

## 1. Indicators

The indicators used in this project were the ones mentioned in the project description, namely Bollinger Band© value, momentum, and volatility. Here is how each was computed.

### a. Bollinger Band

$$bbValue[t] = (price[t] - SMA[t]) / (2 * stdev[t])$$

Where SMA is the 20 day simply moving average of the stock's adjusted close price at time t and stdev is the standard deviation of the stock adjusted close over the last 20 days.

### b. Momentum

The 5 day momentum of the adjusted close price was used as defined by:

$$momentum[t] = \left( \frac{price[t]}{price[t-5]} \right) - 1$$

### c. Volatility

Five day volatility of the adjusted close price was the final feature. This was just the standard deviation of the stock price over the previous 5 days.

## 2. Trading Policy

The trading policy recommended in the project description was used here. If the 5 day return was predicted to go up 1% or more, a buy order was placed. If the 5 day return was predicted to go down 1% or more a sell order was placed. After 5 days, the opposite of the initial order was made. Thus, if a buy order was placed, 5 days later a sell order was placed. If a sell order was placed, 5 days later a buy order was made.

The learner used was a KNN learner with a k value of 3.

## 3. Sine data

### a. In Sample

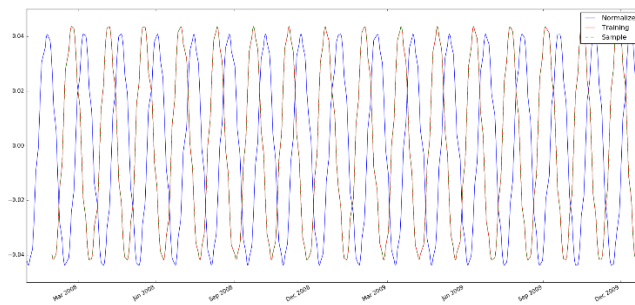


Figure 1: Price, training, and sampled data

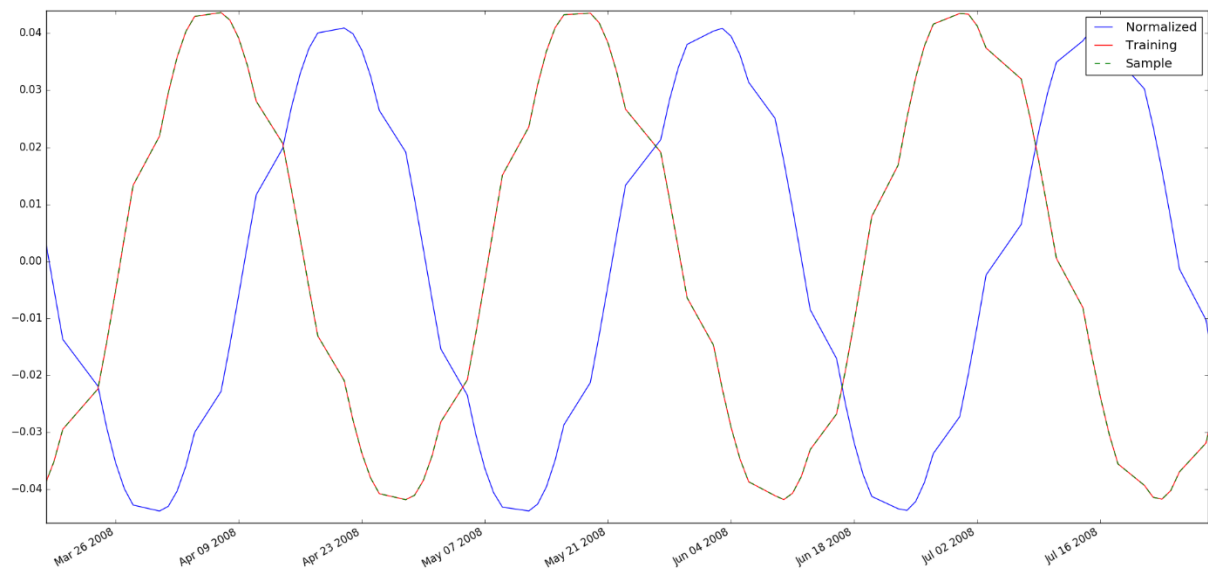


Figure 2: Zoomed view of price, training, and sampled data

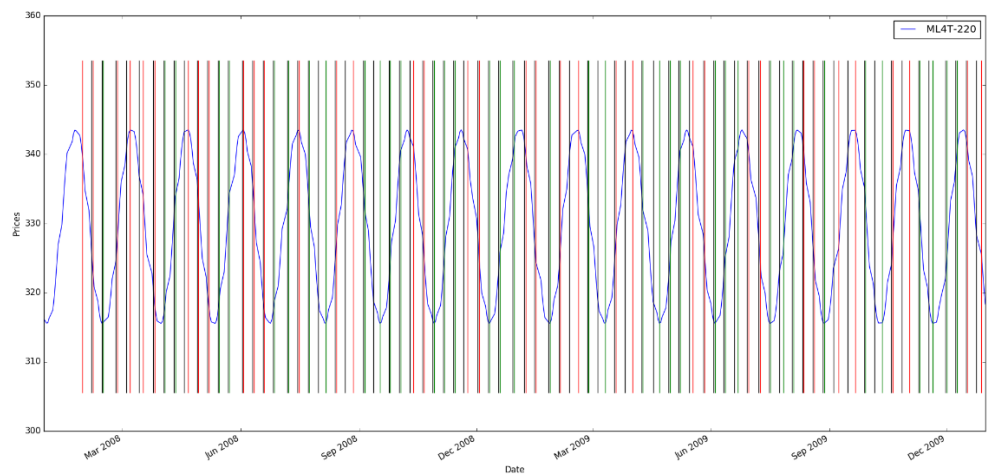


Figure 3: Sine data in-sample trading

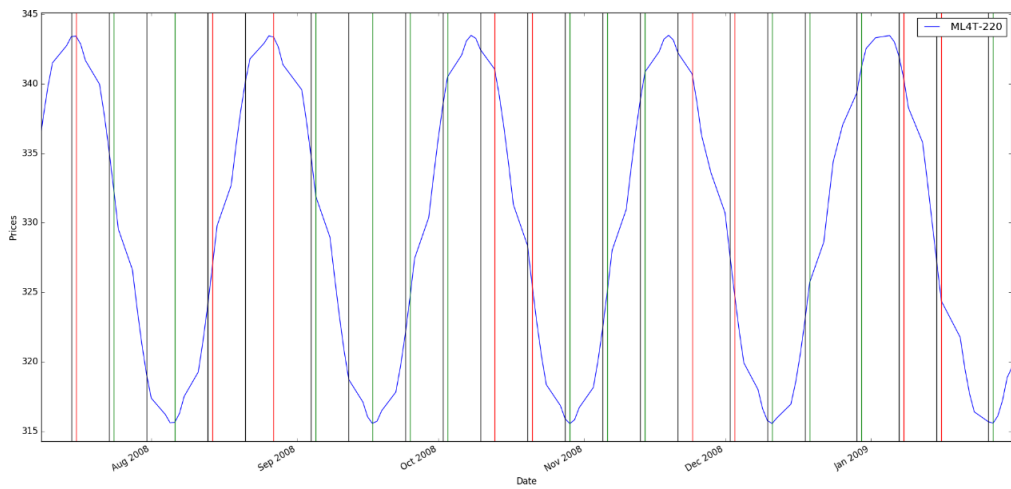


Figure 4: Sine data in-sample trading - zoomed view

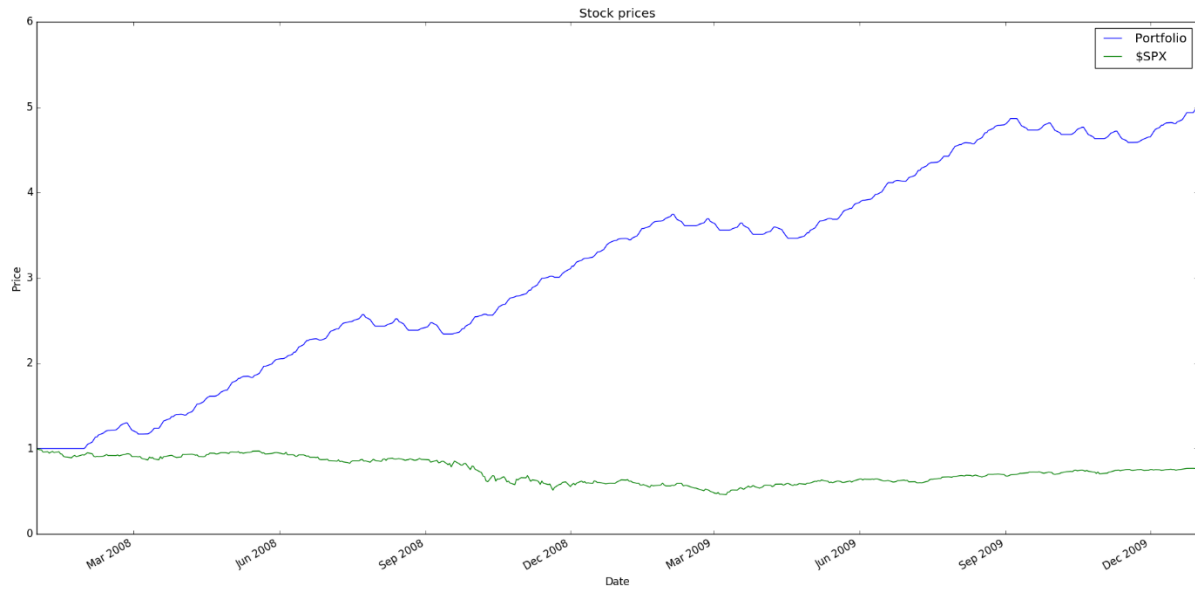


Figure 5: Sine data in-sample backtest performance

Table 1: Sine data in-sample backtest statistics

	KNN Portfolio	\$SPX
Sharpe Ratio	7.25903102978	-0.21996865409
Cumulative Return	4.00684442	-0.240581328829
Standard Deviation	0.00704049843088	0.0219524869863
Avg. Daily Return	0.00321945077007	-0.000304189525556

## b. Out of Sample

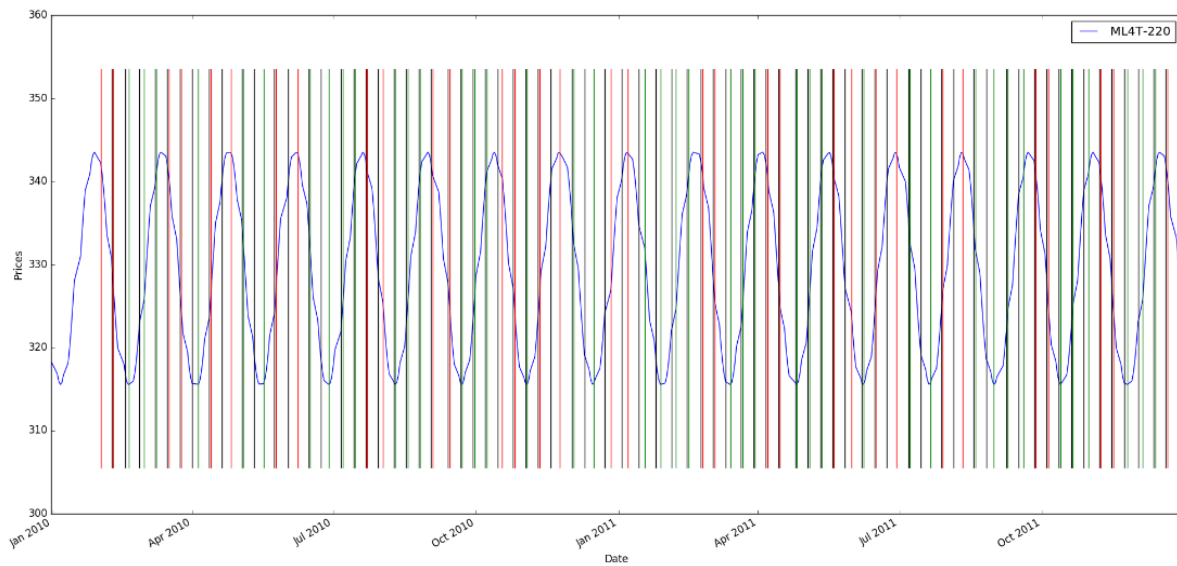


Figure 6: Sine data out-of-sample trading

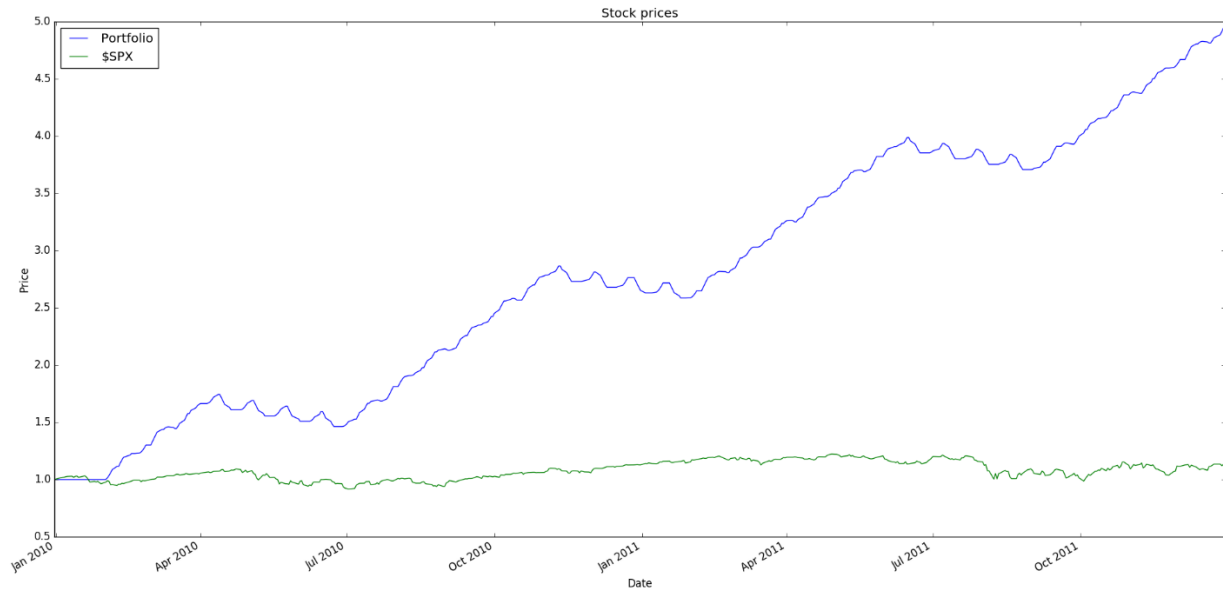


Figure 7: Sine data out-of-sample backtest performance

Table 2: Sine data out-of-sample backtest statistics

	KNN Portfolio	\$SPX
Sharpe Ratio	6.62323125168	0.393165319464
Cumulative Return	3.94422625	0.127791229486
Standard Deviation	0.00768291182691	0.0131086008359
Avg. Daily Return	0.00320549790546	0.000324661859049

### c. Analysis of Results

Figure 1 and Figure 2 show the normalized and shifted price data (so that it is about 0), the training data (5 day future return), and the sampled data used with the KNN learner. The training data can be seen to be shifted 5 days back from the price data. Also, the sampled data practically matches the training data.

The results for the in-sample test were exactly as expected. The KNN learner was able to predict the 5 day return very accurately. The backtest performance for the in-sample set was amazing because the behavior of the stock was very predictable and the KNN learner was able to capitalize on this behavior.

Most of the analysis above also applies to the out-of-sample backtest. Since the “stock price” behaved exactly the same for the out of sample data, the KNN learner was able to accurately predict the future return and the performance in Figure 7 reflects this.

## 4. IBM Data

### a. In Sample

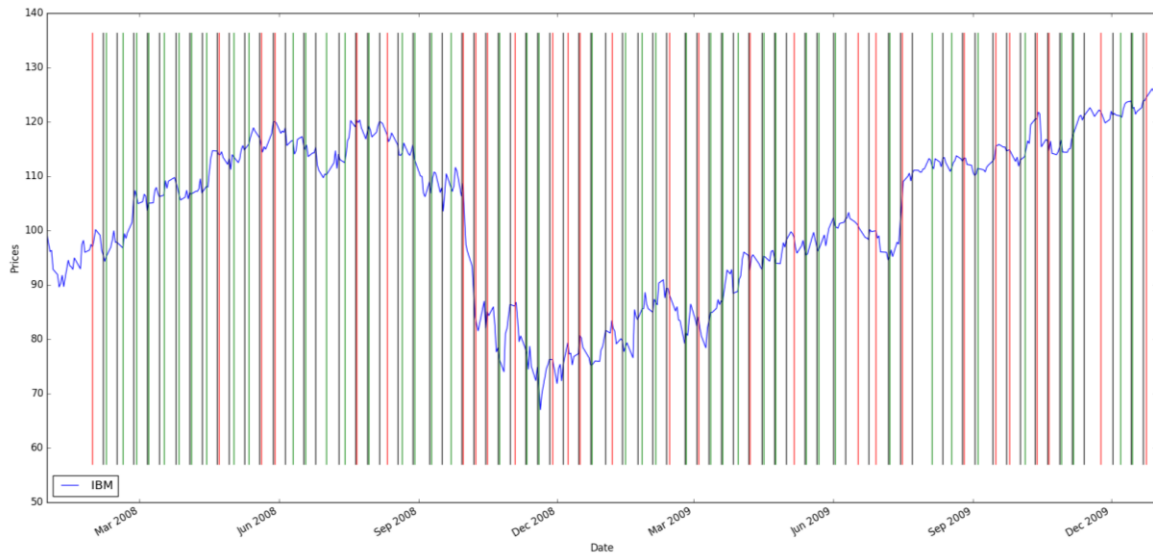


Figure 8: IBM data in-sample trading

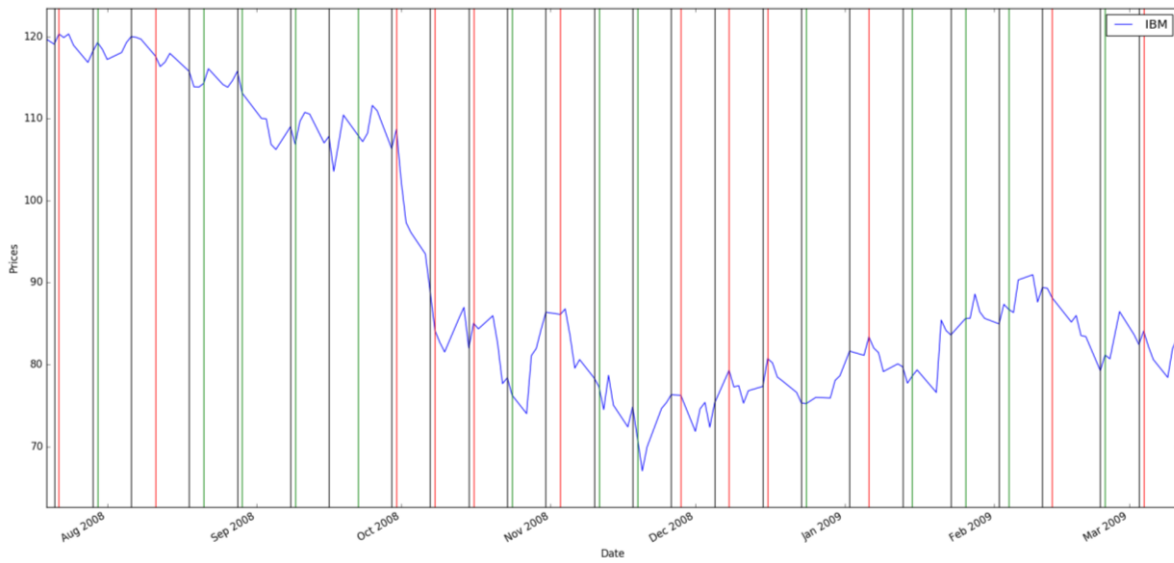


Figure 9: IBM data in-sample trading zoomed view

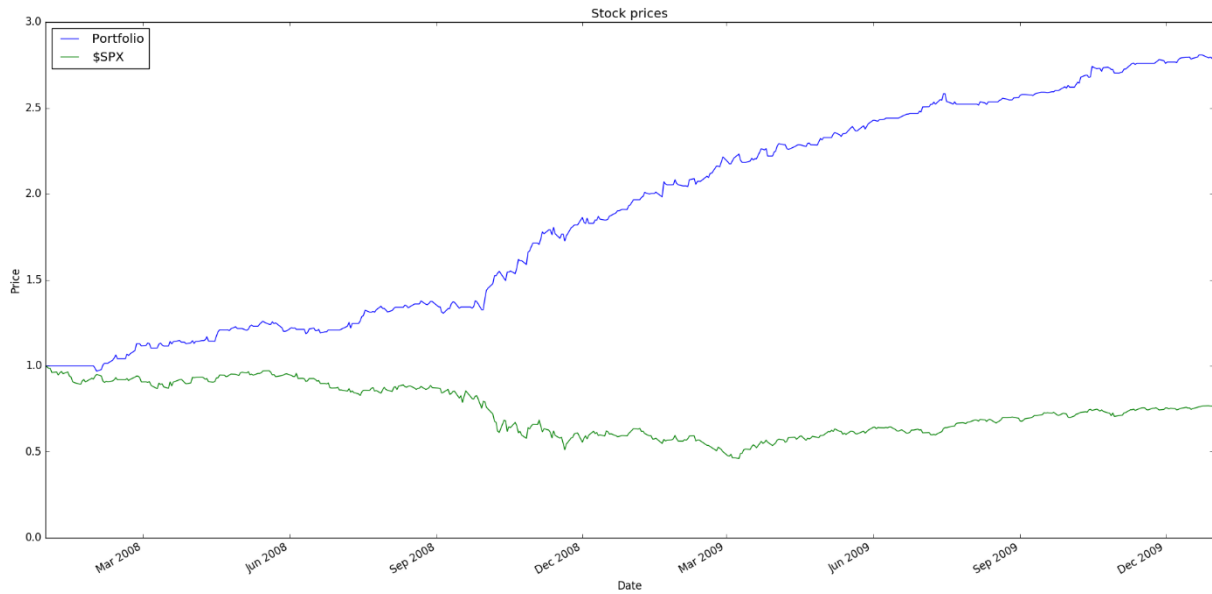


Figure 10: IBM in-sample data backtest performance

Table 3: IBM in-sample backtest performance statistics

	KNN Portfolio	\$SPX
Sharpe Ratio	3.37293130035	-0.21996865409
Cumulative Return	1.8047	-0.240581328829
Standard Deviation	0.00984754069168	0.0219524869863
Avg. Daily Return	0.00209235325655	-0.000304189525556

## b. Out of Sample



Figure 11: IBM out-of-sample trading



Figure 12: IBM out-of-sample data backtest performance

Table 4: IBM out-of-sample backtest performance statistics

	KNN Portfolio	\$SPX
Sharpe Ratio	-0.0358579773671	0.393165319464
Cumulative Return	-0.0911	0.127791229486
Standard Deviation	0.0173479262842	0.0131086008359
Avg. Daily Return	-3.9186194199e-05	0.000324661859049

### c. Analysis of Results

Figure 10 shows the in-sample performance of the KNN trader using IBM as the stock being traded. The performance is phenomenal because the training data was able to accurately predict the future return. This is effectively cheating because the data used to train the KNN learner would not be available when actively trading this stock.

Figure 12 shows the out-of-sample performance of the KNN trader using the training data from the two years prior. This is a more realistic trading example. Unfortunately the trader does not perform as well here throughout the entire sample period as with the in-sample case. For the first year the performance is quite impressive, but following that the market must have shifted in some way such that the indicators and features that were used do not accurately predict the market behavior. While this is unfortunate it is not unexpected. Market behavior can change and this KNN trader does not account for this.

To improve performance, the proper features must be selected. For the sine data in section 3 almost any set of features will do, but using the less predictable market data requires more finesse. This is not an easy task and has taken skilled people much effort to do the same. This is not necessarily a deficiency in the KNN learner. The KNN trader itself performs quite well as shown in Figure 5 and Figure 7 when supplied with the proper data. I would not change the KNN trader, but I would attempt to select better features.

