An Analysis of Backtesting Accuracy

William Guo

July 28, 2017

Rice Undergraduate Data Science Summer Program

Motivations

Background

Financial markets are, generally speaking, very noisy and exhibit non-strong stationarity [1].

- The probability distribution of a financial instrument's returns changes over time.
- Problematic when one wants to understand and predict the future behavior of said instruments.

Forms of stationarity

Strong: The distribution function of vector $(y_1, y_2, \cdots, y_k)$ is equal to that of $(y_{1+h}, y_{2+h}, \cdots, y_{k+h}$ for **all** finite sets of indices $(t_1, t_2, \cdots, t_k) \in \mathbb{Z}$.

Approximately strong: The distribution function of vector $(y_1, y_2, \cdots, y_k)$ is equal to that of $(y_{1+h}, y_{2+h}, \cdots, y_{k+h}$ for **some** finite sets of indices $(t_1, t_2, \cdots, t_k) \in \mathbb{Z}$.

Weak:

$$\begin{cases} 1) & E[y_t] = \mu \quad \forall\ t\ \in \mathbb{Z}^+ \\ 2) & E[y_t^2] = \sigma^2 \quad \forall\ t \in \mathbb{Z}^+ \\ 3) & cov(y_{t_1}, y_{t_2}) = cov(y_{t_1+h}, y_{t_2+h}) \end{cases}$$

Backtesting

- In this context, backtesting is the process of applying a trading strategy to historical data and having the strategy mimic how it would have performed.
- The fundamental assumption of backtesting a market trading strategy is that the market exhibits approximately strong stationarity over an interval of time.

Backtesting Properties

Pros:

- Easy to understand and implement

Cons:

- Historical data alone provides a poor simulation of the past (e.g. no exactly accurate representation of slippage and transaction fees)
- **Market data often does not exhibit strong stationarity**

What Makes a Backtest Reliable?

The main objective of backtesting financial trading strategies is to see how they would have performed in the past.

- If it performed well in the past, then we can (broadly) assume that it will perform well again in the future.
- If this assumption fails to hold, however, the strategy's performance over historical data may be uncorrelated with the strategy's future performance.

Project Goal

Our goal is to test the strength of our assumption that market time series data exhibits approximately strong stationarity by seeing how accurate our results are. We do this by running a pseudo-Monte Carlo analysis on our strategy's returns. More specifically, we don't make any specific assumptions about the market.

- As such, most facets of our testing are fixed, like our trading strategy, stock ticker (IBM), period of time (1993 - 2017), and so forth.

Project Goal (cont.)

We change the following backtesting parameters:

- Different performance measures; the more complicated, the less stationary they will be.
- Sampling frequency of performance measures
- Forecasting training window

It's important to not, however, that these parameters are not drawn randomly; each variable parameter is tested only once. Furthermore, their limits are bounded within conventional windows of time (1-5 years).

Project Design

Outline

- Perform multiple back tests of a simple Bollinger Bands strategy from 1990 to 2017
- Use walk-forward testing to train our forecasting model for different period of time starting at a variety of different points in time.
- Forecast our strategy's performance metrics
- Measure the accuracy of our forecasts

Tooling

Performance metrics

I chose two performance metrics: the information ratio and maximum drawdown. The information ratio represents upside risk, while maximum drawdown represents downside risk.

$$IR = \frac{R_p - R_i}{\sigma_{p-i}}$$

$$MDD = \frac{min(R_p) - max(R_p)}{max(R_p)}$$

where R_p = cumulative portfolio returns, R_i = cumulative benchmark/index returns (in this case, the S&P 500), σ_{p-i} = standard deviation of $R_p - R_i$.

Bollinger Bands strategy

My strategy is an example of a mean reversion strategy.

- If the closing price of a ticker exceeds 2 standard deviations above its 21-day moving average, the market is **overbought** and the price will decrease.
- On the other hand, if the closing price of a ticker falls 2 standard deviations below its 21-day moving average, the market is **oversold** and the price will increase.
- My strategy will enter and long 500 shares of the ticker in question when the market is oversold and exit any long position when the market is overbought.

Forecasting

I used a simple unweighted average forecasting model.

$$\hat{y}_{T+h} = \frac{1}{T}\sum_{t=1}^{T} y_t \quad \forall\ h \in \mathbb{Z}^+ \tag{1}$$

- This model takes the unweighted average of the sample data; all forecasted values are set to equal this average.

Comparison to naive forecasting model

To understand the strength and accuracy of our forecasts, we calculated the mean absolute error for each forecasting model.

$$MAE = \frac{1}{n}\sum_{j=1}^{n}|e_j|$$

- Provides a scale-dependent measure of our forecasts' errors
- Allows us to compare the relative strength of forecasts of a specific metric

Results

Max Drawdown Sampling Frequency

- Separating forecasts by metric sampling frequency reveals no real "best" sampling frequency, but annually sampled data returned best results

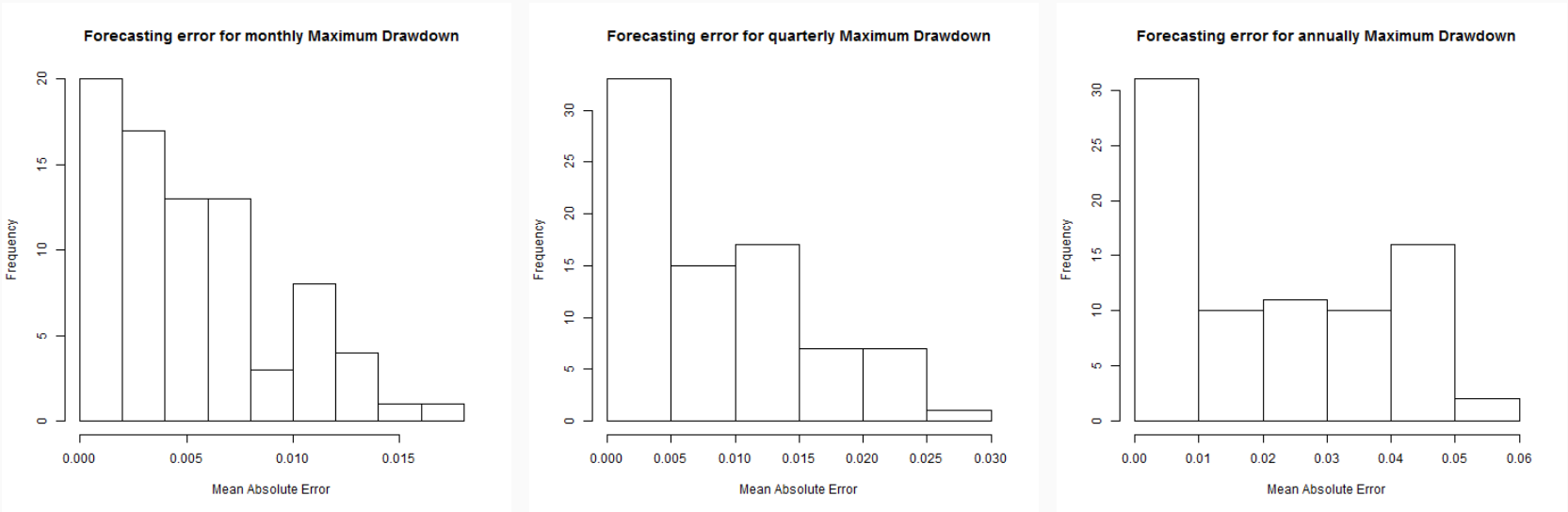

Fig 1: Histograms of forecasting errors separated by metric sampling frequency.

Max Drawdown Testing Length

If we separate the errors by the length of the testing period, we find that all of the error distributions tend to peak around 0.05-0.015.

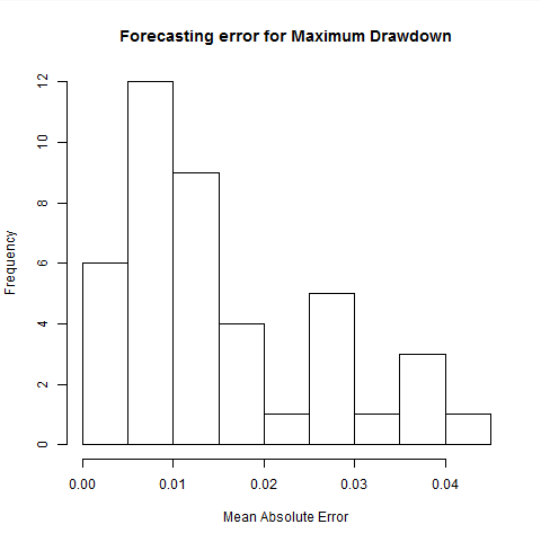

Fig 2: 5 years

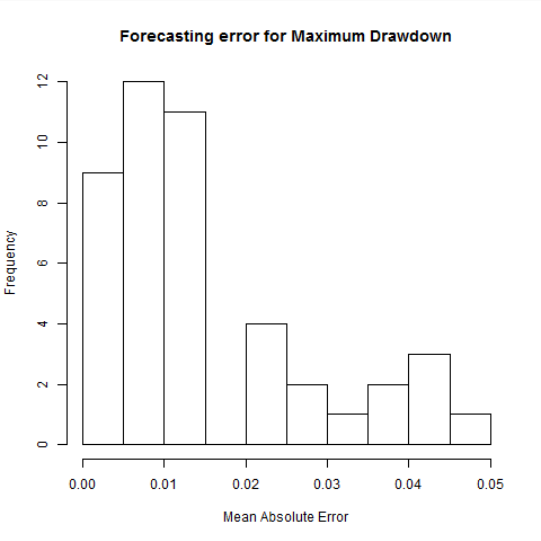

Fig 3: 4 years

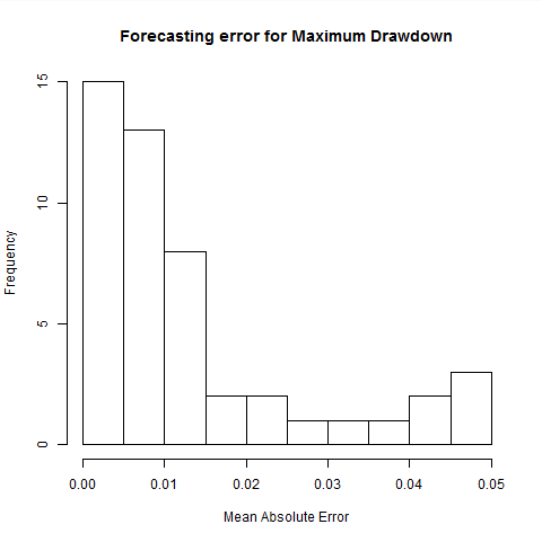

Fig 4: 3 years

Max Drawdown Testing Length (cont.)

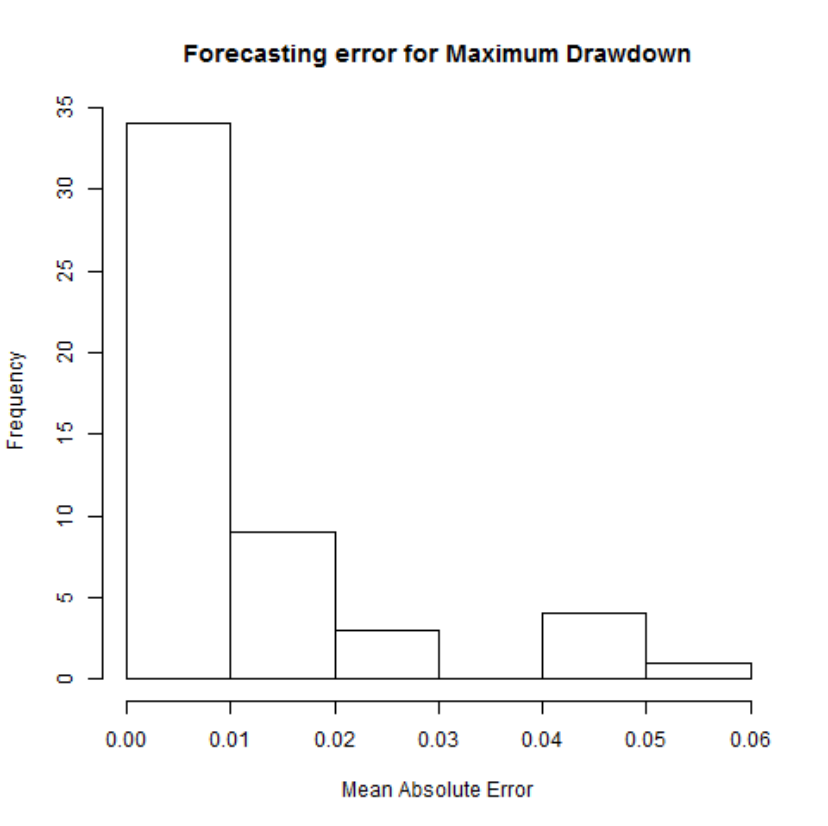

Fig 5: 2 years

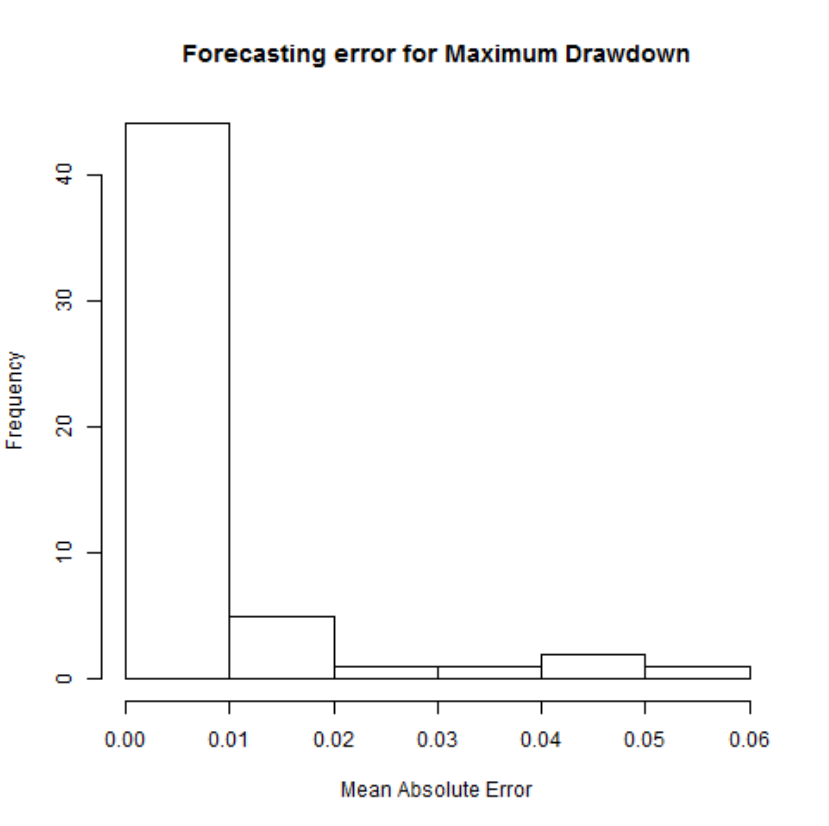

Fig 6: 1 year

Information Ratio Sampling Frequency

Comparing the sampling frequency of the information ratio, there is a distinct increase in the mean absolute error of the forecasts. As with max drawdown, as the sampling frequency decreases, the errors decreases as well.

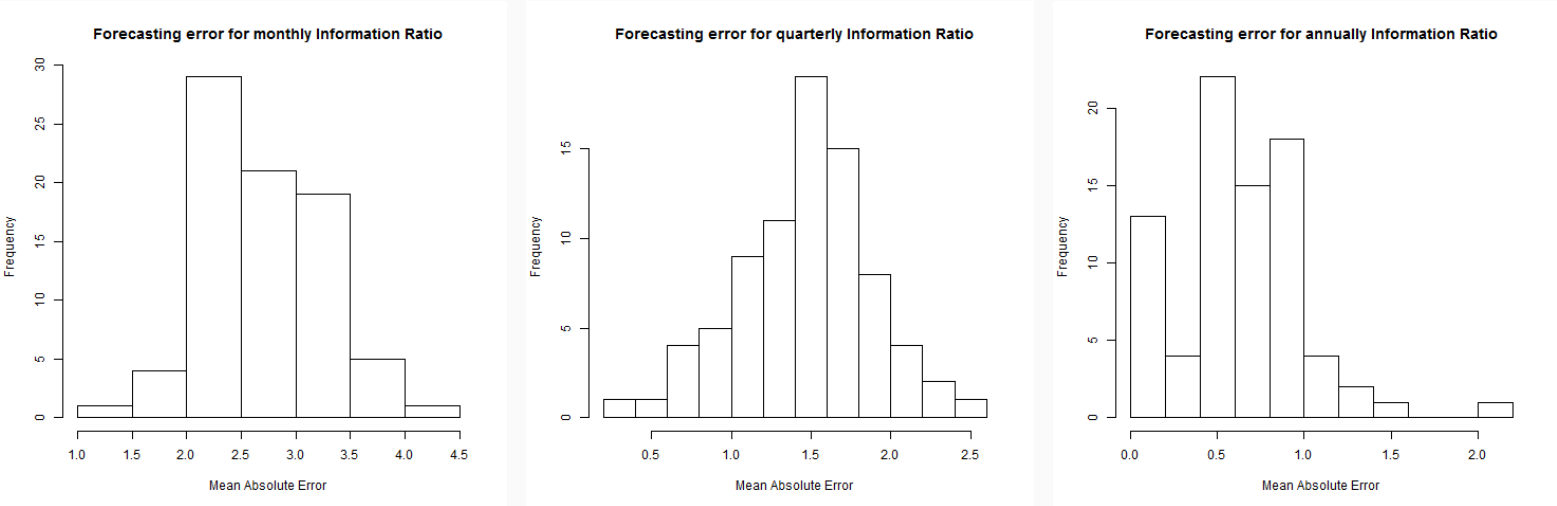

Fig 7: Histograms of forecasting errors separated by metric sampling frequency.

Information Ratio Testing Length

However, for finding an accurate information ratio, the lookback window had a definite impact. The best-performing forecasts, by far, were those trained on 5 years of data. As the training period becomes shorter, we see the error distributions settle around 2.0.

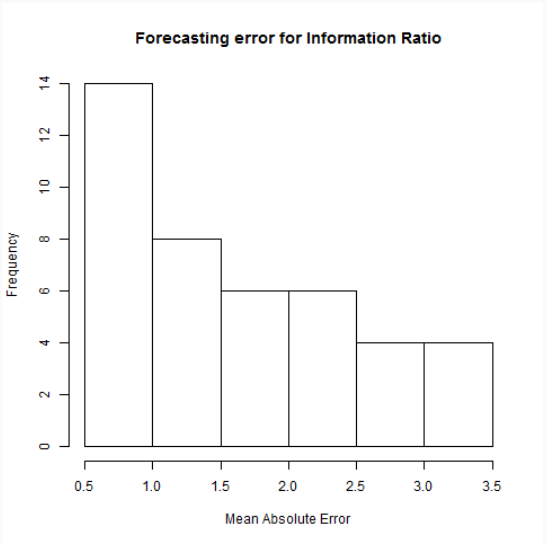

Fig 8: 5 years

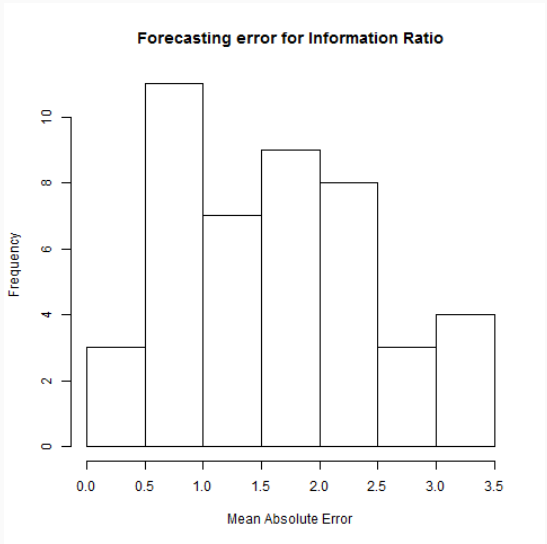

Fig 9: 4 years

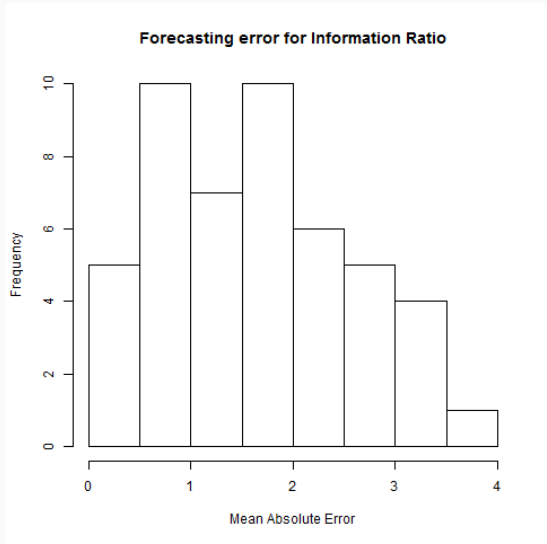

Fig 10: 3 years

Information Ratio Testing Length (cont.)

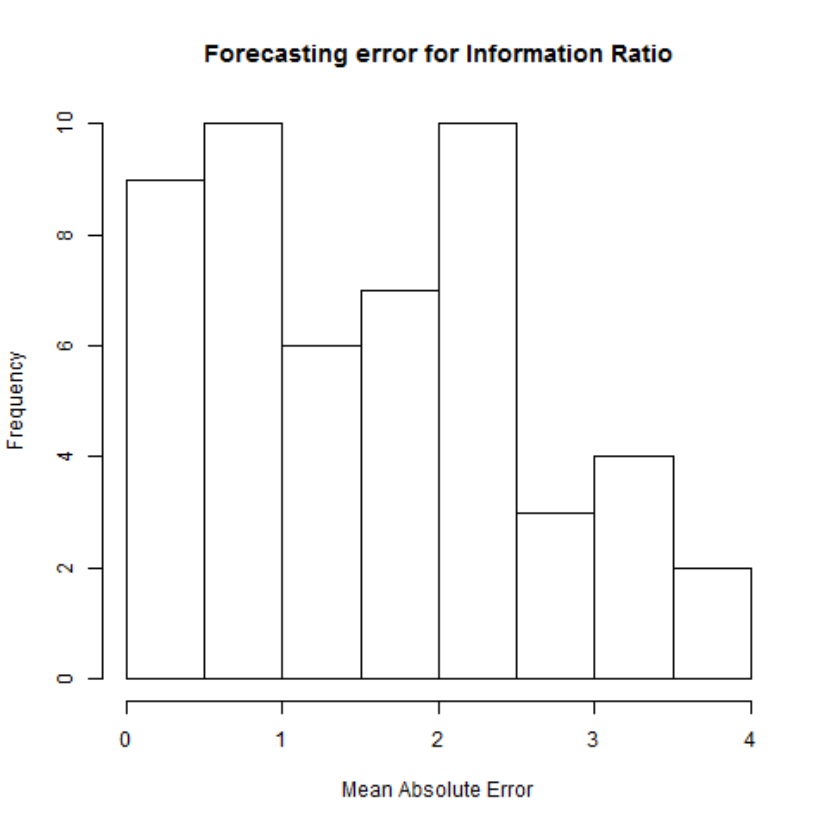

Fig 11: 2 years

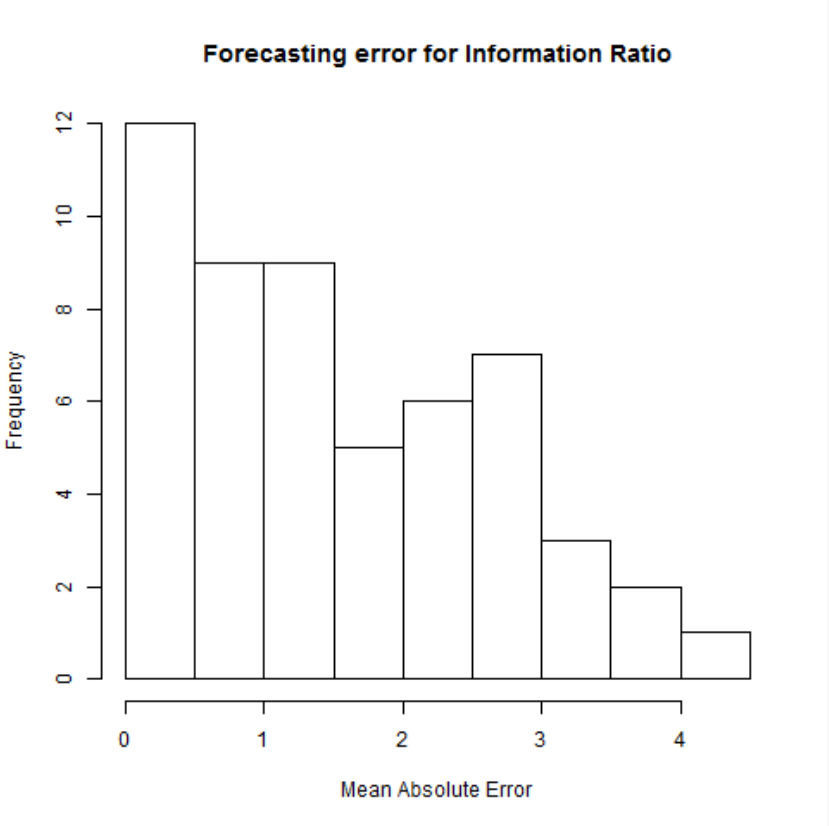

Fig 12: 1 year

Conclusions

Sampling frequency

- In both forecasting the strategy's information ratio as well as its maximum drawdown I found that the annual performance metric samples are much better than quarterly samples, which are in turn better than monthly samples.
- In the short run, the market is much more volatile than in the long run, making forecasting much harder to perform accurately.
- I would suggest sampling one's performance metrics annually at the very least to obtain an accurate assessment of one's strategy in real practice.

Lookback window

The most accurate forecasts of MDD occurred when the forecasting model was trained over only 1 year, while those trained for 3-5 years performed worse but about the same as each other.

- May come as a consequence of two periods of large downside: the end of the dot-com bubble and the 2009 recession.
- Any forecast trained over these time periods would have returned very large drawdowns.
- Longer lookback windows would intersect with these periods of large drawdown more times than shorter lookback windows.
- Running the strategy over bullish periods of time might these results have turned out differently.

Lookback window (cont.)

Forecasts of IR acted oppositely; the longer the training period, the better its forecasts became. In the long run, benchmarked strategy returns become decreasingly volatile, making our training values less dispersed.

Choice of performance metrics

I would not recommend the use of maximum drawdown as an appropriate risk measure for backtesting because it lacks a benchmark. For example, running this strategy over 2007-2010 would yield very different MDD values compared to running this strategy over 2014-2017 because of the state of the worldwide economy. Although a strategy may lose money, it is important to know whether it beat the benchmark or not; the inverse is true as well.

Choice of performance metrics (cont.)

On the other hand, a benchmark-adjusted risk measure like the Information Ratio gives its user a much better understanding of the viability of one's strategy. This is not to say, however, that upside risk measures are inherently better predictors of future performance than downside risk measures, but rather that one must consider the benchmark's performance in regards to one's own.

Future Work

Multivariate strategy

To expand the scope of my conclusions, back testing a multivariate trading strategy would prove useful. In practice, it is uncommon for investors to only trade a single asset; most investors prefer a more diversified portfolio of assets to minimize specific risk [**?**].

More sophisticated forecasting models

There exist many more sophisticated forecasting models, such as autoregressive integrated moving average models, but they also require more assumptions to be made. For example, ARIMA assumes that the residuals are homoscedastic. If the market data has only approximately strong stationarity, however, we cannot assume this. In the case that this assumption and others could be made, though, one can use a more powerful forecasting model.

Different market

In this study, we focused solely on the behavior and predictability of the stock market. Other avenues of interest would include testing our strategy on the bonds or foreign currency exchange (forex) markets, to name a few since the environmental factors and ecology of those markets are inevitably different. Recently the growing practice of High Frequency Trading (HFT) has completely changed the stock market and has begun to leak into forex markets. [**?**]

Challenges

Choice of Backtesting Software

Our initial choice of backtesting software was Zipline, an open-source platform written in Python by Quantopian. Due to its limited behavior and the fact that Zipline is still in developmental stages, I switched to Quantstrat, a powerful and flexible R package.

Figure 1: Cumulative returns of example trading strategy in Zipline

Data Extraction and Organization

Free data sources cannot provide the extent to which paid sources can, nor do they carry information on companies that have gone bankrupt. Instead, I gathered Bloomberg pricing data for every S&P 500 constituent from 1990 to 2017.

Data Extraction and Organization (cont.)

However, information such as every S&P 500 constituent for a period of time is not readily available, nor is downloading historical pricing data for a large amount of companies automated. This process required a great deal of time to manually obtain the data required for each of these steps and write scripts to organize it.

- Organize a list of every S&P 500 constituent (1000+ unique constituents).
- Download daily OHLC prices for each constituent between 1990 and 2017.
- Separate and format each constituent's prices into a single .csv file that Quantstrat can recognize and use.

Future Work

- Testing different strategies
- Using different financial instruments, e.g. bonds, futures
- Forecasting different performance metrics

References

Subha, M. V. *A Study on Stationarity of Global Stock Market Indices.* Journal of Contemporary Research in Management, vol. 5, issue 2.

Thank you!

Mentors

- Professor Ensor
- Michael Weylandt
- James Lenz

Project Sponsors:

- National Science Foundation Grant No. DMS-1547433
- Two Sigma
- Rice University Provost/VP Office