BEATING THE STOCK MARKET USING SEASONAL INVESTING

Project Proposal for SYS 6014 at University of Virginia (Spring 2020)

Ryan Krechel - Updated March 25, 2020

INTRODUCTION

For this project, I intend to create a model for seasonal investing to find a way to beat a buy-and-hold investment strategy. In theory, there are certain market trends that repeat at the same time each year. For instance, the beginning of December tends to be a time when investors sell off stocks to cover the cost of taxes which generally causes a decrease in the stock market. If an investor is aware of this trend, he or she should choose to invest in less aggressive accounts (perhaps bonds) at this time of the year to protect assets and not lose any money. My goal is to find other trends based on historical data and identify a method for strategically picking funds to invest in over a calendar year to maximize returns and beat traditional buy-and-hold approaches

BUILDING THE STRATEGY

This project is built around a day-trading approach. In this way, a strategy is defined as instructions to invest in a particular ETF as a day trade for each trading day over an entire calendar year. The decision maker (a random investor interested in seasonal investing) technically uses this analysis to make a decision every day, but each decision is made from historical data for that particular calendar day. In this way, the decision analysis tool gains new information on a year-to-year basis and not a day-to-day basis. Therefore, the decision maker can essentially define a strategy on December 31 for the following calendar year and stick to it until the following December 31.

DATA SELECTION AND CHOOSING FUNDS

For the analysis, I have chosen 7 Vanguard exchange-traded funds (ETF) that cover differing sectors of the market and differing investment strategies. They each have different inception dates, with the newest being created in 2008. Therefore, to create a level playing field, data was only collected from 2008 to 2019.

The following ETFs were chosen and used for this analysis:

TABLE 1: SUMMARY OF ETFs USED IN THIS ANALYSIS

	ETF Abbreviation	Market Sector	Description	Inception Date	Average Return (2008-2019)
1	VTI	Large Cap Blend	Stock Market ETF	2001	-0.86
2	VXF	Mid Cap Blend	Extended Market ETF	2001	-1.53
3	VBK	Small Cap	Small-Cap Growth ETF	2004	-3.96
4	VEA	International	FTSE Developed Markets ETF	2007	-3.48
5	VWO	Emerging Markets	FTSE Emerging Markets ETF	2005	-1.62
6	VPU	Utilities	Utitlities ETF	2004	-0.50
7	BLV	Bonds	Long-Term Investment Bond	2007	-3.44

Clearly, the average for each fund is actually quite poor. This is largely due to a brutal 2008 recession that drives the averages down, but this fact actually presents the perfect opportunity to prove if this optimization method works since investing in any particular fund does not guarantee a good return based on averages.

DEFINING SUCCESS

In general, the goal is to beat a traditional buy-and-hold strategy, which is defined as choosing to invest in any one the ETFs used in this analysis for the entire calendar year. The following table presents the individual performance of each fund for a common range of years, 2008-2019.

TABLE 2: YEARLY INDIVIDUAL RETURNS SUMMARY

Year	VTI	VXF	VBK	VEA	VWO	VPU	BLV	Max	Average
2008	-33.08	-37.91	-48.57	-88.53	-28.62	-7.25	-8.16	-7.25	-36.02
2009	16.29	23.81	19.07	8.69	18.88	-7.40	-31.82	23.81	6.79
2010	4.64	4.33	-2.05	5.19	5.56	-19.76	-11.27	5.56	-1.91
2011	-4.03	-11.52	-16.52	-4.17	-17.34	-19.52	28.71	28.71	-6.34
2012	6.88	6.85	6.38	15.97	10.42	-3.99	-0.43	15.97	6.00
2013	9.55	6.16	3.43	13.28	-8.66	5.96	-8.58	13.28	3.02
2014	-4.10	-4.93	-7.15	-2.34	-8.04	10.89	9.62	10.89	-0.87
2015	-11.24	-3.29	-8.22	-6.81	-3.27	-5.17	-13.22	-3.27	-7.32
2016	9.00	14.41	9.04	7.97	9.68	21.60	-6.70	21.60	9.29
2017	2.31	-3.16	1.55	12.69	3.93	8.46	5.44	12.69	4.46
2018	-18.71	-21.55	-16.38	-16.04	-11.38	-2.40	-8.33	-2.40	-13.54
2019	12.20	8.48	11.91	12.30	9.40	12.62	3.41	12.64	10.05

Since new data is collected very slowly (it takes an entire calendar year to add a full data set addition), success will be determined by taking a snapshot in time sometime in the past. For instance, data from 2008 to 2018 can be used to create a strategy that the decision maker could have chosen on December 31, 2018 to use during 2019. Since that data has already been collected, it can be used to test if the strategy defined at that time would have worked. The goal is to prove that this method works before actually investing an real money.

It is difficult to define success for a single year, because returns are subject to so many random influences from the world and stock market in general. However, when multiple years are strung together, an investor can get a clearer picture of how sucessful his or her decisions are.

For the sake of this paper, success of the method will be evaluated by comparing a string of yearly returns together and solving for a final value. The following table combines the individual yearly returns of 2013-2019 together into a total net percentage. For instance, if an investor purchased $1000 worth of shares of VTI on December 31, 2012 and looked at their account on December 31, 2019, he or she would have $948, for a total return of -5.15%. The following table represents the returns fo each fund for the same stretch of data.

TABLE 3: AGGREGATE RETURNS FROM 2013-2019

VTI	VXF	VBK	VEA	VWO	VPU	BLV
-5.15%	-7.97%	-8.67%	18.27%	-10.21%	61.53%	-18.90%

To be considered a success, the method described in this report will define strategies for 2013-2019 by adding a new data set each year and recalculating. If the total return is greater than the numbers in the table above, then the methods improves an individual's ability to invest.

ACTION SET

The decision-maker's action set *A* is a complete itemization of all the options available to the decision-maker, from which the decision-maker selects a unique choice. There are 8 available each day, which creates an incredible amount of different strategies. In fact, there are 8^252 different strategies! The options available at each decision mode in this model are shown in the following table.

TABLE 4: ACTION SET

Action	Action Description
1	Invest in VTI
2	Invest in VXF
3	Invest in VBK
4	Invest in VEA
5	Invest in VWO
6	Invest in VPU
7	Invest in BLV
8	Do Not Invest

DATA GENERATING PROCESS

Data will be collected using Yahoo's share price history tool from 2008-2019. Preprocessing will be done on the "open" and "close" prices to convert the share prices into daily losses or gains. This will be produced and uploaded into Matlab code with the corresponding dates to use in the analysis.

Once the raw data is uploaded into Matlab, it will need to be sorted by date and by gain/loss. The code will be written to assess January 2nd as a specific decision node with historical data from all other January 2nd data points. Within this day's worth of data, there will be a set of positive gains and a set of negative losses. In this way, I will be able to track success of each historical success of each ETF for any given day, both in likelihood of positive return and average value of positive/negative return. The code will repeat this process for each calendar day to develop a "strategy" for the year.

PRIOR BELIEFS

Average yearly returns are presented in Table 1, which give an approximation for what an investor can expect from investing yearly in the individual funds. However, the decision maker in this analysis is looking to invest long-term, and it is best to compare to the data for longer stretches of data in Table 3.

For this analysis, the decision maker has prior beliefs that the individual funds average a negative return (based on data from 2008-2019), but also have a aggregate returns shown in Table 3 (for 2013-2019). Again, the goal is to do better than the data in Table 3; to do better than the prior beliefs.

PAYOFFS

Payoffs are defined as the expected return of a particular action. In investing, payoffs are expressed in terms of return on investment, or percentages gained/lost.

The payoffs for this analysis will change based on the data set being observed and will be calculated directly using the historical data. For instance, in a situation where data from 2008 to 2018 is used, up to 11 different data points could exist for each calendar day (depending on where weekends fall because weekends are non-trading days with no data). For each calendar day, the data points will be separated by positive, negative, and net-zero return. The positive days will be averaged to find the positive payoff, and the negative days will be averaged to find the negative payoff.

UTILITY FUNCTION AND DECISION MAKING PROCESS

The utility function is generally defined as follows:

$$EU(ETF) = (AvgPosReturn) \cdot (PercPosReturn) + (AvgNegReturn) \cdot (PercNegReturn)$$

For any given calendar day, this utility function will be calculated for each ETF. The model will make a decision simply based on the highest utility function available for each calendar day.

Here is an example from data collected from 2008 to 2018 for January 2, the first trading day of the year.

TABLE 5: SAMPLE DATA USED TO CALCULATE UTILITY FUNCTION

Decision	Avg Pos Return	Percent Pos Return	Avg Neg Return	Percent Neg Return	Estimated Utility
Invest in VTI	1.32	67%	-0.62	33%	0.6752
Invest in VXF	0.95	67%	-0.63	33%	0.4215
Invest in VBK	1.37	50%	-0.64	50%	0.3652
Invest in VEA	0.87	50%	-0.64	33%	0.2235
Invest in VWO	1.48	67%	-1.49	33%	0.4875
Invest in VPU	0.92	50%	-1.18	50%	-0.1283
Invest in BLV	0.31	67%	-1.52	33%	-0.2965

Therefore, based on the analysis of the data available for January 2, 2008-2018, the optimal ETF to invest in on January 2, 2019 is VTI.

It is important to note that the chance of a positive return and the chance of a negative return does not necessarily have to add up to 1. There is a chance of an ETF returning no net gain or loss, and it is important to include this annomally in the code if it shows up in the data. In fact, this was relavent for the data set above because the change of positive and negative return of investing in VEA was not equal to 1, so there was a day in the data set that had no net gain or loss.

DATA ANALYSIS WITH MATLAB