CS 7646

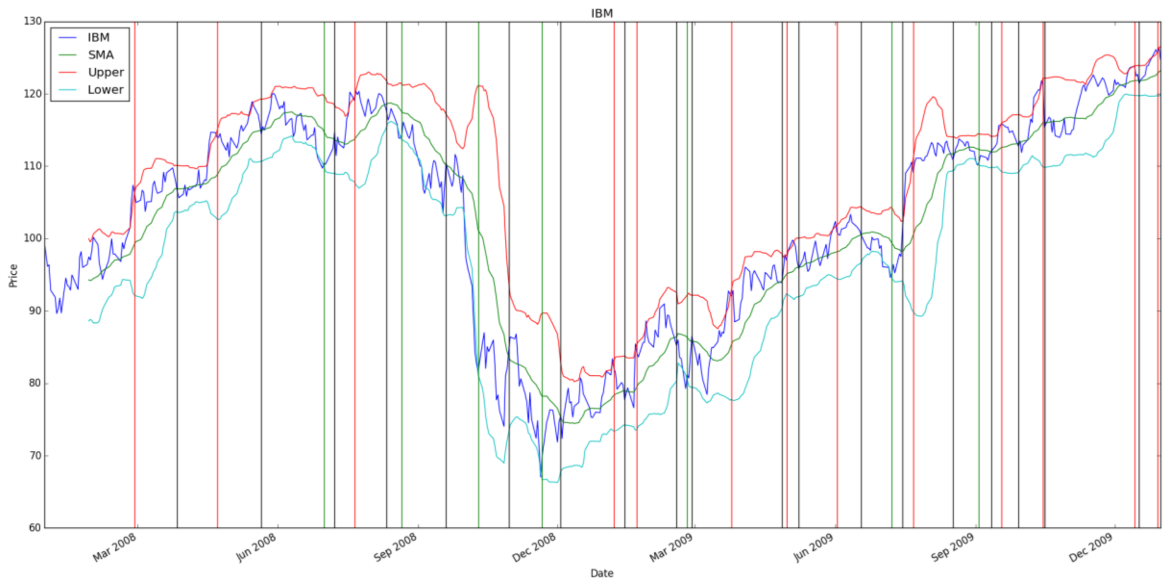
MC2 P2

David Mooney

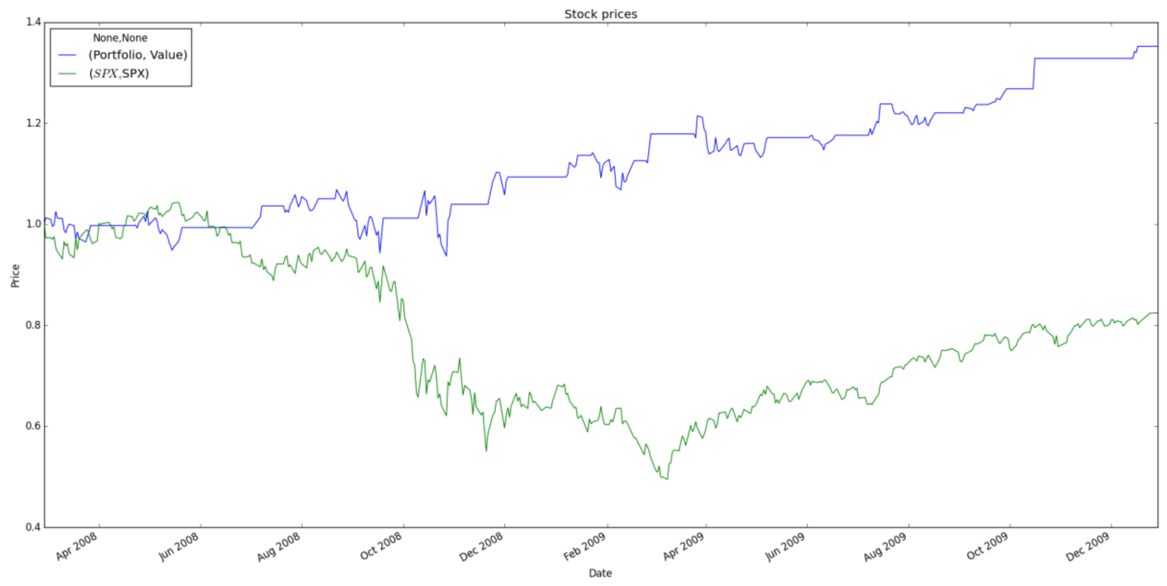
dmooney3

Part 1 Bollinger Bands

Entry and Exit Points



Backtest Chart



Backtest Performance Statistics

Date Range: 2008-02-28 to 2009-12-29

Sharpe Ratio of Fund: Value   1.001959

Cumulative Return of Fund: 0.3524

Standard Deviation of Fund: Value   0.011347

Average Daily Return of Fund: Value   0.000716

Final Portfolio Value: 13524.0

Part 2 My Strategy

My strategy uses the crossing points of short and medium-long term moving averages as signals. The short position would only be taken if the downward crossing signal held for a certain number of days. I experimented with a wide range of values for each variable and settled on a 15-day simple moving average, a 75-day moving average with an 11 day waiting period before taking a short position. This produced 2.34 times the cumulative return of the basic Bollinger strategy in-sample and 2.11 times out-of-sample.

When the 15 day moving average dips below the 75-day moving average, this is a signal to potentially take a short position. We don't actually take the short position unless the following are all true:

1. The 15-day moving average was **above** the 75-day moving average 11 days ago,
2. The 15-day moving average was **below** the 75-day moving average 10 days ago (i.e. it crossed below 10 days ago), and
3. The 15-day moving average is **below** the 75-day moving average today.

If these three hold true, we take a short position whether we are in cash or long. In other words, if we are long, we sell and also take a short position (sell twice).

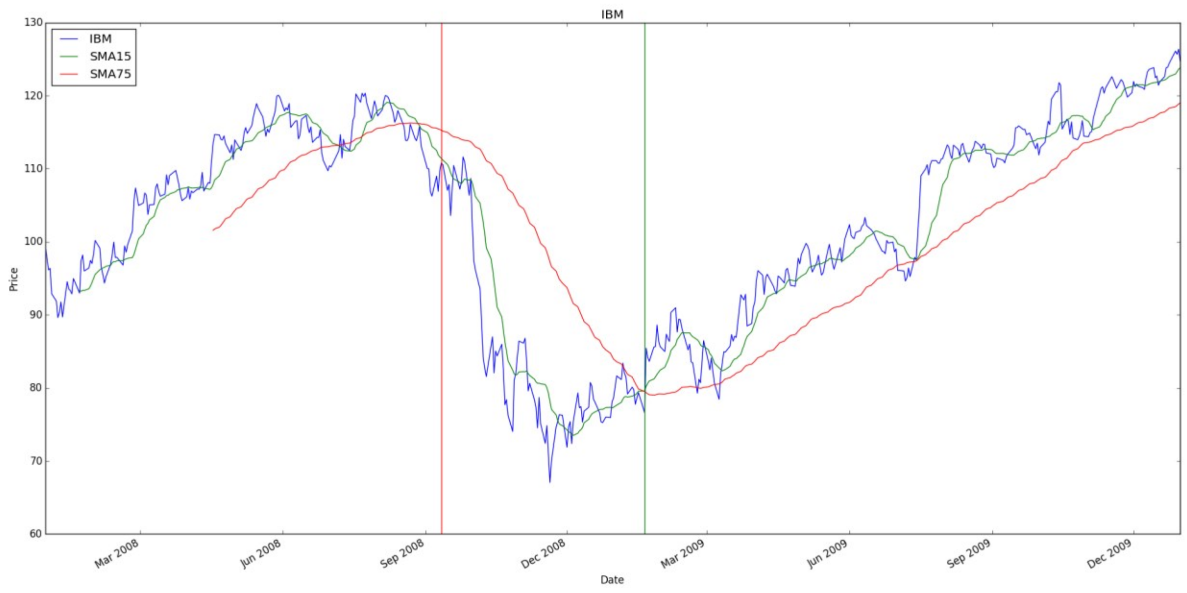
We take a long position as soon as the 15-day moving average crosses back above the 75-day moving average. There is no waiting period. We also never go from cash to a long position. We only go directly from short to long, i.e. buy twice.

If we are still holding any position at the end of the time period, we buy or sell to go to cash.

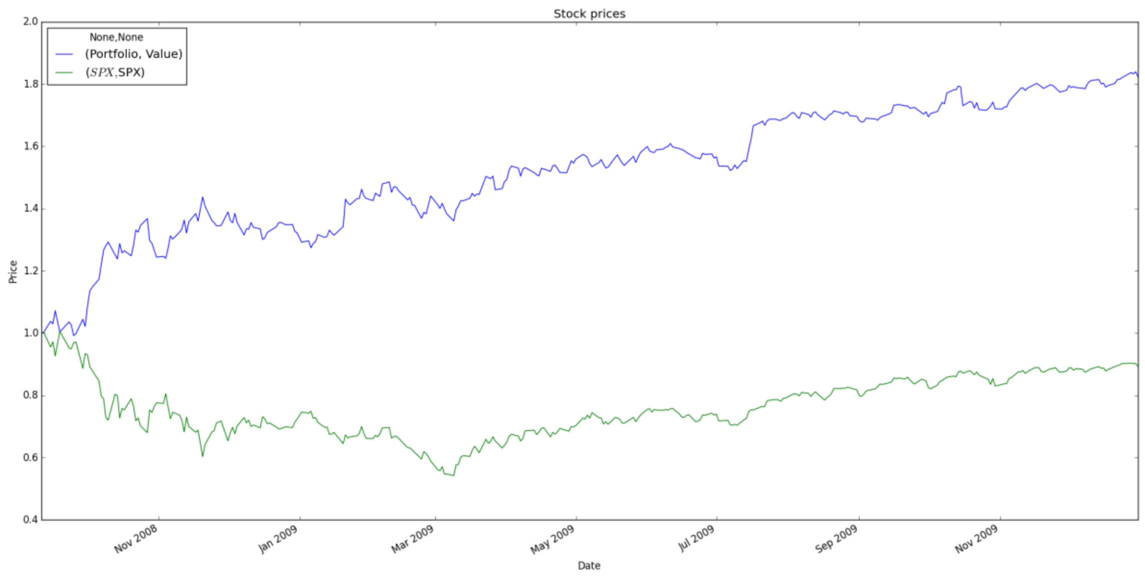
Entry and Exit Points In-Sample

For IBM during 2008 and 2009, this results in just a few trades:

1. Short before the big drop in late 2008
2. Change from short to long (buy twice) in early 2009
3. Sell at the end of the time period.



Backtest Chart In-Sample

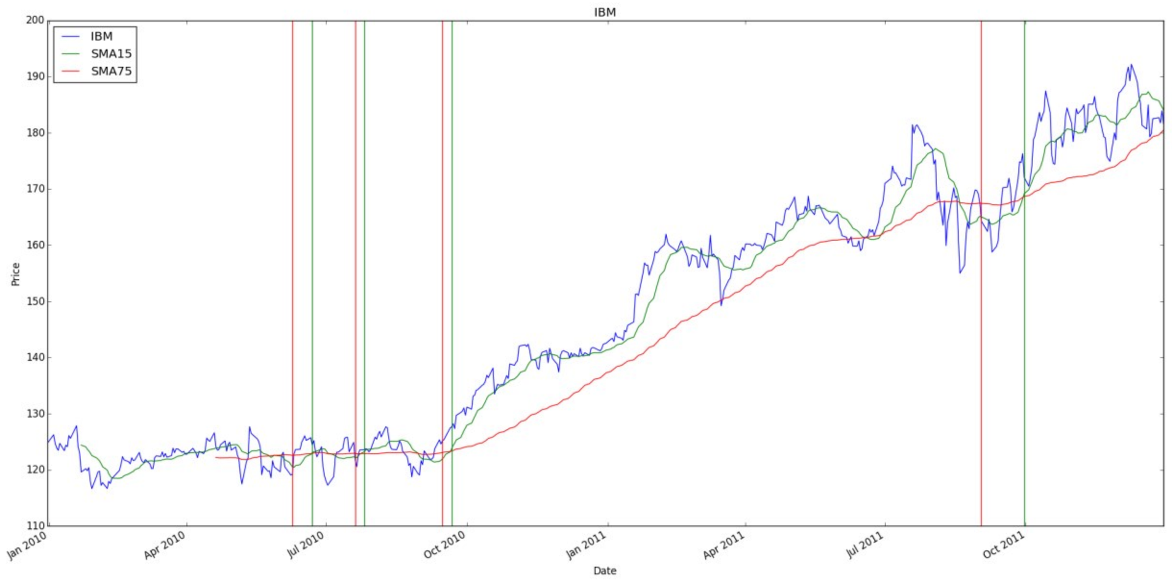


In-Sample Backtest Performance Statistics

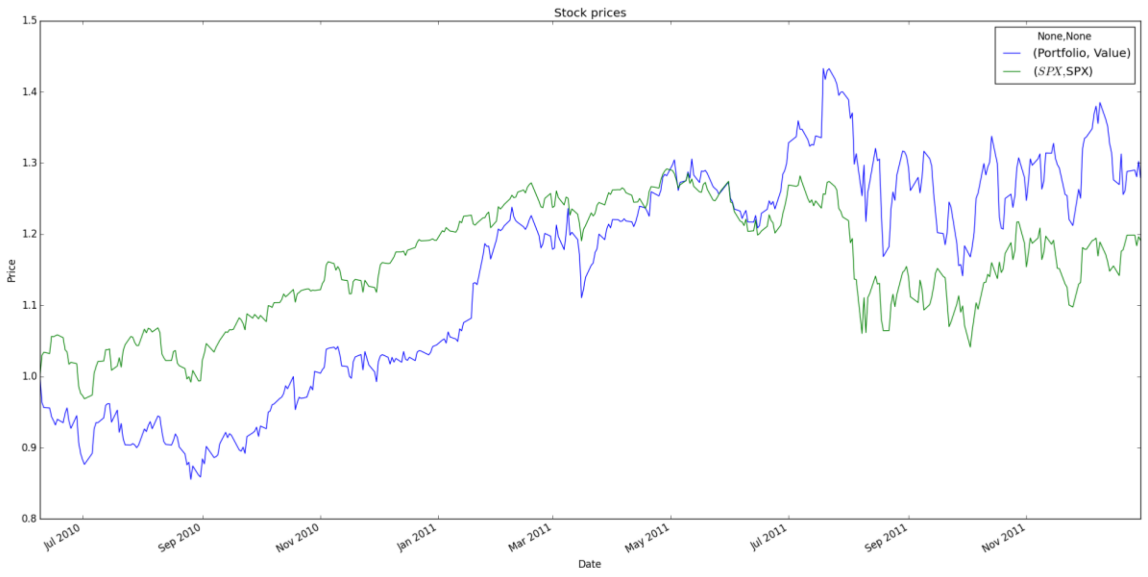
Date Range: 2008-09-11 to 2009-12-31
Sharpe Ratio of Fund: Value   2.090654
Cumulative Return of Fund: 0.8233
Standard Deviation of Fund: Value   0.014686
Average Daily Return of Fund: Value   0.001934
Final Portfolio Value: 18233.0
Performance vs baseline: 2.33626560726

Out-of-Sample Entry and Exit Points

The strategy also works well out of sample. We enter and exit positions and without moving much while the stock goes horizontally. A more sophisticated version of this strategy would probably do better staying in cash if it looked at the horizontal trajectory of the 75-day SMA. However, we correctly take long positions while the stock is moving upward for five straight quarters. This outperforms the basic Bollinger strategy significantly and slightly outperforms SPX.



Out of Sample Backtest Chart



Out of Sample Performance Statistics

Date Range: 2010-06-09 to 2011-12-30

Sharpe Ratio of Fund: Value   0.707273

Cumulative Return of Fund: 0.2795

Standard Deviation of Fund: Value   0.017406
Average Daily Return of Fund: Value   0.000776
Final Portfolio Value: 12795.0
Performance vs baseline: 2.21825396825

Discussion

What do you think of refining and testing your strategy over the same 2 years? Is that a good practice? Why or why not? Does the strategy continue to work as well out of sample? Why?

It is machine learning best practice to use a train on some data and test on another, so what we did here fits with that. The reason for this is to avoid creating a strategy or model that is not general and only works on the test data by attempting to match its behavior exactly. My out of sample performance worked out well but I think that was a very lucky accident. That said, what I did was almost certainly overfitting the sample. I did not try the strategy on anything other than IBM during 2008-2009 and 2010-2011.