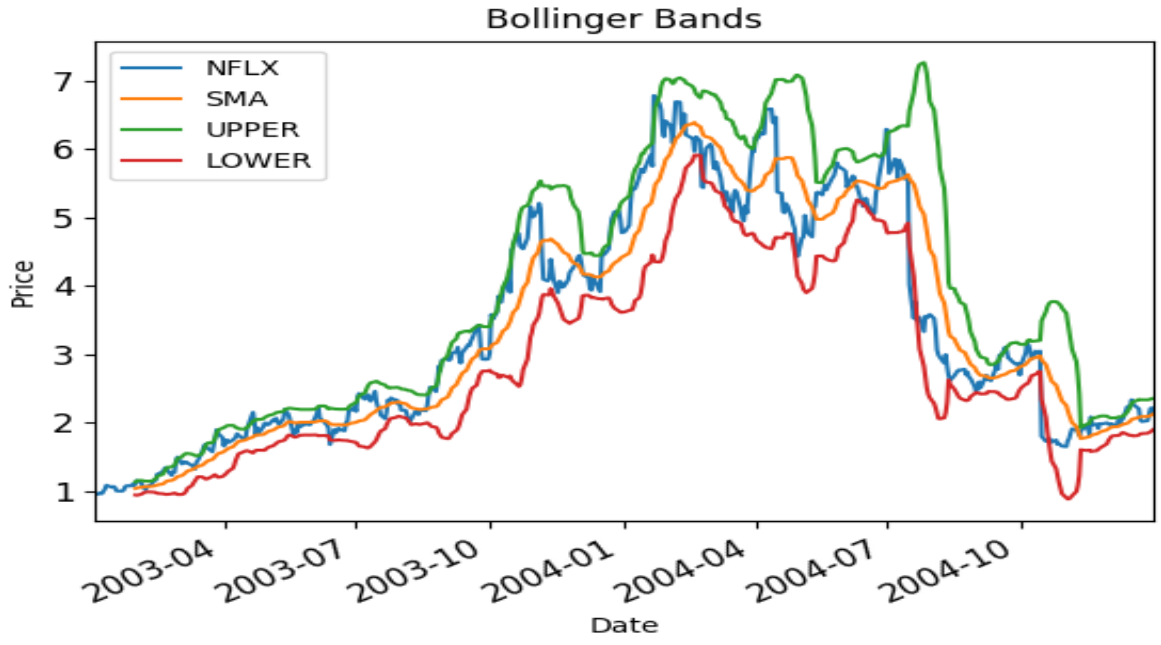
Overview:

The purpose of this project was to learn how to develop a trading strategy using technical indicators. There were three parts to this project and I will describe each part now.

Technical Indicators:

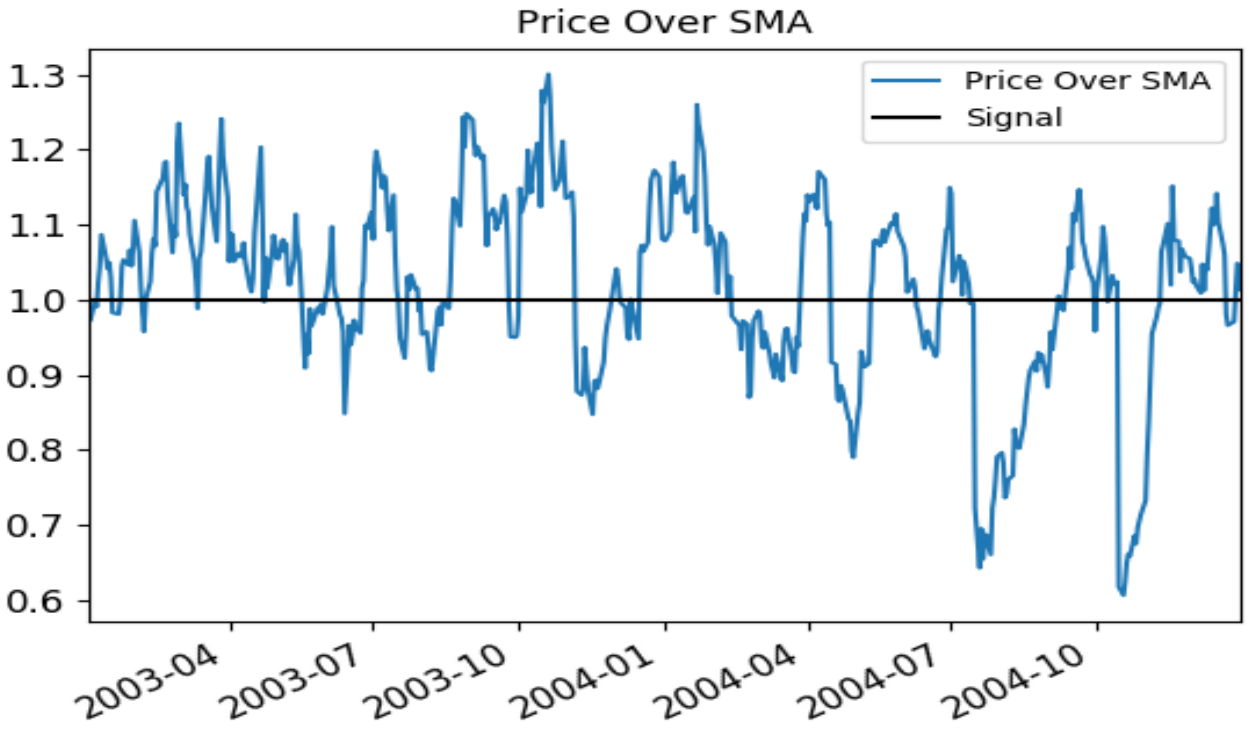
Before I could go about implementing any indicator, I had to implement two basic functions that most indicators called upon: a function to get simple moving average(SMA) and another for exponential moving average(EMA). SMA is the average of every few periods as defined for the method. For example, SMA5 means the 5 days- Simple Moving Average, which is calculated by getting the average of every five days using a rolling window. To calculate this SMA, my getsma() function took in a dataset and a specified window size, which were then passed into the .rolling() function of pandas to get a rolling window average. EMA is like SMA except the average is weighted by how close each previous day in the window is to the latest day, meaning that comparatively, today's price has a higher influence on the average than yesterday. To weigh each day's value, its multiplied by a weight (2/(N+1)) where N is how far away from current day. For example, to get the EMA for the past 5 days, today's price has a multiplier of 2 where as the price from 5 days ago has a multiplier of .4 as today's is more influential, as assumed by indicators that use EMA. To calculate this EMA, my getema() function took in a dataset and a specified window size, which were then passed into the .ewm()(exponential weighted) function of pandas to get a weighted rolling window average.

The first technical indicator that I implemented is Bollinger Bands. Bollinger Bands are calculated by getting the SMA, I used SMA-20 for mine, getting the standard deviation for that window and then adding 2 times that standard deviation for the upper Bollinger Band and subtracting 2 times that standard deviation for the lower Bollinger Band. The following output of Bollinger Bands was calculated for NFLX from 1/1/2003 to 12/31/2004, with the price being normalized to 1.0 at the start of the date range (by dividing price[t] by price[0]):



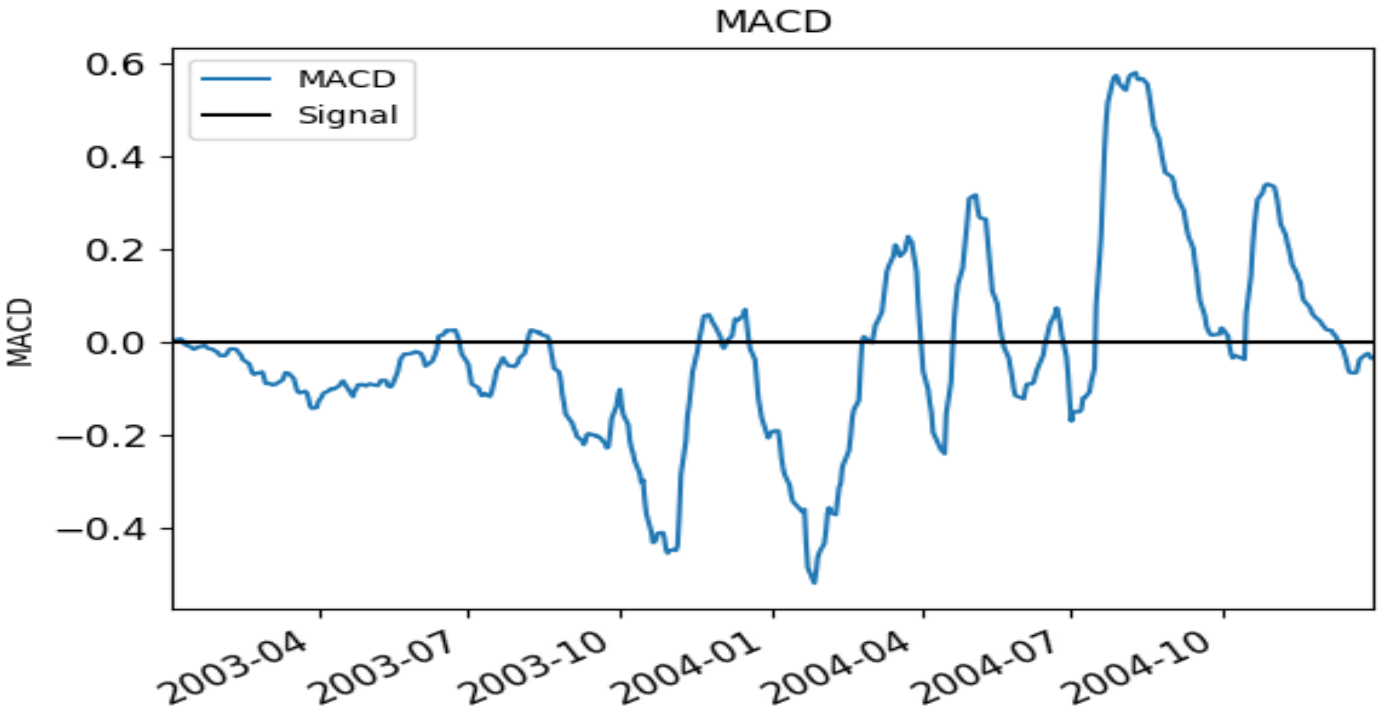
The higher limit works on the assumption that as price nears the upper Bollinger Band, the stock is being overbought and the price is probably reaching levels that won't be supported. On the other side, the lower limit works on the assumption that as price nears the lower Bollinger Band, the stock is being oversold and will start to rise back again soon. With this information, Bollinger Bands can be used by generating a buy signal when the price dips below the lower Bollinger Band and returns up, indicating that the price is going to adjust itself and go high. Similarly, Bollinger Bands can be used to generate a sell or short signal when the price crosses above the upper band and comes back to the upper band, implying that the price is going to adjust itself and go down.

The next indicator I implemented is Price over SMA ratio. This is calculated by dividing the price by the SMA, I used SMA-20, and then comparing that ratio to a signal of 1. The following output of Price to SMA Ratio was calculated for NFLX from 1/1/2003 to 12/31/2004, with the price being normalized to 1.0 at the start of the date range (by dividing price[t] by price[0]):



This indicator represents how the price is doing relative to the average. If the price is above the signal line (1), then that indicates that the price is going up and that would be a good time to buy. If the price is below the signal, then that means that the price is less than the average and is probably going down, so that would be a good time to sell.

The next indicator I implemented is Moving average convergence divergence (MACD). I calculated MACD by subtracting EMA12 from EMA26. For MACD there are various signals but I implemented the zero crossover (a line at y=0). The MACD follows the momentum trend and shows the relationship between two moving averages of prices. The following output of MACD was calculated for NFLX from 1/1/2003 to 12/31/2004, with the price being normalized to 1.0 at the start of the date range (by dividing price[t] by price[0]):

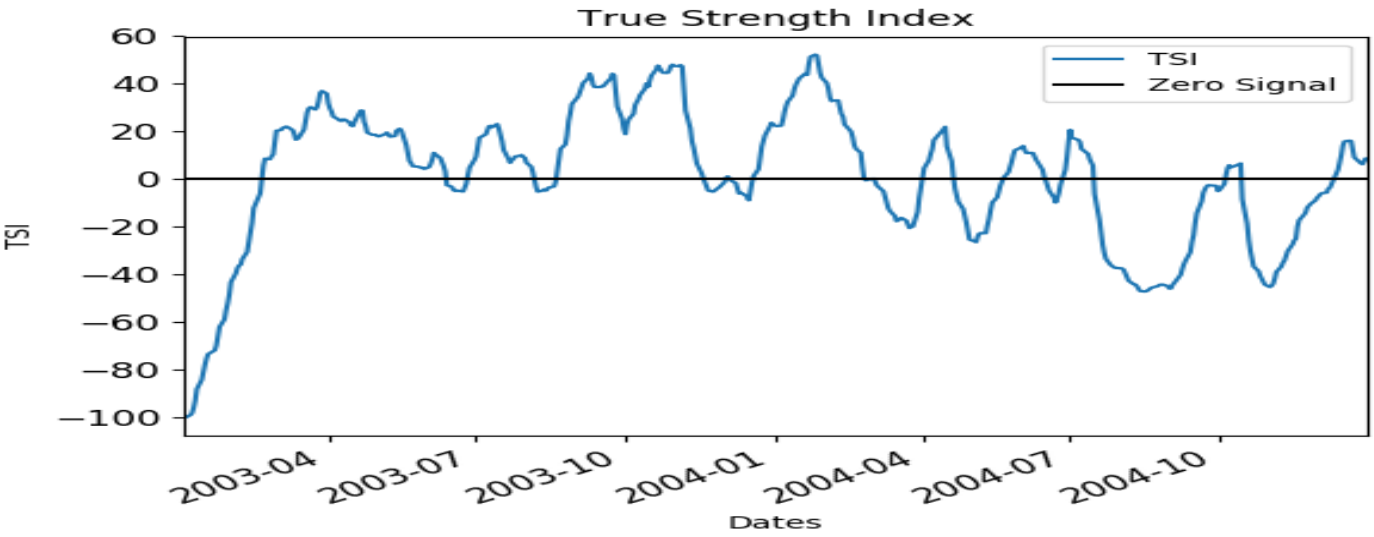


MACD tells us how two moving averages are doing in relation to each other. The idea is that the smaller moving average (SMA12) is more price responsive and the longer one (SMA 26) is less reactive. The convergence part of MACD is when the two averages come together (converge) which is basically a small magnitude of MACD. Divergence is when the two averages move away from each other. If the MACD has a positive crossover at 0, that means that EMA12 became higher than EMA26, meaning that the price is doing better recently than it was a while back, indicating a buy signal since the uptrend is increasing. Conversely, a negative MACD crossover at 0 indicates that the price is not doing as well it was before, indicating a sell signal, because the downtrend is increasing.

My last indicator was the True Strength Indicator (TSI). The TSI requires multiple steps broken down into two parts: double smoothed price change and double smoothed absolute price change. For part one you first get the price change or PC (todays' price minus yesterdays), then get the EMA25 of that PC (known as the first smoothing), and then get the 13-day EMA of that 25-day EMA to get the double smoothing. Part two is very similar to part one, except you must take the absolute value of the price change, and then absolute first smoothing = EMA25(|PC|) and absolute second smoothing = EMA13(absolute first smoothing). To get the TSI value at this point, you do elementwise division of part one's second smoothing by part two's absolute second smoothing, and then multiply by 100. The table presents the formula concisely:

Double Smoothed PC	Double Smoothed Absolute PC
PC = Current Price minus Prior Price First Smoothing = 25-period EMA of PC Second Smoothing = 13-period EMA of 25-period EMA of PC	|PC| = ABS(Current Price minus Prior Price) First Smoothing = 25-period EMA of |PC| Second Smoothing = 13-period EMA of 25-period EMA of |PC|
TSI = 100 x (Double Smoothed PC / Double Smoothed Absolute PC)	

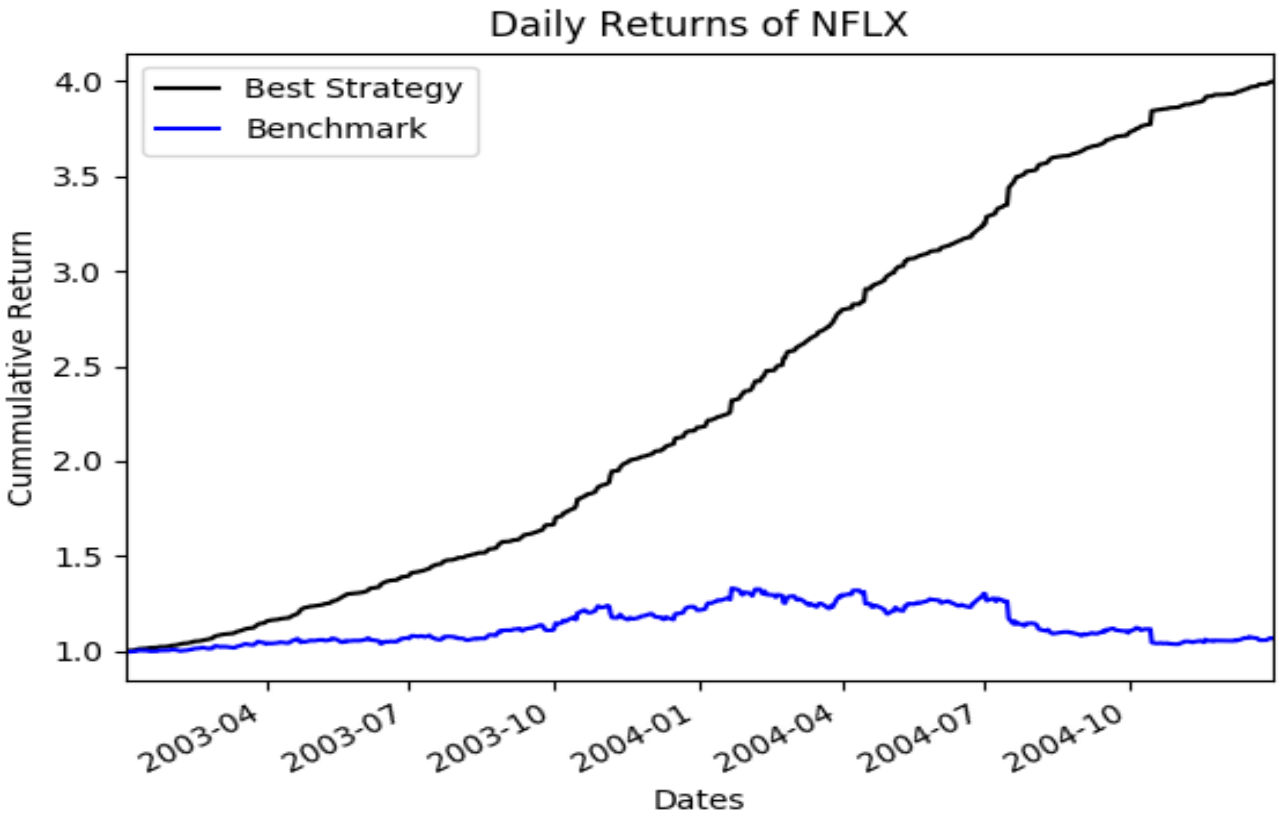
The following output of TSI was calculated for NFLX from 1/1/2003 to 12/31/2004, with the price being normalized to 1.0 at the start of the date range (by dividing price[t] by price[0]):



The TSI combines the short-term momentum (a leading indicator) with moving averages (a lagging indicator) to effectively showcase trends. If TSI is positive and rising that indicates a buy signal since the uptrend is increasing. If TSI is negative and decreasing that indicates a sell signal since the downtrend is increasing.

Best Possible Strategy:

For best possible strategy, I do a simple one-day look ahead. To achieve this, I get the price change each day by subtracting tomorrow's price from today's and setting that as today's effective price. I then iterated through the days and if today's effective price is positive I sold everything I had and shorted what I could because that meant today's actual price is greater than tomorrow's. This is so that when price goes down tomorrow, I could make money with the stock I shorted today and not lose money on the stocks that I had today. Conversely, if today's effective price is negative I bought as many stocks as I could because that means today's actual price is less than tomorrow's and thus, I want to buy today and sell tomorrow for profit. I then compared this to a benchmark strategy of investing long on the first day and this is the result I got for NFLX from 1/1/2003 to 12/31/2004, with the price being normalized to 1.0 at the start of the date range (by dividing price[t] by price[0]): I had no commission or transaction cost for the best possible strategy.



These are the portfolio statistics that I got for both benchmark and best possible strategy approaches:

	Benchmark	Best Possible
Cumulative Return	0.0661	2.997
Std(Daily returns)	0.00854838899046	0.00337160407852
Avg(Daily returns)	0.000164136118256	0.00276412220003

Manual Strategy:

For my manual strategy, I used a combination of the Bollinger Bands and MACD indicators. Before I looped through each date and decided whether or not to buy or sell on that date, I first calculated MACD and Bollinger Bands by passing the stocks data frame to those methods in indicators.py. From that, I got a data frame for each indicator that had a date, corresponding to each date that the stock had a price, and the values of the features of that indicator on that day. For MACD, that was just the MACD, or the EMA26-EMA12, value on the date. For Bollinger Bands, that was the SMA-20, the upper band value (SMA plus 2 standard deviations), and the lower band value (SMA minus 2 standard deviations). Then I iterated the dates and had a nested set of if statements that checked certain conditions of Bollinger Bands to decide if I should sell:

- If the price crossed (went above) the upper band on that date and the exit flag was not set, then I raised an exit-flag (set exit-flag equal True), because when the price comes back down through the upper band, I want to sell

- If the price is less than the upper band on that date and the exit flag was set, then I went out of the stock and sold all 1000 shares, if I owned any, and shorted another 1000, because according to my Bollinger Band indicator, this is where price falls, as explained in indicators section. I then reset the exit-flag to off (set exit-flag equal False).

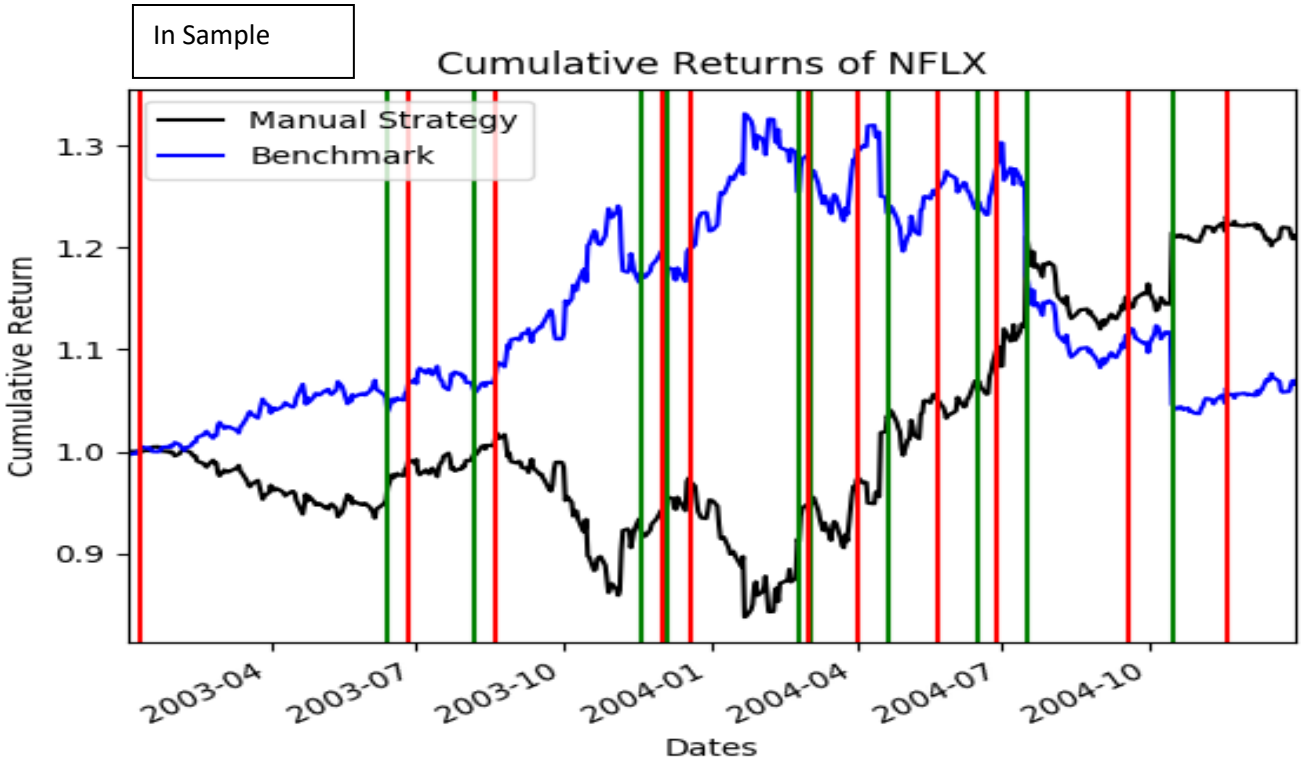
I then checked these condition of Bollinger bands to decide I should buy:

- If price went below the lower bound on that date and the entry flag was not set, then I set the entry flag to true because when the price comes back up through the lower band, I want to buy
- If the price is above the lower band on that date and entry flag was set, then I went into the stock and bought as much as I could within the constraints, because according to my Bollinger Band indicator, this is indicating a rise in price, as explained in indicators section.

I then checked these conditions on MACD to decide to buy or sell:

- If today's MACD is positive and yesterday's MACD (I saved yesterday's date in a local variable and passed it into the MACD data frame) was negative, then I went into the stock and bought as much as I could within the constraints. This is because that is a positive MACD zero crossover, which is a buy signal, with an indication of rising prices, as explained in indicators section.
- If today's MACD is negative and yesterday's MACD was positive, then I went out of the stock and sold all 1000 shares, if I owned any, and shorted another 1000. This is because my negative MACD zero crossover was giving me a selling signal, by indicating a drop in price, as explained in indicators section.

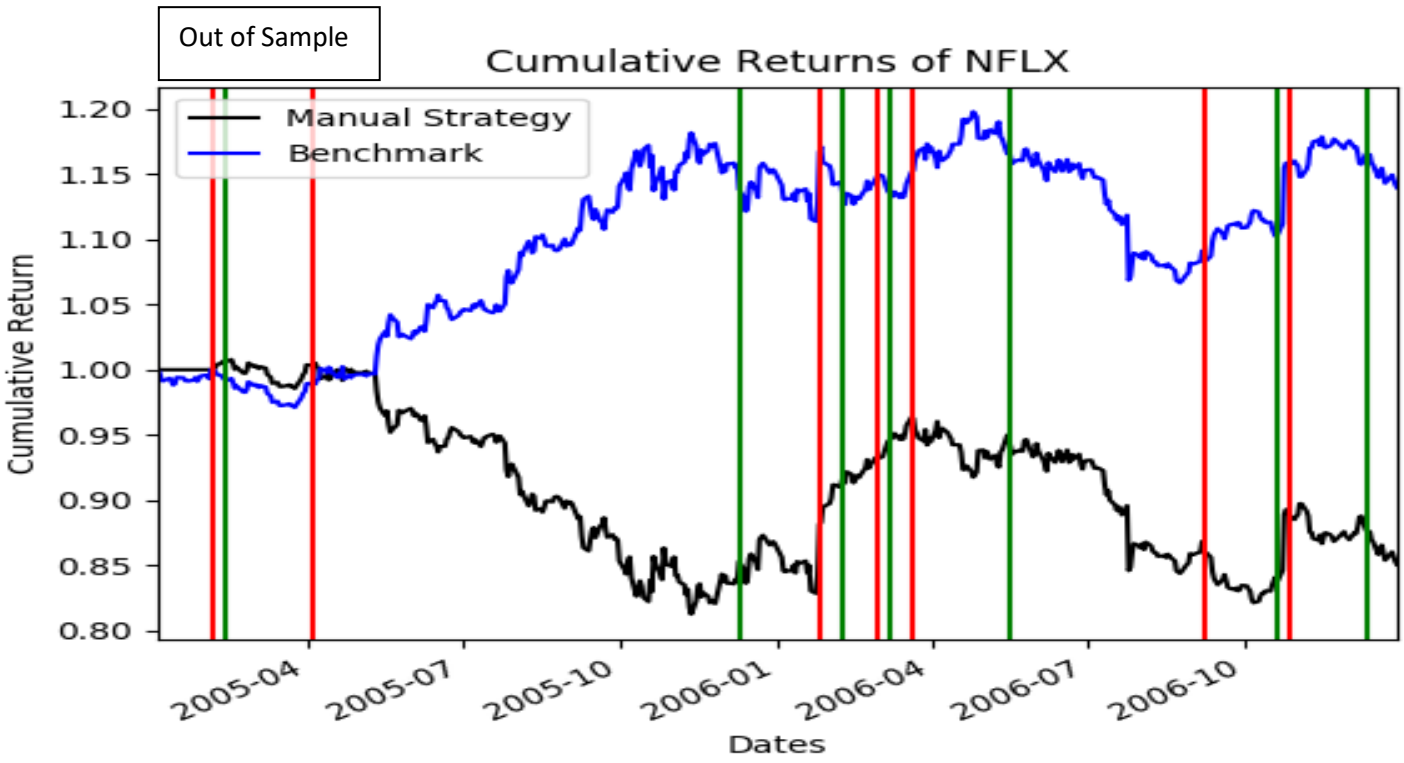
I then compared this to a benchmark strategy of investing long on the first day and this is the results I got for NFLX from 1/1/2003 to 12/31/2004, with the price being normalized to 1.0 at the start of the date range (by dividing price[t] by price[0]): I had a commission of $9.95 and impact of 0.005



The green lines represent long entry points and the red points indicate short entry points.

Comparative Analysis:

Although my manual strategy made some pretty head-scratching decisions (the third red line), by the end of the date periods, I was able to optimize the strategy to out-perform the benchmark. I then tested my manual strategy with my out of sample test, same stock (NFLX) from time period January 1, 2005 to December 31 2006 with the same commission and trade penalty and this was my output:



	In Sample		Out of Sample	
	Benchmark	Manual Strategy	Benchmark	Manual Strategy
Cumulative Return	0.0661254913769	0.2119915	0.139497020178	-0.149423
Std(Daily returns)	0.00855114085899	0.0102890368933	0.00600670005472	0.00762279156024
Avg(Daily returns)	0.000164207543753	0.000434561374452	0.000278170449764	-.00029340133205

This table summarizes the performance of the stock, and the manual strategy for both in sample and out of sample period. Apart from the obvious difference that the manual strategy performed better in the in-sample period than the out of sample period, there are a few interesting features in this table. In the in sample, my manual strategy does cause a higher variance in the portfolio, with a higher standard deviation of daily returns. This makes sense for both periods as the multiple trades caused by my strategy should cause more fluctuations in the portfolio and thus increase variance. The ugliest bit of the table is that the average daily returns in the out of sample period is negative, meaning on average, my trading strategy loses money every day. This can be attributed to the fact that my indicators probably

don’t generalize well and are not as effective at predicting price trends as I thought they would be. The performance of the benchmark is a lot better in the out of sample period, with a higher cumulative return (more than double) and higher average daily return (almost double). This means that the out of sample period had better up trends that the benchmark was able to take advantage of, with an overall increase in the price of the stock, most likely. The fact that the out of sample manual strategy did so bad, even with such positive growth in the price of the stock, indicates that overall, I devised a really bad strategy. I tried to optimize my manual strategy by using various indicators and the two methods that I ended up using did perform the best in sample so it’s really unfortunate that they did not generalize to work for out of sample. Looking back at the chart for the in-sample period, I can see that the strategy performed better by the end because of a few really good trades towards the end of the period. A way I would expand on this project in the future is to figure out how I was able to have good trades there and what worked, and hopefully extrapolate that to the beginning of the in sample and into the out of sample period.