MC3-Project-2 - ML Trading Strategy

Magahet Mendiola

November, 2015

Overview

We will evaluate the use of machine learning algorithms in implementing trading strategies. Specifically, we will evaluate the use of KNN on forecasting future asset prices given a set of technical indicators.

Regression Task

The following indicators were used as features in the regression task:

1. Bollinger Band Value w/ 20-day SMA/STDEV
2. 5 Day Price Momentum
3. 20 Day Rolling Standard Deviation

The regression task was set to predict the 5 day future asset price change. KNN was used with K=3 and euclidean distance as the metric.

Trading Strategy

The predicted future price change was used to create a trading strategy that would enter a long position of 100 shares whenever the predicted change went over 1%, entered a short position of -100 shares when predicted change dropped below -1%, and exited any long/short positions when predicted change fell between 1% and -1%.

ML4T-399 Dataset

2008-2009 Time Period

Regression Results

Figure 1 illustrates actual 5 day future price, predicted future price, and current price on the ML4T-399 dataset. We see that future prices, both actual and predicted, correctly lag 5 trading days behind current prices. We also see that predicted future prices and actual future prices visually match. This is validated by low cumulative error between predicted and actual Y values (RMSE: 0.0004, corr: 0.99).

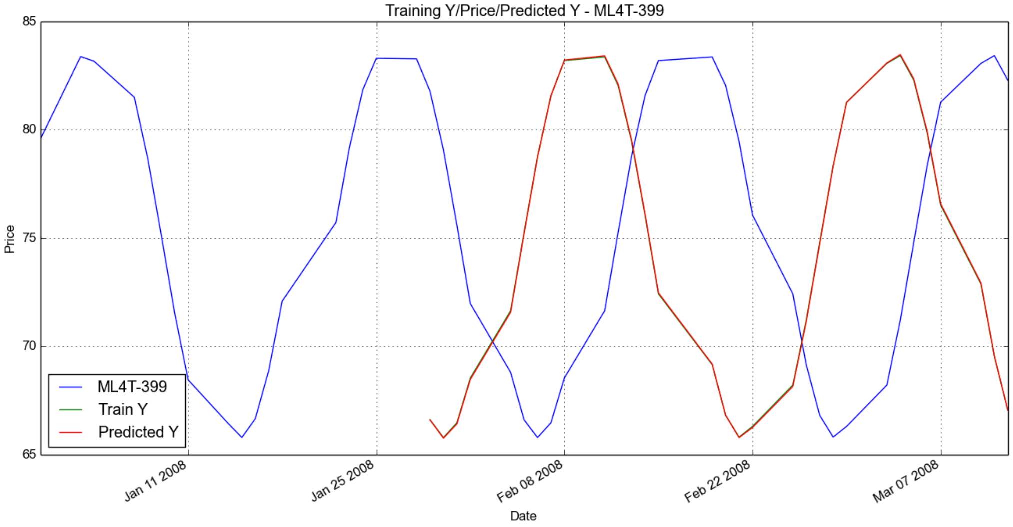

Figure 1: Training Y/Price/Predicted Y - ML4T-399 '08-'09

Generated Trades

Figure 2 shows the entry and exits points for period 2008-2009. The plot has asset prices on top and the predicted 5 day price change on the bottom. We can see how the trading strategy is entering long positions as this value moves above 0.01. Likewise, short positions are shown in red when the predicted change falls below -0.01. We can see in the top plot that this strategy does capture much of the movement, both up and down, in the asset price. However, it could be tuned further as it currently looses some value in the threshold window 0.01 to -0.01. It would be possible to remove this window completely and trade strictly on movements above and below a change of 0.0. This would likely cause our regression trading strategy to perform more poorly on data without such obvious trends.

Trading Strategy Backtest

Figure 3 shows the performance of our trading strategy for ML4T-399 2008-2009. We see a cumulative return of 7.7 (-0.2 on SPY). The Sharpe ratio of our portfolio is 9.3 (-0.2 on SPY).

2010 Time Period

Generated Trades

The KNN learner was trained on ML4T-399 data during the 2008-2009 time period. Pricing data from 2010 was then loaded and technical indicators calculated on this dataset. Using the 2008-2009 trained learner, queries were made with the 2010 attributes. As before, trades were generated using the +/- 0.01 trading signal.

Figure 4 shows the results of these generated trades. As before, the strategy does well at capturing the movement in asset price. Error on this test set is: RMSE: 0.0018, Corr: 0.99, indicating that our KNN learner performed quite well in predicting price changes with test samples outside the original training data.

Trading Strategy Backtest

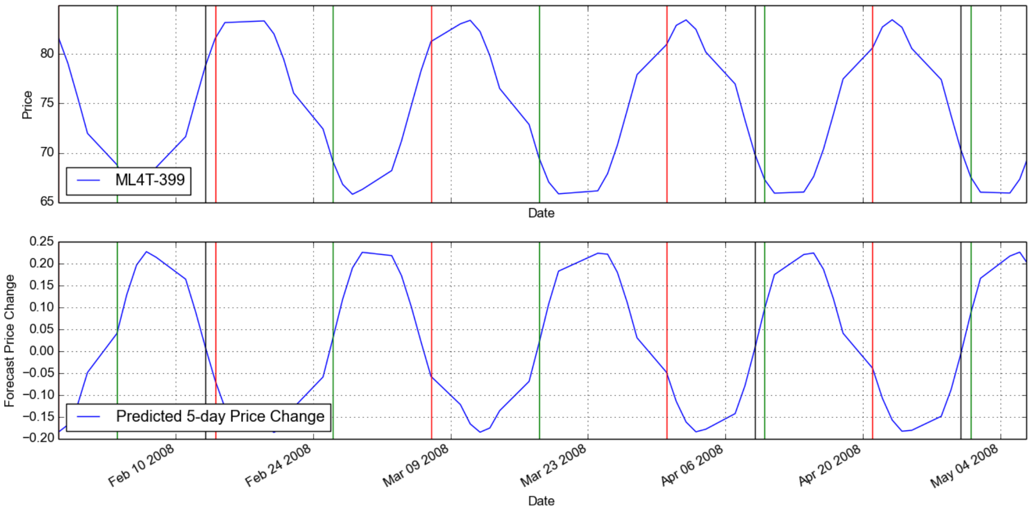

Figure 2: Entries/Exits - ML4T-399 '08-'09

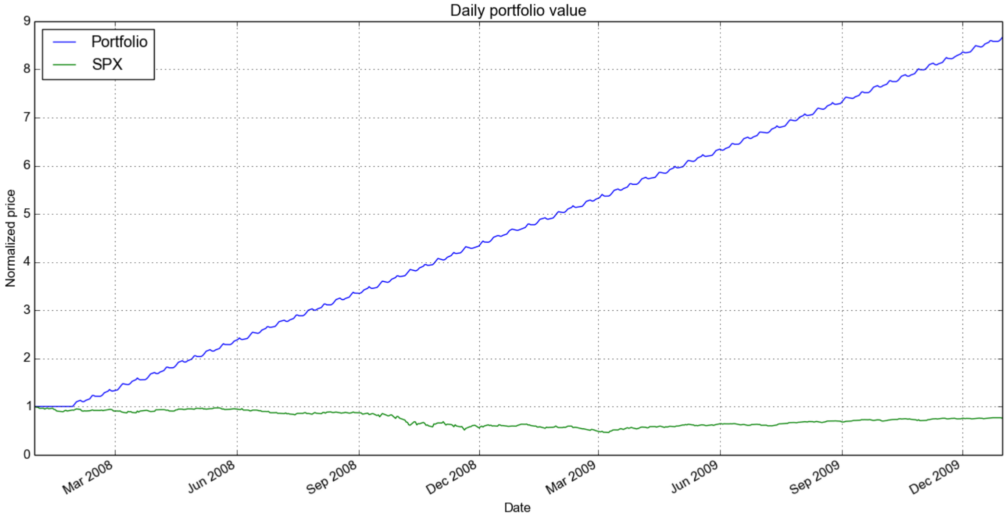

Figure 3: Backtest - ML4T-399 '08-'09

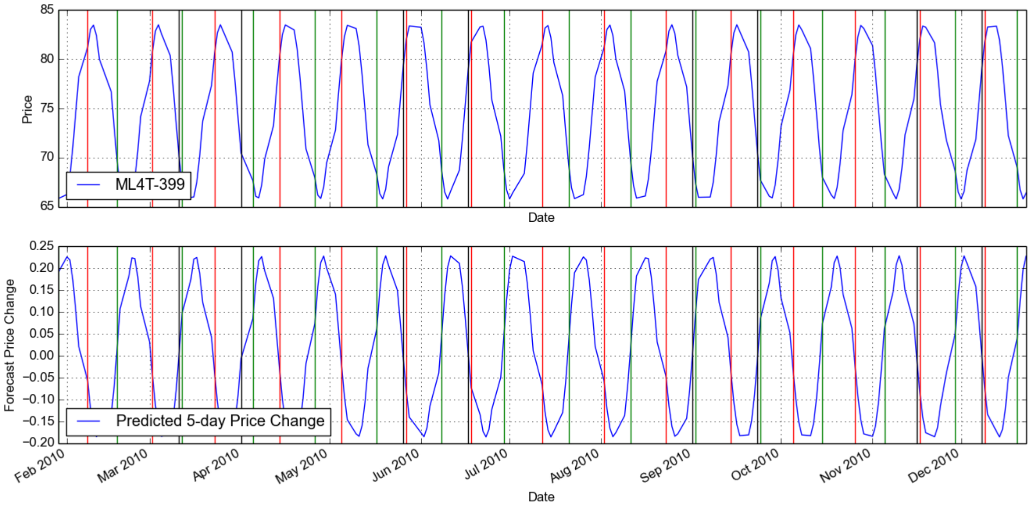

Figure 4: Entries/Exits - ML4T-399 '10

Figure 5 shows the performance results of our 2010 portfolio as created with the KNN based trading strategy. We see a cumulative return of 3.7 (0.1 on SPY) and a Sharpe ratio of 10.4 (0.8 on SPY).

ML4T-399 Dataset

2008-2009 Time Period

Generated Trades

Using the same learning and trading strategy on IBM during 2008-2009 shows positive, though not overly impressive, results. Figure 6 shows generated orders, IBM prices, and predicted price change. We see that the trading indicator is working properly, as trades are generated as the predicted price change moves above and below the thresholds. In sample error results show an RMSE of 0.03 and Corr of 0.697. This is not as close as the ML4T-399 predictions, but still quite reasonable.

RMSE, in this case, can be informally interpreted as the confidence in predicted price change. Intuitively, we can say that the 5 day price change will be a given value as determined by our KNN regression learner. We can also say the predicted change will be off by an average of 0.03. This error could be used to improved the trading signal used by the trading strategy. We could trade only on moves greater than RMSE, which would prevent trades based on predicted values within the error bounds.

Trading Strategy Backtest

IBM Data In Sample Backtest

Figure 7 shows the portfolio performance using our KNN based trading strategy on IBM over 2008-2009. We see reasonable results, with a cumulative return of 2.0 (-0.2 on SPY) and a Sharpe ratio of 3.7 (-0.2 on SPY). These results show the positive effect of trading on the predicted price from our learner.

2010 Time Period

Generated Trades

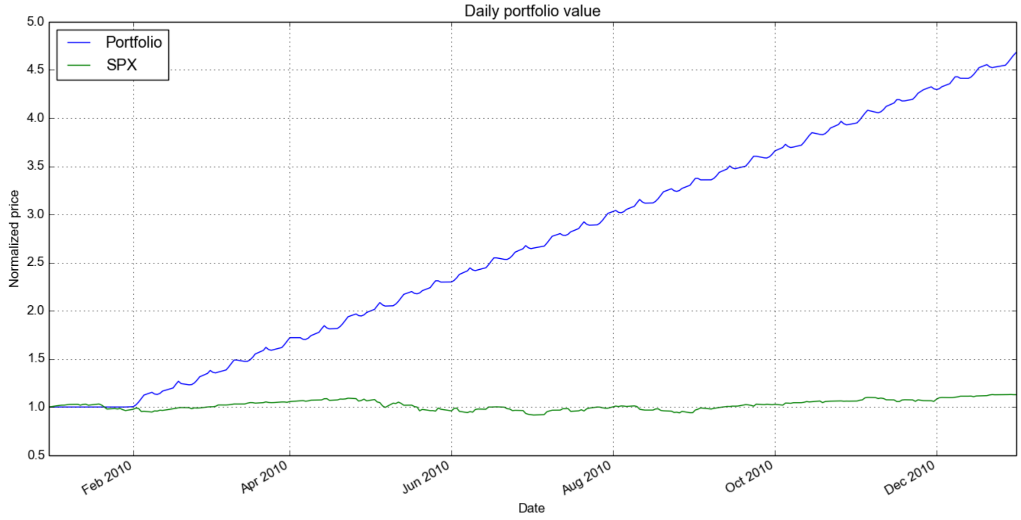

Figure 5: Backtest - ML4T-399 '10

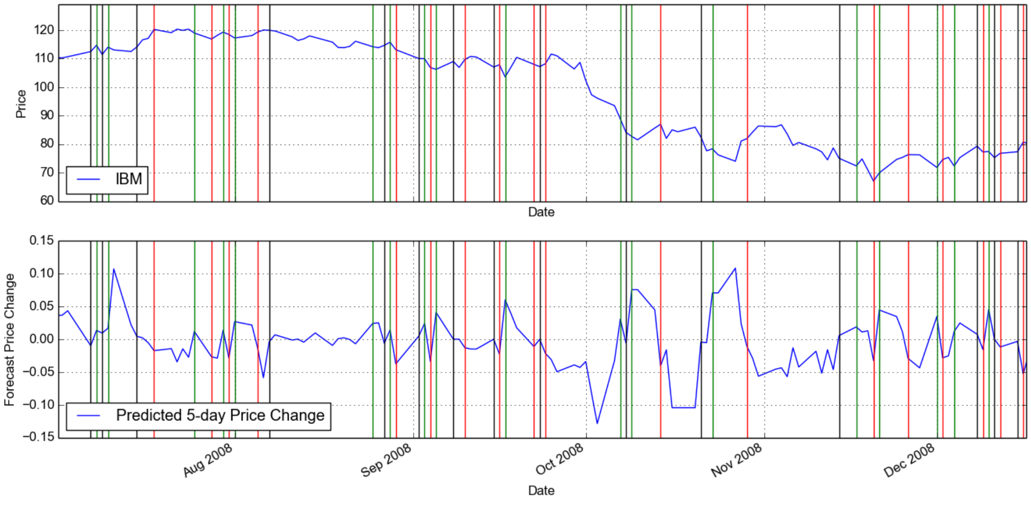

Figure 6: Training Y/Price/Predicted Y - IBM '08-'09

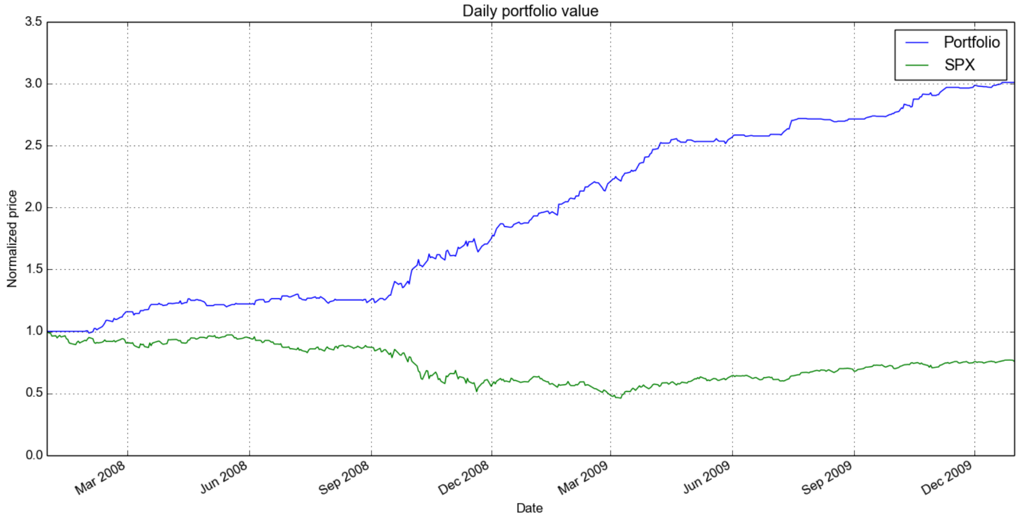

Figure 7: Backtest - IBM '08-'09

The KNN learner, trained with 2008-2009 samples, was used to predict and generate trades for the 2010 time period. We see the resulting orders, along with the predicted price change, on Figure 8. The price change plot shows that our learner predicts a number of large price changes that never actually take place. This is likely due to changing relationships between our trading indicators and predicted price change from 2008-2009 to 2010. Signals which predicted major price changes in a highly volatile market seem to create false signals in a time period with mostly stable prices.

RMSE and Cor between predicted and actual price changes are 0.04 and 0.14 respectively on the 2010 data. This illustrates how poorly our learner predicted price changes after training on 2008-2009 data.

Trading Strategy Backtest

Results on the KNN learner trading in 2010 was predictably very poor given the out of sample error results. Cumulative return was 0.099, compared to 0.128 on SPY. The limitation of our trading strategy is likely due to our learner training on relationships within the data that then changed between the two time periods. This clearly illustrates the danger of assuming historical relationships between technical indicators and predicted asset prices will continue to hold in the future.

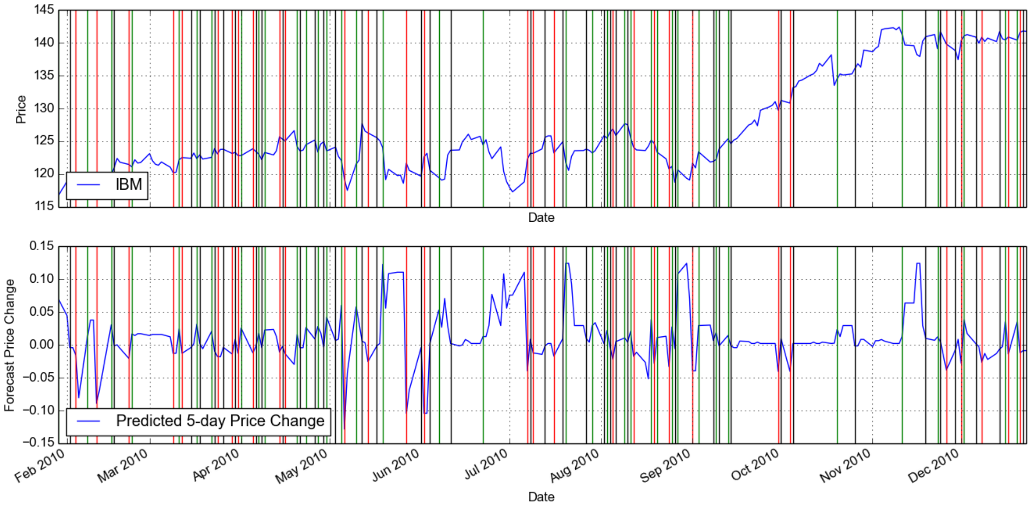

Figure 8: Entries/Exits - IBM ’10

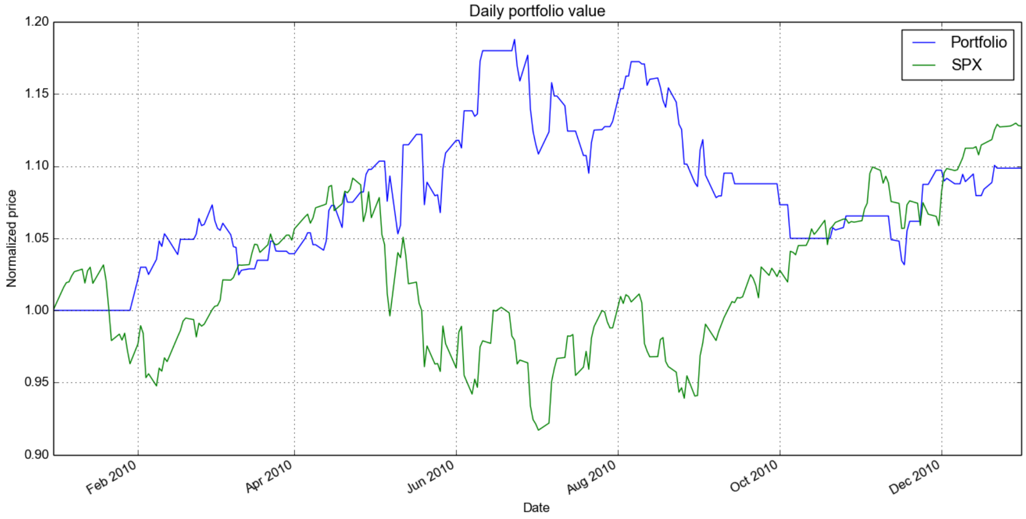

Figure 9: Backtest - IBM ’10