INTRODUCTION

This project evaluated a rules-based manual, rules based long-short, non0leveraged strategy (MS) for stock trading. The MS was constructed by identifying technical indicators[1], which derived from historical stock data that provide a signal for buying or selling a security. For this analysis, I chose to utilize the following indicators: Simple Moving Average (SMA), Bollinger Bands® (BBI) and Chaikin Money Flow (CMF). These indicators provided the basis for developing a long-short strategy that was trained over an in-sample period and tested on an out of sample interval. The training strategy maximized portfolio statistics: Sharpe ratio and cumulative return over the in-sample period. For this report, the portfolio consisted of: a starting value of $100,000 in cash, and long/short positions in a single security, JPM. The in sample/development period is: January 1, 2008 to December 31, 2009. The out of sample/testing period is: January 1, 2010 to December 31, 2011.

To baseline and provide an upper bound for the MS, Benchmark (BM), and Theoretically Optimal (TOS) portfolios were developed respectively. Both the BM and TOS consisted of the same securities as the MS, however, the trading strategies were different. The BM portfolio employed a buy and hold strategy. That is, a long position (buy) was initiated on the first trading day of the interval and held until the end date of the interval. For the TOS, information about future stock prices was utilized to make trades in the present. Both strategies did not incorporate commissions or impact on the portfolio statistics.

Allowable positions for all strategies (MS, TOS, BM) were: 1000 shares long, 1000 shares short, 0 shares.

I. Technical Indicators

This section will describe my implementation of indicators.py. This file defined a function, `indicators`, which applied three technical indicators over an interval of historical price data [sd, ed] for securities [syms], and returned a Pandas dataframe. This section will focus on the development of technical indicators. Further discussion of how the technical indicators were utilized to signal long/short positions will be covered in Part III (Manual Rule-Based Trading Strategy).

```
def indicators(syms, sd=dt.datetime, ed=dt.datetime, window=14)
```

The technical indicators I chose to utilize: Simple Moving Average (SMA), Bollinger Bands® (BBI) and Chaikin Money Flow (CMF) provided signals for a long/short trading policy. Since the interval (one year) for both the in-sample and out-of-sample evaluation is relatively short, I chose a shorter lookback window of 14 days to account for a short-term trading strategy. That is, I wanted more frequent trading activity over this sample periods.

Simple Moving Average (SMA): This indicator smoothes out price action by filtering out the "noise" from daily (random) price fluctuations. I used SMA to identify a trend in a stock's price movement. I developed the SMA indicator by obtaining the [quotient of the portfolio price over the mean price of the portfolio over the lookback window] minus one. To utilize the indicator over the in and out-of-sample trading intervals, I used the Pandas rolling mean function. SMA is an oscillator that fluctuates between -0.5 and +0.5.

```
sma = (prices.loc[:,'PORTFOLIO'] /
      prices.loc[:,'PORTFOLIO'].rolling(window=window,
      min_periods=window).mean()) - 1
```

Bollinger Bands® (BBI): BBI is an indicator of volatility and ranges between +-2 standard deviations of the SMA. I used BBI to identify overbought, or oversold conditions in a security. I developed BBI by obtaining the quotient of SMA over two standard deviations of the portfolio over the lookback window. To utilize the indicator over the in and out-of-sample window, I used the Pandas rolling standard deviation function. BBI is an oscillator that fluctuates between -1 and +1.

```
bbi = (prices.loc[:,'PORTFOLIO'] - indicators_df.loc[:,'RM']) /
      (prices.loc[:,'PORTFOLIO'].rolling(window=window, center=False).std() *2)
```

Chaikin Money Flow (CMF): CMF is a volume-weighted average of security accumulation and distribution over a specified period, i.e., the lookback window of 14 days. CMF is an oscillator that fluctuates between -1 and +1.

First, calculate the Money Flow Multiplier for each period. Second, multiply this value by the period's volume to find Money Flow Volume. Third, sum Money Flow Volume for the 20 periods and divide by the 20-period sum of volume.

I developed CMF by first calculating the Money Flow Multiplier = [(Close - Low) - (High - Close)] / (High - Low). I then calculated the Money Flow Volume = Money Flow Multiplier * Volume for the period (lookback window). The last step was to utilize the Pandas rolling sum function to calculate a 14 period CMF = 14-period Sum of Money Flow Volume / 14 period Sum of Volume.

```
STATS["MF_Multiplier"] =      ((STATS["CLOSE"]  - STATS["LOW"])  -\
                         (STATS["HIGH"] - STATS["CLOSE"])) /\
                         (STATS["HIGH"] - STATS["LOW"])
STATS["MF_Volume"] =     STATS["MF_Multiplier"] * STATS["VOL"]
indicators_df['CMF']                                                   =
      STATS["MF_Volume"].rolling(window=window,
                         center=False).sum() /
                         STATS["VOL"].rolling(window=window,
                         center=False).sum()
```

Figure 1: Dataframe of technical indicators returned from *indicators.indicators.py*:

	JPM	PORTFOLIO	MM	SMA	RM	BBU	BBL	Norm_RM	BB_Ind	CMF
2010-01-04	40.87	40.87	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2010-01-05	41.67	41.67	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2010-01-06	41.89	41.89	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2010-01-07	42.72	42.72	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2010-01-08	42.62	42.62	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2010-01-11	42.48	42.48	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2010-01-12	41.48	41.48	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2010-01-13	42.21	42.21	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2010-01-14	42.63	42.63	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2010-01-15	41.67	41.67	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2010-01-19	41.28	41.28	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2010-01-20	41.40	41.40	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2010-01-21	38.67	38.67	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2010-01-22	37.35	37.35	-0.086127	-0.096798	41.352857	44.445184	38.260530	1.000000	-1.294448	-0.060650
2010-01-25	37.40	37.40	-0.102472	-0.090135	41.105000	44.851175	37.358825	0.994006	-0.989009	-0.118034
2010-01-26	36.67	36.67	-0.124612	-0.100075	40.747857	45.156738	36.338977	0.985370	-0.924919	-0.208585
2010-01-27	37.52	37.52	-0.121723	-0.072107	40.435714	45.107223	35.764205	0.977822	-0.624148	-0.154270
2010-01-28	37.66	37.66	-0.116377	-0.060245	40.074286	44.767408	35.381163	0.969081	-0.514431	-0.169141
2010-01-29	37.14	37.14	-0.125706	-0.064079	39.682857	44.375466	34.990248	0.959616	-0.541886	-0.248087
2010-02-01	37.80	37.80	-0.088717	-0.039355	39.348571	43.845530	34.851613	0.951532	-0.344360	-0.210753

II. Theoretically Optimal Strategy

I developed TheoreticallyOptimalStrategy.py to obtain the greatest possible cumulative return at the end of the sample period. This strategy allowed the learner to peek into the future to utilize this information to execute trades in the present. Essentially, the learner knew the price of the security tomorrow when placing a trade today. Given this information, I developed the learner to carry the max number of shares (+1000) as a long position when the price would rise the next day. Conversely, the learner would carry the max number of shares (-1000) as a short position when the price for the security would fall the next day. To achieve this, I was buying or selling 2000 shares each day for the sample period. The one caveat was the first day of the sample period. For this day, I went long or short 1000 shares based on the price of the stock on the second day of the sample period. For this activity, commissions and impact were $0.00, and 0.0 respectively.

Figure 2: Statistics for Theoretical Optimal Strategy (TOS) and Benchmark (BM) portfolios. Each strategy used a portfolio consisting of a single security, JPM. Statistics computed for the in-sample period are: Cumulative Return (CR), Average Daily Return (ADR), Standard Deviation Daily Return (SDDR) and Sharpe Ratio (SR). The statistics for TOS do not include expense: commission=0.00, impact=0.00. Statistics for the BM portfolio were computed with zero expense, as well as with expense: commission=9.95, impact=0.005.

STRATEGY	CR	ADR	SDDR	SR
TOS	5.7861	0.003817	0.004548	13.32277
BM	0.0123	0.000168	0.017004	0.156918
BM (Commission and Impact)	-0.03792	7.51E-05	0.017468	0.068229

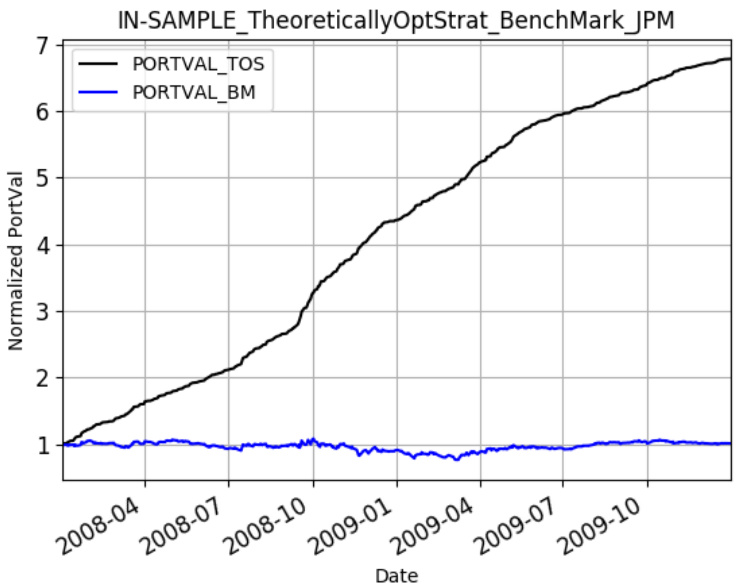


Figure 3: Plot of the BM portfolio (with expenses) normalized to 1.0 at the start (Blue line) and the value of the TOS portfolio normalized to 1.0 at the start (Black line) over the in-sample period.

III. Manual Rule-Based Trading Strategy

ManualStrategy.py applied indicators in a contrarian approach. That is, I was looking for oversold, or overbought signals in the security. Initially, I tried a momentum-based investment approach, looking for positive money flow CMF as a leading indicator, confirmed by momentum, or a crossing of short and long term SMA trend lines. I was not able to generate sufficient performance (cumulative return and Sharpe ratio) during the in-sample training period (2008-2009). Perhaps this was a period when value investing provided a strong influence on the market.

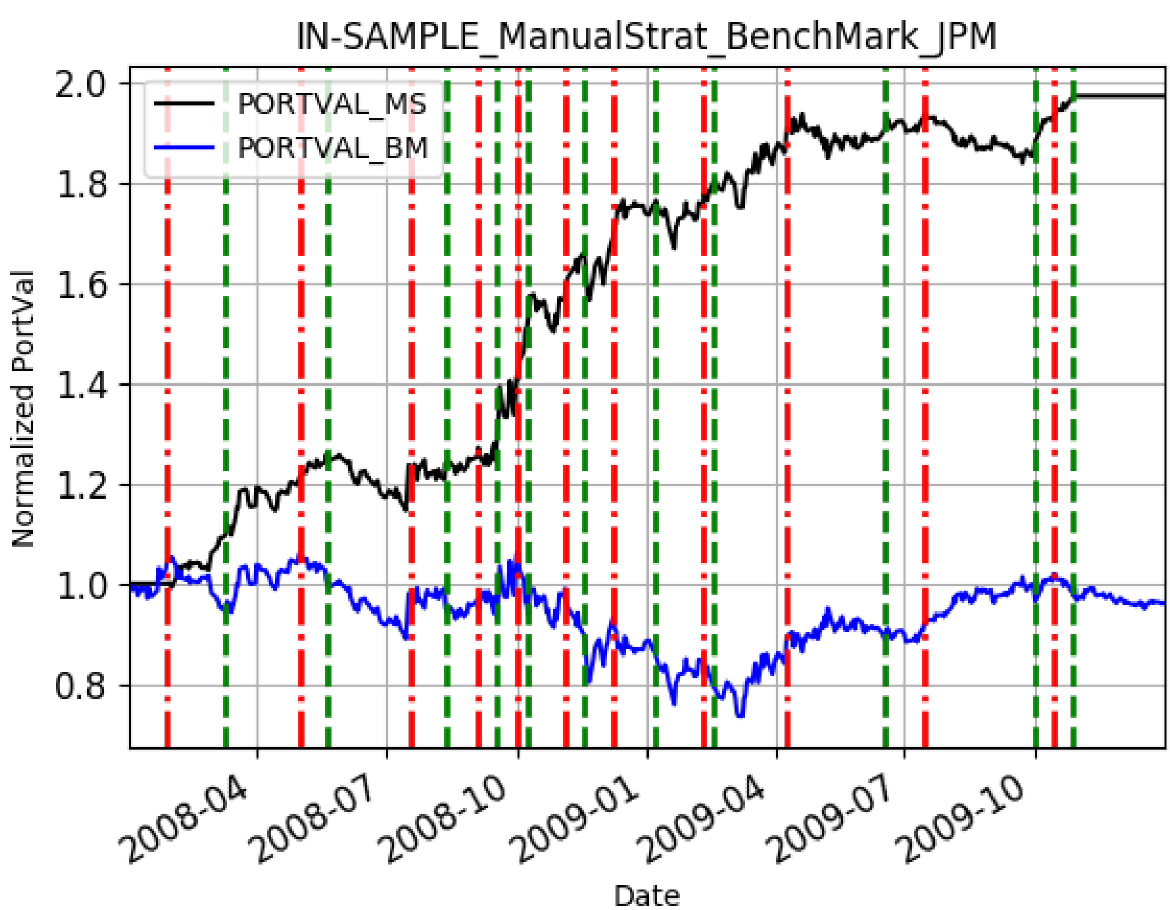
To develop the contrarian approach, I used the indicators dataframe (I_DF) (***Figure 1***) that was returned from *indicators.indicators.py*. Long or short signals were identified by concatenating the indicators from I_DF. A long signal was indicated when: [(CMF<-0.01) & (SMA<-0.05) & (BBI<-0.7)]. Notice that the thresholds for all the signals are negative, which may indicate an oversold condition in the security. Conversely, a short position was indicated when: [(CMF>0.1) & (SMA>0.05) & (BBI>0.7)]. Notice that the thresholds for all the signals are positive, which may indicate an overbought condition in the security. Utilizing oversold or overbought signals to buy, or sell securities respectively, is, by definition, a contrarian approach to investing. Like the theoretically optimal strategy, my MS started with an initial position of +-1000 shares, and then went long/short 2000 shares for all subsequent trades. That is, I never had a position of zero shares in the security.

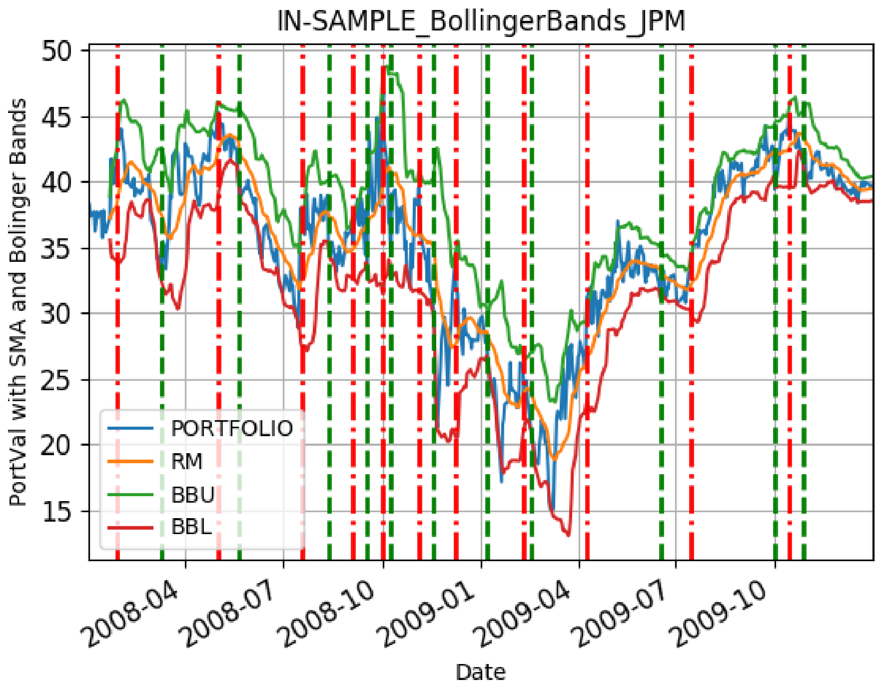
```
orders_df[  (indicators_df['CMF']<-0.01) &
            (indicators_df['SMA']<-0.05) &
            (indicators_df['BB_Ind']<-0.7)] = 1000
orders_df[  (indicators_df['CMF']>0.1) &
            (indicators_df['SMA']>0.05) &
            (indicators_df['BB_Ind']>0.7)] = -1000
```

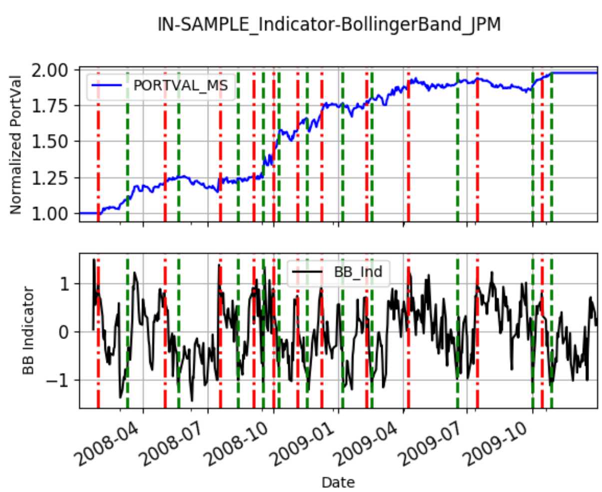
Figure 4: Statistics for Manual Strategy (MS) and Benchmark (BM) portfolios. Each strategy used a portfolio consisting of a single security, JPM. Statistics computed for the in-sample period are: Cumulative Return (CR), Average Daily Return (ADR), Standard Deviation Daily Return (SDDR) and Sharpe Ratio (SR). The statistics for MS include expense: commission=9.95, impact=0.005. Statistics for the BM portfolio were computed with the same expense, as well as zero expense: commission=0.00, impact=0.00.

STRATEGY	CR	ADR	SDDR	SR
MS	0.972403	0.0016	0.01122	2.264393
BM	0.0123	0.000168	0.017004	0.156918
BM (Commission and Impact)	-0.03792	7.51E-05	0.017468	0.068229

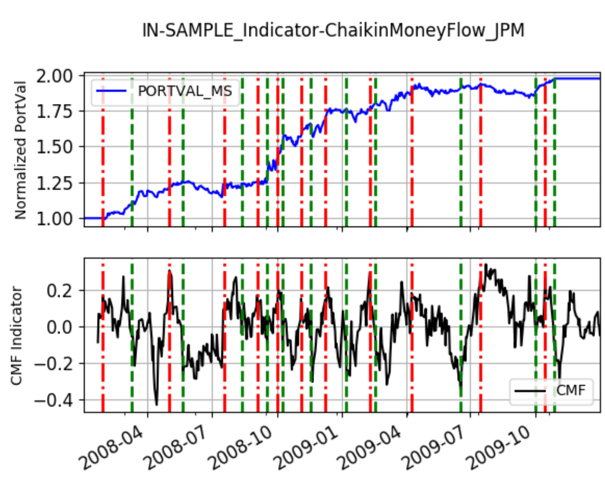
Figure 5: Plot of the BM portfolio including expenses: commission=9.95, impact=0.005 normalized to 1.0 at the start (Blue line) and the value of the rules-based MS portfolio normalized to 1.0 at the start (Black line) over the in-sample period. Vertical (green, --) lines indicate LONG entry points, while vertical (red, -.) lines indicate SHORT entry points.

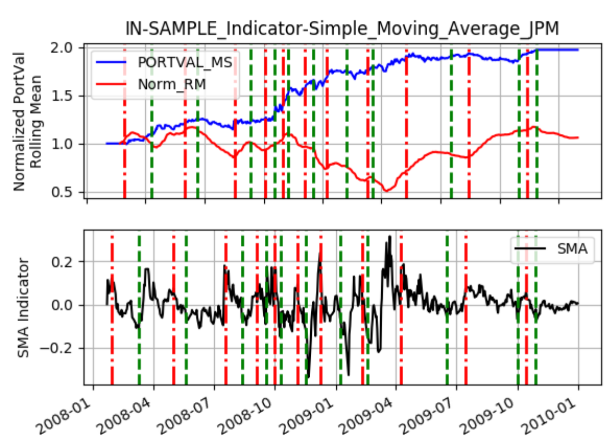






Figures [6, 7, 8]: Plots of indicators identifying long/short portfolio positions over the in-sample period. From top: Bollinger Bands® with Rolling Mean, BBI, CMF, SMA.





IV. Conclusion: Comparative Analysis

This section will evaluate the performance of the MS in the out-of-sample period. The MS was trained over the in-sample period by manually adjusting the indicators (as discussed in Section III) to optimize portfolio statistics (performance). I chose to focus on optimizing Cumulative Return (CR) and Sharpe Ratio (SR). Indicators thresholds were adjusted over repeated runs over the in-sample period, noting the threshold values, and CR and SR for each run. Once an optimal value for CR and SR were observed, I ran this strategy one time over the out-of-sample period. I was satisfied that my MS outperformed the BM portfolio over the out-of-sample period.

Figure 9: Statistics for Manual Strategy (MS) and Benchmark (BM) portfolios and Theoretically Optimal Strategy (TOS). Each strategy used a portfolio consisting of a single security, JPM. Statistics computed for both the in and out-of-sample periods are: Cumulative Return (CR), Average Daily Return (ADR), Standard Deviation Daily Return (SDDR) and Sharpe Ratio (SR). The statistics for all strategies in both periods include expense: commission=9.95, impact=0.005.

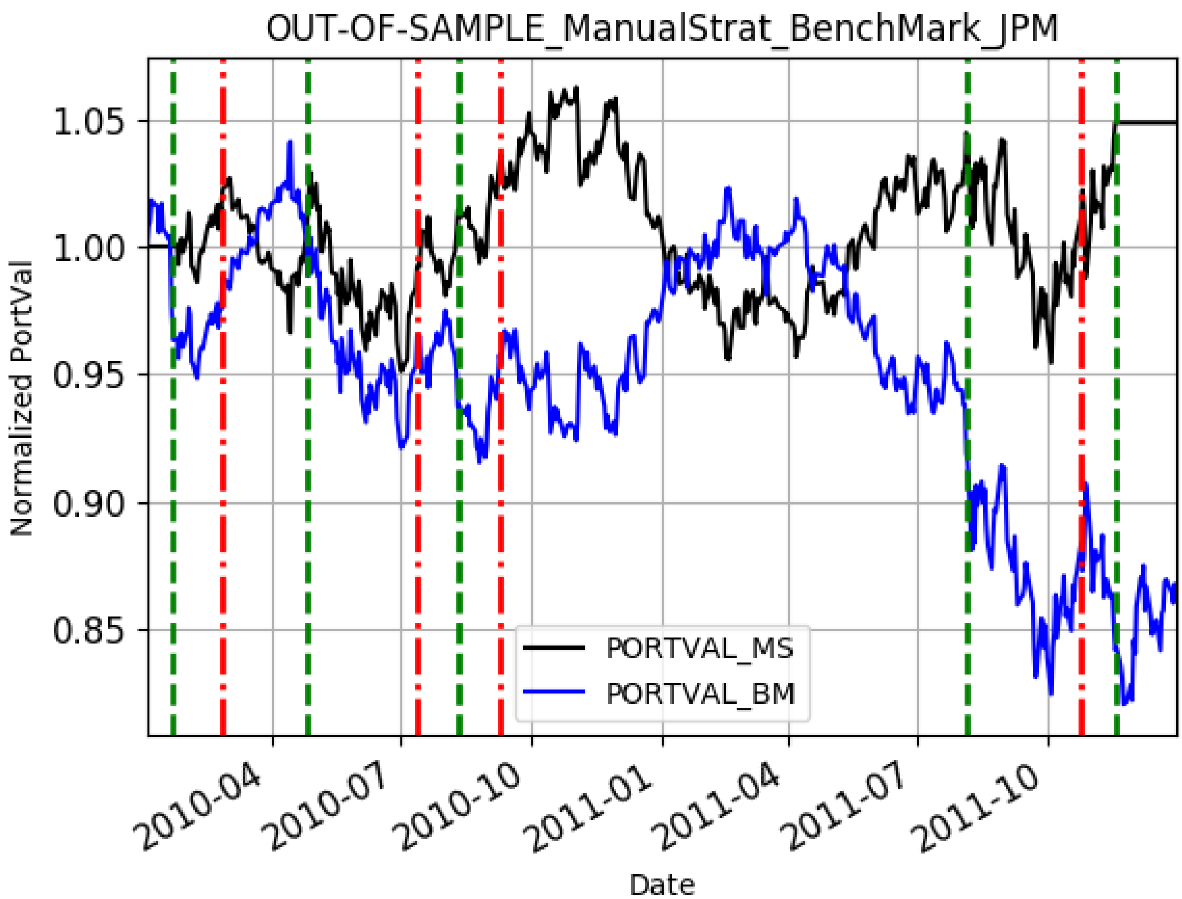
STRATEGY	CR	ADR	SDDR	SR
MS (OOS)	0.048486	0.000133	0.007847	0.269944
TOS (OOS)	3.1202	0.002823	0.002734	16.38975
BM (OOS)	-0.13374	-0.00025	0.008781	-0.44633
MS (IS)	0.972403	0.0016	0.01122	2.264393
TOS (IS)	5.7861	0.003817	0.004548	13.32277
BM (IS)	0.0123	0.000168	0.017004	0.156918

Observing the statistics in ***Figure 9***, the MS over the in-sample period significantly outperformed (CR, SR) the MS over the out-of-sample period. While this is not ideal, it can be explained. Note that during that over the same periods, both the TOS and BM also underperformed (CR, SR).

Consider the Capital Assets Pricing Model (CAPM), where market performance has a strong influence of the portfolio's performance. The MS was trained over an in-sample interval that consisted of a bear market, while the out-of-sample interval was the start of a favorable market. The contrarian strategy utilized by the MS over the bear market may not perform well in a favorable market.

Looking at ***Figure 10***, note that the MS exhibited significant diverges from the BM over the intervals: [07.2010 - 01.2011] and [05.2011 - ED]. The strategy carried a short position (~10.2010) over most of the first interval. The long position at ~08.11 did not have a huge impact on performance, however, the short at ~11.2011 added a strong finish to the period. The plots in the appendix ***Section V*** provide additional insight into how the MS utilized the indicators to develop a trading strategy. As discussed in ***Section III***, my MS traded +-2000 shares (following an initial position of +-1000 shares). It's possible better performance could have been obtained from the MS by holding a neutral position (zero shares), rather than exclusively trading to achieve the thresholds of the allowable positions [-1000, +1000].

Figure 10: Plot of the BM portfolio including expenses: commission=9.95, impact=0.005 normalized to 1.0 at the start (Blue line) and the value of the rules-based MS portfolio normalized to 1.0 at the start (Black line) over the out-of-sample period. Vertical (green, --) lines indicate LONG entry points, while vertical (red, -.) lines indicate SHORT entry points.



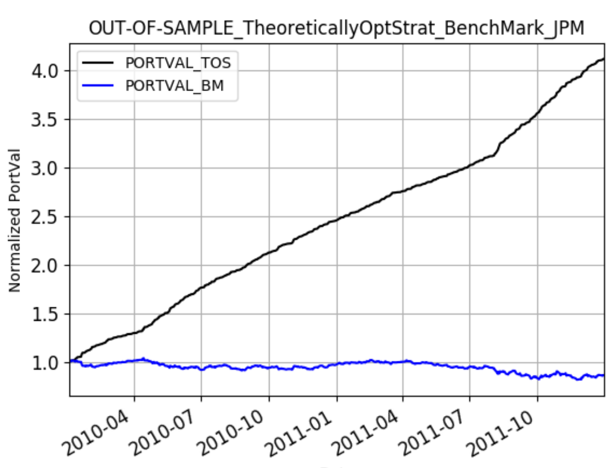
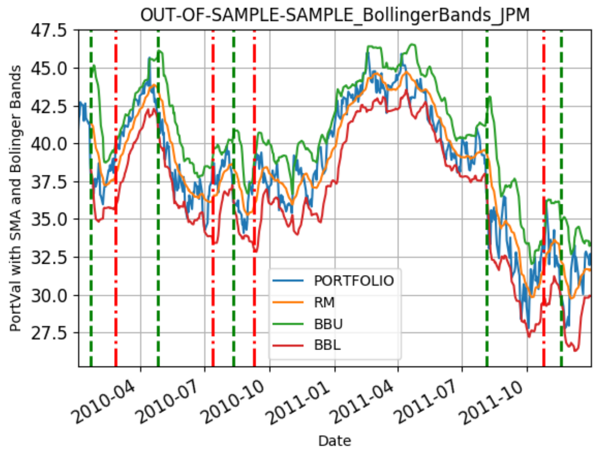
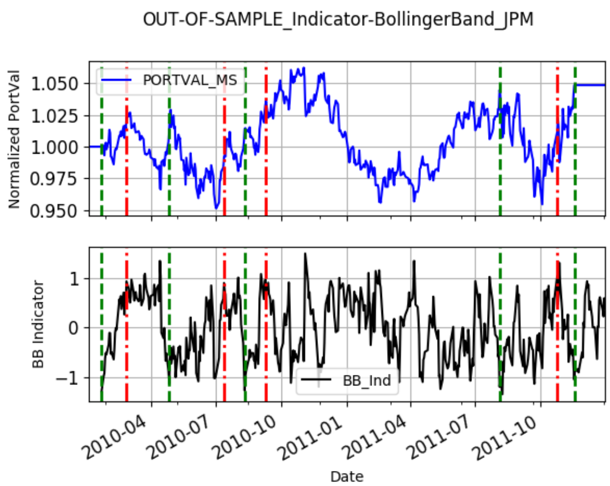


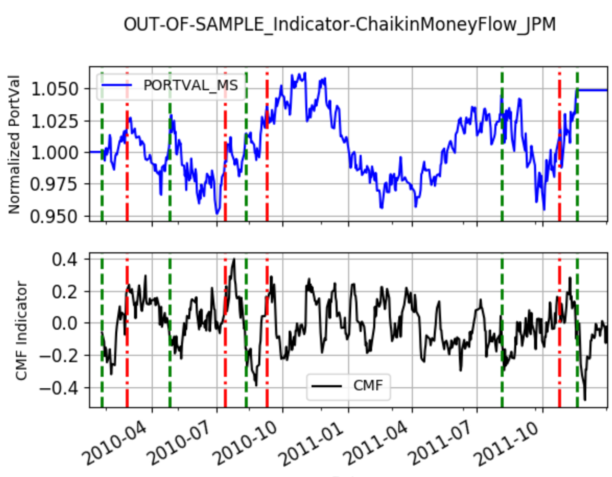
Figure 11: Plot of the BM portfolio (with expenses) normalized to 1.0 at the start (Blue line) and the value of the TOS portfolio normalized to 1.0 at the start (Black line) over the out-of-sample period.

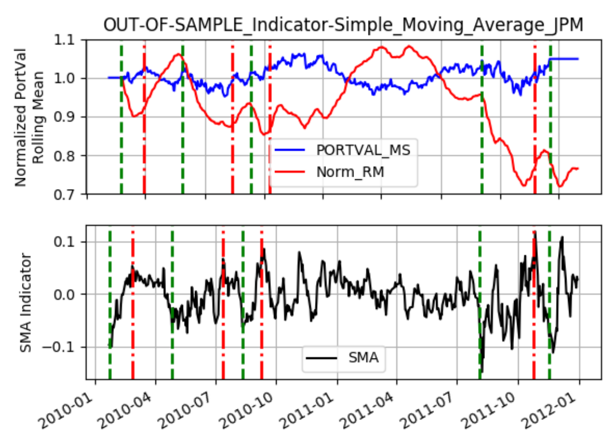
V. APPENDIX: Additional Charts for Comparative Analysis



Figures [12, 13, 14]: Plots of indicators identifying long/short portfolio positions over the out-of-sample period. From top: Bollinger Bands® with Rolling Mean, BBI, CMF, SMA.







---

[1] https://en.wikipedia.org/wiki/Technical_indicator