Thesis

John Gilheany

2017-10-07

Contents

Chapter 1

Opening Comments

I woud like to thank David Kane and Michael Parzen for their contribution and help on my thesis. This would not have been possible without their assistance.

Chapter 2

Introduction

I would like to start by introducing the main index this thesis will be focusing on, USMV. The MSCI Minimum Volatility Index (USMV) is intended to have a lower beta, lower volatility, lower cap bias, and contain more stocks with less risk than its parent index. It is rebalanced twice a year, on the last trading days of May and November. The index typically has around 180 constituents, with an average of 20 new additions and 14 deletions every 6 months when rebalancing occurs. Over the last five, years, the number of additions has ranged from 12 to 25, while the deletions have been between 10 and 19. Changes to the index are usually announced nine trading days before they are set to take place.

Using the Barra Open Optimizer, USMV creates a minimum variance portfolio of low risk stocks, as a subset from its parent index, EUSA. Using this estimated security covariance matrix, the MSCI Minimum Volatility Index is the product of the lowest absolute volatility, considering the constraints. Moreover, these additions are simply a relabeling of existing stocks in the parent index, and do not include new additions to the parent index. The low-risk stocks chosen to be in USMV are determined by a set of constraints, like maintaining a certain sector or country weight relative to the parent index.

There are many specific constraints to this index. The first is that an individual stock cannot exceed 1.5% or 20 times the weight of the stock in the parent index. The minimum weight of a security in the index is also capped at 0.05%. USMV also aims to keep the weight of specific countries within a 5% range of the weight in the parent index, or 3 times the weight of the country in the parent index. Sector weights of USMV also cannot deviate more than 5% from the sector weights in the parent index. One way turnover of the index is also maxed at 10%. Thus, taking into account these constraints, the Barra Open Optimizer creates the lowest absolute volatility portfolio possible.

Chapter 3

Literature Review

3.1 High Returns from Low Risk By Pim van Vliet and Jan De Koning

One of the most widely believed tenants of finance is the concept that with more risk comes more reward. However, looking at historical market returns, this does not appear to be the case. Over an 86-year period from 1929, low volatility stocks outperformed high volatility stocks by a factor of 18. If both portfolios started off with the same $100, the low volatility portfolio end value would be $395,000, while the high volatility portfolio would be worth just $21,000. Low risk stocks returned 10.2% annually whereas the high risk stocks returned just 6.4% annually. This difference of 3.8% is striking, and presents an anomaly in the field of finance.

This begs the question of how a portfolio of lower volatility stocks can outperform higher volatility stocks over a long period of time. The primary way this occurs is that the low volatility portfolio loses less during times of financial stress. For example, in 1932 following the Great Depression, it was observed that the high volatility portfolio shrunk from $100 in value to $5 in value, while the low volatility portfolio shrunk from $100 to $30. Since the low volatility portfolio is able to lose less money, it is able to grow capital more effectively than the high volatility portfolio. In this example, the annualized volatility of the low risk portfolio was 13%, and the annualized volatility of the high risk portfolio was around 2.5 times that, at 36%. In addition to being more risky, the high volatility portfolio was outperformed by 18 times.

Thus, it seems very counterintuitive that fund managers and investors would not only invest in low risk stocks. Part of understanding this comes from interpreting what risk is defined as in the financial community. Risk is not necessarily defined as losing money, as it may be for an individual, but instead underperforming a benchmark. Volatility is also an important concept to understand. Volatility is an important measure of financial risk, as it comes from the price fluctuations of a stock or investment. Volatility is also one of the best indicators of bankruptcy. Taking some risk does pay off, as the relationship between risk and return starts off slightly positive before leveling off and becoming negative. Many researchers focus on short-term periods when analyzing stock returns instead of longer term for a couple of reasons. The first reason many focus on “single period returns”, which in most academic studies is just a one month period, is because this takes away the significance of compounding. The longer the investment period, the more risk one takes in hurting long term returns through compounding. By not fully including the magic “return upon return” effect of compounding, a high-risk portfolio in this book performs more than 6% better per year. For example, if in month a portfolio worth $100 drops 50% to $50, then the next month increases 50% to $75, the investment return is dependent on how one divides the time period. Looking at it on a monthly basis, even though the portfolio lost $25, the net return would be -50% +50%, or 0%. Looking at it on a long term basis, the net return was -25%.

David Blitz, the head of quantitative equity research at Robeco, discusses this different perspective as

somewhat due to the need to benchmark the performance of an investment manager. This is a concept known as "relative" risk. In the examples above, everything has been in respect to absolute risk - that is how much money is being gained or lost due to overall stock movements, with regard to the starting amount of money invested. Volatility, in itself, captures these changes in the price of a stock, and is an absolute risk measurement. Many institutional investors do not look at risk on an absolute level, as a retiree or mom and pop investor may, but instead look at the risk of a portfolio with respect to market or some other widely accepted benchmark. For these investors, the risk is not as much about losing money, rather is more about lagging the market or their peers. Investing is very much a relative game. If a portfolio drops 20% while the market drops 40%, this is seen as a much better outcome than if a portfolio goes up 20% while the market goes up 40%. Thus, a portfolio that moves closely with the market has a very low relative risk. This risk can be calculated as volatility by looking at the relative price movements, instead of the absolute price movements.

Investment professionals focus on relative risk for a number of reasons, one of which is the fact that they are not managing their own money. They need to prove to their bosses and clients that they are above average in their job. If a particular benchmark cannot be beaten by these investors, clients may ask why pay for them to manage their money when they could put it in a low-fee or no-fee mutual fund. This is one of the reasons that institutional investors need to compare their performance to some benchmark. Thus, the focus for investors is return and relative risk. Adding low risk stocks to the portfolio causes relative risk to increase a lot, making it an unappealing investment because low absolute risk inherently causes high relative risk. A low risk portfolio only makes sense if absolute risk is what one cares about. Thus, for those investors who don't care about relative risk and just absolute risk, low risk stocks are a great investing opportunity.

In addition to the reasons mentioned, there are several additional reasons why some investors are not attracted to low-risk stocks. Eric Falkenstein, a renowned author in the low volatility investing realm, wrote that "envy is at the root of the investment paradox." Some investors don't recognize the significance of compounding returns. Others do, but are unable to utilize the paradox due to relative risk and career pressures. Analysts who choose big winners are more likely to get recognized than those who pick safer stocks with lower upside potential, and funds that pick the right high risk stocks also see more reward in an increase in AUM. Moreover, some people do not invest in low risk stocks because they have less appeal of high risk stocks, where they think they can make money easily and quickly. These high risk stocks are more "sexy" and have a "lottery ticket" element that attracts investors with the appeal of a big payday.

3.2 Betting Against Correlation: Testing Theories of the Low-Risk Effect

A recurring phenomena in finance, is the observation of the "low-risk effect." This is the idea that lower risk or lower volatility stocks, tend to have higher alpha than higher risk or higher volatility stocks. In trying to understand the reason this anomaly occurs, are two possible explanations. The first looks at at whether this is caused by leverage constraints, meaning measurement using systematic risk. The second focuses on the behavioral effects, or idiosyncratic risks. One of the main issues with prior research, is that a lot of the low-risk factors are correlated and interrelated, making it hard to isolate certain factors or effects. In this paper, the global data was used, with a couple new factors meant to control for existing factors.

Previous studies, like one by Adrian, Etula, and Muir in 2014, showed a link between return to the BAB (Betting Against Beta) factor and financial intermediary leverage. Many of these factors, though, including BAB, generally exhibit the "low-risk effect" and are thus very hard to differentiate between. Thus, this paper decided to do just that, by breaking down BAB into two other factors: betting against correlation (BAC) and betting against volatility (BAV). BAC is accomplished through longing stocks with low correlation to the market, and shorting those with high correlation to the market, while trying to match the volatilities of both the long and short portfolios. BAV is achieved in a similar manner, except instead of longing and shorting correlation, volatility is used, and correlation is kept constant.

To address the behavioral explanation, the paper looks at some prior factors from studies done by Ang,

Hodrick, Xing, and Zhang in 2006 and 2009. The first study found stocks with low idiosyncratic volatility (IVOL) have a greater risk-adjusted return, while the second found that a low maximum return (LMAX), a measure of idiosyncratic skewness, is associated with greater risk-adjusted returns. This paper kept the focus on LMAX and IVOL, but added another factor, scaled MAX (SMAX), which longs stocks with a low MAX return divided by ex ante volatility, and then shorts stocks with a high MAX return divided by ex ante volatility. This focuses on the lottery demand, holding volatility relatively constant and only focusing on the distribution of the returns. Margin debt held by investors, and investor sentiment were also noted.

In the paper, 58,415 stocks from the MSCI World Index, from 24 countries between January 1926 and December 2015 were covered. BAB and BAC ended up being very successful in controlling for the other factors that could influence the "low-risk effect." For all stocks, the BAC factor produced a significant six-factor alpha that was nearly independent of the other low-risk factors studied. In terms of explaining the behavioral side with factors, SMAX was the only truly great, resilient measure used. The rest generally had higher turnover, and were consequently very susceptible to microstructure noise. SMAX attained positive risk-adjusted returns in the U.S. but negative risk-adjusted returns globally, which was seen with some other idiosyncratic risk factors. The paper showed that systematic low-risk factor generally tended to outperform behavioral risk factors, especially when considering turnover and time period length. All in all, the low-risk effect was believed to be driven by multiple factor effects, meaning both leverage constraints and the demand for lottery could play a role in effecting this. However, leverage constraint effects were a bit stronger, especially internationally.

3.3 Price Response to Factor Index Additions and Deletions

Some of the driving fundamental assumptions of finance is the flat demand curve for stocks, where risk is the main driver and each stock has a perfect substitute. However, this concept has been questioned for the past few years, with literature picking up on stocks with show supply shocks and checking how this affects their price. The literature has shown several instances where large block sales of stock has negatively affected its price. This was often due to information contamination, which is new, significant information about the company in the market. This information often reflects fundamental changes in the company, and if is is negative, will understandably trigger block sales. Thus, the price change is less due to the supply shock, and moreso due to the fundamental change in the company's value (like a scandal or earnings report).

However, interesting patterns that have not yet been fully explained have been observed regarding S&P 500 company addition and deletions. When companies are added or removed from the index, it is purely mechanical, and usually not due to some drastic fundamental change in the company. Assuming the market is efficient, the demand for stocks should not change due to being added or removed from an index, but several studies have shown that it does. Harris and Gurel (1986), Shleifer (1986), Beneish and Whaley (1996), Chen, Noronha, and Singal (2004) all show how new additions to the S&P come with higher than normal returns for that company. Though they agree on the price movement, the studies tend to have a hard time agreeing on the reason for this price movement. Some possible explanations include compensation for providing liquidity, better monitoring for investors when a company is added to a reputable and large index, and higher analyst coverage leading to more information and analysis available on the company. One primary concern is whether or not index reshuffling is an information-free event - that is, whether a company being added or removed adds information to the market about the company.

In this paper, the authors look at factor index rebalancing for an information-free event. Factor indices are part of a parent index of many other stocks, and are constructed in a mechanical way that is publically available and usually based on ranking stocks off a particular ratio of characteristic. Looking at the MSCI Minimum Volatility index, stocks returns were recorded for the stocks that had been added/dropped. It was found that the cumulative return from announcement to the effective day was 1.07% for stocks added with a significant t-statistic of 7.16, with 62% of the stocks exhibiting a positive cumulative abnormal return. Of the 1.07% increase, 0.63% of it was gained the following day, indicating that a large part of the increase is from an increase in demand from index funds. 0.31% of the return is lost five days after the rebalancing, but generally the price tends to stabilize afterwards after ten days. Thus, 68% of the price increase is permanent,

while the other 32% is temporary and lost after a few days. This can be due to a number of reasons including a liquidity premium charged by the stock's owner or arbitrage activity. Average trading volume was also significantly more for stocks that were recently added to the index. Between the announcement and actual day of adding the stock, the average trading volume was 30% higher than normal, with a significant t-statistic of 3.81. Moreover, there is a 74% increase in volume for the day prior to the actual adding of the security. A very similar phenomena occurs with stocks set to be dropped from the MSCI Minimum Volatility Index. From the announcement of a stock being dropped to the day before it is actually deleted from the index, the total cumulative abnormal return is -0.91%, and -0.57% of this comes the day right before. After the stocks are deleted, 64% of them have a negative return the following day, and only 0.49% of the -0.91% is regained after three weeks. Trading volume also spikes 46% on the day prior to removal from the index. After three weeks, it returns back to within 1% of the normal trading volume.

These findings imply that once a security is added to a factor index, the demand curve shifts to the right, moving the equilibrium. The trading volume change is likely due to index funds buying or selling massive amounts of the stocks that will be added or removed. Moreover, it was found that the amount of the return is also directly related to the weighting of the volume of stocks entering or leaving the factor index. All in all, these findings suggest an index arbitrage opportunity if the index additions or deletions can be predicted.

3.4 Benchmarks as Limits to Arbitrage: Understanding the Low-Volatility Anomaly

Low-beta and low-volatility stocks have outperformed high-beta, high-volatility stocks from 1968-2008, combining great returns with low downturns. This can be explained by behavioral models of security prices. First is the idea that investors have a preference for "lotteries" and a bias of overconfidence creates a higher demand for higher-volatility securities. Second is the idea that this arbitrage opportunity is very limited, as the need to benchmark creates a greater demand for higher risk stocks, while discouraging investments in low-beta, low-volatility stocks. Several other papers have tried to explain this seemingly misunderstood phenomena, including Karceski in 2002, where is was noted that an extrapolation bias could cause mutual fund managers to care more about overperforming in a bull market, than underperforming in a bear market. This understandably, increases the market's appetite for risky stocks with high reward potential. This paper focuses on the distortions caused specifically by benchmarking.

In the paper, the authors used 41 years of CRSP data, ranging from January 1968 – December 2008. Using the top 1,000 stocks by market cap, the five year trailing volatilities were calculated and returns were tracked. A dollar investment from 1968 made its way to $59.55, or $10.12 in real terms when accounting for inflation. On the other hand, the highest volatility portfolio went from a dollar in value to 58 cents during the period, with a real value of just around 10 cents when considering inflation. When using beta as a measure for risk, the finding was very similar. In the lowest-beta portfolio, a dollar grew to $60.46 in nominal value, or $10.28 in real value after inflation. The highest-beta portfolio grew from a dollar to $3.77 in nominal terms, or $0.64 in real terms after inflation. This held true for large cap companies, but the discrepancy was even higher for smaller cap companies. Generally, the low-risk portfolios also grew much more steadily and constant, without many drawdowns. To add on to this as well, the portfolio values did not include transaction costs. The high-risk portfolios cost more to rebalance on a monthly level, as was done in the paper, than the low-risk portfolios, indicating this anomaly is more pronounced that initially reported. These findings are not novel, but this paper attempts to explain them in a new way. Many theories add evidence for disproving the Capital Asset Pricing Model (CAPM), and even suggest that beta may not be the correct measure of risk. Other models relating risk and return, however, have had difficulty gaining acceptance and widespread usage in the finance industry.

One theory explored in detail is an investor's irrational preference for high-volatility stocks. First, many investors have a natural preference for lotteries, even though there is a general aversion towards loss. If a stock has a positive skew, that is a larger probability of a large positive payoff than probability of a small payoff, investors typically are very interested. Though skew is not the same as volatility, in their paper in

2010, Boyer, Mitton, and Vorkink make a strong case for how expected skewness is a proxy for volatility. Another idea is representativeness, or that Bayes' rule and probability theory are often not natural to people. One example of this is, selectively looking at a few speculative investments that have turned out to be massive successes, without considering the numerous failures. Overconfidence has also been tied to a preference for volatile stocks; optimists are generally more aggressive than pessimists.

The need for benchmarking, especially among institutional investors, is also believed to heavily play into the anomaly. In fact, from the period when institutional investors managed 30% of all money to 60%, the anomaly intensified. This still begs the question as to why institutional investors do not buy more low-volatility stocks, and the answer has to do with benchmarking. The typical fund manager is judged for his/her "information ratio" (IR) which is the expected return difference between the manager and the expected return of the S&P 500, divided by the volatility of this return difference (tracking error). The goal of the investment manager is to maximize this information ratio, best as possible, through picking stocks. In 2009, Sensoy showed that over 61% of U.S. mutual fund managers are benchmarked against the S&P 500, while over 94% are benchmarked to some U.S. index benchmark. Moreover, SEC rules require mutual funds to compare their performance to some benchmark. This intuitively makes sense, as it allows investors to assess the skill and ability of managers in an unbiased way, and also allows fund managers a chance to differentiate themselves.

Investment managers without leverage will try to find mispriced stocks with a beta very close to market risk (beta of 1), overweighting positive-alpha stocks while underweighting negative-alpha stocks. When comparing the Sharpe ratio of large cap stocks for a low-volatility portfolio, it was quite high at 0.38. However, IR was a very low 0.08, showing this would be very tough for a fund manager to invest in. While Beta and volatility are undoubtedly very correlated, the study showed that beta is more related to the anomaly than volatility, especially with large cap stocks, which is what most fund managers disproportionately focus their invests in. Both volatility and beta play a significant role in this anomaly in smaller cap stocks. Looking at the period from 1968-2008, the top value strategy portfolios had an IR of 0.51, and top momentum strategy portfolios had an IR of 0.64. This is extremely high compared to the IR of low-volatility stocks in this period, which ranged from 0.08 to 0.17.

Overall, irrational investor preference for lotteries and high volatility stocks, as well as investment managers' focus on benchmarks and IR flatten or even invert the relationship between risk and return as we know it. This has been shown by the study, and prior observations that the anomaly intensified with the increase in AUM of fund managers in the U.S. These reasons appear perennial, so the anomaly will likely not be going away anytime soon. One other factor to this is compounding of the low-volatility portfolio. Since it suffers fewer drawdowns, this means the portfolio is able to experience more stable and upward growth.

3.5 Betting Against Beta

In this paper, a real-world resembling model is created with leverage and margin constraints in 55,600 stocks from 20 global stock, bond, credit, and futures markets. Some agents in this model cannot use any leverage, and some have limited margin constraints, much like many investors and fund managers.

Many mutual funds, pension funds, and individual investors are constrained by the amount of leverage they can take on, such that instead of investing in a portfolio yielding the highest Sharpe Ratio, they are forced to overweight portfolios with higher risk stocks. This suggests fund managers hold high-beta stocks to a lower risk adjusted return standard than low-beta stocks, which would require leverage. Thus, if one cannot leverage or has significant leverage constraints, then this agent will overweight riskier securities. The model was able to empirically show this in the equities, bonds, and futures markets. This was done by sorting portfolios by betas, and realizing alphas and sharpe ratios declining with increases in portfolio beta.

Moreover, if one can leverage without constraint, then they would underweight high-beta assets and overweight low-beta assets. Betting against beta (BAB) factors help explain this better. A BAB factor is a portfolio longing low-beta securities (leveraged to a beta of 1), shorting high-beta assets (deleveraged to a beta of 1). Hence, a BAB factor is market neutral. The model in the paper predicts that this portfolio will have a positive return, that increases with the spread in the betas and tightness of leverage constraints. Thus,

longing low-beta and shorting high-beta yields significant, and positive risk-adjusted returns. This was observed in the model by looking at U.S., developing, and international equity markets and observing that the BAB factor yielded a Sharpe ratio that was double its value effect, and 40% greater than momentum. The BAB factor had very high risk adjusted returns, and during four twenty-year periods between 1926 and 2012, produced significant positive returns. This generally held across other asset classes, including credit and treasury bond markets.

When a leverage constraint is met or surpassed, and the agent needs to deleverage, the BAB factor portfolio experiences negative returns, but its expected future returns increase. This was once again shown with a time series with spreads of various funding constraints.

Another central idea of the model was that increased funding liquidity risk compresses betas toward one. This was proven by looking at the volatility of funding constraints as funding liquidity risk, and the end result was that the dispersion of betas when funding liquidity risk is high, was much lower than when funding liquidity risk is low.

Finally, the model showed that investors that are more constrained are forced to overweight riskier securities, while investors without such constraints can overweight lower-risk securities. Studying a number of stock portfolios from constrained investors, most fund managers and individual investors' portfolios have a beta greater than one. On the flip side, many PE firms that perform an LBO get firms with a beta below one, and apply leverage. Great investor Warren Buffett even bets against beta, as many of his investments are leveraged, low-beta stocks.

3.6 The Low-Risk Anomaly: A Decomposition into Micro and Macro Effects

The low beta anomaly can be broken up into micro and macro effects. The micro effects include picking low beta stocks, while the macro effects are picking low beta countries or industries. In this paper, the micro effects were recorded by creating long-short portfolios of stocks, holding constant country and industry risk. The macro effects were observed through long-short portfolios of various countries and industries, holding stock level risk constant. Studying a number of stocks within 29 industries and 31 different developed countries, the macro and micro effects were observed, and both together were shown to play an important role in the low risk anomaly.

Micro selection of stocks, that is constructing a low risk portfolio of stocks, holding country and industry constant, showed that risk could be significantly decreased without a significant decrease in return. Macro selection, especially with regards to the country chosen, causes greater returns with small differences in risk. These findings have pretty significant implications, as this indicates that people seeking arbitrage opportunities through mispricing of industry or sector ETFs may not be as feasible or profitable as one may think. There is more of an opportunity exploiting the micro effects of individual stock selection.

It was found that using industry beta to predict future stock betas was possible, but not as effective as just using historical stock betas. However, industry beta information without stock information does improve risk-adjusted returns, just not as to the same level as with stock information. The paper also goes into detail trying to isolate pure industry effects and pure stock effects. Pure industry effects are the average differences between high and low beta industries, while holding constant stock risk. Pure stock risk is the opposite of that, calculating the average difference between high bet and low beta stocks, keeping industry risk constant. In the end, finding low-risk portfolios using selection of low risk stocks keeping industry constant was around four times more effective than using industries and keeping stock risk constant. Using the historical betas of both together, however, has more predictive power than either one alone.

Next, looking at 31 developed countries including Canada, France, Germany, Japan, and Singapore, the paper worked to decompose the low risk anomaly into country and stock specific effects. Similar to the industry findings, country beta was able to predict stock betas to a certain extent, but not as well as historical stock betas were. Looking only at country betas yielded around half the risk reduction and two-thirds the

risk adjusted return improvement, as compared to stock betas. This study implies that predicting risk of individual stocks is in itself very hard when only given data on country or industry risk, but when given all the data can have much more predictive power.

Chapter 4

Data Collection and Summary Statistics

4.1 EUSA and USMV Data Compilation

Data was downloaded from www.ishares.com for EUSA (iShares MSCI USA Equal Weighted ETF) and USMV (iShares Edge MSCI Min Vol USA ETF), from Oct 31, 2011 to December 31, 2016. iShares are a type of ETF managed by BlackRock, and www.ishares.com contains the month end data for the two ETFs of interest in this dissertation. The data sets included information for the constituents of each ETF for a period in time, and some other characteristics of them, including: ticker, company name, asset class, weight of the stock relative to the entire index, price per share, number of shares, market value of the position, notional value of the position, sector, sedol number, isin number, exchange that the stock is listed on, and the month end date for the data. On the website, iShares had data for the positions and constituents of each ETF, for the last trading day of every month. Thus, an R function was created, one for each ETF, that would combine each month end data set into one aggregated one. Thus, each month end data set was individually downloaded, then aggregated to create the data sets "usa" and "minvol". These were stored in the data-raw folder, for safekeeping.

4.2 EUSA and USMV Data Cleaning

I began the data cleaning process by removing cash and cash related assets, since this is not important for our purposes. After having a quick overview of the data, there were many issues with each respective data set that needed to be fixed before the analysis could begin. As USMV is a subset of EUSA, the issues were very similar, and those that existed in USMV, generally existed in USMV as well. The issues could be broke down into 3 main types.

4.2.1 Non-US Exchanges

First, looking at unique exchanges of the data on R, it was seen that there were many foreign exchanges like the Swiss Exchange and the Mexican Exchange, which did not make sense, given the ETF constituents are supposed to be listed on US-based exchanges. These could be broke up into two more groups: companies that were incorretly listed overseas and are actually listed on US exchanges, and companies that also are actually listed on US exchanges but instead had their overseas exchange tickers listed.

The first type of error was from companies that were listed on either the NYSE and NASDAQ, but were curiously listed on an foreign exchange instead, but had their US ticker used. One example was BAC, Bank of America, which is listed on the NYSE, but was listed on the Swiss Stock Exchange in the data set. The price for BAC in the data set corresponded to the price of BAC in the NYSE, although it was listed on the Swiss Exchange. Moreover, I checked to see if BAC corresponded to Bank of America on the Swiss Exchange, and it did not. Thus, after several checks, I was able to conclude that BAC in the dtaa set was incorrectly listed on the Swiss Exchange, and should have been listed on the NYSE instead. Since the ticker would still be able to be read into WRDS, these cases were left as is.

The next type of error was from companies listed on foreign exchanges that are listed on a US exchange as well, but their non-US ticker used. One example of this was Aflac, Inc. which was listed by its ticker "8686" on the Tokyo stock exchange. This was immediately a red flag due to the numbers in the ticker. This numeric ticker corresponded to Aflac, Inc. on the Tokyo exchange, but when checking the recorded price of the stock for corresponding dates, it matched up with the Aflac, Inc. stock on the NYSE, with ticker "AFL". Thus, when this happened, each company was treated on a case by case basis. In this case, since the stock price corresponded to AFL, the ticker name was changed from "8686" to "AFL". This would ensure the data could be properly read in from WRDS.

Overall, even with these numerous errors, it was a good sign because it implied that the data was generally correct (no internationally listed companies), but just recorded incorrectly. Thus, after making these changes, it was safe to assume the data was for the most part accurate.

4.2.2 Unrecognized Tickers

Another general type of error was when the ticker was not read into WRDS, causing all the prices for that ticker and company to be NA. This was evaluated, once again, on a case by case basis, by observing which tickers WRDS did not recognize, and looking at the company name to understand why. Sometimes, the issue was very obvious. One example of a clear discrepancy was when the ticker had an asterisk at the end of it. After careful digging, the asterisk did not seem to mean anything, and it is unclear why some tickers contained it. One example was "AAPL*". This caused issues for reading the data in from WRDS, because that ticker was not read in as "AAPL" due to the asterisk.

Another example of the ticker not being read in properly was when it contained numbers. Alflac was an example that was mentioned previously, but another one that applied here was "AG4" which was the ticker for Allergan Since NYSE and NASDAQ tickers do not contain numbers, this was a clear red flag. After some research, it appeared AG4 is the ticker for Allergan on the Deutsche Boerse AG Stock Exchange. However, the prices corresponded to Allergan's on the NYSE. Thus, this change in ticker was made. Overall, though each category is unique, there has been a lot of overlap, and often times correcting one type of error would fix other errors too. For example here, many tickers that include numbers will not be read in, and this is usually because the ticker corresponds with the same company but on a foreign exchange.

4.2.3 Price Discrepancies

The general methodology to make sure a change in ticker was appropriate was to check the price of the stock at a specific date, in the usa data set, and then comparing it to the new ticker I was going to assign it. If the price matched, the change was made. If the price did not match up and was very different, then I looked to see if a stock-split might be the cause of this. If there was no evidence of a stock-split, then the stock further analyzed to see what the issue was. In addition to looking and when prices did not match up with tickers and companies for certain dates, monthly returns were calculated for each stock during the times they were in the index, and any abnormal returns (magnitude greater than 30%) were look at manually. One example of this was Netflix's stock 7:1 stock split in 2015. The monthly data showed a price of 656.94 on 2015-05-29 to a price of 114.31 on 2015-07-31, just one month later. This amounts to recorded loss of 82.5%. Since this surpassed the threshold set, it was look at in more detail. After some research, it was shown there was in

fact a 7:1 stock split, so the price of the stock on 2015-07-31 was adjusted to 800.17, and the appropriate calculations were done. Thus in this case, the ticker was left alone, but just the price was adjusted.

Tickers that could not be determined were removed. In the end, the ticker named "1015736" and Orchard Supply Hardware Stores were removed from the data set. These together accounted for less than 0.2% of the data from one month-end date.

4.3 EUSA and USMV Data Overview

To get a sense of the EUSA data, summary statistics are shown below:

```
##      ticker                            name             asset.class
##  CB     :  101   3M CO                  :   63   Cash        :    0
##  AGN    :   67   ABBOTT LABORATORIES    :   63   Equity      :38598
##  NLSN   :   64   ACCENTURE PLC          :   63   Money Market:    0
##  A      :   63   ACTIVISION BLIZZARD INC:   63
##  AAP    :   63   ADOBE SYSTEM INC       :   63
##  AAPL   :   63   ADVANCE AUTO PARTS INC :   63
##  (Other):38177   (Other)                :38220
##      weight           price            shares        market.value
##  Min.   :0.0000   Min.   :  0.56   Min.   :     0   Min.   :2.000e+03
##  1st Qu.:0.0536   1st Qu.: 35.07   1st Qu.:   798   1st Qu.:5.289e+07
##  Median :0.1208   Median : 55.09   Median :  1546   Median :8.282e+07
##  Mean   :0.1629   Mean   : 71.91   Mean   :  3365   Mean   :1.610e+08
##  3rd Qu.:0.1636   3rd Qu.: 83.52   3rd Qu.:  3181   3rd Qu.:1.410e+08
##  Max.   :4.6773   Max.   :953.00   Max.   :133289   Max.   :6.804e+09
##                   NA's   :46
##  notional.value                          sector         sedol
##  Min.   :     40.5   Financials            :6825   2000019:   63
##  1st Qu.:  54595.6   Consumer Discretionary:6595   2002305:   63
##  Median :  74215.6   Information Technology:5487   2005973:   63
##  Mean   :  89651.9   Industrials           :4687   2008154:   63
##  3rd Qu.: 116297.5   Health Care           :4067   2011602:   63
##  Max.   :2106050.4   Energy                :3208   2018175:   63
##  NA's   :22878       (Other)               :7729   (Other):38220
##            isin
##  AN8068571086:   63
##  BMG0450A1053:   63
##  BMG0692U1099:   63
##  BMG169621056:   63
##  BMG3223R1088:   63
##  BMG491BT1088:   63
##  (Other)     :38220
##                                                 exchange
##  New York Stock Exchange Inc.                      :27499
##  NASDAQ                                            : 9238
##  Boerse Berlin                                     :  427
##  Deutsche Boerse Ag                                :  394
##  Bolsa Mexicana De Valores (Mexican Stock Exchange):  266
##  (Other)                                           :  658
##  NA's                                              :  116
##       date
##  Min.   :2011-10-31
```

```
##  1st Qu.:2013-02-28
##  Median :2014-06-30
##  Mean   :2014-06-12
##  3rd Qu.:2015-09-30
##  Max.   :2016-12-30
##
```

To get a sense of the USMV data, summary statistics are shown below:

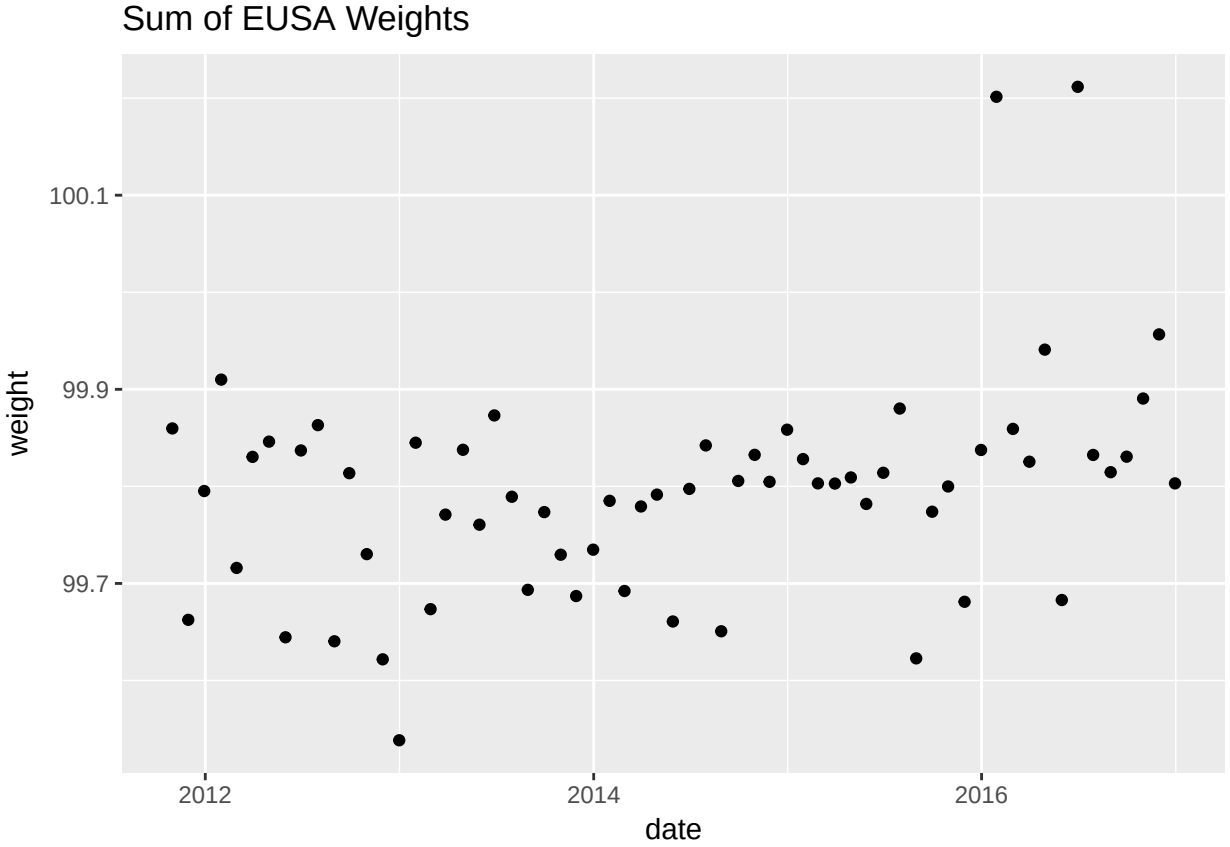
```
##      ticker                                name            asset.class
##  CB     :  81   ABBOTT LABORATORIES          :  63   Cash        :   0
##  ABT    :  63   ALTRIA GROUP INC             :  63   Equity      :9229
##  ACGL   :  63   ARCH CAPITAL GROUP LTD       :  63   Money Market:   0
##  ADP    :  63   AT&T INC                     :  63
##  AMT    :  63   AUTOMATIC DATA PROCESSING INC:  63
##  AZO    :  63   AUTOZONE INC                 :  63
##  (Other):8833   (Other)                      :8851
##      weight           price            shares          market.value
##  Min.   :0.0002   Min.   :  0.56   Min.   :       6   Min.   :     1138
##  1st Qu.:0.2790   1st Qu.: 48.52   1st Qu.:   60888   1st Qu.:  5109636
##  Median :0.6016   Median : 71.52   Median :  233116   Median : 18493183
##  Mean   :0.6810   Mean   : 93.10   Mean   :  462436   Mean   : 30802130
##  3rd Qu.:1.0191   3rd Qu.:100.79   3rd Qu.:  567713   3rd Qu.: 39743039
##  Max.   :2.8287   Max.   :953.00   Max.   :15574666   Max.   :240885300
##                   NA's   :6
##  notional.value                          sector         sedol
##  Min.   :    22212   Health Care           :1631   2002305:  63
##  1st Qu.: 19047994   Financials            :1548   2005973:  63
##  Median : 39501565   Information Technology:1482   2065308:  63
##  Mean   : 52545119   Consumer Staples      :1222   2065955:  63
##  3rd Qu.: 70790821   Consumer Discretionary:1053   2073390:  63
##  Max.   :240885300   Utilities             : 616   2077905:  63
##  NA's   :5026        (Other)               :1677   (Other):8851
##             isin                                exchange
##  BMG0450A1053:  63   New York Stock Exchange Inc.:7010
##  BMG3223R1088:  63   NASDAQ                      :1890
##  BMG7496G1033:  63   Deutsche Boerse Ag          :  58
##  US00206R1023:  63   Spot Regulated Market - Bvb :  58
##  US0028241000:  63   Boerse Berlin               :  32
##  US02209S1033:  63   (Other)                     : 164
##  (Other)     :8851   NA's                        :  17
##       date
##  Min.   :2011-10-31
##  1st Qu.:2013-04-30
##  Median :2014-09-30
##  Mean   :2014-08-13
##  3rd Qu.:2015-11-30
##  Max.   :2016-12-30
##
```

4.4 EUSA and USMV Data Check

4.4.1 Weights

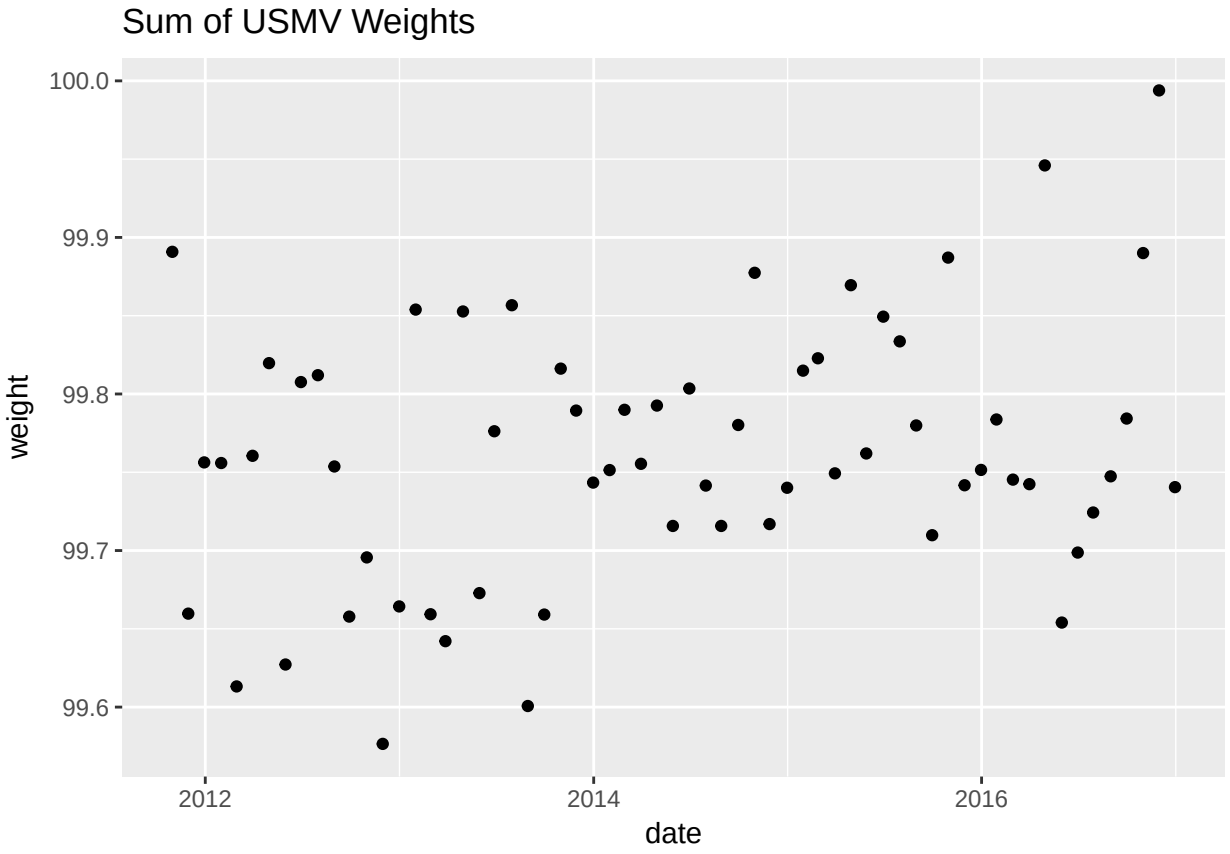
Thus, after cleaning all the data, I wanted to check how accurate the data set actually was. First, the total monthly weights for EUSA and USMV were plotted over time. Since cash and a few tickers were removed, it was not expected for the ticker weights to add up to 1 each month, but something very close to 1 was expected. The monthly change in weights for EUSA is shown below.

```
##       date                weight
##  Min.   :2011-10-31   Min.   : 99.54
##  1st Qu.:2013-02-14   1st Qu.: 99.73
##  Median :2014-05-30   Median : 99.80
##  Mean   :2014-05-30   Mean   : 99.79
##  3rd Qu.:2015-09-15   3rd Qu.: 99.84
##  Max.   :2016-12-30   Max.   :100.21
```



As we can see in the scatterplor above for EUSA, the weights are very close to 100%, generally within 0.2%. The minimum weight is 99.54%, while the largest weight is 100.21%. The mean weight is 99.79%. The monthly change in weights for USMV is shown below.

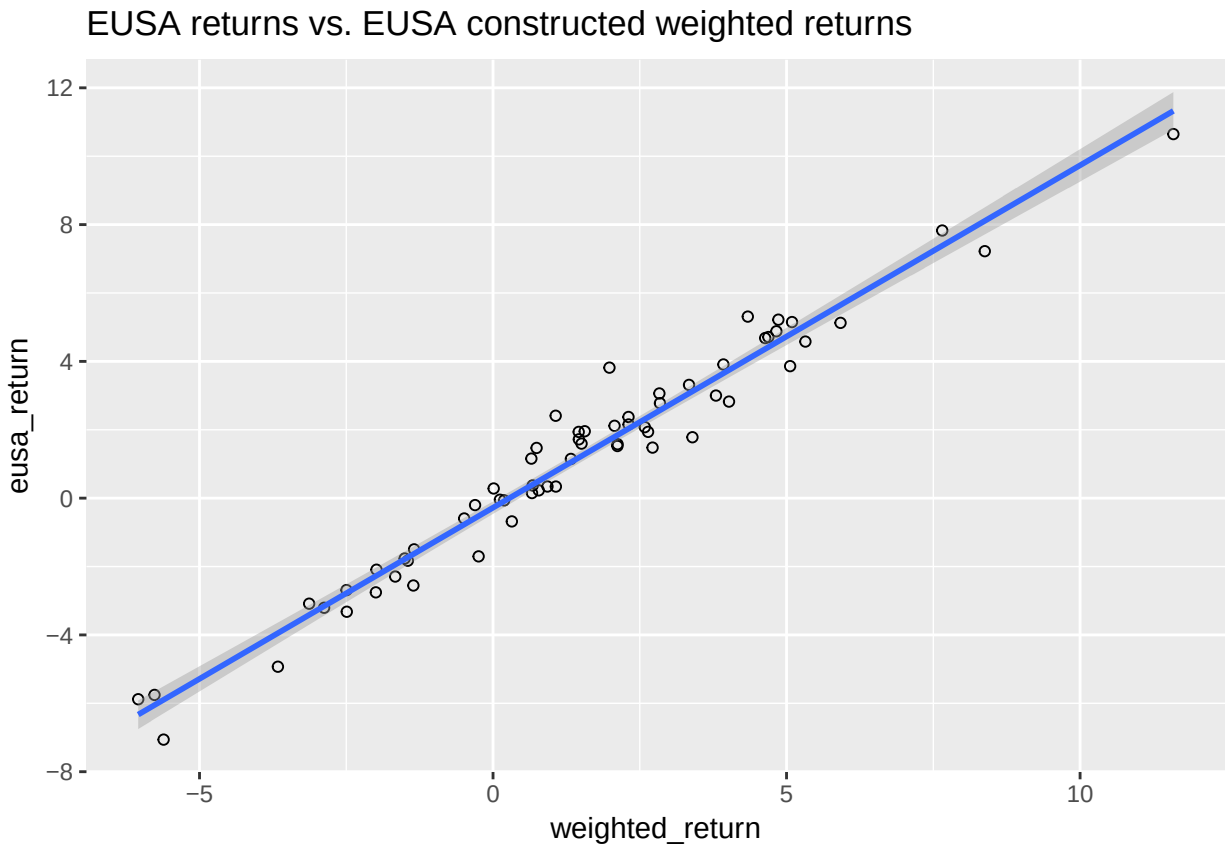
```
##       date                weight
##  Min.   :2011-10-31   Min.   :99.58
##  1st Qu.:2013-02-14   1st Qu.:99.72
##  Median :2014-05-30   Median :99.76
##  Mean   :2014-05-30   Mean   :99.76
##  3rd Qu.:2015-09-15   3rd Qu.:99.81
##  Max.   :2016-12-30   Max.   :99.99
```



As we can see in the scatterplor above, the weights for USMV are very close to 100%, and no value exceeds 100%. The minimum weight is 99.58%, while the largest weight is 99.99%. The mean weight is 99.76%. Overall, these look pretty solid and imply the data is trustable.

4.4.2 Comparing actual ETF returns to constructed ETF returns for EUSA and USMV

Before taking the data as accurate, though, some checks were done first. This was accomplished by comparing the weighted returns of the constructed index we had for our data (looking at each constituent's monthly return, multiplied by its weight), and comparing it to the actual ETF return. Thus, we wanted to check how our weighted returns compared to the ETF returns for both EUSA and USMV. Though we did not expect it to be perfectly correlated, we wanted to aim for at least a 98% or higher correlation between the weighted returns we calculated, and the ETF returns, on a monthly basis. The results for EUSA are shown below.

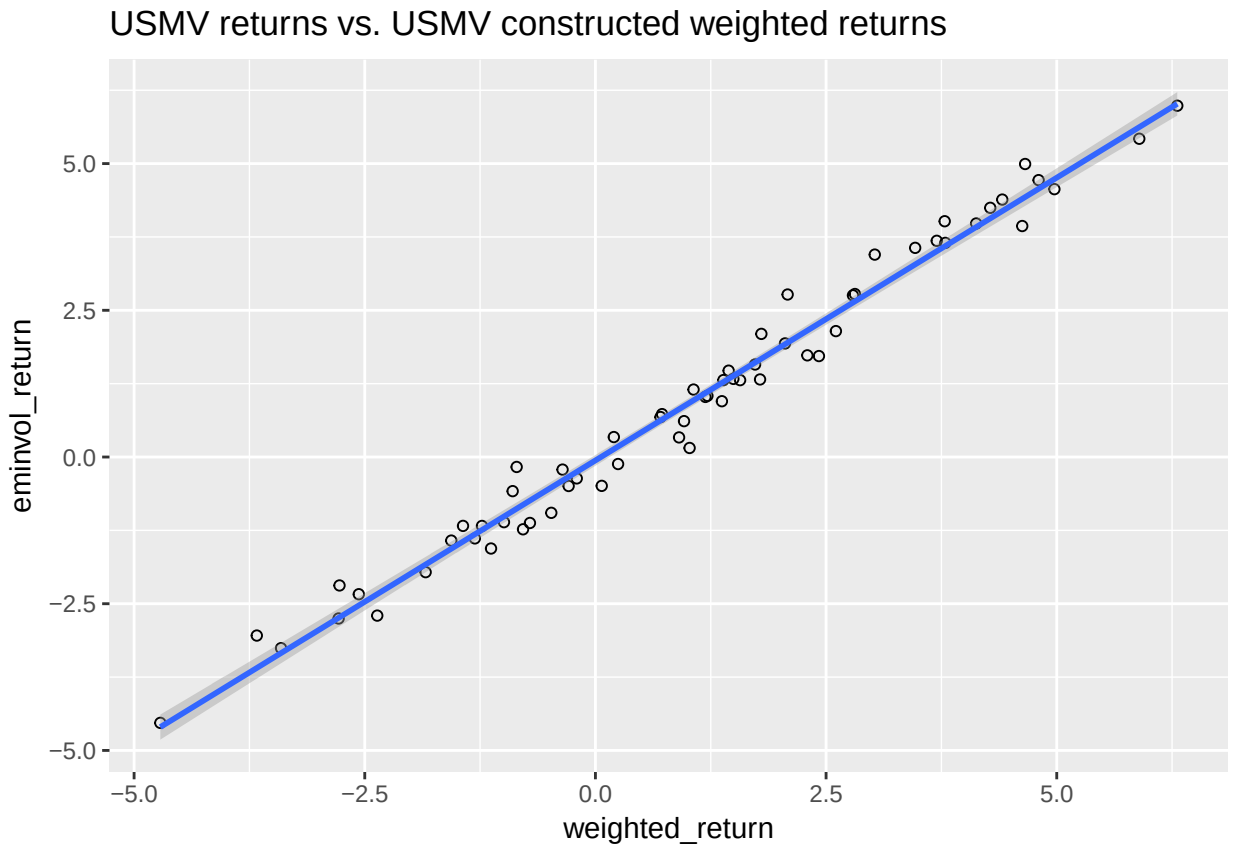



```
##                          [,1]
## Delt.1.arithmetic 0.9805503
```

As we can see, the returns seem pretty consistent and have a correlation greater than 0.98.

Shown below is the data for USMV.

```
# USA data
library(ggplot2)
data(usa)
data(minvol)
data(returns2)
ggplot(returns2, aes(x=weighted_return, y=eminvol_return)) +
    geom_point(shape=1) +  geom_smooth(method=lm) + ggtitle("USMV returns vs. USMV constructed weighted
```



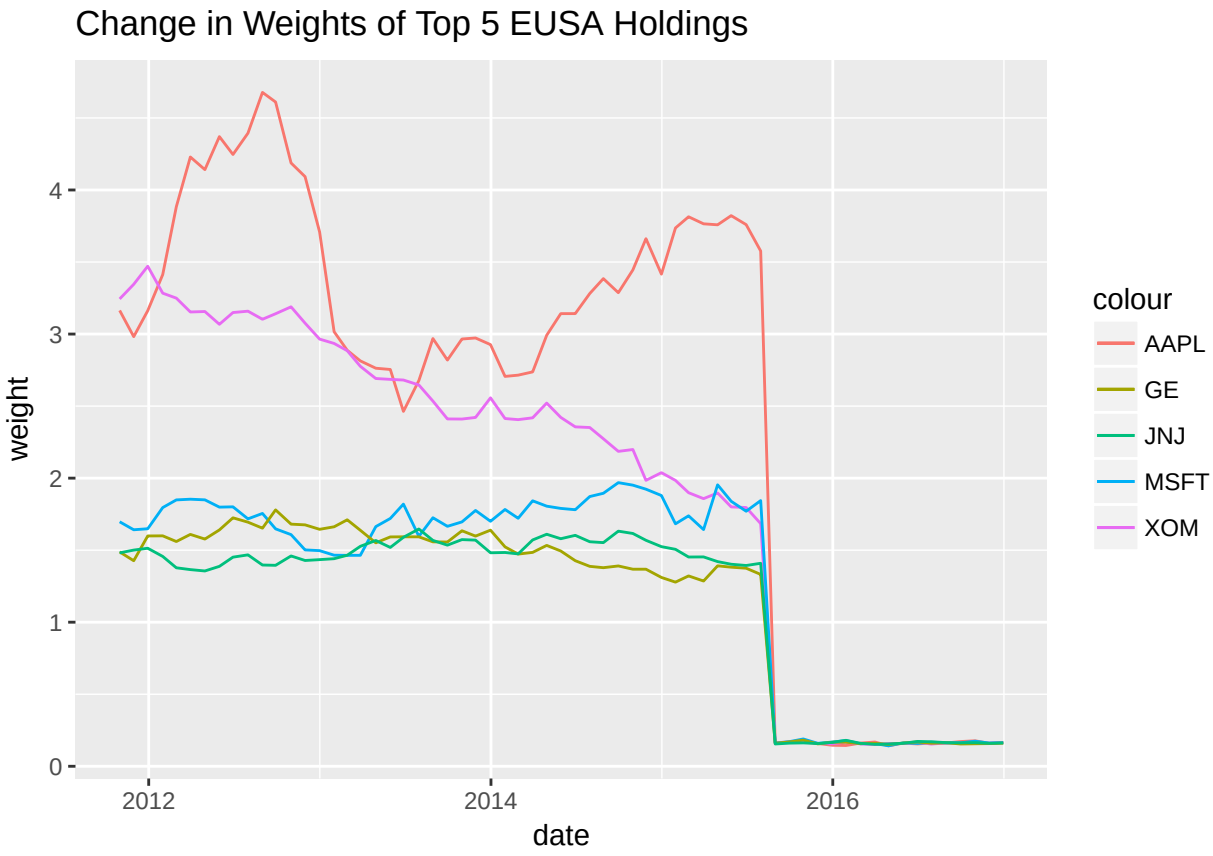

```
# Correlation between USMV returns and USMV constructed weighted returns
cor(returns2$eminvol_return, returns2$weighted_return)
```

```
##                         [,1]
## Delt.1.arithmetic 0.9906848
```

The correlation is 0.99, which is also very good.

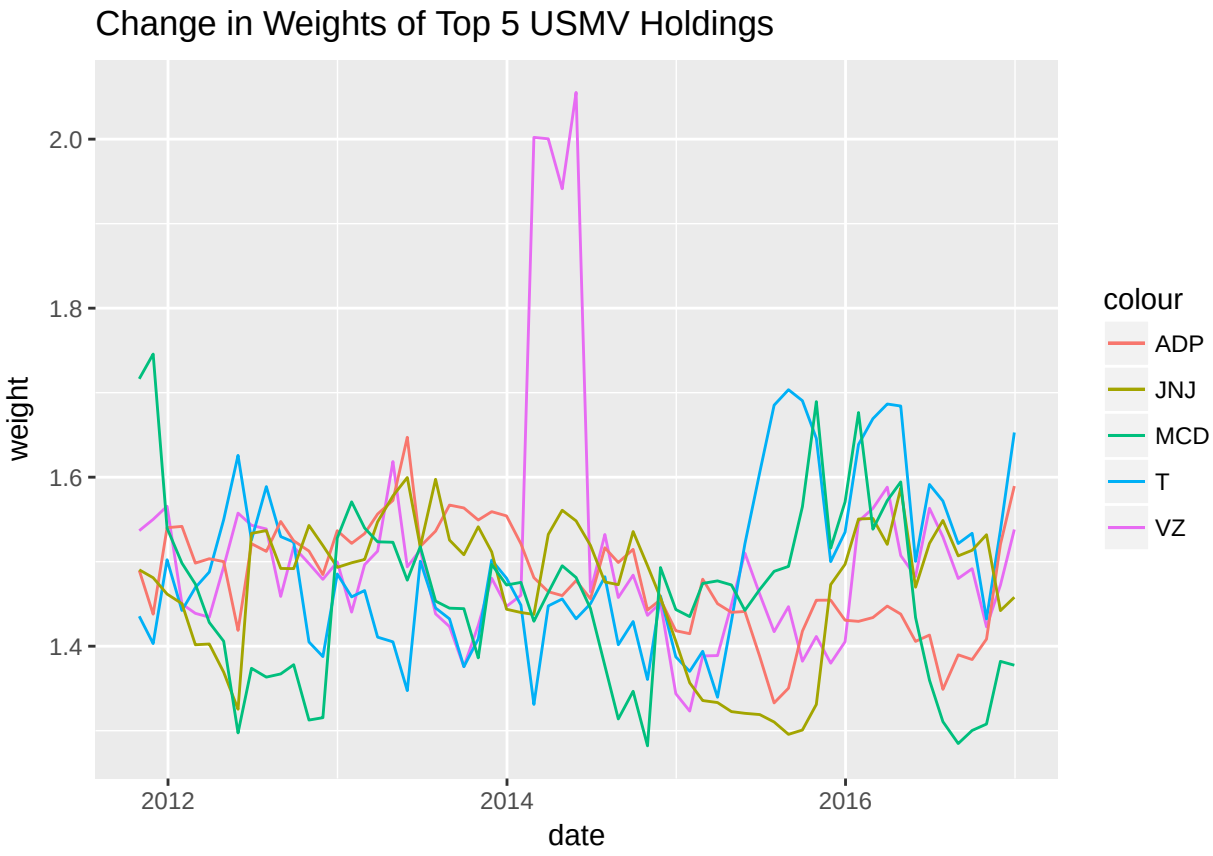
4.4.3 Change in 5 largest holdings by average weight for EUSA and USMV

The next thing we want to see is how the top 5 largest holdings, by average weight, in each index have changed in weighting over time. For EUSA, the 5 largest holdings were AAPL, XOM, MSFT, GE, and JNJ. Their change in weights are shown below.



Shown above, for EUSA, we have some very interesting findings. The weights of the 5 companies are all very high, then suddenly all spike. Verifying this in the data, showed that for all 5 companies, holdings dropped significantly between 2015-07-31 and 2015-08-31. The reason for this is not entirely clear, but the general ETF started performing poorly around this time too. In July of 2015 the price per share was 45.20, then it dropped to 42.60 the following month, and dropped again to 40.50 in August 2015. Perhaps these large companies were doing poorly, and MSCI decided to try underweighting them.

For USMV, the 5 largest holdings were VZ, T, ADP, JNJ, and MCD. Their change in weights are shown below. As we can see below, with the exception of Verizon, the holdings generally remain between 1 and 1.6 percent of the overall portfolio.


Change in Weights of Top 5 USMV Holdings
weight
2.0
1.8
1.6
1.4
2012
2014
2016
date
colour
ADP
JNJ
MCD
T
VZ

Chapter 5

Data Analysis

5.1 Sector Weights

First, sector weights were calculated over time for both EUSA and USMV. Plots were made by sector and displayed to compare the relative weights of EUSA and USMV.

Sector Weight Summary Statistics:

```
data(usa_percent)
```

```
## Warning in data(usa_percent): data set 'usa_percent' not found
```

```
data(minvol_percent)
```

```
## Warning in data(minvol_percent): data set 'minvol_percent' not found
```

```
## Summary statistics of EUSA sector weights
head(usa_percent)
```

```
##               sector_name sector_count total     percent       date
## 1 Cash and/or Derivatives            2   631 0.003169572 2017-01-05
## 2  Consumer Discretionary          106   631 0.167987322 2017-01-05
## 3        Consumer Staples           43   631 0.068145800 2017-01-05
## 4                  Energy           44   631 0.069730586 2017-01-05
## 5              Financials           84   631 0.133122029 2017-01-05
## 6             Health Care           71   631 0.112519810 2017-01-05
```

```
tail(usa_percent)
```

```
##                sector_name sector_count total     percent       date
## 827 Information Technology           79   585 0.135042735 2011-10-31
## 828              Materials           33   585 0.056410256 2011-10-31
## 829            Real Estate            0   585 0.000000000 2011-10-31
## 830         S-T Securities            1   585 0.001709402 2011-10-31
## 831     Telecommunications           12   585 0.020512821 2011-10-31
## 832              Utilities           35   585 0.059829060 2011-10-31
```

```
summary(usa_percent)
```

```
##                    sector_name  sector_count          total
##  Cash and/or Derivatives: 64   Length:832         Length:832
##  Consumer Discretionary : 64   Class :character   Class :character
```

```
##  Consumer Staples       : 64   Mode  :character   Mode  :character
##  Energy                 : 64
##  Financials             : 64
##  Health Care            : 64
##  (Other)                :448
##     percent             date
##  Min.   :0.00000   Min.   :2011-10-31
##  1st Qu.:0.01426   1st Qu.:2013-02-21
##  Median :0.06820   Median :2014-06-14
##  Mean   :0.07692   Mean   :2014-06-14
##  3rd Qu.:0.12236   3rd Qu.:2015-10-07
##  Max.   :0.19293   Max.   :2017-01-05
##
```

```
## Summary statistics of USMV sector weights
head(minvol_percent)
```

```
##               sector_name sector_count total    percent       date
## 1 Cash and/or Derivatives            2   186 0.01075269 2017-01-05
## 2  Consumer Discretionary           18   186 0.09677419 2017-01-05
## 3        Consumer Staples           24   186 0.12903226 2017-01-05
## 4                  Energy            4   186 0.02150538 2017-01-05
## 5              Financials           21   186 0.11290323 2017-01-05
## 6             Health Care           33   186 0.17741935 2017-01-05
```

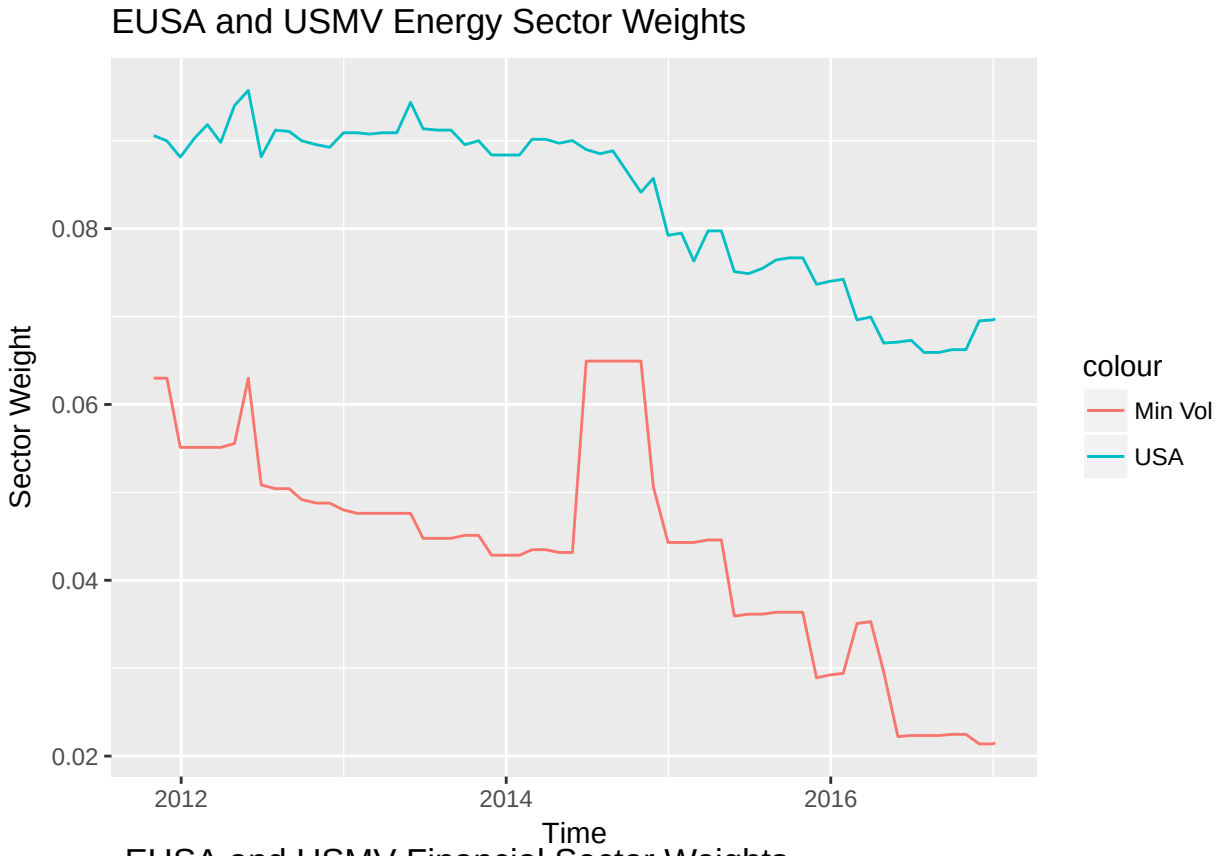
```
tail(minvol_percent)
```

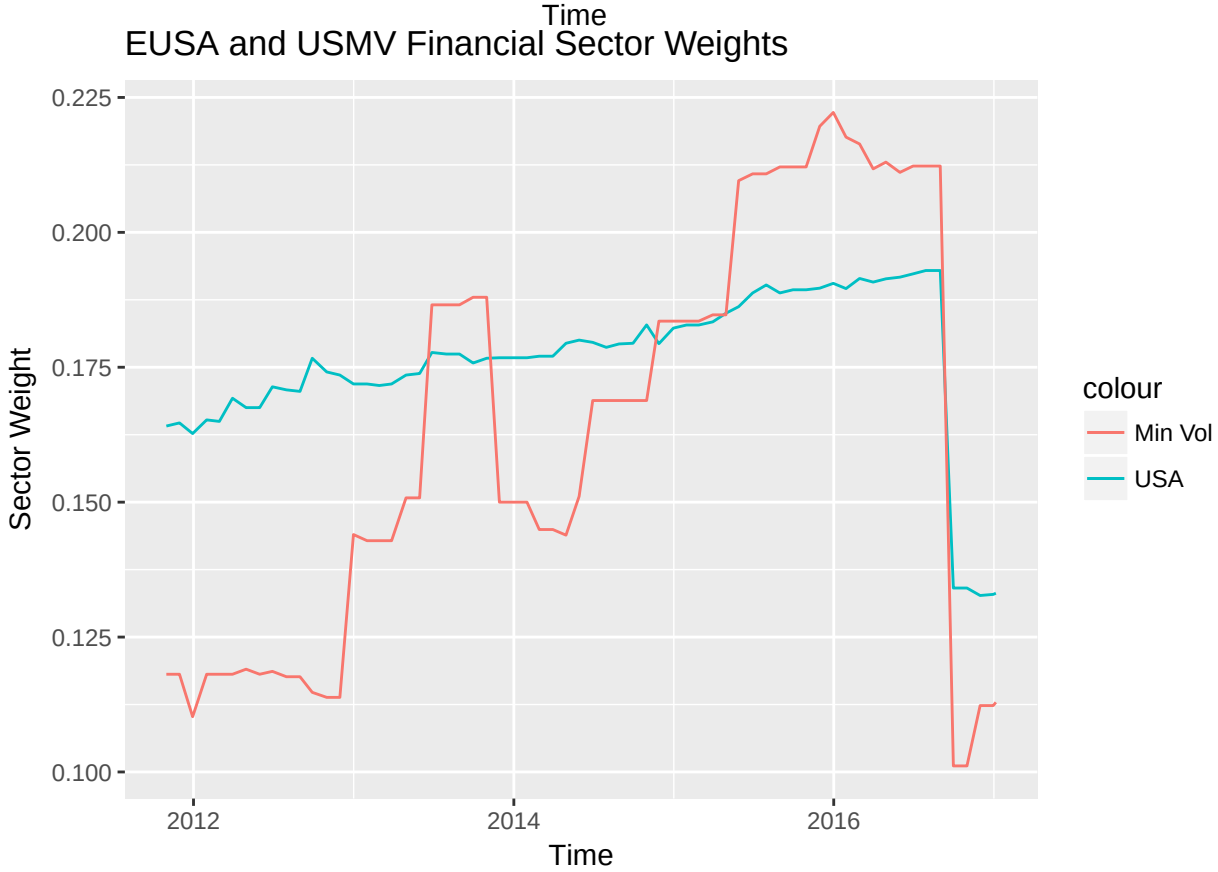
```
##                sector_name sector_count total     percent       date
## 827 Information Technology           19   127 0.149606299 2011-10-31
## 828              Materials            4   127 0.031496063 2011-10-31
## 829            Real Estate            0   127 0.000000000 2011-10-31
## 830         S-T Securities            1   127 0.007874016 2011-10-31
## 831     Telecommunications            7   127 0.055118110 2011-10-31
## 832              Utilities            9   127 0.070866142 2011-10-31
```

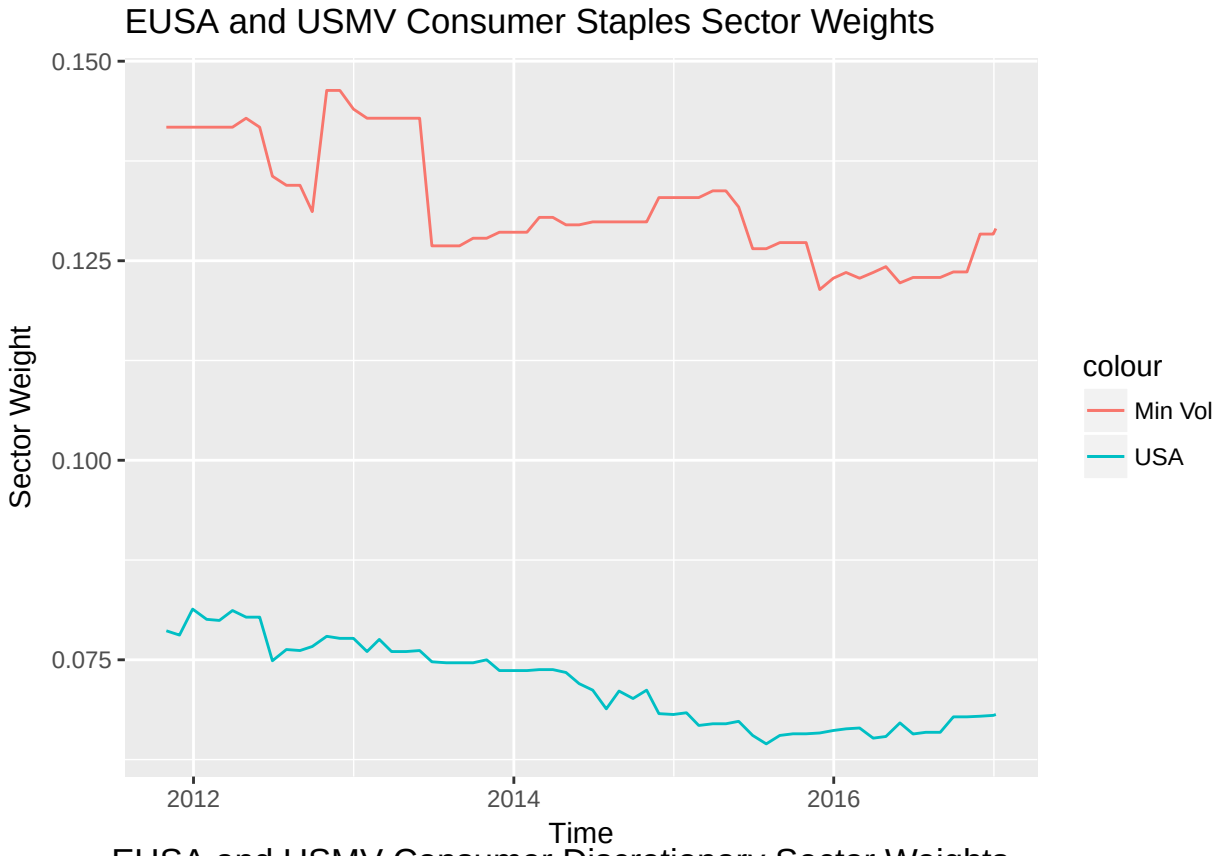
```
summary(minvol_percent)
```

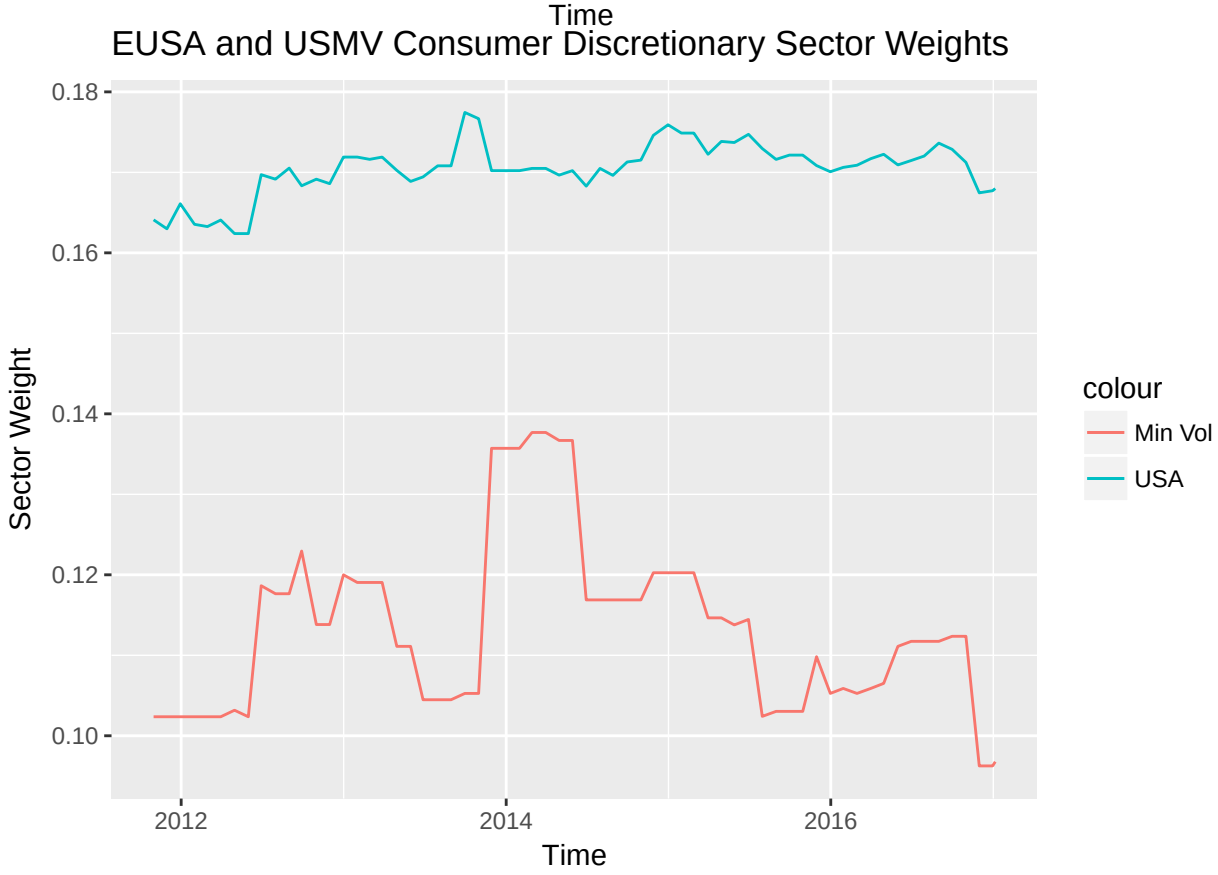
```
##                   sector_name  sector_count          total
##  Cash and/or Derivatives: 64   Length:832         Length:832
##  Consumer Discretionary : 64   Class :character   Class :character
##  Consumer Staples       : 64   Mode  :character   Mode  :character
##  Energy                 : 64
##  Financials             : 64
##  Health Care            : 64
##  (Other)                :448
##     percent             date
##  Min.   :0.00000   Min.   :2011-10-31
##  1st Qu.:0.02235   1st Qu.:2013-02-21
##  Median :0.06349   Median :2014-06-14
##  Mean   :0.07692   Mean   :2014-06-14
##  3rd Qu.:0.13043   3rd Qu.:2015-10-07
##  Max.   :0.22222   Max.   :2017-01-05
##
```

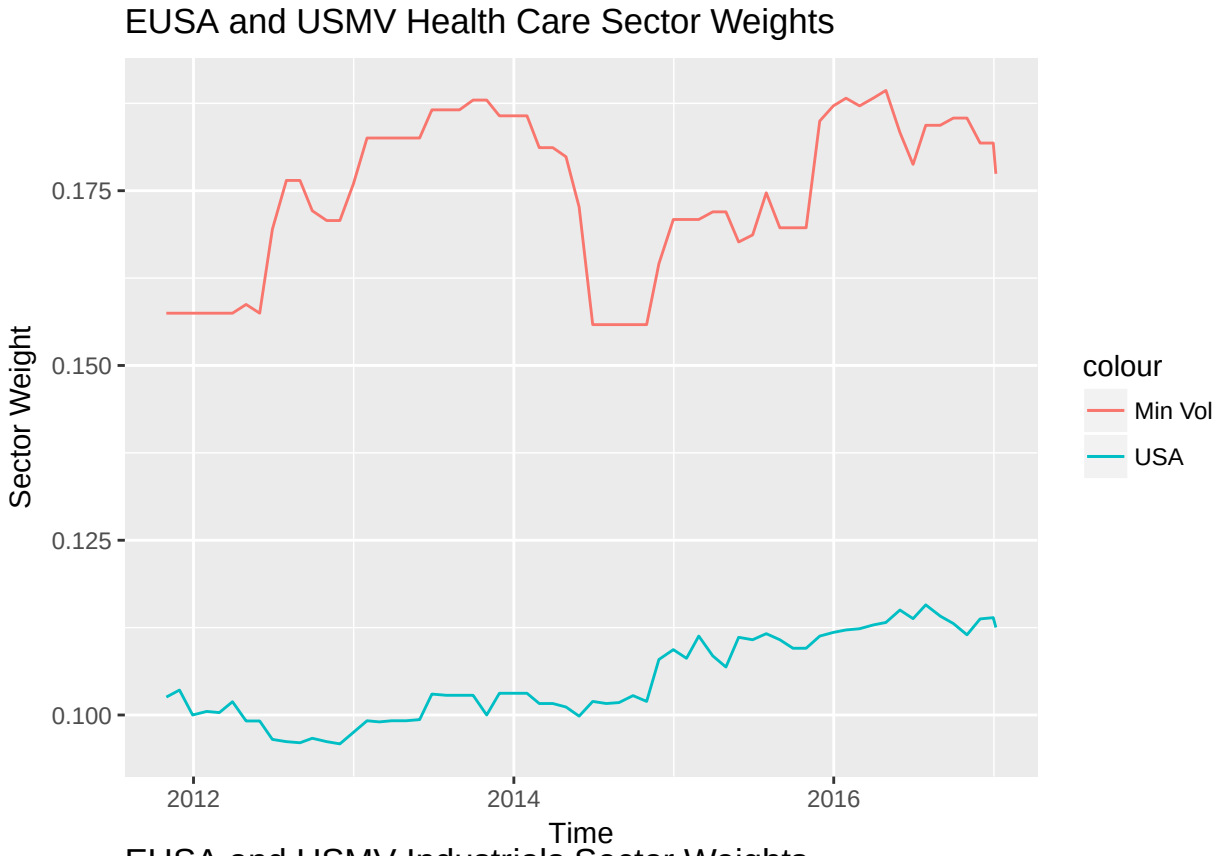
Sector Weights for EUSA and USMV:

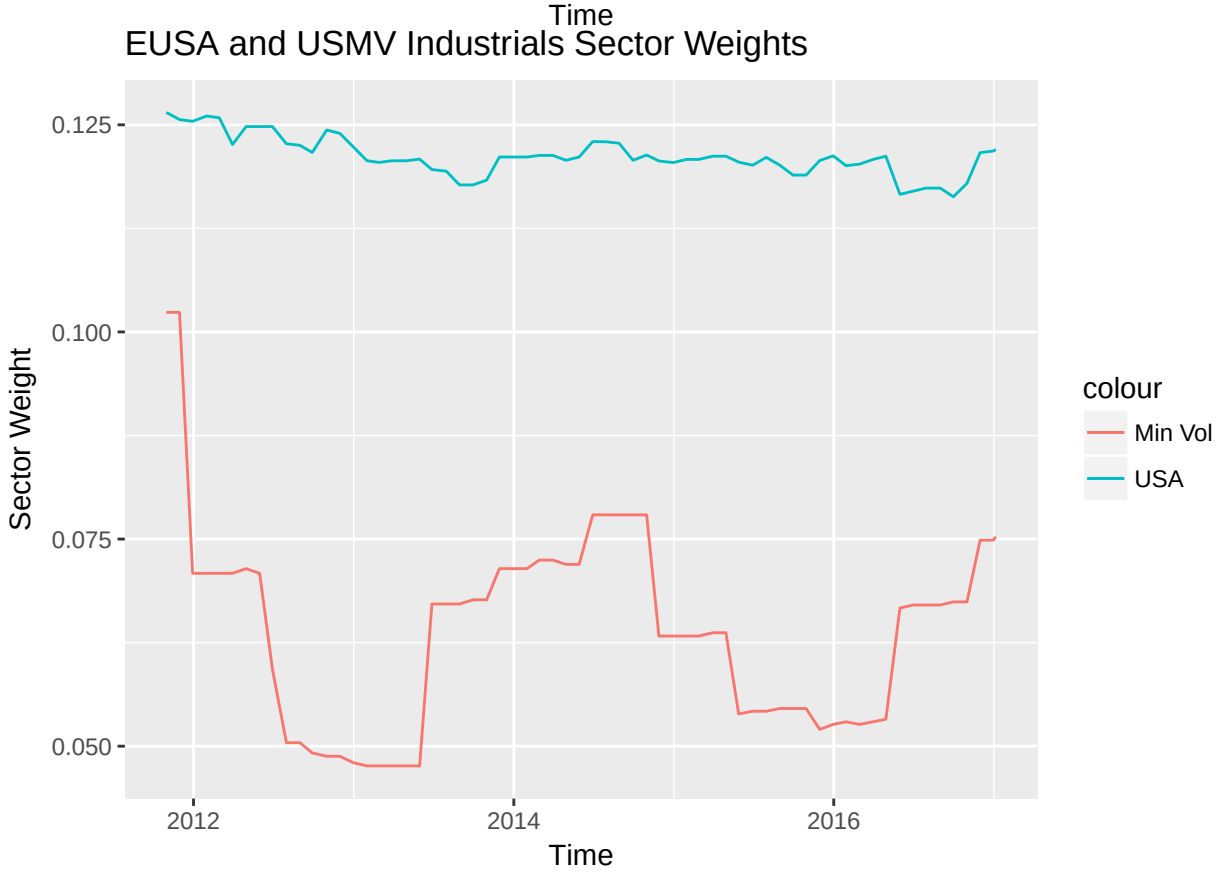

EUSA and USMV Energy Sector Weights
Sector Weight
0.08
0.06
0.04
0.02
2012
2014
2016
Time
colour
Min Vol
USA

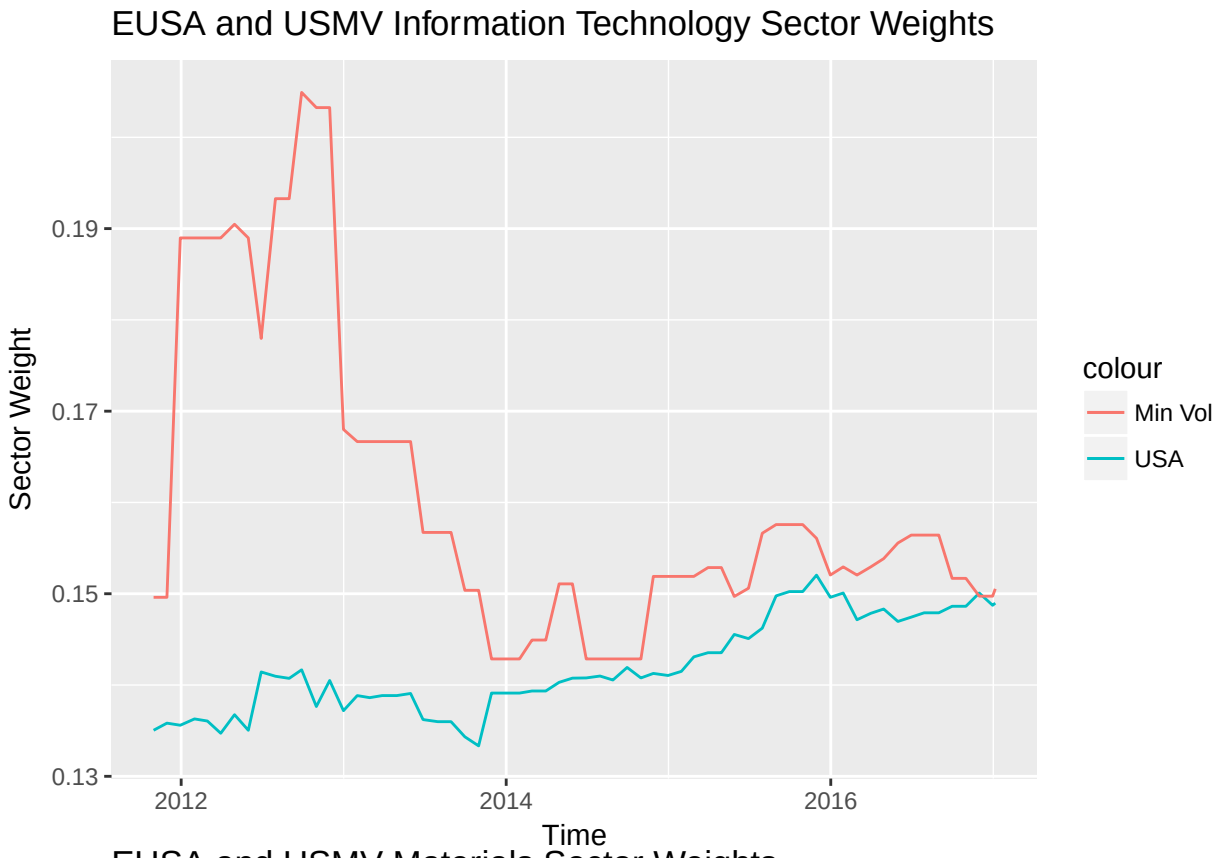

EUSA and USMV Financial Sector Weights
Sector Weight
0.225
0.200
0.175
0.150
0.125
0.100
2012
2014
2016
Time
colour
Min Vol
USA

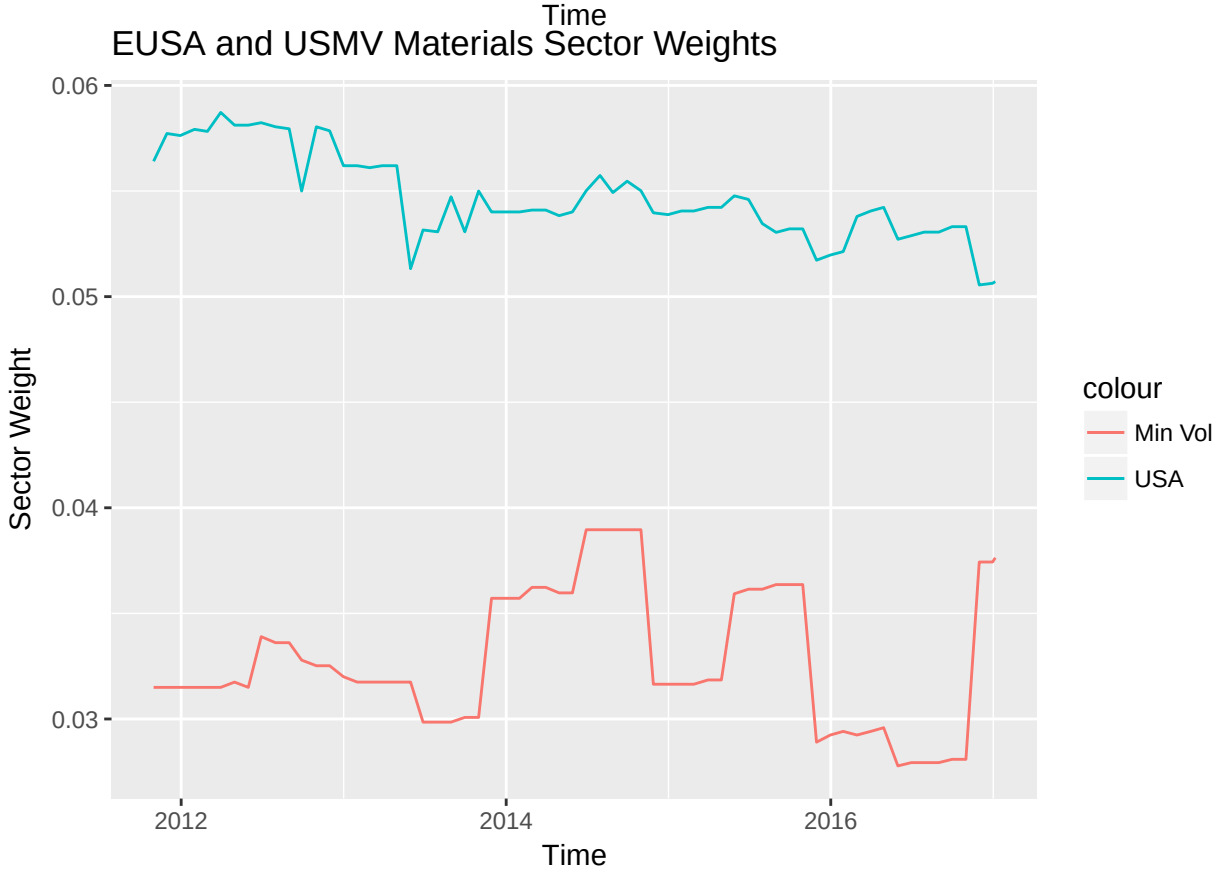

EUSA and USMV Consumer Staples Sector Weights
Sector Weight
0.150
0.125
0.100
0.075
2012
2014
2016
Time
colour
Min Vol
USA

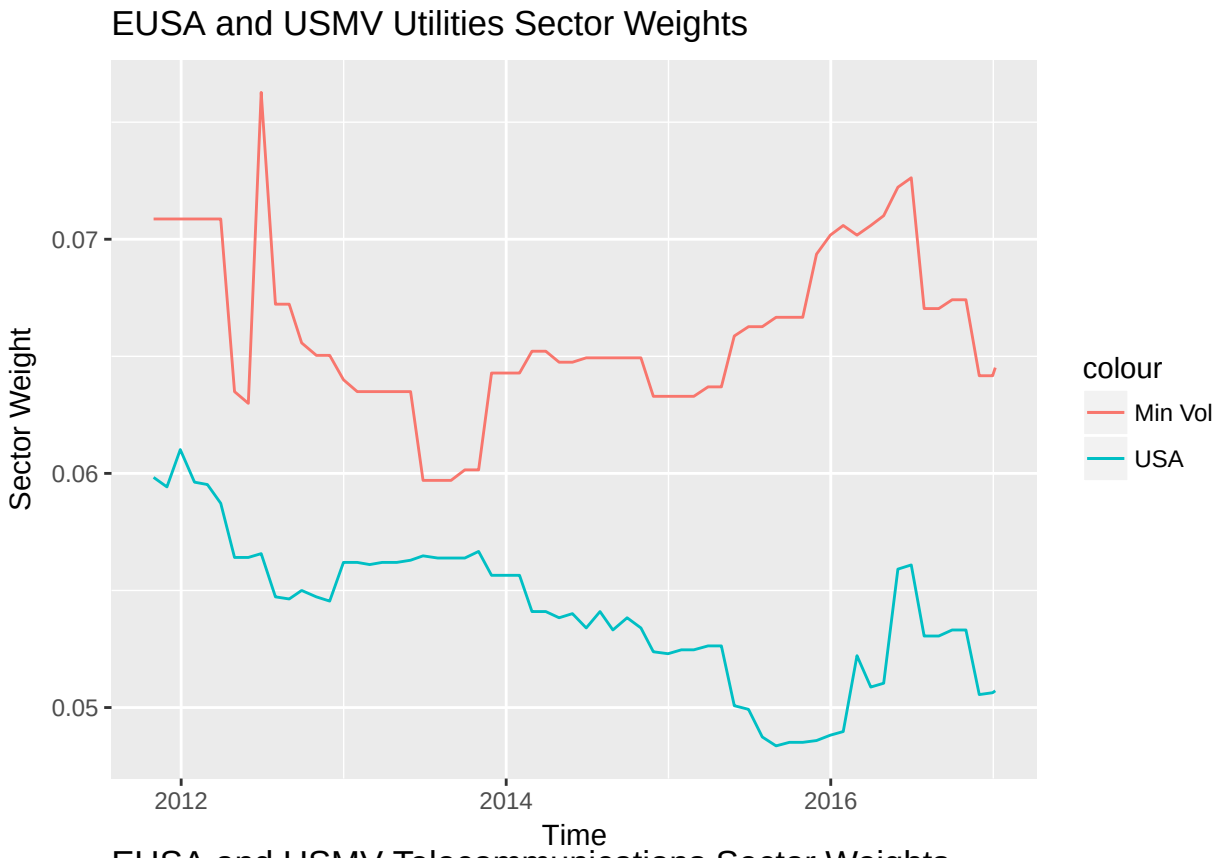

EUSA and USMV Consumer Discretionary Sector Weights
Sector Weight
0.18
0.16
0.14
0.12
0.10
2012
2014
2016
Time
colour
Min Vol
USA

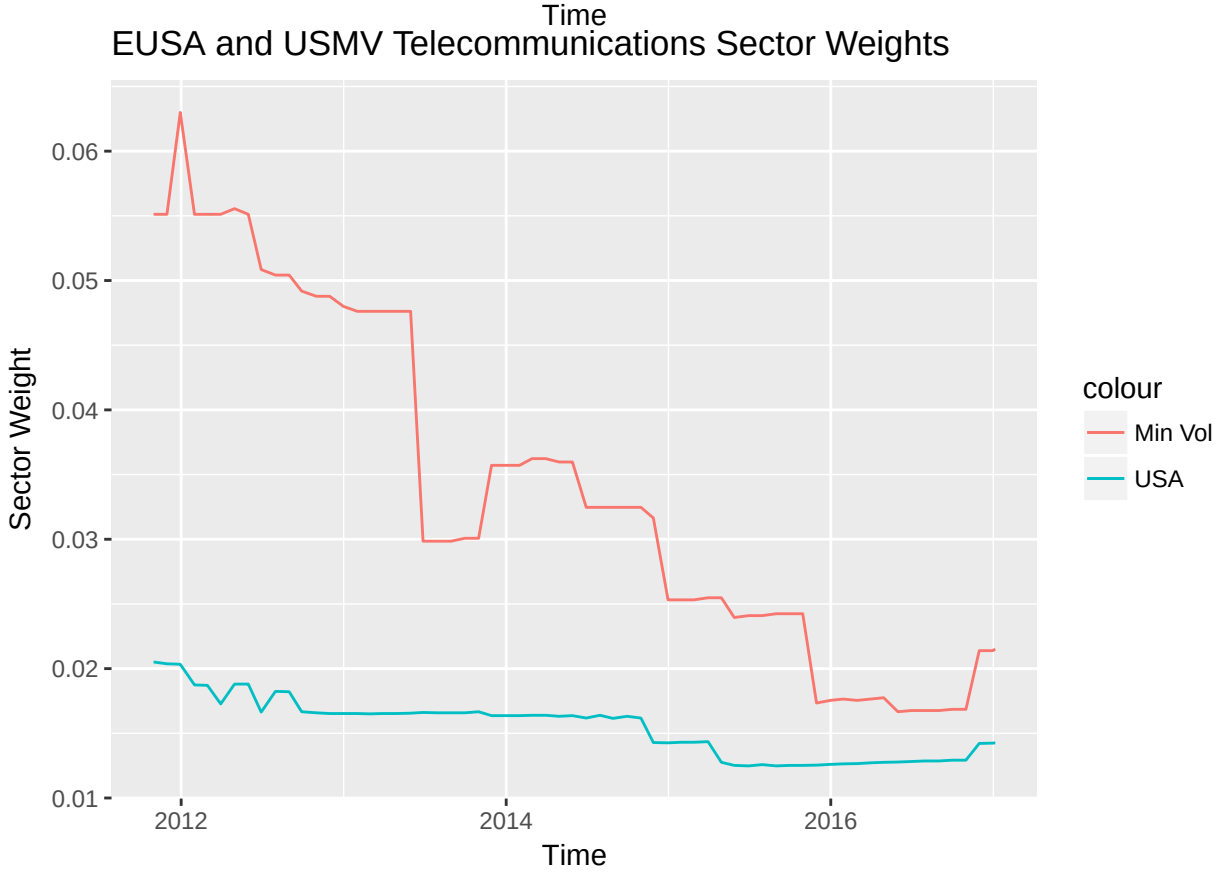

EUSA and USMV Health Care Sector Weights
Sector Weight
0.175
0.150
0.125
0.100
2012
2014
2016
Time
colour
Min Vol
USA


EUSA and USMV Industrials Sector Weights
Sector Weight
0.125
0.100
0.075
0.050
2012
2014
2016
Time
colour
Min Vol
USA


EUSA and USMV Information Technology Sector Weights
Sector Weight
0.19
0.17
0.15
0.13
2012
2014
2016
Time
colour
Min Vol
USA


EUSA and USMV Materials Sector Weights
Sector Weight
0.06
0.05
0.04
0.03
2012
2014
2016
Time
colour
Min Vol
USA


EUSA and USMV Utilities Sector Weights
Sector Weight
0.07
0.06
0.05
2012
2014
2016
Time
colour
Min Vol
USA


EUSA and USMV Telecommunications Sector Weights
Sector Weight
0.06
0.05
0.04
0.03
0.02
0.01
2012
2014
2016
Time
colour
Min Vol
USA

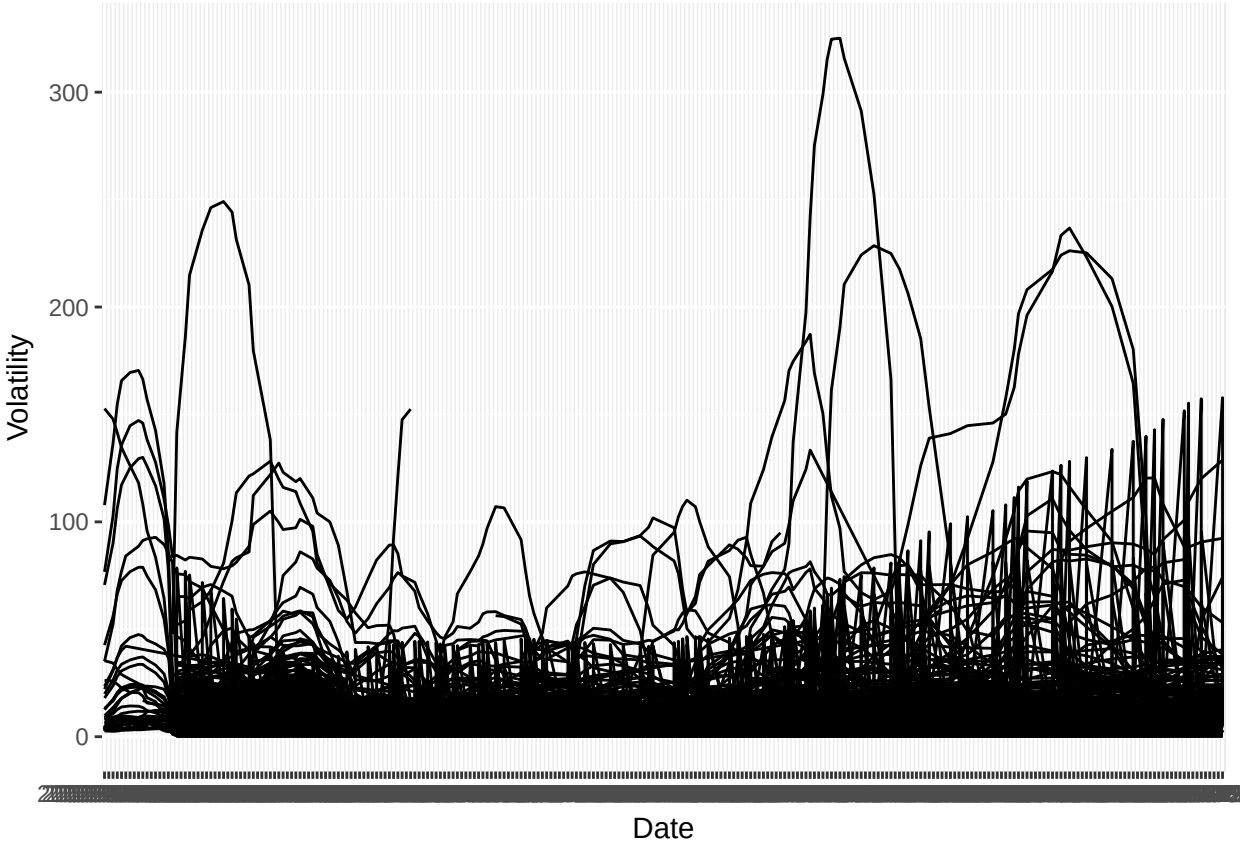
5.2 EUSA Constituent Trailing Volatilities

Data was collected from the past 10 years of the EUSA index. The data was collected from 12/31/2006 to 12/30/2016, was collected from WRDS for the 908 historical constituents of the USA Equal Weight (EUSA) index, of which USMV is derived. Each tickers' 252-day (annual) trailing volatility was calculated and a month end spagetti plot was produced.

```
##         Date Ticker Volatility
## 1 2008-01-31      A   2.289449
## 2 2008-02-29      A   2.357208
## 3 2008-03-31      A   2.660299
## 4 2008-04-30      A   3.017754
## 5 2008-05-30      A   3.028868
## 6 2008-06-30      A   2.921781
```

```
##             Date Ticker Volatility
## 82257 2016-07-29    ZTS   2.718338
## 82258 2016-08-31    ZTS   3.150629
## 82259 2016-09-30    ZTS   3.363564
## 82260 2016-10-31    ZTS   3.372530
## 82261 2016-11-30    ZTS   3.436391
## 82262 2016-12-30    ZTS   3.664053
```

```
##          Date           Ticker        Volatility
##  2014-09-30:  774   GE     :  432   Min.   :  0.0492
##  2014-10-31:  774   LSI    :  246   1st Qu.:  2.4407
##  2014-11-28:  774   UA     :  234   Median :  4.3057
##  2014-12-31:  773   CBS    :  228   Mean   :  6.9432
##  2014-06-30:  771   LEN    :  228   3rd Qu.:  7.4648
##  2014-07-31:  771   MKC    :  228   Max.   :325.1242
##  (Other)   :77625   (Other):80666
```



5.3 EUSA Constituent Trailing Betas

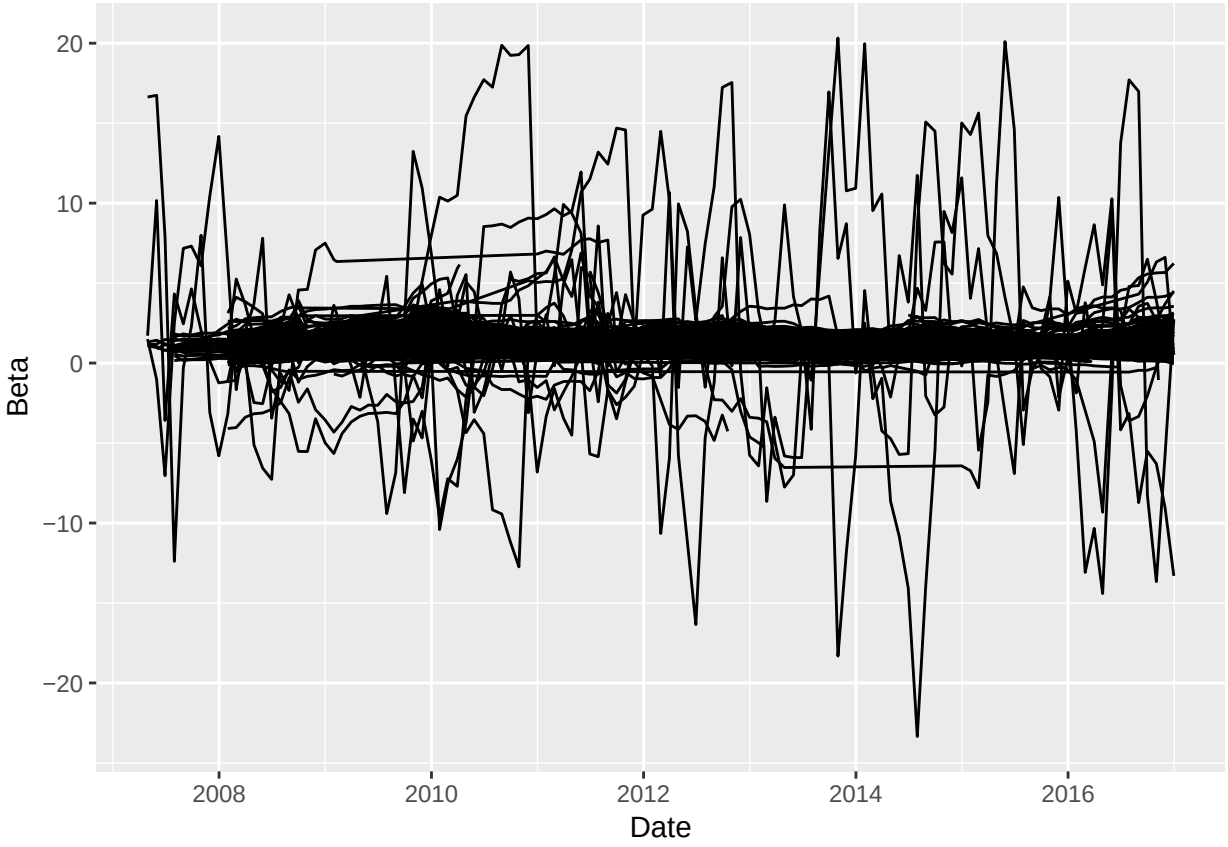
Data was collected from the past 10 years of the EUSA index. The data was collected from 12/31/2006 to 12/30/2016, was collected from WRDS for the 908 historical constituents of the USA Equal Weight (EUSA) index, of which USMV is derived. Each tickers' 252-day (annual) trailing beta was calculated and a month end spagetti plot was produced.

```
##           Date Ticker      Beta
## 1         <NA>   <NA>        NA
## 271 2008-01-31      A 0.9538067
## 291 2008-02-29      A 0.9473060
## 311 2008-03-31      A 0.9376670
## 333 2008-04-30      A 0.9588268
## 354 2008-05-30      A 0.9680630
```

```
##              Date Ticker      Beta
## 879817 2016-07-29    ZTS 1.0224673
## 902806 2016-08-31    ZTS 1.0302957
## 923805 2016-09-30    ZTS 0.9779760
## 944802 2016-10-31    ZTS 0.9830767
## 965801 2016-11-30    ZTS 0.9220281
## 986798 2016-12-30    ZTS 0.9549808
```

```
##       Date               Ticker              Beta
##  Min.   :2007-04-30   Length:78532       Min.   :-23.3438
##  1st Qu.:2010-04-30   Class :character   1st Qu.:  0.8204
```

```
##  Median :2012-07-31   Mode  :character   Median :  1.0523
##  Mean   :2012-07-21                      Mean   :  1.0993
##  3rd Qu.:2014-10-31                      3rd Qu.:  1.3295
##  Max.   :2016-12-30                      Max.   : 20.3256
##  NA's   :1                               NA's   :1
```



5.4 EUSA Constituent Price to Book Ratios

Data was collected from the past 10 years of the EUSA index. The data was collected from 12/31/2006 to 12/30/2016, was collected from WRDS for the 908 historical constituents of the USA Equal Weight (EUSA) index, of which USMV is derived. Each tickers' Price to Book ratio was calculated in two ways, to ensure accuracy.

```
##    gvkey       Date Year Ticker Total_Assets BV_per_share
## 1 126554 2007-10-31 2007      A    7.554e+09       8.7405
## 2 126554 2008-10-31 2008      A    7.437e+09       7.3114
## 3 126554 2009-10-31 2009      A    7.612e+09       7.2397
## 4 126554 2010-10-31 2010      A    9.696e+09       9.3256
## 5 126554 2011-10-31 2011      A    9.057e+09      12.4371
## 6 126554 2011-10-31 2011      A    9.049e+09           NA
##   Shares_Outstanding Total_Liabilities Market_Value Share_Price Book_Value
## 1          370000000         4.320e+09  13634500000       36.85 3233985000
## 2          350000000         4.878e+09   7766500000       22.19 2558990000
## 3          346148000         5.106e+09   8563701500       24.74 2506007676
## 4          346144000         6.460e+09  12045811200       34.80 3228000486
## 5          346382000         4.741e+09  12840380700       37.07 4307987572
```

```
## 6                 NA                NA  12840380700          NA         NA
##       PBR1     PBR2
## 1 4.216006 4.216006
## 2 3.034986 3.034986
## 3 3.417269 3.417269
## 4 3.731663 3.731663
## 5 2.980598 2.980598
## 6       NA       NA
```

```
##       gvkey       Date Year Ticker Total_Assets BV_per_share
## 17182 13721 2014-12-31 2014    ZTS    6.607e+09       2.6151
## 17183 13721 2014-12-31 2014    ZTS    6.588e+09           NA
## 17184 13721 2015-12-31 2015    ZTS    7.913e+09       2.1472
## 17185 13721 2015-12-31 2015    ZTS    7.913e+09           NA
## 17186 13721 2016-12-31 2016    ZTS    7.649e+09       3.0171
## 17187 13721 2016-12-31 2016    ZTS    7.649e+09           NA
##       Shares_Outstanding Total_Liabilities Market_Value Share_Price
## 17182          501328000         5.270e+09  21572143800       43.03
## 17183                 NA                NA  21572143800          NA
## 17184          497400000         6.822e+09  23835408000       47.92
## 17185                 NA                NA  23835408000          NA
## 17186          492855000         6.150e+09  26382528200       53.53
## 17187                 NA                NA  26382528200          NA
##       Book_Value     PBR1     PBR2
## 17182 1311022853 16.45444 16.45444
## 17183         NA       NA       NA
## 17184 1068017280 22.31744 22.31744
## 17185         NA       NA       NA
## 17186 1486992820 17.74220 17.74220
## 17187         NA       NA       NA
```

```
##      gvkey                Date           Year          Ticker
##  Min.   :  1045   2013-12-31:1283   Min.   :2006   ACGL   :   33
##  1st Qu.:  7146   2012-12-31:1277   1st Qu.:2009   AET    :   33
##  Median : 14824   2014-12-31:1275   Median :2011   AFL    :   33
##  Mean   : 50895   2011-12-31:1265   Mean   :2011   AIZ    :   33
##  3rd Qu.: 65556   2015-12-31:1253   3rd Qu.:2014   AMG    :   33
##  Max.   :294524   2010-12-31:1222   Max.   :2016   ANTM   :   33
##                   (Other)   :9612   NA's   :33     (Other):16989
##   Total_Assets        BV_per_share      Shares_Outstanding
##  Min.   :0.000e+00   Min.   :-1489600   Min.   :0.000e+00
##  1st Qu.:3.912e+09   1st Qu.:       8   1st Qu.:1.010e+08
##  Median :9.538e+09   Median :      15   Median :1.973e+08
##  Mean   :4.725e+10   Mean   :    5311   Mean   :4.654e+08
##  3rd Qu.:2.632e+10   3rd Qu.:      27   3rd Qu.:4.344e+08
##  Max.   :2.573e+12   Max.   :16297416   Max.   :2.906e+10
##  NA's   :2568        NA's   :9284       NA's   :9188
##  Total_Liabilities    Market_Value        Share_Price
##  Min.   :0.000e+00   Min.   :3.545e+06   Min.   :   0.027
##  1st Qu.:2.135e+09   1st Qu.:4.445e+09   1st Qu.:  25.985
##  Median :6.310e+09   Median :8.851e+09   Median :  43.180
##  Mean   :4.451e+10   Mean   :2.166e+10   Mean   :  57.339
##  3rd Qu.:1.865e+10   3rd Qu.:1.942e+10   3rd Qu.:  67.955
##  Max.   :2.341e+12   Max.   :6.266e+11   Max.   :1466.060
##  NA's   :7785        NA's   :2764        NA's   :10064
```

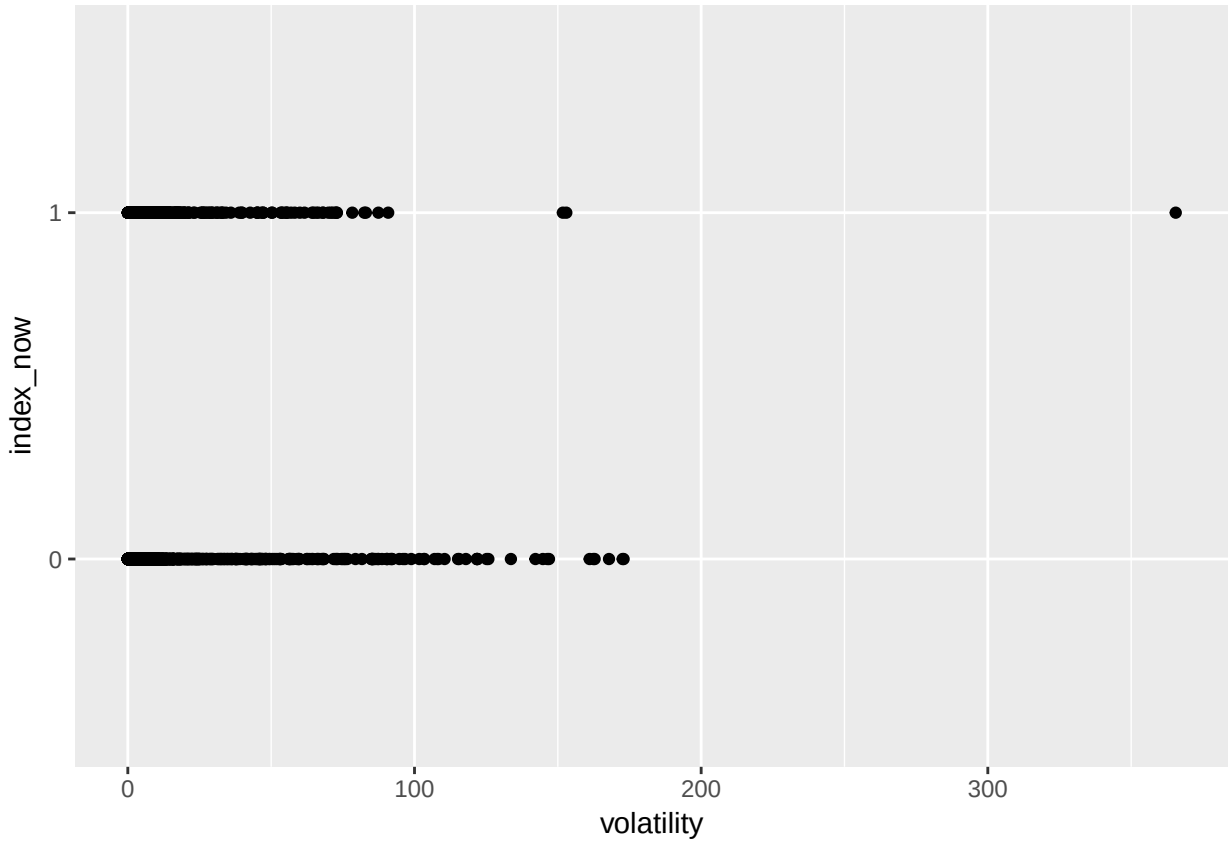
```
##    Book_Value             PBR1              PBR2          
##  Min.   :-8.615e+10   Min.   :-687.634   Min.   :-687.634  
##  1st Qu.: 1.313e+09   1st Qu.:   1.595   1st Qu.:   1.595  
##  Median : 3.121e+09   Median :   2.626   Median :   2.626  
##  Mean   : 8.170e+09   Mean   :   4.503   Mean   :   4.503  
##  3rd Qu.: 7.376e+09   3rd Qu.:   4.416   3rd Qu.:   4.416  
##  Max.   : 2.416e+11   Max.   :1575.000   Max.   :1575.000  
##  NA's   :9284         NA's   :10079      NA's   :10079
```

Chapter 6

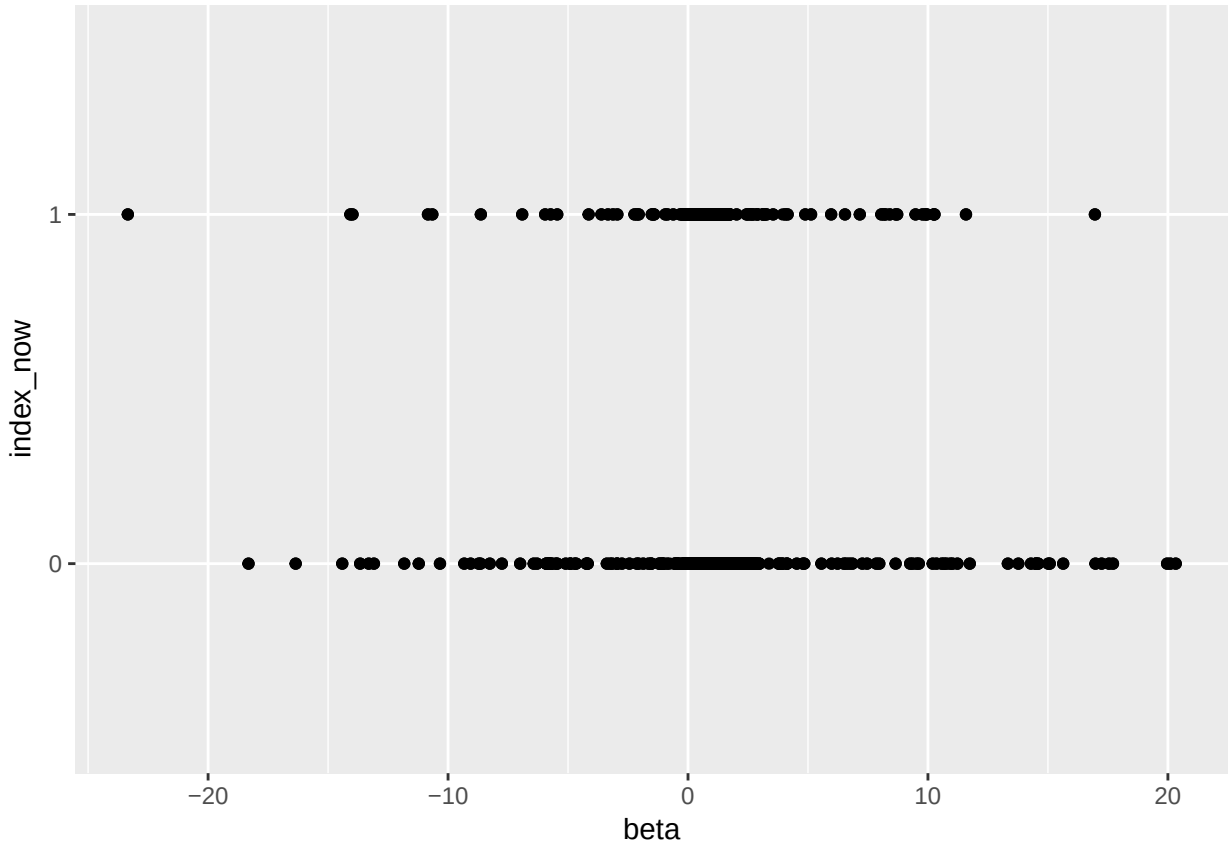
Data Distribution

Here, the distribution of the predictor variable, if the stock is in the index now, with respect to each response variable will be shown.

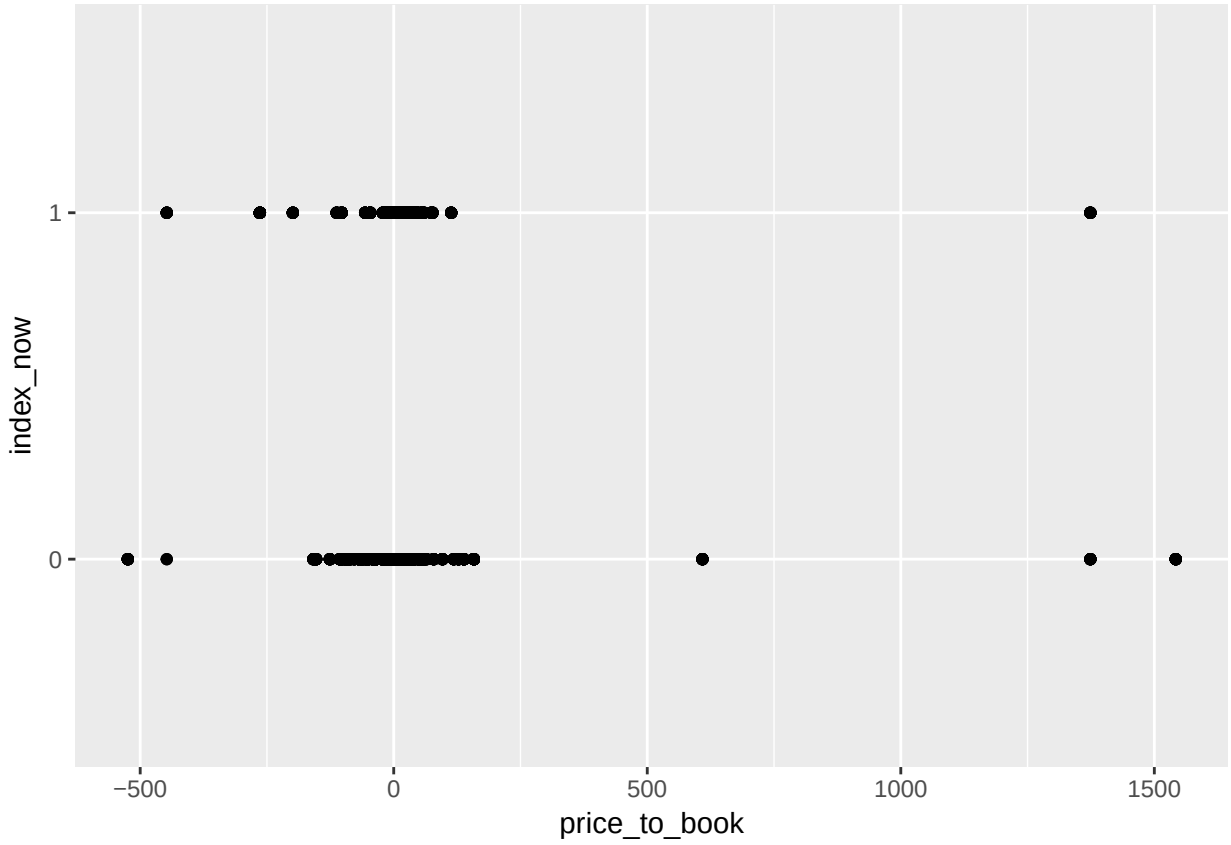
6.1 Index now vs. Trailing Volatility



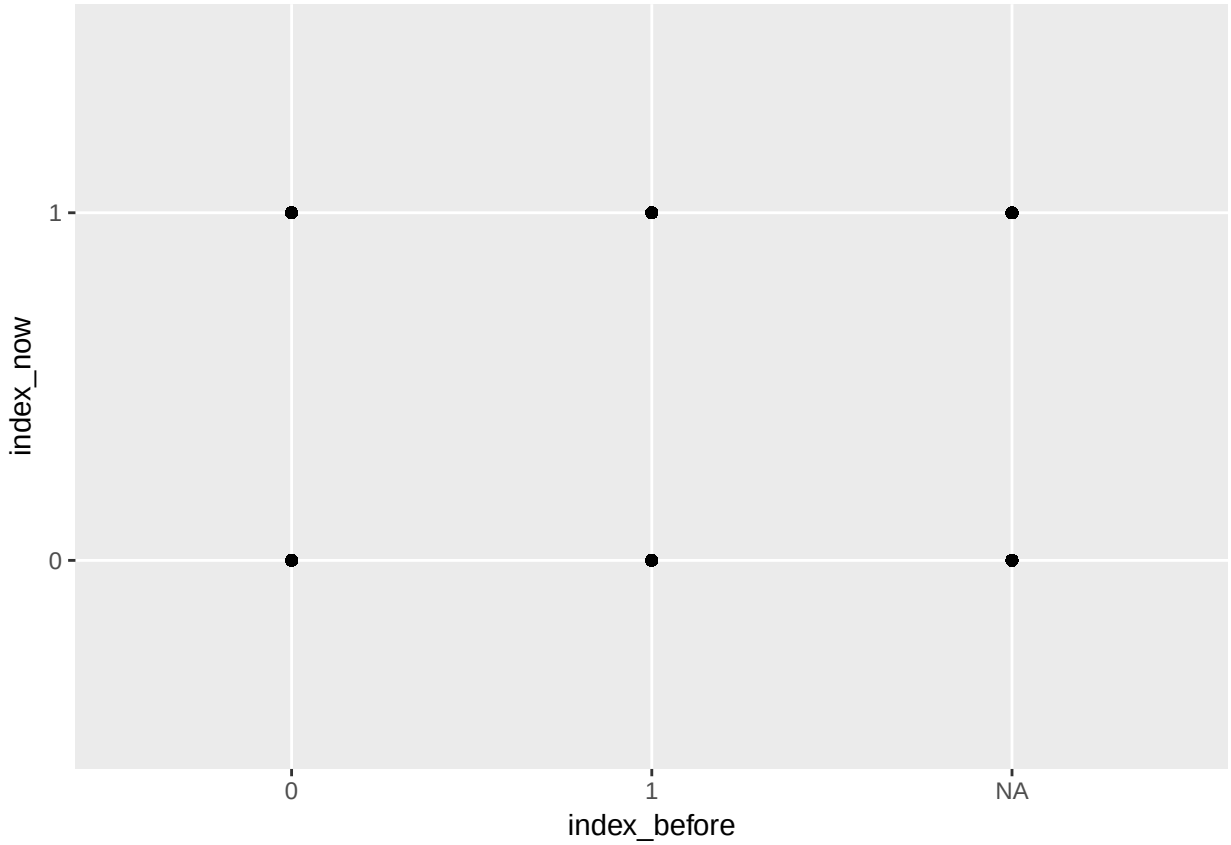
6.2 Index now vs. Trailing Beta



6.3 Index now vs. Price to Book Ratio



6.4 Index now vs. Index 6 months ago



Chapter 7

Model

Once the final data set was created and cleaned, with a number of response variables including trailing beta, trailing vol, and price to book value, and the associated outcome, which was measured by whether or not a stock was in the Min Vol index or not (1 if in, 0 if not in).

A snippet of the final data set is shown below

```
head(trainingData)
```

```
## # A tibble: 6 x 8
## # Groups:   ticker [6]
##         date ticker      beta volatility price_to_book weight index_now
##       <date> <fctr>     <dbl>      <dbl>         <dbl>  <dbl>    <fctr>
## 1 2013-09-30     CL 1.0451272  0.6023338    14.8676397 0.7228         1
## 2 2013-05-31    MKC 4.1521567  1.7358281     5.0811086 0.9495         1
## 3 2014-12-31    SJM 0.7885774  1.9425497     1.9548413 0.0497         1
## 4 2014-07-31    SPG 0.4961165  0.9387595     4.5435197 0.3643         1
## 5 2014-08-29     RE 0.7246354  0.6597441     0.7347133 0.5964         1
## 6 2016-03-31    EXR 0.6859407  1.7823979     2.7733161 0.0890         1
## # ... with 1 more variables: index_before <fctr>
```

```
tail(trainingData)
```

```
## # A tibble: 6 x 8
## # Groups:   ticker [6]
##         date ticker      beta volatility price_to_book weight index_now
##       <date> <fctr>     <dbl>      <dbl>         <dbl>  <dbl>    <fctr>
## 1 2012-08-31    MOS 1.3932857  1.0350886      1.692923      0         0
## 2 2013-01-31    HON 1.2007977  0.7540130      3.902511      0         0
## 3 2014-04-30   GILD 1.5797310  3.4051551     10.079184      0         0
## 4 2014-10-31     UA 1.4545868  0.9678980      3.496854      0         0
## 5 2016-10-31   XLNX 1.0518379  0.3432995      4.645901      0         0
## 6 2015-12-31     XL 0.7457147  0.8823093      1.719251      0         0
## # ... with 1 more variables: index_before <fctr>
```

```
summary(trainingData)
```

```
##       date                ticker           beta
##  Min.   :2012-01-31   MKC    :   86   Min.   :-23.3438
##  1st Qu.:2013-07-31   CB     :   52   1st Qu.:  0.7327
##  Median :2014-10-31   ADP    :   45   Median :  0.9262
```

```
##  Mean   :2014-10-01   K      :   45   Mean   :  0.9541
##  3rd Qu.:2015-11-30   SBAC   :   45   3rd Qu.:  1.1567
##  Max.   :2016-12-30   XEL    :   45   Max.   : 17.5440
##                       (Other):10880
##    volatility        price_to_book          weight       index_now
##  Min.   :  0.01073   Min.   :-524.357   Min.   :0.0000   0:5490
##  1st Qu.:  0.53315   1st Qu.:   1.726   1st Qu.:0.0000   1:5708
##  Median :  0.90939   Median :   2.954   Median :0.0526
##  Mean   :  1.81575   Mean   :   4.742   Mean   :0.3455
##  3rd Qu.:  1.57163   3rd Qu.:   4.957   3rd Qu.:0.6123
##  Max.   :152.96132   Max.   :1542.215   Max.   :2.7535
##
##  index_before
##  0:5988
##  1:5210
##
##
##
##
##
```

Given the nature of the data, a logit regression will be ran. Looking at all of the historical data and stock various characteristics, this would model the log odds of a stock being in the minimum volatility index as a combination of the linear predictors mentioned. Several models will be run in a panel, including one by certain months, and one by the entire pool of data.

7.1 Model 1: Entire Data Set (Monthly)

The first logit model that will be run is for the entire pool of monthly data.

7.1.1 Data Cleaning - Checking for Class Bias

Ideally, the proportion of stocks in and out of the USMV index should approximately be the same. Checking this, we can see that this is not the case. However, just around 24% of the data is from stocks that ereurrently in the index, so there is a class bias. As a result, we must sample the observations in approximately equal proportions to get better models.

```
table(monthly_final$index_now)
```

```
##
##     0     1
## 28639  8994
```

7.1.2 Create Training and Test Samples

One way to address the problem of class bias is to draw the 0's and 1's for the trainingData (development sample) in equal proportions. In doing so, we will put rest of the inputData not included for training into testData (validation sample). As a result, the size of development sample will be smaller that validation, which is okay, because, there are large number of observations.

```
# Create Training Data
input_ones <- monthly_final[which(monthly_final$index_now == 1), ]  # all 1's
```

```
input_zeros <- monthly_final[which(monthly_final$index_now == 0), ]  # all 0's
set.seed(100)  # for repeatability of samples
input_ones_training_rows <- sample(1:nrow(input_ones), 0.7*nrow(input_ones))  # 1's for training
input_zeros_training_rows <- sample(1:nrow(input_zeros), 0.7*nrow(input_ones))  # 0's for training. Pic
training_ones <- input_ones[input_ones_training_rows, ]
training_zeros <- input_zeros[input_zeros_training_rows, ]
trainingData <- rbind(training_ones, training_zeros)  # row bind the 1's and 0's
# Create Test Data
test_ones <- input_ones[-input_ones_training_rows, ]
test_zeros <- input_zeros[-input_zeros_training_rows, ]
testData <- rbind(test_ones, test_zeros)  # row bind the 1's and 0's
# Remove NA values in index_before
testData <- subset(testData, !is.na(index_before))
trainingData <- subset(trainingData, !is.na(index_before))
```

Now we can check class bias to see if it is more balanced. It is very close to being evenly weighted now.

```
table(trainingData$index_now)
```

```
##
##    0    1
## 5490 5708
```

7.1.3 Logistic Regression Model

Now the model can be run:

```
# Model 1
logit1 <- glm(index_now ~  volatility + beta + price_to_book + index_before, data=trainingData, family=

# Summary of Model 1
summary(logit1)
```

```
##
## Call:
## glm(formula = index_now ~ volatility + beta + price_to_book +
##     index_before, family = binomial(link = "logit"), data = trainingData)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0173  -0.4514   0.1744   0.1891   2.6170
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -1.8759922  0.0592045 -31.687   <2e-16 ***
## volatility    -0.0043563  0.0054966  -0.793    0.428
## beta          -0.3127464  0.0371146  -8.426   <2e-16 ***
## price_to_book  0.0003945  0.0006032   0.654    0.513
## index_before1  6.1554851  0.1126370  54.649   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 15519  on 11197  degrees of freedom
## Residual deviance:  4772  on 11193  degrees of freedom
## AIC: 4782
##
## Number of Fisher Scoring iterations: 6
```

```
# Coefficient Interpretation
## Log Odds
exp(coef(logit1))
```

```
##   (Intercept)    volatility          beta price_to_book index_before1
##     0.1532029     0.9956532     0.7314354     1.0003946   471.2953929
```

```
## Probability
(exp(coef(logit1))) / (1+(exp(coef(logit1))))
```

```
##   (Intercept)    volatility          beta price_to_book index_before1
##     0.1328499     0.4989109     0.4224445     0.5000986     0.9978827
```

Looking at the monthly data is not a true representation of the results, because the index is rebalanced once every six months - not once a month.

7.1.4 Interpretation of Model

The model can be interpreted as:
$\ln[\frac{p}{1-p}]$ = -1.86 - 0.0044 x vol - 0.31 x beta + 0.00039 x price_to_book + 6.16 x index_before
$\frac{p}{1-p}$ = exp(-1.86 - 0.0044 x vol - 0.31 x beta + 0.00039 x price_to_book + 6.16 x index_before)
The coefficients can be interpreted as:
- Volatility: The odds ratio of being added to the index is 0.996 times smaller, given a one unit increase in volatility. This response variable is not statistically significant.
- Beta: The odds ratio of being added to the index is 0.731 times smaller, given a one unit increase in beta. This response variable is statistically significant.
- Price to Book: The odds ratio of being added to the index is 1.0051 times greater, given a one unit increase in price to book ratio. This response variable is not statistically significant.
- Index before: The odds ratio of being added to the index is 410.261 times greater if the stock was in the index 6 months ago. This response variable is statistically significant.

7.1.5 Sanity Check

To take a sample stock to understand the model, we can look at a stock that was not in the USMV index on 12-30-2016, as see how accurate our model would be in predicting the probability of this stock being in the index. We can take AAL (American Airlines), which had a beta of 1.6312867, volatility of 0.8067945, price to book ratio of 4.6943413, and was not in the USMV index 6 months ago. This stock ended up not being in the minimum volatility index on 12-30-2016, so we would expect a probability to be relatively low.
- Odds Ratio:
$\frac{p}{1-p}$ = exp(-3.094 - 0.0032 x 0.8067945 - 0.25 x 1.6312867 + 0.00051 x 4.6943413 + 6.017 x 0)
$\frac{p}{1-p}$ = 0.03013677

- Probability:
p = (exp(-3.094 - 0.0032 x 0.8067945 - 0.25 x 1.6312867 + 0.00051 x 4.6943413 + 6.017 x 0) / (1+exp(-3.094 - 0.0032 x 0.8067945 - 0.25 x 1.6312867 + 0.00051 x 4.6943413 + 6.017 x 0)))
p = 0.02925511

The odds of AAL being in the index on 12-30-2016 is 0.03013677, and this translates to a probability of 2.93%. As expected, already knowing that the stock was not in the index, this low probability seems reasonable.
To further understand the model, we can look at a stock that was was in the USMV index on 12-30-2016, as see how accurate our model would be in predicting the probability of this stock being in the index. We can take AAPL (Apple), which had a beta of 1.0099644, volatility of 0.6118842, price to book ratio of 4.7037726, and it was in the USMV index 6 months ago. This stock ended up being in the minimum volatility index on 12-30-2016, so we would expect a probability to be relatively high
- Odds Ratio:
$\frac{p}{1-p}$ = exp(-3.094 - 0.0032 x 0.6118842 - 0.25 x 1.0099644 + 0.00051 x 4.7037726 + 6.017 x 1)
$\frac{p}{1-p}$ = 14.45369
- Probability:
p = (exp(-3.094 - 0.0032 x 0.6118842 - 0.25 x 1.0099644 + 0.00051 x 4.7037726 + 6.017 x 1) / (1+exp(-3.094 - 0.0032 x 0.6118842 - 0.25 x 1.0099644 + 0.00051 x 4.7037726 + 6.017 x 1)))
p = 0.9352905

The odds of AAL being in the index on 12-30-2016 is 14.45369, and this translates to a probability of 93.53%. As expected, already knowing that the stock was in the index, this high probability seems reasonable.

7.1.6 Model Quality

To test the quality of the model, several tests were done:
Predictive Power
The default cutoff prediction probability score is 0.5 or the ratio of 1's and 0's in the training data. But sometimes, tuning the probability cutoff can improve the accuracy in both the development and validation samples. The InformationValue::optimalCutoff function provides ways to find the optimal cutoff to improve the prediction of 1's, 0's, both 1's and 0's and to reduce the misclassification error. Here, the optimal cut off is 0.74.

```
library(InformationValue)
optCutOff <- optimalCutoff(testData$index_now, predicted)[1]
```

*VIF**_
Like in case of linear regression, we should check for multicollinearity in the model. As seen below, all X variables in the model have VIF well below 4.

```
library(car)
vif(logit1)
```

```
##    volatility          beta price_to_book  index_before
##      1.016126      1.018653      1.000327      1.005611
```

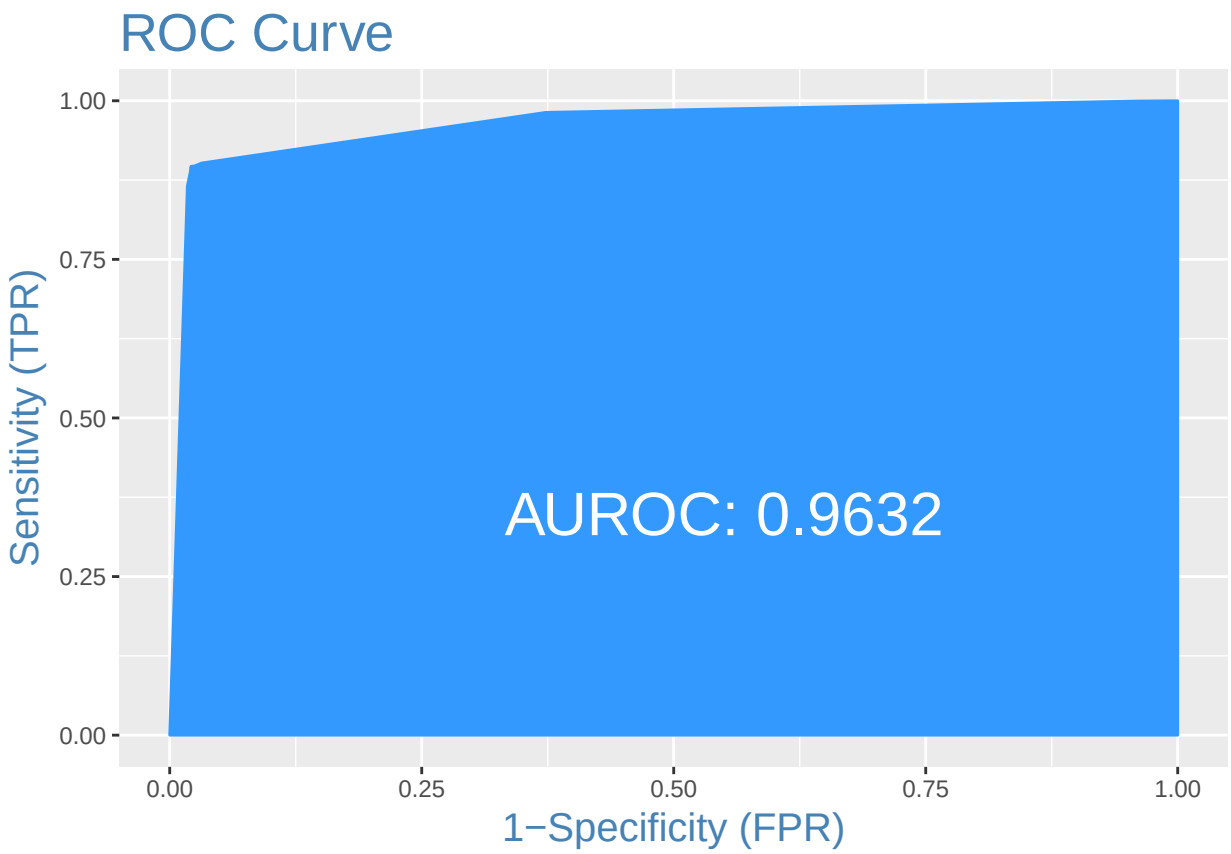
Misclassification Error
Misclassification error is the percentage mismatch of predicted vs actuals, irrespective of 1's or 0's. The lower the misclassification error, the better the model. Here it is 3.1%, which is quite low, and thus good.

```
predicted <- plogis(predict(logit1, testData))
misClassError(testData$index_now, predicted)
```

```
## [1] 0.0309
```

ROC
Receiver Operating Characteristics Curve traces the percentage of true positives accurately predicted by a given logit model as the prediction probability cutoff is lowered from 1 to 0. For a good model, as the cutoff is lowered, it should mark more of actual 1's as positives and lesser of actual 0's as 1's. So for a good model, the curve should rise steeply, indicating that the TPR (Y-Axis) increases faster than the FPR (X-Axis) as the cutoff score decreases. Greater the area under the ROC curve, better the predictive ability of the model. Here, it is 96.3%.

```
plotROC(testData$index_now, predicted)
```



Concordance
Ideally, the model-calculated-probability-scores of all actual Positive's, (aka Ones) should be greater than the model-calculated-probability-scores of ALL the Negatives (aka Zeroes). Such a model is said to be perfectly concordant and a highly reliable one. This phenomenon can be measured by Concordance and Discordance.

In simpler words, of all combinations of 1-0 pairs (actuals), Concordance is the percentage of pairs, whose scores of actual positive's are greater than the scores of actual negative's. For a perfect model, this will be 100%. So, the higher the concordance, the better is the quality of model. This model with a concordance of 97.2% is a good quality model.

```
Concordance(testData$index_now, predicted)
```

```
## $Concordance
## [1] 0.9724405
##
## $Discordance
## [1] 0.02755952
##
## $Tied
```

```
## [1] -4.510281e-17
##
## $Pairs
## [1] 47632140
```

Specificity and Sensitivity
- Sensitivity (or True Positive Rate) is the percentage of 1's (actuals) correctly predicted by the model, while, specificity is the percentage of 0's (actuals) correctly predicted. In this model, it was found to be 89.6%.
- Specificity can also be calculated as 1 - False Positive Rate. In this model, it was found to be 97.9%.

```
sensitivity(testData$index_now, predicted, threshold = optCutOff)
```

```
## [1] 0.8956805
```

```
specificity(testData$index_now, predicted, threshold = optCutOff)
```

```
## [1] 0.9789284
```

Confusion Matrix
In the confusion matrix, the columns are actuals, while rows are predicteds

```
confusionMatrix(testData$index_now, predicted, threshold = optCutOff)
```

```
##       0    1
## 0 19001  256
## 1   409 2198
```

7.2 Model 2: November Model

Since the index is rebalanced twice a year (once in November and once in May), it makes sense to look at a model for each of these individual months. Thus, a subset of the data was taken for November, and the same procedures done at with Model 1.

```
# Subset data for dates from November only
november_final <- filter(monthly_final, date == "2011-11-30" | date == "2012-11-30"| date == "2013-11-29
# Remove NA values from set
november_final <- subset(november_final, !is.na(index_before))
```

7.2.1 Data Cleaning - Checking for Class Bias

Ideally, the proportion of stocks in and out of the USMV index should approximately be the same. Checking this, we can see that this is not the case. However, just around 26% of the data is from stocks that ereurrently in the index, so there is a class bias. As a result, we must sample the observations in approximately equal proportions to get a better model.

```
table(november_final$index_now)
```

```
##
##    0    1
## 2161  750
```

7.2.2 Create Training and Test Samples

One way to address the problem of class bias is to draw the 0's and 1's for the trainingData (development sample) in equal proportions. In doing so, we will put rest of the inputData not included for training into testData (validation sample). As a result, the size of development sample will be smaller that validation, which is okay, because, there are large number of observations.

```
# Create Training Data
input_ones2 <- november_final[which(november_final$index_now == 1), ]  # all 1's
input_zeros2 <- november_final[which(november_final$index_now == 0), ]  # all 0's
set.seed(100)  # for repeatability of samples
input_ones_training_rows2 <- sample(1:nrow(input_ones2), 0.7*nrow(input_ones2))  # 1's for training
input_zeros_training_rows2 <- sample(1:nrow(input_zeros2), 0.7*nrow(input_ones2))  # 0's for training.
training_ones2 <- input_ones2[input_ones_training_rows2, ]
training_zeros2 <- input_zeros2[input_zeros_training_rows2, ]
trainingData2 <- rbind(training_ones2, training_zeros2)  # row bind the 1's and 0's
# Create Test Data
test_ones2 <- input_ones2[-input_ones_training_rows2, ]
test_zeros2 <- input_zeros2[-input_zeros_training_rows2, ]
testData2 <- rbind(test_ones2, test_zeros2)  # row bind the 1's and 0's
```

Now we can check class bias to see if it is more balanced. It is evenly weighted now, with each being represented by 525 observations.

```
table(trainingData2$index_now)
```

```
##
##   0   1
## 525 525
```

7.2.3 Logistic Regression Model

Now the model can be run:

```
# Model 2
logit2 <- glm(index_now ~  volatility + beta + price_to_book + index_before, data=trainingData2, family=

# Summary of Model 2
summary(logit2)
```

```
##
## Call:
## glm(formula = index_now ~ volatility + beta + price_to_book +
##     index_before, family = binomial(link = "logit"), data = trainingData2)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.98863  -0.51069  -0.04623   0.27163   2.05545
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -1.4577528  0.1834842  -7.945 1.94e-15 ***
## volatility     0.0604954  0.0330935   1.828 0.067548 .
## beta          -0.4945911  0.1331544  -3.714 0.000204 ***
## price_to_book -0.0001288  0.0017214  -0.075 0.940368
```

```
## index_before1  5.0806517  0.2776866  18.296  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1455.61  on 1049  degrees of freedom
## Residual deviance:  585.48  on 1045  degrees of freedom
## AIC: 595.48
##
## Number of Fisher Scoring iterations: 6
```

```
# Coefficient Interpretation
## Log Odds
exp(coef(logit2))
```

```
##   (Intercept)    volatility          beta price_to_book index_before1
##     0.2327588     1.0623627     0.6098202     0.9998712   160.8788646
```

```
## Probability
(exp(coef(logit2))) / (1+(exp(coef(logit2))))
```

```
##   (Intercept)    volatility          beta price_to_book index_before1
##     0.1888113     0.5151192     0.3788126     0.4999678     0.9938225
```

Looking at the November model will be helpful for someone looking to predict index rebalancing between June and October.

7.2.4 Interpretation of Model

The model can be interpreted as:
$\ln[\frac{p}{1-p}]$ = -1.46 + 0.061 x vol - 0.49 x beta - 0.00013 x price_to_book + 5.08 x index_before
$\frac{p}{1-p}$ = exp(-1.46 + 0.061 x vol - 0.49 x beta - 0.00013 x price_to_book + 5.08 x index_before)
The coefficients can be interpreted as:
- Volatility: The odds ratio of being added to the index is 1.063 times greater, given a one unit increase in volatility. This response variable is statistically significant, at an alpha level of 0.1.
- Beta: The odds ratio of being added to the index is 0.61 times smaller, given a one unit increase in beta. This response variable is statistically significant.
- Price to Book: The odds ratio of being added to the index is 0.99 times smaller, given a one unit increase in price to book ratio. This response variable is not statistically significant.
- Index before: The odds ratio of being added to the index is 160.88 times greater if the stock was in the index 6 months ago. This response variable is statistically significant.

7.2.5 Sanity Check

Will do later, if useful.

7.2.6 Model Quality

To test the quality of the model, several tests were done:
Predictive Power

The default cutoff prediction probability score is 0.5 or the ratio of 1's and 0's in the training data. But sometimes, tuning the probability cutoff can improve the accuracy in both the development and validation samples. The InformationValue::optimalCutoff function provides ways to find the optimal cutoff to improve the prediction of 1's, 0's, both 1's and 0's and to reduce the misclassification error. Here, the optimal cut off is 0.95.

```
library(InformationValue)
optCutOff2 <- optimalCutoff(testData2$index_now, predicted2)[1]
```

*VIF**_
Like in case of linear regression, we should check for multicollinearity in the model. As seen below, all X variables in the model have VIF well below 4.

```
library(car)
vif(logit2)
```

```
##    volatility          beta price_to_book  index_before
##      1.107794      1.106221      1.000799      1.003120
```

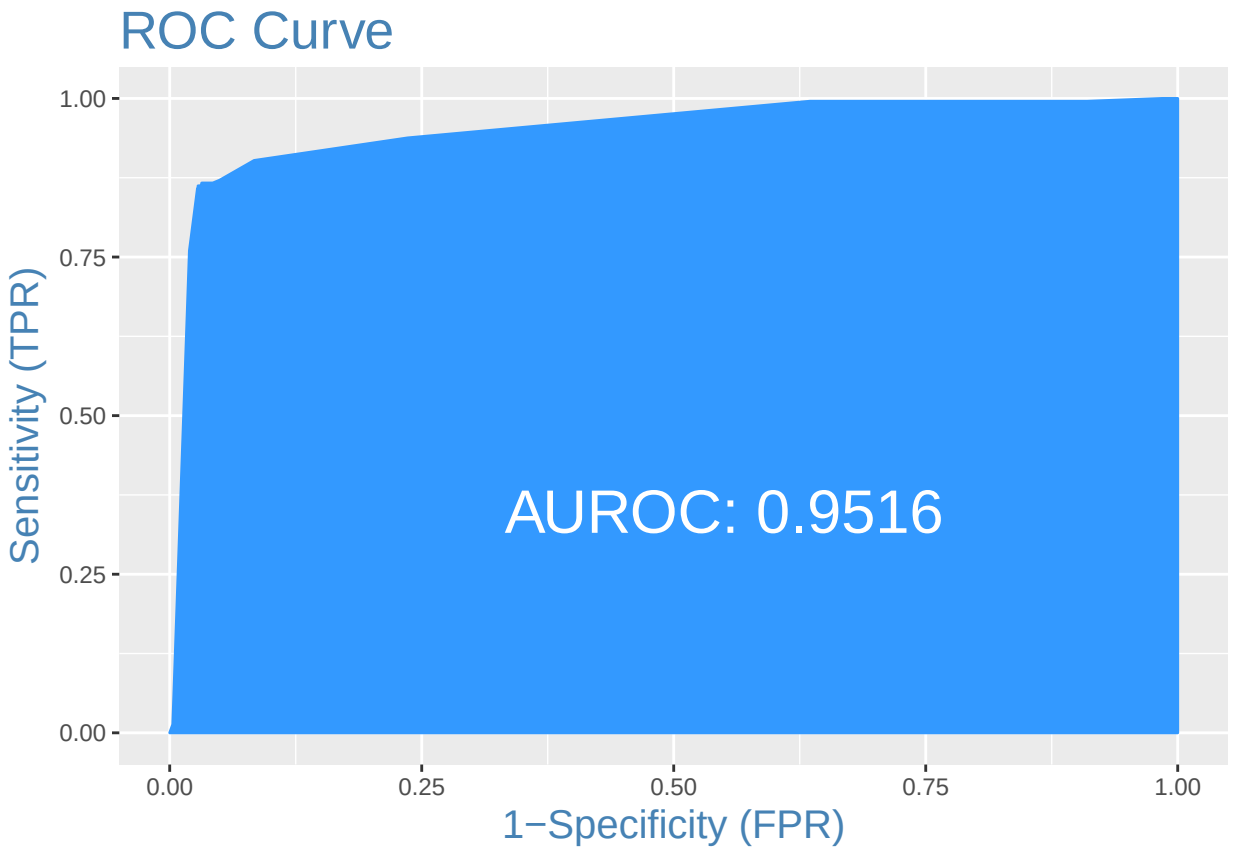
Misclassification Error
Misclassification error is the percentage mismatch of predicted vs actuals, irrespective of 1's or 0's. The lower the misclassification error, the better the model. Here it is 4.4%, which is quite low, and good.

```
predicted2 <- plogis(predict(logit2, testData2))
misClassError(testData2$index_now, predicted2)
```

```
## [1] 0.0435
```

ROC
Receiver Operating Characteristics Curve traces the percentage of true positives accurately predicted by a given logit model as the prediction probability cutoff is lowered from 1 to 0. For a good model, as the cutoff is lowered, it should mark more of actual 1's as positives and lesser of actual 0's as 1's. So for a good model, the curve should rise steeply, indicating that the TPR (Y-Axis) increases faster than the FPR (X-Axis) as the cutoff score decreases. Greater the area under the ROC curve, better the predictive ability of the model. Here, it is 95.2%.

```
plotROC(testData2$index_now, predicted2)
```



Concordance
Ideally, the model-calculated-probability-scores of all actual Positive's, (aka Ones) should be greater than the model-calculated-probability-scores of ALL the Negatives (aka Zeroes). Such a model is said to be perfectly concordant and a highly reliable one. This phenomenon can be measured by Concordance and Discordance.

In simpler words, of all combinations of 1-0 pairs (actuals), Concordance is the percentage of pairs, whose scores of actual positive's are greater than the scores of actual negative's. For a perfect model, this will be 100%. So, the higher the concordance, the better is the quality of model. This model with a concordance of 95.5% is a good quality model.

```
Concordance(testData2$index_now, predicted2)
```

```
## $Concordance
## [1] 0.9558843
##
## $Discordance
## [1] 0.04411573
##
## $Tied
## [1] -6.938894e-18
##
## $Pairs
## [1] 368100
```

Specificity and Sensitivity
- Sensitivity (or True Positive Rate) is the percentage of 1's (actuals) correctly predicted by the model, while, specificity is the percentage of 0's (actuals) correctly predicted. In this model, it was found to be 85.8%.
- Specificity can also be calculated as 1 - False Positive Rate. In this model, it was found to be 97.3%.

```
sensitivity(testData2$index_now, predicted2, threshold = optCutOff2)
```

```
## [1] 0.8577778
```

```
specificity(testData2$index_now, predicted2, threshold = optCutOff2)
```

```
## [1] 0.9731051
```

Confusion Matrix
In the confusion matrix, the columns are actuals, while rows are predicteds

```
confusionMatrix(testData2$index_now, predicted2, threshold = optCutOff2)
```

```
##      0   1
## 0 1592  32
## 1   44 193
```

7.3 Model 3: May Model

Since the index is rebalanced twice a year (once in November and once in May), it makes sense to look at a model for each of these individual months. Thus, a subset of the data was taken for May, and the same procedures done at with Model 1.

```
# Subset data for dates from May only
may_final <- filter(monthly_final, date == "2012-05-31" |  date == "2013-05-31"| date == "2014-05-30" |
# Remove NA values from set
may_final <- subset(may_final, !is.na(index_before))
```

7.3.1 Data Cleaning - Checking for Class Bias

Ideally, the proportion of stocks in and out of the USMV index should approximately be the same. Checking this, we can see that this is not the case. However, just around 24% of the data is from stocks that ereurrently in the index, so there is a class bias. As a result, we must sample the observations in approximately equal proportions to get a better model.

```
table(may_final$index_now)
```

```
##
##    0    1
## 2188  705
```

7.3.2 Create Training and Test Samples

One way to address the problem of class bias is to draw the 0's and 1's for the trainingData (development sample) in equal proportions. In doing so, we will put rest of the inputData not included for training into testData (validation sample). As a result, the size of development sample will be smaller that validation, which is okay, because, there are large number of observations.

```
# Create Training Data
input_ones3 <- may_final[which(may_final$index_now == 1), ]  # all 1's
input_zeros3 <- may_final[which(may_final$index_now == 0), ]  # all 0's
set.seed(100)  # for repeatability of samples
input_ones_training_rows3 <- sample(1:nrow(input_ones3), 0.7*nrow(input_ones3))  # 1's for training
```

```
input_zeros_training_rows3 <- sample(1:nrow(input_zeros3), 0.7*nrow(input_ones3))  # 0's for training.
training_ones3 <- input_ones3[input_ones_training_rows3, ]
training_zeros3 <- input_zeros3[input_zeros_training_rows3, ]
trainingData3 <- rbind(training_ones3, training_zeros3)  # row bind the 1's and 0's
# Create Test Data
test_ones3 <- input_ones3[-input_ones_training_rows3, ]
test_zeros3 <- input_zeros3[-input_zeros_training_rows3, ]
testData3 <- rbind(test_ones3, test_zeros3)  # row bind the 1's and 0's
```

Now we can check class bias to see if it is more balanced. It is evenly weighted now, with each being represented by 493 observations.

```
table(trainingData3$index_now)
```

```
##
##   0   1
## 493 493
```

7.3.3 Logistic Regression Model

Now the model can be run:

```
# Model 3
logit3 <- glm(index_now ~  volatility + beta + price_to_book + index_before, data=trainingData3, family=

# Summary of Model 3
summary(logit3)
```

```
##
## Call:
## glm(formula = index_now ~ volatility + beta + price_to_book +
##     index_before, family = binomial(link = "logit"), data = trainingData3)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -3.04816  -0.38611   0.00159   0.13885   2.34011
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -1.744382   0.249805  -6.983 2.89e-12 ***
## volatility    -0.039892   0.028585  -1.396 0.162842
## beta          -0.642378   0.165440  -3.883 0.000103 ***
## price_to_book -0.012360   0.007939  -1.557 0.119498
## index_before1  7.013927   0.497441  14.100  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1366.89  on 985  degrees of freedom
## Residual deviance:  327.87  on 981  degrees of freedom
## AIC: 337.87
##
## Number of Fisher Scoring iterations: 7
```

```
# Coefficient Interpretation
## Log Odds
exp(coef(logit3))
```

```
##   (Intercept)    volatility          beta price_to_book index_before1
##     0.1747529     0.9608932     0.5260398     0.9877159  1112.0132792
```

```
## Probability
(exp(coef(logit3))) / (1+(exp(coef(logit3))))
```

```
##   (Intercept)    volatility          beta price_to_book index_before1
##     0.1487572     0.4900283     0.3447091     0.4969100     0.9991015
```

Looking at the May model will be helpful for someone looking to predict index rebalancing between December and April.

7.3.4 Interpretation of Model

The model can be interpreted as:
$\ln[\frac{p}{1-p}]$ = -1.74 - 0.04 x vol - 0.64 x beta - 0.012 x price_to_book + 7.014 x index_before
$\frac{p}{1-p}$ = exp(-1.74 - 0.04 x vol - 0.64 x beta - 0.012 x price_to_book + 7.014 x index_before)
The coefficients can be interpreted as:
- Volatility: The odds ratio of being added to the index is 0.96 times smaller, given a one unit increase in volatility. This response variable is not statistically significant.
- Beta: The odds ratio of being added to the index is 0.52 times smaller, given a one unit increase in beta. This response variable is statistically significant.
- Price to Book: The odds ratio of being added to the index is 0.99 times smaller, given a one unit increase in price to book ratio. This response variable is not statistically significant.
- Index before: The odds ratio of being added to the index is 1112.01 times greater if the stock was in the index 6 months ago. This response variable is statistically significant.

7.3.5 Sanity Check

Will do later if useful.

7.3.6 Model Quality

To test the quality of the model, several tests were done:
Predictive Power
The default cutoff prediction probability score is 0.5 or the ratio of 1's and 0's in the training data. But sometimes, tuning the probability cutoff can improve the accuracy in both the development and validation samples. The InformationValue::optimalCutoff function provides ways to find the optimal cutoff to improve the prediction of 1's, 0's, both 1's and 0's and to reduce the misclassification error. Here, the optimal cut off is 0.98.

```
library(InformationValue)
optCutOff3 <- optimalCutoff(testData3$index_now, predicted3)[1]
```

*VIF**_

Like in case of linear regression, we should check for multicollinearity in the model. As seen below, all X variables in the model have VIF well below 4.

```
library(car)
vif(logit3)
```

```
##    volatility          beta price_to_book  index_before
##      1.124595      1.161319      1.006899      1.074031
```

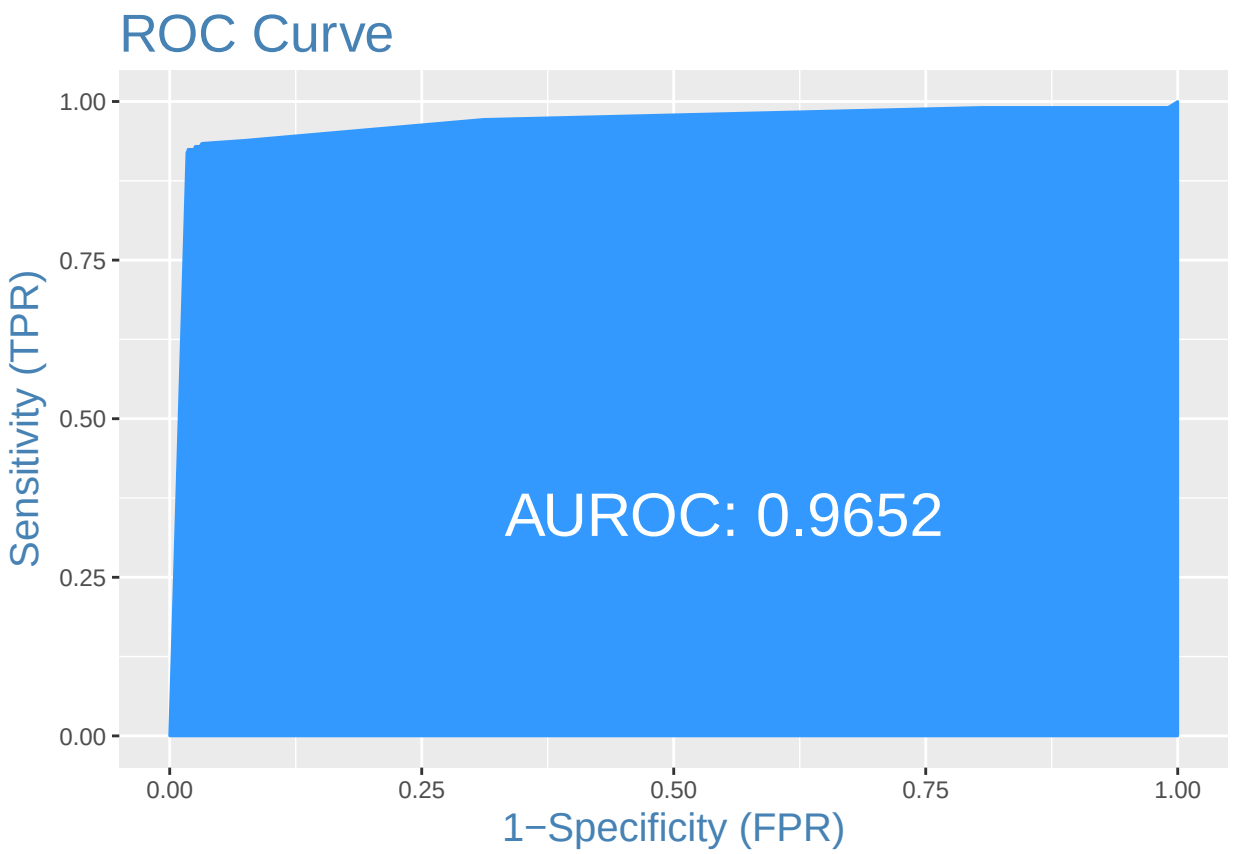
Misclassification Error
Misclassification error is the percentage mismatch of predicted vs actuals, irrespective of 1's or 0's. The lower the misclassification error, the better the model. Here it is 2.6%, which is quite low, and good.

```
predicted3 <- plogis(predict(logit3, testData3))
misClassError(testData3$index_now, predicted3)
```

```
## [1] 0.0262
```

ROC
Receiver Operating Characteristics Curve traces the percentage of true positives accurately predicted by a given logit model as the prediction probability cutoff is lowered from 1 to 0. For a good model, as the cutoff is lowered, it should mark more of actual 1's as positives and lesser of actual 0's as 1's. So for a good model, the curve should rise steeply, indicating that the TPR (Y-Axis) increases faster than the FPR (X-Axis) as the cutoff score decreases. Greater the area under the ROC curve, better the predictive ability of the model. Here, it is 96.5%.

```
plotROC(testData3$index_now, predicted3)
```



Concordance

Ideally, the model-calculated-probability-scores of all actual Positive's, (aka Ones) should be greater than the model-calculated-probability-scores of ALL the Negatives (aka Zeroes). Such a model is said to be perfectly concordant and a highly reliable one. This phenomenon can be measured by Concordance and Discordance.

In simpler words, of all combinations of 1-0 pairs (actuals), Concordance is the percentage of pairs, whose scores of actual positive's are greater than the scores of actual negative's. For a perfect model, this will be 100%. So, the higher the concordance, the better is the quality of model. This model with a concordance of 97.3% is a good quality model.

```
Concordance(testData3$index_now, predicted3)
```

```
## $Concordance
## [1] 0.9732621
##
## $Discordance
## [1] 0.02673791
##
## $Tied
## [1] -3.469447e-18
##
## $Pairs
## [1] 359340
```

Specificity and Sensitivity
- Sensitivity (or True Positive Rate) is the percentage of 1's (actuals) correctly predicted by the model, while, specificity is the percentage of 0's (actuals) correctly predicted. In this model, it was found to be 92.0%.
- Specificity can also be calculated as 1 - False Positive Rate. In this model, it was found to be 98.3%.

```
sensitivity(testData3$index_now, predicted3, threshold = optCutOff3)
```

```
## [1] 0.9198113
```

```
specificity(testData3$index_now, predicted3, threshold = optCutOff3)
```

```
## [1] 0.9828909
```

Confusion Matrix
In the confusion matrix, the columns are actuals, while rows are predicteds

```
confusionMatrix(testData3$index_now, predicted3, threshold = optCutOff3)
```

```
##      0   1
## 0 1666  17
## 1   29 195
```

7.4 Model 4: Total Rebalancing (November & May) Model

Since the index is rebalanced twice a year (once in November and once in May), it makes sense to look at a model for both of these months. Thus, a subset of the data was taken for May and November, by combining the data sets from Model 2 and Model 3.

```
# Subset data for dates from May only
both_final <- rbind(may_final, november_final)
```

7.4.1 Data Cleaning - Checking for Class Bias

Ideally, the proportion of stocks in and out of the USMV index should approximately be the same. Checking this, we can see that this is not the case. However, just around 25% of the data is from stocks that ereurrently in the index, so there is a class bias. As a result, we must sample the observations in approximately equal proportions to get a better model.

```
table(both_final$index_now)
```

```
##
##    0    1
## 4349 1455
```

7.4.2 Create Training and Test Samples

One way to address the problem of class bias is to draw the 0's and 1's for the trainingData (development sample) in equal proportions. In doing so, we will put rest of the inputData not included for training into testData (validation sample). As a result, the size of development sample will be smaller that validation, which is okay, because, there are large number of observations.

```
# Create Training Data
input_ones4 <- both_final[which(both_final$index_now == 1), ]  # all 1's
input_zeros4 <- both_final[which(both_final$index_now == 0), ]  # all 0's
set.seed(100)  # for repeatability of samples
input_ones_training_rows4 <- sample(1:nrow(input_ones4), 0.7*nrow(input_ones4))  # 1's for training
input_zeros_training_rows4 <- sample(1:nrow(input_zeros4), 0.7*nrow(input_ones4))  # 0's for training.
training_ones4 <- input_ones4[input_ones_training_rows4, ]
training_zeros4 <- input_zeros4[input_zeros_training_rows4, ]
trainingData4 <- rbind(training_ones4, training_zeros4)  # row bind the 1's and 0's
# Create Test Data
test_ones4 <- input_ones4[-input_ones_training_rows4, ]
test_zeros4 <- input_zeros4[-input_zeros_training_rows4, ]
testData4 <- rbind(test_ones4, test_zeros4)  # row bind the 1's and 0's
```

Now we can check class bias to see if it is more balanced. It is evenly weighted now, with each being represented by 1018 observations.

```
table(trainingData4$index_now)
```

```
##
##    0    1
## 1018 1018
```

7.4.3 Logistic Regression Model

Now the model can be run:

```
# Model 4
logit4 <- glm(index_now ~  volatility + beta + price_to_book + index_before, data=trainingData4, family=

# Summary of Model 3
summary(logit4)
```

```
##
## Call:
```

```
## glm(formula = index_now ~ volatility + beta + price_to_book +
##     index_before, family = binomial(link = "logit"), data = trainingData4)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.76973  -0.46138   0.00366   0.21208   2.16144
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -1.840874   0.124092 -14.835  < 2e-16 ***
## volatility     0.002945   0.009558   0.308    0.758
## beta          -0.309455   0.071228  -4.345  1.4e-05 ***
## price_to_book -0.001894   0.001547  -1.224    0.221
## index_before1  5.886884   0.241988  24.327  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2822.50  on 2035  degrees of freedom
## Residual deviance:  931.66  on 2031  degrees of freedom
## AIC: 941.66
##
## Number of Fisher Scoring iterations: 6
```

```
# Coefficient Interpretation
## Log Odds
exp(coef(logit4))
```

```
##   (Intercept)    volatility          beta price_to_book index_before1
##     0.1586787     1.0029492     0.7338469     0.9981080   360.2808848
```

```
## Probability
(exp(coef(logit4))) / (1+(exp(coef(logit4))))
```

```
##   (Intercept)    volatility          beta price_to_book index_before1
##     0.1369480     0.5007362     0.4232478     0.4995265     0.9972321
```

Looking at this model will be helpful for someone looking to predict index rebalancing, generally, for both months.

7.4.4 Interpretation of Model

The model can be interpreted as:
$\ln[\frac{p}{1-p}]$ = -1.84 + 0.003 x vol - 0.31 x beta - 0.0019 x price_to_book + 5.89 x index_before
$\frac{p}{1-p}$ = exp(-1.84 + 0.003 x vol - 0.31 x beta - 0.0019 x price_to_book + 5.89 x index_before)
The coefficients can be interpreted as:
- Volatility: The odds ratio of being added to the index is 1.0029 times greater, given a one unit increase in volatility. This response variable is not statistically significant.
- Beta: The odds ratio of being added to the index is 0.73 times smaller, given a one unit increase in beta. This response variable is statistically significant.
- Price to Book: The odds ratio of being added to the index is 0.99 times smaller, given a one unit increase in price to book ratio. This response variable is not statistically significant.
- Index before: The odds ratio of being added to the index is 360.28 times greater if the stock was in the index 6 months ago. This response variable is statistically significant.

7.4.5 Sanity Check

Will do later if useful.

7.4.6 Model Quality

To test the quality of the model, several tests were done:
Predictive Power
The default cutoff prediction probability score is 0.5 or the ratio of 1's and 0's in the training data. But sometimes, tuning the probability cutoff can improve the accuracy in both the development and validation samples. The InformationValue::optimalCutoff function provides ways to find the optimal cutoff to improve the prediction of 1's, 0's, both 1's and 0's and to reduce the misclassification error. Here, the optimal cut off is 0.77.

```
library(InformationValue)
optCutOff4 <- optimalCutoff(testData4$index_now, predicted4)[1]
```

*VIF**_
Like in case of linear regression, we should check for multicollinearity in the model. As seen below, all X variables in the model have VIF well below 4.

```
library(car)
vif(logit4)
```

```
##    volatility          beta price_to_book  index_before
##      1.022471      1.033581      1.007330      1.022806
```

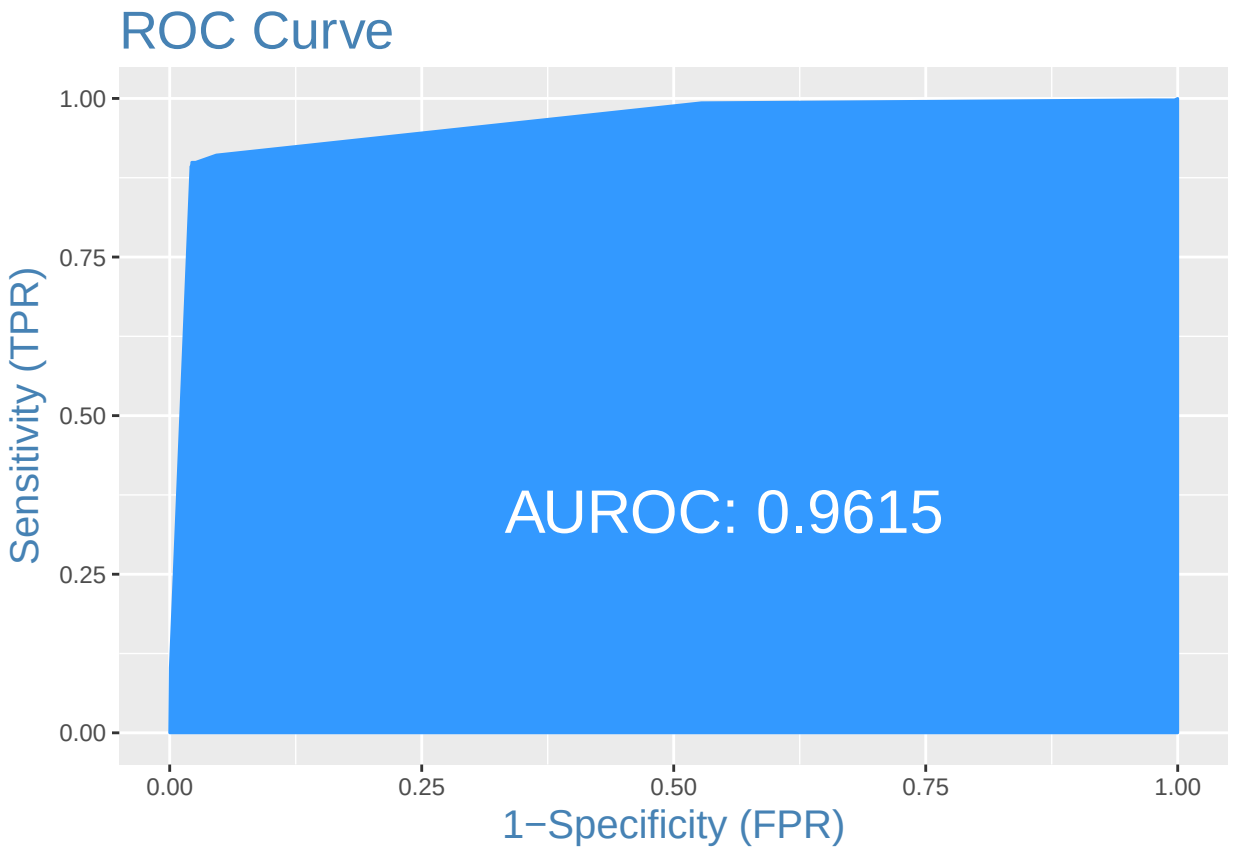
Misclassification Error
Misclassification error is the percentage mismatch of predicted vs actuals, irrespective of 1's or 0's. The lower the misclassification error, the better the model. Here it is 3.2%, which is quite low, and good.

```
predicted4 <- plogis(predict(logit4, testData4))
misClassError(testData4$index_now, predicted4)
```

```
## [1] 0.0321
```

ROC
Receiver Operating Characteristics Curve traces the percentage of true positives accurately predicted by a given logit model as the prediction probability cutoff is lowered from 1 to 0. For a good model, as the cutoff is lowered, it should mark more of actual 1's as positives and lesser of actual 0's as 1's. So for a good model, the curve should rise steeply, indicating that the TPR (Y-Axis) increases faster than the FPR (X-Axis) as the cutoff score decreases. Greater the area under the ROC curve, better the predictive ability of the model. Here, it is 96.2%.

```
plotROC(testData4$index_now, predicted4)
```



Concordance
Ideally, the model-calculated-probability-scores of all actual Positive's, (aka Ones) should be greater than the model-calculated-probability-scores of ALL the Negatives (aka Zeroes). Such a model is said to be perfectly concordant and a highly reliable one. This phenomenon can be measured by Concordance and Discordance.

In simpler words, of all combinations of 1-0 pairs (actuals), Concordance is the percentage of pairs, whose scores of actual positive's are greater than the scores of actual negative's. For a perfect model, this will be 100%. So, the higher the concordance, the better is the quality of model. This model with a concordance of 97.1% is a good quality model.

```
Concordance(testData4$index_now, predicted4)
```

```
## $Concordance
## [1] 0.9714409
##
## $Discordance
## [1] 0.02855912
##
## $Tied
## [1] -4.857226e-17
##
## $Pairs
## [1] 1455647
```

Specificity and Sensitivity
- Sensitivity (or True Positive Rate) is the percentage of 1's (actuals) correctly predicted by the model, while, specificity is the percentage of 0's (actuals) correctly predicted. In this model, it was found to be 89.9%.
- Specificity can also be calculated as 1 - False Positive Rate. In this model, it was found to be 97.8%.

```
sensitivity(testData4$index_now, predicted4, threshold = optCutOff4)
```

```
## [1] 0.8993135
```

```
specificity(testData4$index_now, predicted4, threshold = optCutOff4)
```

```
## [1] 0.9780847
```

Confusion Matrix
In the confusion matrix, the columns are actuals, while rows are predicteds

```
confusionMatrix(testData4$index_now, predicted4, threshold = optCutOff4)
```

```
##      0   1
## 0 3258  44
## 1   73 393
```

Chapter 8

Conclusion

All in all, the 4 models were comprable in terms of statistical resiliency and predictive power. Each model may have a different usage, based of each model's strengths and weaknesses, and what goals the investor has in mind for the model. For example, an investor looking to capture an arbitrage opportunity in November, might be best suited in looking at the November specific model. Someone looking for abritrage opportunities throughout the year during both rebalances might look at the combined May and November model.

8.1 Side by Side Model Comparison

```
## Warning: 'rbind_list' is deprecated.
## Use 'bind_rows()' instead.
## See help("Deprecated")
```

```
## `mutate_each()` is deprecated.
## Use `mutate_all()`, `mutate_at()` or `mutate_if()` instead.
## To map `funs` over a selection of variables, use `mutate_at()`
```

```
## # A tibble: 10 x 6
##             term       key   `1`   `2`   `3`   `4`
##  *         <chr>     <chr> <dbl> <dbl> <dbl> <dbl>
##  1   (Intercept)  estimate -1.88 -1.46 -1.74 -1.84
##  2   (Intercept) std.error  0.06  0.18  0.25  0.12
##  3          beta  estimate -0.31 -0.49 -0.64 -0.31
##  4          beta std.error  0.04  0.13  0.17  0.07
##  5 index_before1  estimate  6.16  5.08  7.01  5.89
##  6 index_before1 std.error  0.11  0.28  0.50  0.24
##  7 price_to_book  estimate  0.00  0.00 -0.01  0.00
##  8 price_to_book std.error  0.00  0.00  0.01  0.00
##  9    volatility  estimate  0.00  0.06 -0.04  0.00
## 10    volatility std.error  0.01  0.03  0.03  0.01
```

As seen, each model gave out pretty similar coefficient values for the various response variables. Beta ranged between -0.31 and -0.64, index_before ranged from 5.08 to 7.01, price to book ranged from 0.00 to -0.01, and volatility ranged between -0.04 and 0.06.

Chapter 9

Discussion

After comparing all of the models, it makes sense to discuss the applications of these various models to the real world, and to finance.

9.1 Understanding of Relationships

Through these models, we can get a better understanding of the relationships between the predictor variables, and whether or not the stock is in the Min Vol index. In general, each model suggested an increase in beta will reduce the likelihood of a stock being in the min vol index, with all else held constant. This makes sense, as beta is one measure of risk and volatility. Moreover, it is a widely used metric in finance, so it is not surpising that it is a statistically significant variable. Moreover, the most significant variable was whether or not the stock was in the index before. This makes a lot of sense, as a stock currently in the index presumably has many min vol characteristics from before, that must be signficantly altered if it were to be removed. Moreover, stocks that were in the index previously were many times more likely to be in the index currently, than stocks that had previously not been in the index. This variable was also statistically significant. Surpisingly, volatility was not statistically signficant, though the index itself is called the "Minimum Volatility" Index. Moreover, price to book was also an insignficant variable, which does make sense. Each model was able to quantify these relationships, and help us better understand what

9.2 Arbitrage

Each model was able to take various attributes of a stock, and calculate a probability for it currently being in the index. Using the optimal cutoffs, we were able to get a sense of the probability value that would be significant in determining when a stock would be in or out of the index. For example, at a cutoff of 0.9, this would tell us that we could reasonably expect stocks with a probability of over 90% to be in the index, and stocks with less than a 90% probability to not be in the index. With this information, there are many different arbitrage opportunities. One could long stocks currently not in the index that have a probability greater than the optimal cutoff for that model. This would represent the stocks with the greatest chance of being added to the index, that are currently not in the index. If correct, prior studies would suggest that the stock price would consequently increase from this happening. Moreover, one could short stocks that are currently in the index, that have a probability value less than the cutoff. This could lead to an arbritrage opportunity if the stock is removed from the index, as one is short it.

Chapter 10

Bibliography

Vliet, P. V., & Koning, J. D. (2017). High returns from low risk: a remarkable stock market paradox.

Asness, C. S., Frazzini, A., Gormsen, N. J., & Pedersen, L. H. (2016). Betting Against Correlation: Testing Theories of the Low-Risk Effect.

Huij, J., & Kyosev, G. (2016). Price Response to Factor Index Additions and Deletions.

Baker, M., Bradley, B., & Wurgler, J. (2011). Benchmarks as limits to arbitrage: Understanding the low-volatility anomaly. Financial Analysts Journal, 67(1), 40-54.

Frazzini, A., & Pedersen, L. H. (2014). Betting against beta. Journal of Financial Economics, 111(1), 1-25.

Baker, M., Bradley, B., & Taliaferro, R. (2014). The low-risk anomaly: A decomposition into micro and macro effects. Financial Analysts Journal, 70(2), 43-58.