ORIE 4300: Optimization Modeling

Project 1

130-30 Fund vs **Long Only Investment Strategy**

Can Technical Analysis help?

Chris Nieves (cdn25)

Kunal Rajani (kr386)

Roni Yosofov (ry96)

March 16th 2012

Contents

1. Introduction

In this project we chose to concentrate on the performance of two types of investment strategies in the Markowitz framework. The first investment strategy is disallowing the option of short selling, which means that the investor can go long only. The second strategy is 130-30 Fund investment strategy that leverages the investment portfolio by 30%, with the usage of sell shoring. 130-30 Fund is used by many Hedge Funds and is considered to be a widely used strategy, so our goal was to see if it is also true in the Markowitz framework. To add extra analysis, we used a Technical Indicator called Bloomberg Trender. The indicator gives us signals of buy/sell every time we need to rebalance. We ran the investment strategies with and without the Technical Indicator and analyzed the performance of the four investments.

We gathered data (e.g. close, high and low prices) from different 22 ETFs provided by Bloomberg Terminal. The following is the list of the ETFs:

Ticker	Name	Inception
XLB	Materials Select Sector SPDR Fund	12/16/1998
XLE	Energy Select Sector SPDR Fund	12/16/1998
XLF	Financial Select Sector SPDR Fund	12/16/1998
XLI	Industrial Select Sector SPDR Fund	12/16/1998
XLK	Technology Select Sector SPDR Fund	12/16/1998
XLP	Consumer Staples Select Sector SPDR Fund	12/16/1998
XLU	Utilities Select Sector SPDR Fund	12/16/1998
XLV	Health Care Select Sector SPDR Fund	12/16/1998
XLY	Health Care Select Sector SPDR Fund	12/16/1998
IJR	iShares S&P SmallCap 600 Index	5/22/2000
VXX	iPath S&P 500 VIX Short Term Futures	1/29/2009
EWZ	iShares MSCI Brazil Index Fund	7/10/2000
RSX	Market Vectors TR Russia ETF	4/24/2007
EPI	WisdomTree India Earnings Fund	2/22/2008
FXI	iShares FTSE China 25 Index Fund	10/5/2004
ILF	iShares Latin America 40 Index	10/25/2001
UCO	ProShares Ultra DJ-AIG Crude Oil	11/24/2008
GLD	SPDR Gold Trust	11/18/2004
SLV	iShares Silver Trust	4/21/2006
UNG	United States Natural Gas Fund	4/18/2007
TLT	iShares Barclays 20 Year Treasury Bond Fund	7/22/2002
SHY	iShares Barclays 1-3 Year Treasury Bond Fund	7/22/2002

Table 1.1 ETF's chosen for Analysis

We chose these ETFs as they are widely traded and they cover many different sectors. This allowed us to gauge the wellness of the market, overall. Since some of the ETFs we chose

started trading only in the past four or five years, the prices we have in the data base is from 1/2/2009 to 1/30/2012.

Next, in 2009 there was a lot of volatility and sharp down-slop in the market; therefore we chose our present day to be 3/12/2010 (300) which provides more stable future periods to run our analysis. The following graph shows the huge drop of some of the ETFs had in 2009.

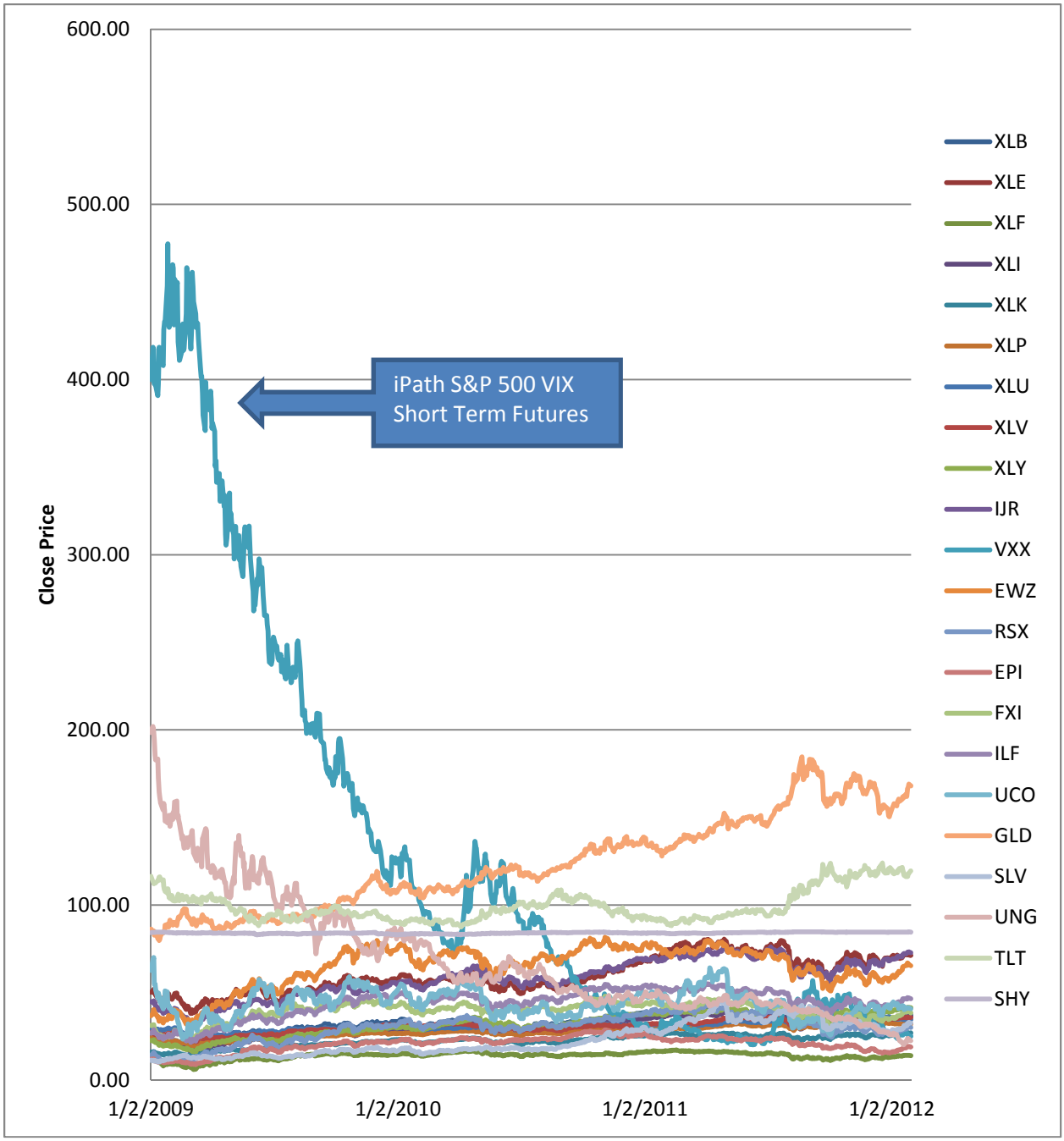


Figure 1.1 Trends in ETF prices over the past 3 years

Lastly, 130-30 Fund is used mostly by active investment managers, therefore, we will need to rebalance our positions every month (every 20 trading days) with the hope to generate positive alpha (excess return due to active management). For further analysis we may want to try and change our rebalancing periods to see if more frequent rebalancing will help us perform better. And, since we want to diversify our portfolios somehow, we won't accept more than 30% of total investment for long ETFs or -20% of total investment for short ETFs. Eventually, we will analyze the performance of the four investments by the growth of the wealth (initial well was chosen to be $100,000) and the standard deviation of the wealth change through the rebalancing period. Next, we will explain shorty how was our Markowitz constructed, how the Technical Indicator was implemented, how some sensitivity analysis impacted our results, and lastly which portfolio performed best during our default choice of parameters.

1.1 Model

Min: $delta*X^{T}VX + (-\mu^{T}X + trxn^{T}*(X^{+} - X^{-}))$

Such that:
$lb <= X <= U$
$\sum X = 1$
$\sum X^{long} = long$
$\sum (Tech\_ind\_long)^{T}*X^{long} = long$
$\sum X^{short} = short$
$\sum (Tech\_ind\_short)^{T}*X^{short} = short$

The above model can be translated into words as:
Minimize the variance of the portfolio and the transaction costs while maximizing the returns.
Constrained with the weights being bounded and the total long and short positions adding up to the specified parameters.

We have the following parameters in the model**:**

- Current date – Contains the date when we form the portfolio
- Look back period – contains the no. of months we consider to obtain historical prices for the mean variance analysis
- Delta – the risk aversion of investors that multiplies the variance factor in the Markowitz analysis
- Transaction costs – the costs associated with each transaction as a ratio of the total wealth
- Long/Short – specify the combined long/short position over all ETFs

Technical Indicator – This is a trender that is obtained by an algorithm that is used widely in the industry to identify trends in the stock prices. The trender recommends stocks to be bought or sold based on the trends.

1.2 Trender Overview

The goal of technical analysis is to predict the future direction of prices based off of historical market

data. Even though the Efficient Market Hypothesis alleges that this type of analysis is fruitless, many investors still utilize this type of analysis in their decision-making. We will be examining the performance of one such technical indicator, based off the Trender in Bloomberg. This indicator is designed to predict whether the stock is currently in an uptrend or a downtrend.

Price History:

The Trender works best when given daily price information. Besides the daily closing prices $p_C(t)$, our calculations will also require the historical daily high and low prices: $p_H(t)$, $p_L(t)$.

Definitions:

Midpoint: $MP(t) = \frac{p_H(t) + p_L(t)}{2}$

True Range: $TR(t) = max\{\, p_H(t) - p_L(t), |p_H(t) - p_C(t-1)|, |p_L(t) - p_C(t-1)| \,\}$

14 Day Exponential Moving Average: $EMAvg(X_T) = \sum_{j=T-14}^{T-1} w_t * X_t \quad with\ w_t = \frac{\alpha^{T-t}}{\sum_{j=0}^{14} \alpha^{T-14+j} * X_j}$

Standard deviation of the previous 14 values of the EMAvg(TR): $\sigma_{EMAvg(TR)}$

Support and Resistance Lines:

$$Trender\_UP = EMAvg(MP) + \frac{EMAvg(TR)}{2} + K * \sigma_{EMAvg(TR)}$$

$$Trender\_DOWN = EMAvg(MP) - \frac{EMAvg(TR)}{2} - K * \sigma_{EMAvg(TR)}$$

If the stock currently has a positive signal, it will continue to have a positive signal until its price closes below the current Trender_DOWN line value. In the example below with EWZ, this occurs on 6/15/2009. At this time, a new negative signal is sent out and the Trender_UP line is now the resistance line. Then we see that at 7/15/2009, EWZ closes above the Trender_UP line, and a new positive signal is generated. As long as the resistance line is not crossed, the Trender simply maintains its previous signal.

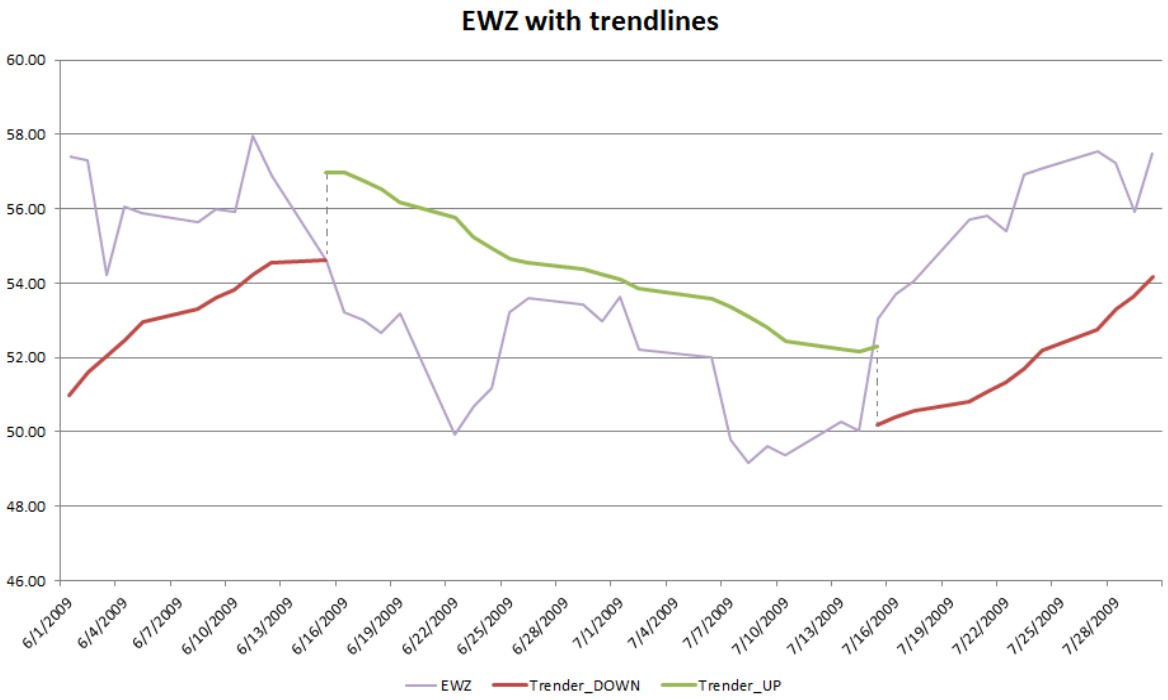


Figure 1.2 Trender

2. Sensitivity Analysis

Before we start the analysis, we would like to list all of default parameters that we assumed before running our code:

- **trxn_cost is set to 1%** of the price of the ETF (weighted). This is the transaction cost we are charged for rebalancing our current portfolio (we start with 0 weight in all ETFs).
- **Lookback_period is 60 days**. This is the period we look back in order to work with the 'stats' function to calculate mean and covariance of the returns.
- **current is 300 (3/12/2010)**. This is our present time. We gathered data from 1 to 774 and 300 is a period that is past the bear and highly volatile market.
- **delta is equal to 1**. Delta indicates risk aversion and this parameter is the more risk averse the investor.
- **upper** is 0.3 and it indicates what the maximal weight we can put in one ETF.
- **lower** is -0.2 and it indicated what the minimal weight we can put in one ETF.
- **Alpha** is set to 0.96, which indicates how much the weights in the Exponential Moving Average are discounted each day you go back. So an observation at t-1 has a weight that is 96% of the weight of time t's observation.
- **Sensitivity** is set to be 1 by default, which is the maximum level of sensitivity. Increasing this parameter (1<X<10) reduces how sensitive the Trender is.

To do the sensitivity analysis, we changed some of these parameters and tried to see how they influenced our total performance.

2.1 Transaction Costs

In this sensitivity analysis we will change the transaction cost and see how the rebalancing frequency is changed through time.

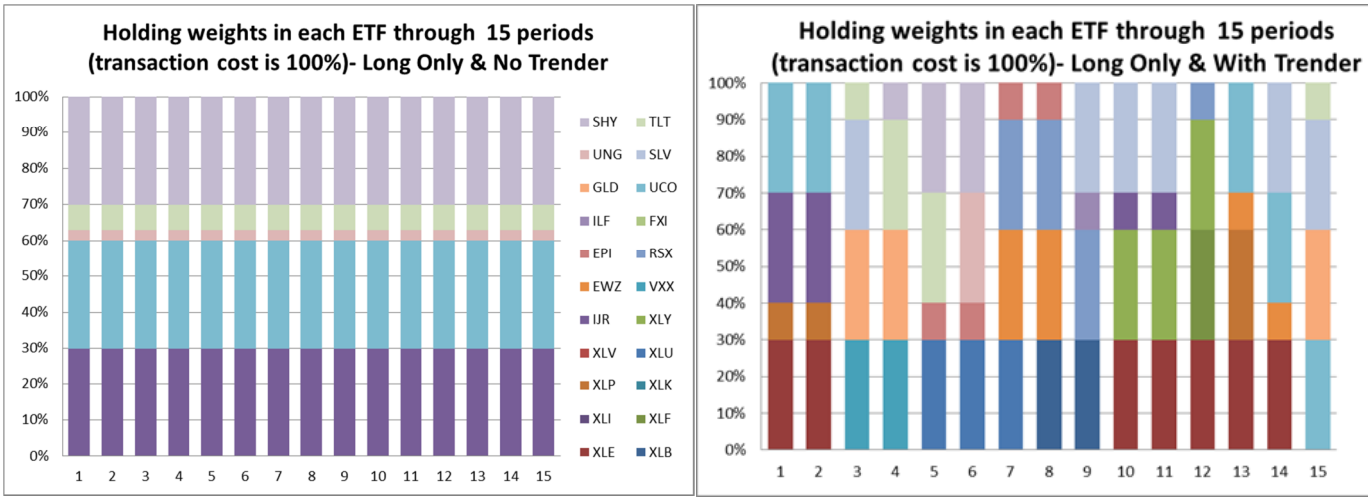


Figure 2.1 Rebalancing for transaction cost = 100% of change in portfolio

Our default transaction cost is 1% of the total size of the rebalancing and we will take an extreme number of 100% transaction cost to analyze extreme changes. We decided to look only into the Long Only investment strategies to simplify the complexity.

Here we can clearly see that no rebalancing is done throughout the 15 holding periods, and this is independent of anything. It makes sense, though, because we can long and short any of the ETFs and due to the very high transaction costs, we are reluctant to do so.

Here we are using the Technical Indicator, so it has the priority to the Markowitz in telling which stocks we can buy and which we can sell. If the signals are constant throughout the periods, we won't rebalance that specific ETF. On the other hand, if the Trender changes signals on any specific rebalancing period, we will change our position and then the Markowitz function will decide how to rebalance.

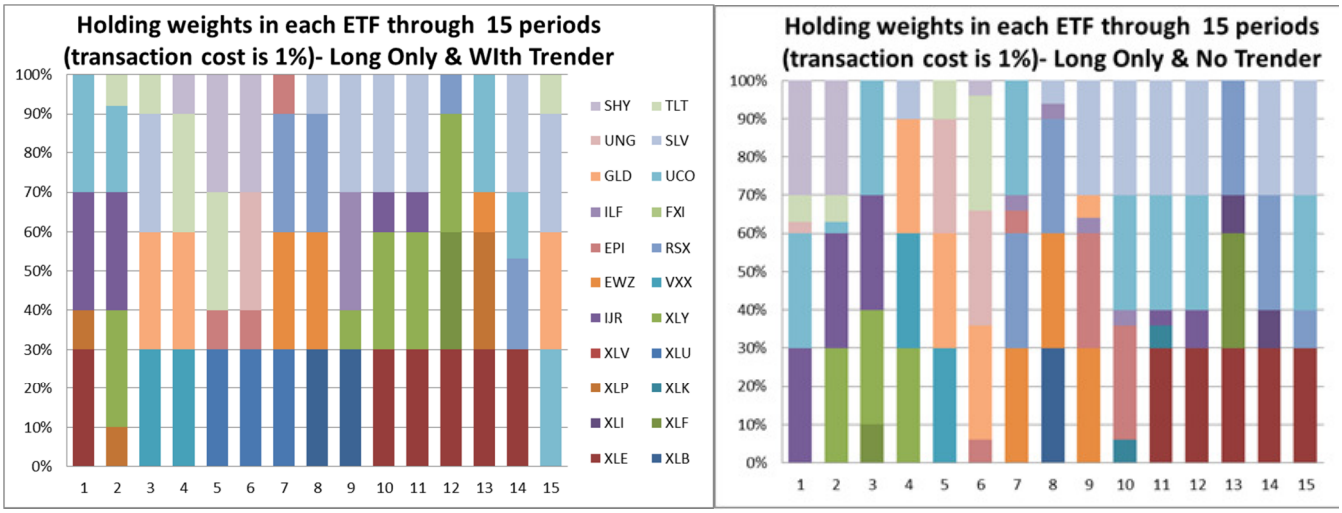


Figure 2.2 Rebalancing for Transaction Cost = 1% of change in portfolio

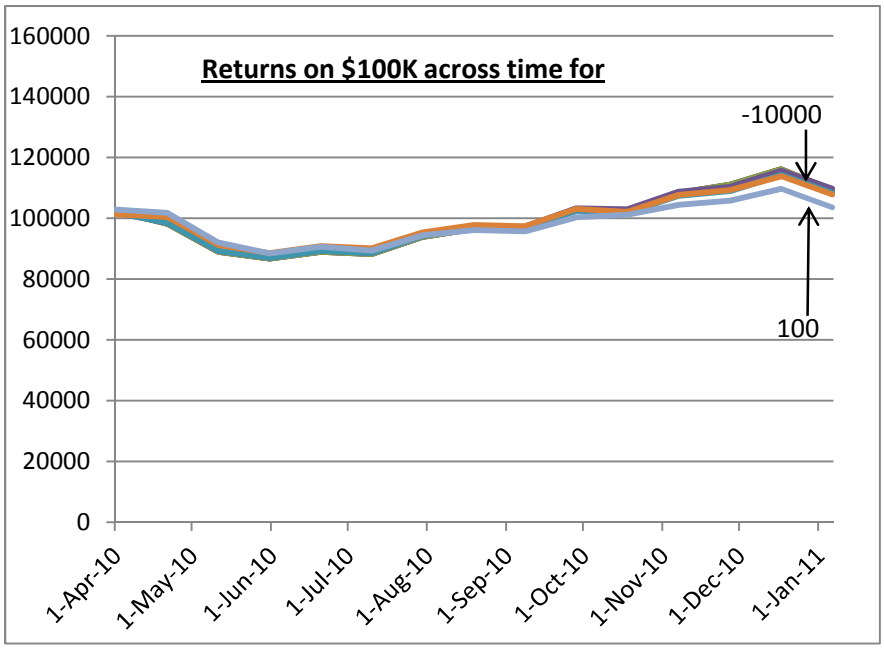
When the transaction cost is only 1%, we can see that there is attempt to stay constant in the first graph when the Trender doesn't change the signal. However, in the second graph we can see that rebalancing is pretty heavy at times.

The conclusion is pretty obvious that higher transaction costs, will force the investors to be less active and eager in rebalancing, unless the Trender says to rebalance.

2.2 Delta

We ran the Markowitz optimization on our data by keeping all parameters constant except delta to study its impact on the outcomes. We restricted ourselves to a long portfolio only with the upper bounds of 0.1 for each security to encourage diversification. We found that there were hardly any differences within small values of delta so we analyzed to get a directional read by looking at large variations.

We see that the expected returns for higher delta are slightly lower as the optimizer works towards minimizing the variance that has a high impact on the overall objective function as compared to the other linear terms, i.e. returns and transaction cost. This can also be checked by the standardized variations (without the delta) that show that for high deltas the rebalancing is done so that the variance of the portfolio is minimum.



Delta	X^TVX: Actual Variance
-10000	0.040537
-1000	0.040537
-100	0.040537
-10	0.036723
0	0.036723
10	0.016987
100	0.002008

Figure 2.3. Returns vs Delta

Keeping this in mind we chose the value of delta to be equal to one for the rest of our analysis.

2.3 Time Analysis

In this part of our sensitivity analysis, we tried to see if choosing a different and more volatile starting point can assist any of the four portfolios to generate higher returns. The default date is 3/12/2010 which is the number 300 in the database. This period is past the high volatility periods. As a comparison we used the date 5/27/2009 (=100), which is a much more volatile period (Figure 1.1).

We can clearly see from Figure 2.4 that our four investment strategies did better during the less volatile periods. One reason why the investments did better during less volatile and bear market is because Markowitz and the Trender use 'back-testing' and perhaps they are not very accurate when the state of the economy changes signals frequently.

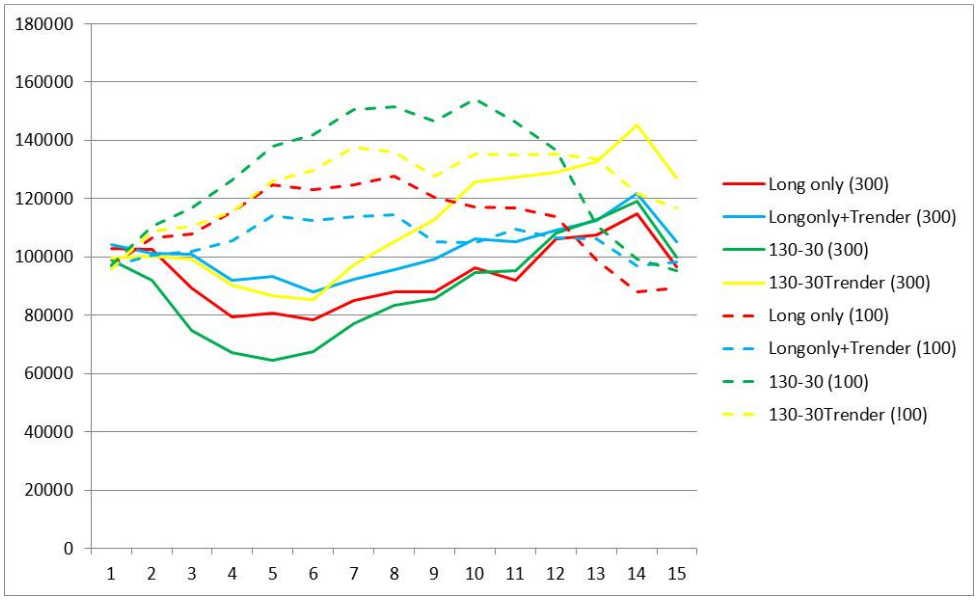


Figure 2.4 Returns for different strategies in different time periods

Starting period:	**5/27/2009 (=100)**	**3/12/2010 (=300)**
Long only	11211.59847	12931.86887
Long only & Trender	8923.058322	6137.581219
130-30 Fund	17068.15861	21295.89371
130-30Fund + Trender	18883.06317	12461.54935

Table 2.1 Standard deviation of our wealth (initial wealth is $100,000)

We can see that Long Only and 130-30 Fund have higher standard deviation in the first column; the reason can be because of the volatile period. However, if we look at these strategies with the Trender, the first column has lower standard deviations. We can say that during volatile periods the Trender doesn't use the mean-variance to signal, and therefore, the Markowitz function has more limited choice.

2.4 Sensitivity Analysis of the Trender

From our analysis, we conclude there are benefits as well as drawbacks to using the Trender for portfolio management.

We saw evidence that there were months where the Trender benefited tremendously from correctly identifying a downtrend in a particular set of ETFs. On a number of occasions, the funds without the Trender entered into downtrending ETFs only to suffer double digit losses. Using the Trender can prevent and/or limit large losses in these situations. Additionally, with the 130/30 structure, these ETFs can be shorted in the hopes that the trend continues.

On the other hand, there were a few examples of the non-Trender portfolios greatly outperforming the Trender portfolios. If the Trender sends a negative signal when the ETF is

actually trending upwards, the Trender-based portfolios will miss out on the upside for these excluded ETFs.

The Trender function has a parameter "sensitivity" which is used to determine how wide the distance between support and resistance lines should be. A wider range means that the price changes need to be more extreme before a new signal is output. Note that 1, the lowest possible value for sensitivity, actually corresponds to the narrowest band, thus most sensitive to price changes. Thus increasing the parameter will actually decrease sensitivity.

A false negative (no new signal when the trend has actually changed) is more likely for larger X. This results in holding ETFs long after they begin a price decline or waiting too long to buy during a boom. In the extreme this can result in "buy high, sell low", which is not desirable.

A false positive (signaling a new trend when the old trend continues) is more likely when X=1, since the Trender is most sensitive. This leads to situations where ETFs are not held before a rally or are held when a downtrend continues. In addition, if Trender is too sensitive, you will need to pay much more in transaction costs.

Below are the results for X = 1,2,3,4,5 for both the Long-Only strategy and the 130-30 strategy. The long only strategy seems to be an increasing function of X up until X=3, which is the maximum return of these five parameter values. With a return of 8.46%, the Trender with X=3 outperforms our original portfolio by over 3%.

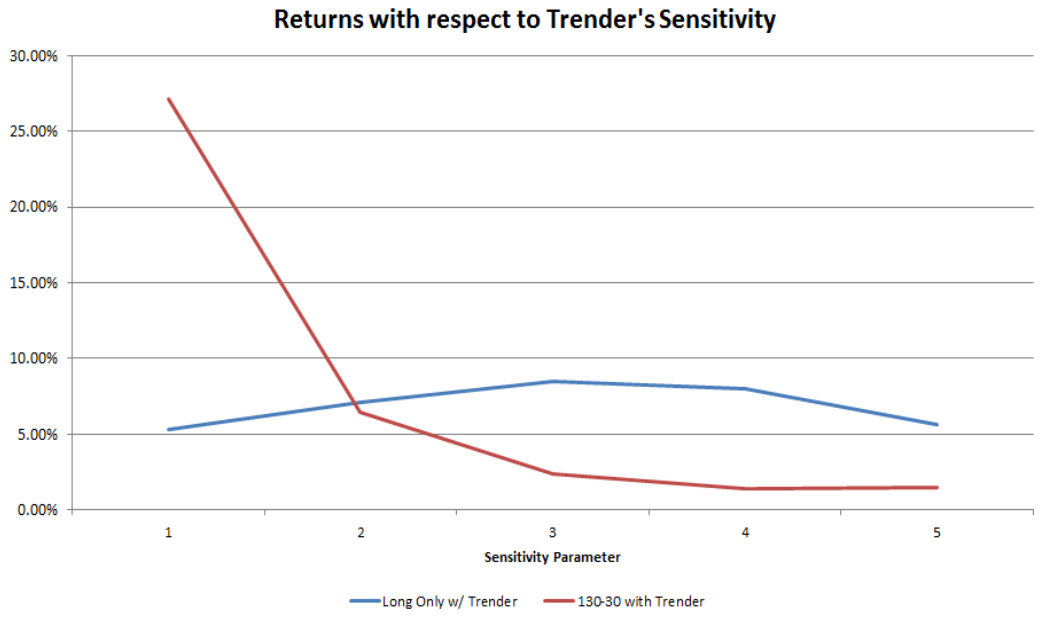


Figure 2.5 Returns with Trender Sensitivity

However, the 130/30 fund seems to perform exponentially worse as the sensitivity is diminished (increase X). In moving from X=1 to X=2, the 130/30 fund's returns get cut by almost 25% for this investment horizon.

Therefore, we believe the Trender should be very sensitive for 130/30 funds since leveraging increases the importance of identifying ETFs for both long and short positions.

The Trender portfolios have lower volatility in their monthly earnings than the comparable portfolio without the technincal indicator. The standard deviation of the long only portfolio decreases from 8.6% to 6.0% when the technical indicator is used. Similarly, the 130-30 fund's standard deviation of returns drops from 10.1% to 7.3% when the Trender is used. This makes sense, since the Trender limits the amount of exposure a portfolio gets to extreme price flucuations.

As shown below, the standard deviation decreases as the trender gets more sensitive. When the Trender is its most sensitive (X=1), the returns also exhibit the lowest standard deviation. The more sensitive the Trender is, the less exposure to the portfolio gets to price swings.

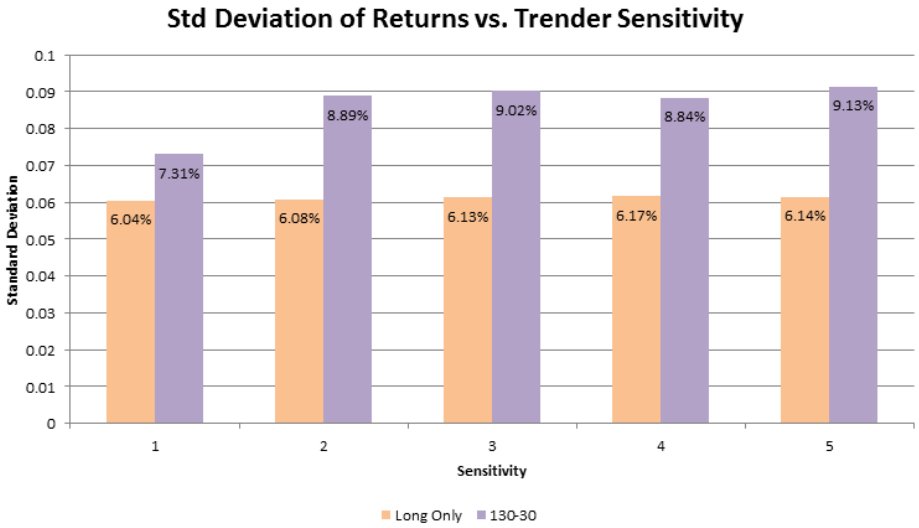


Figure 2.6 Std Dev. Of returns with sensitivity

3. Impact of the Trender

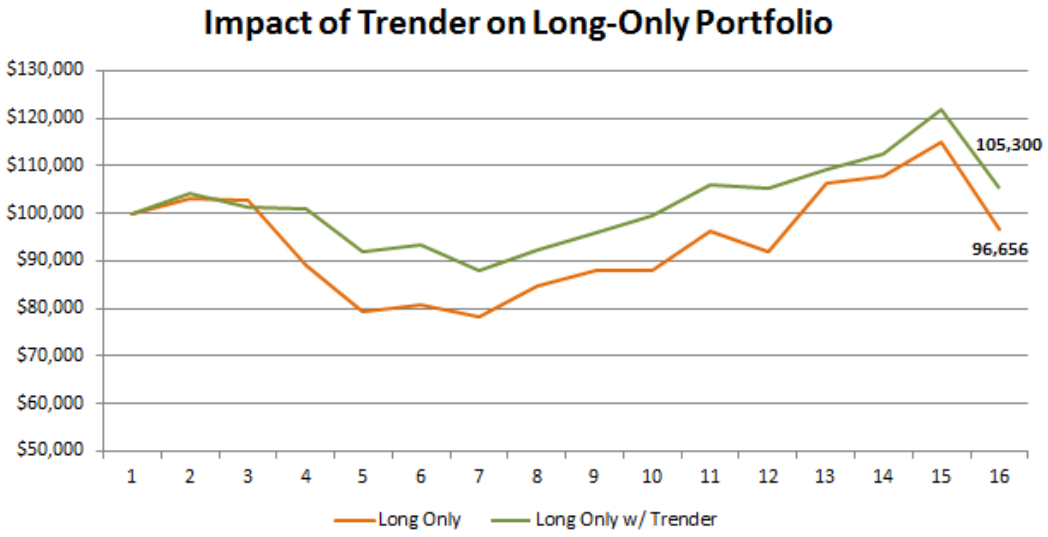


Figure 3.1 Long Only Portfolio

Above, we show the growth of two $100,000 investments beginning at March 12, 2010. The portfolio weights are optimized roughly every month (20 trading days) with Markowitz's mean-variance optimization. The only difference between the two is that the "Long Only w/ Trender" investment has an additional constraint that we cannot hold any stock that has a negative signal from the Trender at the time of rebalancing.

As we can see above, the Trender leads to a significant improvement. After the second month, the Trender-based investment remains strictly more valuable at the end of each periods.

After the 15 investment periods (5/19/2011), the Trender-based investment achieved a 5.3% return, ending with a value of $105,300. The portfolio without Trender fared significantly worse declining 3.34% to end with a final value of $96,656.

Period 3 (5/10/2010 - 6/8/2010)

Long Only without Trender

Ticker	Description	Weight	Period 3 Return
XLF	Financials	0.097	-11.12%
XLY	Consumer Staples	0.300	-7.91%
IJR	Small-Cap Stocks	0.300	-10.41%
VXX	Volatilty	0.003	-4.15%
UCO	Crude Oil	0.300	-21.53%

Long Only with Trender

Ticker	Description	Weight	Period 3 Return
VXX	Volatility	0.300	-4.15%
GLD	Gold	0.300	2.92%
SLV	Silver	0.300	-1.21%
TLT	US Long Term Bonds	0.100	4.75%

Table 3.1 Weights of Trender for a Long Only Portfolio

In period 3, the long-only portfolio lost 13.05% due to an overly optimistic portfolio which had large exposures to equities and crude oil. The Trender was able to identify that these markets were in a downturn, and instead invested in asset classes that typically do well in a downturn

such as Gold and long-term US Treasuries. This conservative strategy resulted in a tiny loss of 0.26% which represents an improvement of nearly 13%.

While the Trender clearly has its benefits, there is always the risk that it will falsely identify a downtrend. In this case, the Trender-based portfolio can miss out on large losses. The Trender portfolio returned 3.86% during Period 12 (1/25/2011 – 2/23/2011). However, the other portfolio returned 15.67% in this period, beating the Trender portfolio by nearly 12%. The unrestricted portfolio invested 60% split between crude oil and silver, both of which wound up gaining double digit returns. However, the Trender-based portfolio was not able to invest in SLV and UCO since the Trender identified them as being in a downtrend. As shown on the left, the downtrend immediately prior to Period 12 (in red) prevented these large gains from being realized by the Trender portfolio.

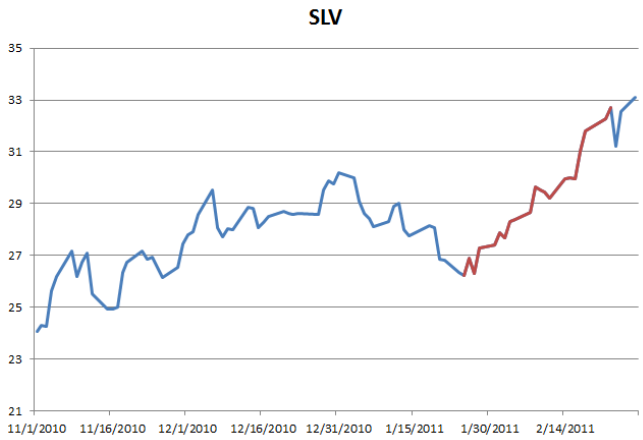


Figure 3.2 Trender Risk

	Long Only Returns		
	w/ Trender	w/o Trender	difference
1	4.35%	2.97%	1.39%
2	-2.99%	-0.31%	-2.68%
3	-0.26%	-13.05%	12.79%
4	-8.83%	-10.98%	2.15%
5	1.47%	1.55%	-0.08%
6	-5.81%	-2.79%	-3.02%
7	4.83%	8.28%	-3.46%
8	3.87%	3.54%	0.33%
9	3.77%	0.02%	3.75%
10	6.82%	9.33%	-2.51%
11	-0.91%	-4.41%	3.50%
12	3.86%	15.67%	-11.81%
13	3.10%	1.30%	1.80%
14	8.11%	6.75%	1.36%
15	-13.55%	-15.94%	2.39%
OVERALL	5.30%	-3.34%	8.64%

Table 3.2 The long-only monthly returns

Below we show the growth of two $100,000 investments beginning at March 12, 2010. Both investments employ a 130-30 structure. However, in the green line, we use the Trender technical indicator to determine the two subsets of ETFs can be considered for a long or short position.

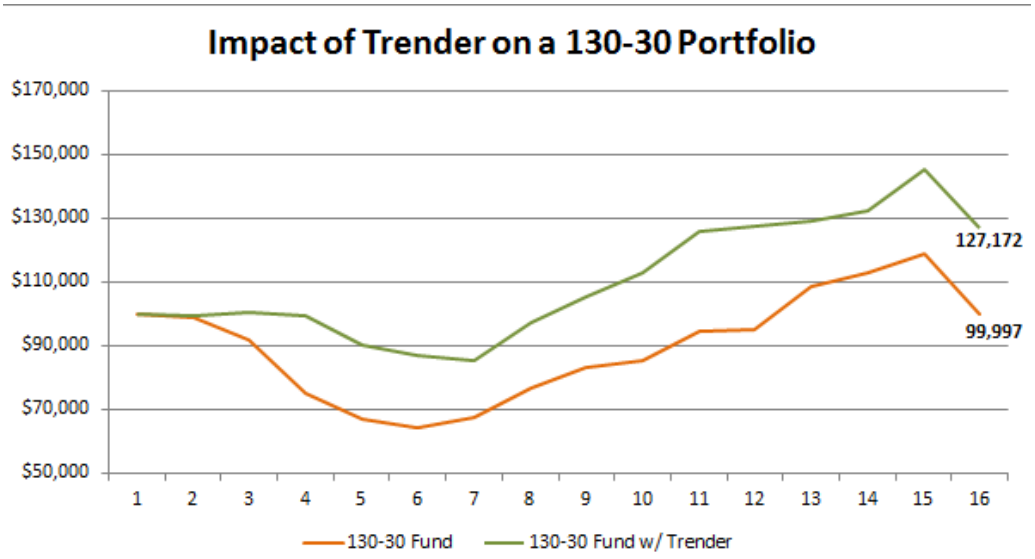


Figure 3.3 130-30 Portfolio

Like for the long only comparison, the Trender-based portfolio's value is strictly higher at each month's close after initialization. By the end of 15 months, the Trender portfolio grew to $127,172 (+27.2%) while the non-Trender portfolio essentially broke even, finishing at $99,997.

As was the case with the long-only portfolios, Period 3 had the largest monthly outperformance by the Trender portfolio, at 17.57%. The portfolio without Trender lost 18.66% as opposed to the 1.09% when the Trender was used. Note that among the trend-based portfolios, the return is slightly lower in the 130/30 structure due to a short position in the natural gas ETF (UNG). In general, the differences between the Trender portfolio and the non-Trender are more spread out for both gains and losses as a result of the increased leverage. The monthly returns are summarized below.

	130-30 Returns		
	w/ Trender	w/o Trender	difference
1	-0.35%	-0.91%	0.55%
2	0.74%	-7.12%	7.86%
3	-1.09%	-18.66%	17.57%
4	-9.05%	-10.46%	1.42%
5	-3.88%	-3.74%	-0.15%
6	-1.47%	4.76%	-6.23%
7	13.73%	13.90%	-0.18%
8	8.15%	8.15%	0.00%
9	7.37%	2.75%	4.63%
10	11.43%	10.68%	0.75%
11	1.15%	0.54%	0.62%
12	1.51%	13.84%	-12.33%
13	2.58%	4.08%	-1.50%
14	9.65%	5.62%	4.04%
15	-12.50%	-16.06%	3.56%
OVERALL	27.17%	0.00%	27.18%

Fig. 3.3 130-30 monthly returns

4. Conclusion

This analysis was carried out to implement and compare between the following strategies:

1) 130/30 strategy on the ETFs optimized within the Markowitz framework
2) 130/30 strategy again with a technical indicator and again optimized using Markowitz.
3) Markowitz optimization without the 130/30 or the technical indicator constraint

We selected a diverse set of equities that spanned all industrial sectors in the US and we also considered a few indices from the emerging markets, commodities and fixed income

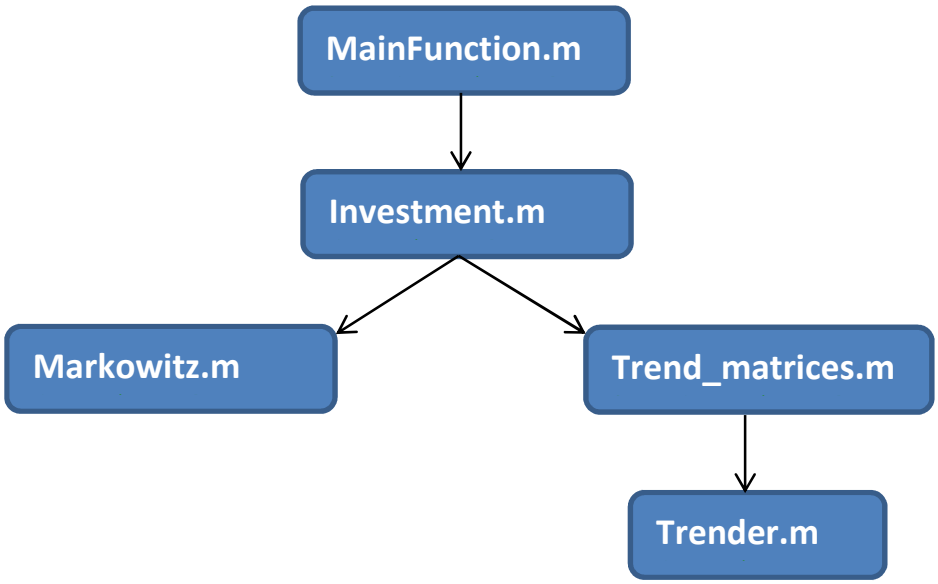
First we looked at the sensitivities of our parameters to our analysis:

- Time period – We looked at the returns from two separate time periods and decided to continue our analysis beginning March 2010.
- Delta – We studied the impact of changing delta on the returns but didn't observe a high sensitivity. Since the impact isn't too high we chose delta to be 1.
- Transaction cost – The transaction cost are quite considerable for our analysis since we look to rebalance our portfolio with every change in the trender signal. Varying such costs brings quite noticeable changes to our positions at each time.
- Trender Sensitivity – The Trender depends on how extreme price movement is needed for a signal to be generated. We find that volatility tends to decrease as the Trender is set more sensitive. Returns were highest for the 130-30 fund with the greatest sensitivity while the long-only portfolio benefited from being slightly less sensitive.

Figure 4.1 Comparison of Returns and Risk of Long Only and 130-30 vs Trender

From the above observations and analysis we can conclude that the trender that we developed has proven to be a successful technical indicator and helps identify certain attributes that escape the algorithm that Markowitz follows. Even for strategies such as the 130/30, the trender is able to identify profitable stocks to both long and short so that maximum returns can be achieved with minimum risk.

5. Code



MainFunction.m

```
% Investments(total long position, upper bound, lower bound, use trender?);

clear all;

[Strat13030 var1030 weights1]=Investment(1.3,0.3,-0.2,0);
[Trender13030 vartrender weights2]=Investment(1.3,0.3,-0.2,1);
[LongOnly varlong weights3]=Investment(1,0.3,-0.3,0);
[LongOnlyTrend varlongTrend weights4]=Investment(1,0.3,-0.3,1);

xlswrite('Returns.xls',[LongOnly;LongOnlyTrend;Strat13030;Trender13030],
'Returns', 'A1');
xlswrite('Returns.xls',[varlong;varlongTrend;var1030;vartrender], 'Variance',
'A1');
xlswrite('Returns.xls',[weights1], 'Weights-13030', 'A1');
xlswrite('Returns.xls',[weights2], 'Weights-13030-Trender', 'A1');
xlswrite('Returns.xls',[weights3], 'Weights-LongOnly', 'A1');
xlswrite('Returns.xls',[weights4], 'Weights-LongOnly-Trender', 'A1');
```

Investment.m

```
function [Strategy var MarkowitzStrat]=Investment(long1,x,y,xxx,d)

load ETFS.mat;

trxn_cost = repmat(0.01,22,1); %transaction cost
upper = repmat(x,22,1);        % long weights in one ETF
lower = repmat(y,22,1);        % short weights in one ETF
port = repmat(0,44,1);         % the weights in the initial portfolio
tech_ind_long = ones(22,1);    % default values if the trender is not
```

```
                                % used. Basically, all signals say buy
tech_ind_short = ones(22,1);    % same idea but all signals also say sell
                                % so it's up to Markowitz to decide
lookback_period = 3;
current = 300; % 100=5/27/2009
               % 200=10/16/2009
               % 300=3/12/2010
               % 400=8/4/2010
               % 600=5/19/2011
               % 700=10/11/2011
delta = 1;     % indicates how much risk averse the investor is
no_etf = 22;

MarkowitzStrat=zeros(44,1);
var = 0;
port_val=zeros(22,1);
e1=ones(22,1);
horizon=15; % how many times we want to rebalance

alpha = 0.96; % used in the indicator, to specify the exp-mean
sensitivity = 1; % shows how sensitive the signals be
                 % the higher the number the.......
if xxx==1  % the value xxx is 1, it means we are using the Trender
    [trender_BUY, trender_SELL] = trend_matrices(px_high,
px_low,px_close,alpha,sensitivity);
end
% the loop runs the Markowitz the number of times we rebalance
% and it calculates the real returns on each asset each period
 for i= 1:horizon
    long = long1;
    short = 1-long;
    start1 = 20*(i-1) + current;
     if xxx==1  % the technical indicator changes the buy/sell signals
                % every time we want to rebalance
        tech_ind_long=trender_BUY(start1,:)';
        tech_ind_short=trender_SELL(start1,:)';
        if(sum(trender_BUY(start1,:),2) < long/x)
              tech_ind_long = ones(22,1)
        end
        if(y<0)
        if(sum(trender_SELL(start1,:),2) < short/(-y))
              tech_ind_short = ones(22,1)
        end
        end
    end
    [temp var1] = markowitz(start1 - 20*lookback_period, start1-1, delta,
trxn_cost, upper, lower, port, long, short, tech_ind_long, tech_ind_short);
    port=temp;
    MarkowitzStrat=[MarkowitzStrat,temp];
    var = [var,var1];
    total_port = temp(1:22,1) + temp((22+1):44,1);
    ret = (px_close(start1+20,:)-px_close(start1,:))./px_close(start1,:);
    temp1 = total_port.*(ret-)';
    port_val=[port_val,temp1];

 end
var = var(2:horizon+1);
MarkowitzStrat=MarkowitzStrat(:,2:horizon+1);
MarkowitzStrat=round(MarkowitzStrat*10000)/10000;
```

```
port_val=port_val(:,2:horizon+1);
Strategy=cumprod((sum(port_val,1))+1).*100000;
end
```

Markowitz.m

```
%the function below calculates the optimal weight allocation to the
%portfolio whose historical mean and cov have been passed for each security
%mu is nX1 vector of historical mean
%V is nXn matrix of historical covariances
%trxn_cost is a nX1 vector of the transactions cost, can be the same
%upper and lower are the bounds of the weights, nX1 vectors
%the input portfolio is 2nX1 of the form: (port+ port-)' where port+ is a
%set of long positions and port- is a set of short positions

%It optimizes by maximizing the return and minimizing the variance (depending
on delta which defines risk aversion) under
%the following constraints:
%transaction cost - specified by the scalar trxn_cost
%upper bound - an

function [new_port var] = markowitz(start, stop, delta, trxn_cost, upper,
lower, port, long, short, tech_ind_long, tech_ind_short)
[mu,V]=stats(start,stop);

n=length(mu); %this gives the no. of securities being considered

H = [delta*V zeros(n,6*n);  % for minimizing variance
    zeros(6*n,7*n)];        % for rest of the variables
f = [-mu zeros(1,2*n) trxn_cost' -trxn_cost' zeros(1,2*n)];

A = [eye(n) zeros(n,6*n);   % for Xnew<= upper
    -eye(n) zeros(n,6*n);   % for Xnew>= lower
    zeros(n,n) -eye(n) zeros(n,5*n);    % for Xnew+ >= 0
    zeros(n,2*n) eye(n) zeros(n,4*n)    % for Xnew- <=0
    zeros(n,3*n) -eye(n) zeros(n,3*n);  % for X+>=0
    zeros(n,4*n) eye(n) zeros(n,2*n)    % for X-<=0;

    ];

b = [upper;
     -lower;
     zeros(4*n,1)
    ];

Aeq = [ones(1,n) zeros(1,6*n);                          %sum of Xnew
       zeros(1,n) ones(1,n) zeros(1,5*n);               %sum of Xnew+
       zeros(1,n) ones(1,n).*tech_ind_long' zeros(1,5*n);     %sum of
Xnew+*tech_ind
       zeros(1,2*n) ones(1,n) zeros(1,4*n);             %sum of Xnew-
       zeros(1,2*n) ones(1,n).*tech_ind_short' zeros(1,4*n);   %sum of Xnew-
*tech_ind
       eye(n) -eye(n) -eye(n) zeros(n,4*n);             %Xnew= Xnew+ + Xnew-
```

```
       eye(n) zeros(n,2*n) -eye(n) -eye(n) -eye(n) -eye(n);     % Xnew=Xold+
+ Xold- + X+ + X-
       zeros(2*n,5*n) eye(2*n)                          %Xold+ = port+ and
Xold- = port-
      ];

beq = [1;
        long;           % scalar that represents the total long position
        long;
        short;          % scalar that denotes the total short position
        short;
        zeros(2*n,1);   %for Xnew - Xnew+ - Xnew- = 0
        port            %the 2nX1 vector of the previous portfolio
        ];

x = quadprog(H,f,A,b,Aeq,beq);
new_port = x(n+1:3*n);
var = sqrt(delta*x'*H*x);
```

Trend_matrices.m

```
% A function that returns a matrix of buy signals and sell signals for
% each day from 1..n for a series of ETFs 1..d.
%
% Inputs:
%      px_high - A matrix of highest daily prices where the (i,j) element
%                represents the jth ETF's highest price for day i.
%      px_low - A matrix of lowest daily prices where the (i,j) element
%               represents the jth ETF's lowest price for day i.
%      px_close - A matrix of closing daily prices where the (i,j) element
%                 represents the jth ETF's closing price for day i.
%      sensitivity, alpha - parameters used for the trender()
%
% Outputs:
%      trender_BUY - (i,j)  Positive signal from indicator for jth ETF on
%                     ith day if =1 and non-positive when =0.
%      trender_BUY - (i,j)  Negative signal from indicator for jth ETF on
%                     ith day if =1 and non-positive when =0.
%
function [trender_BUY, trender_SELL] = trend_matrices(px_high, px_low,
px_close,alpha,sensitivity)
% Stores the dimensions of inputs
dim = size(px_close);
n = dim(1);
d = dim(2);

% Create the empty nxd matrices for storing the positive/negative signals
trender_BUY = zeros(n,d);
trender_SELL = zeros(n,d);

% Calculates the trender signal history (days 1..n) for each ETF
for j=1:d
    close = px_close(1:n,j);
    high = px_high(1:n,j);
    low = px_low(1:n,j);
    [this_BUY, this_SELL] = trender(high,low,close,alpha,sensitivity);
```

```
    trender_BUY(1:n,j) = this_BUY;
    trender_SELL(1:n,j) = this_SELL;
end
```

Trender.m

```
% A function that returns a column vector of buy signals and a column
% vector of sell signals for days 1..n based on technical analysis
% of a security's 14 day price history.
%
% Inputs:
%     high  - a column vector of highest recorded price for days 1..n
%     low   - a column vector of lowest recorded price for days 1..n
%     close - a column vector of the closing price for days 1..n
%     alpha - a constant used to determine how much to discount information
%             as you go futher back. An alpha of 0.9 means that you weigh
%             yesterday's values 90% of the weight given to the current
%             day.  (0 < alpha < 1)
%     sensitivity - a constant used to determine how sensitive the trigger
%                   for the buy/sell signal is. Values range from 1 to 10.
%                   Values closer to one reflect short term price movements
%                   while values closer to ten are more representative of a
%                   long run trend.
% Outputs:
%      buy - binary vector where 1 represents a buy recommendation and
%            0 represents a sell recommendation
%      sell - binary vector where 1 represents a sell recommendation and
%            0 represents a buy recommendation.
%
function [buy, sell]= trender(high,low,close,alpha,sensitivity)
n = length(close);
buy = zeros(n,1);
sell = zeros(n,1);

% calculates the weights needed for a 14 day exponential moving average
p = 1:14;
raw = alpha.^p;
wt = raw/(sum(raw));

% calculates the Exponential Moving Average of the Midpoint (for t > 15)
MP = (high+low)/2;
EMAvg_MP = filter(wt,1,MP);
EMAvg_MP = [zeros(15,1); EMAvg_MP(16:n)];

% calculates the Exponential Moving Average of the True Range (for t > 15)
yesterday_close = [close(1); close(1:n-1)];
TR = max(high-low, max(abs(high - yesterday_close), abs(low -
yesterday_close)));
EMAvg_TR = filter(wt,1,TR);
EMAvg_TR = [zeros(15,1); EMAvg_TR(16:n)];

% calculates the standard deviation of EMavg_TR for past 14 days (t > 29)
std_dev_TR = zeros(n,1);
for t= 30:n
    partition = EMAvg_TR(t-14:t-1);
    std_dev_TR(t) = std(partition);
end
```

```
% Computes the Trender_Up values for days 16..n.
%   If a price closes above the Trender Up value for the day,
%   then a BUY signal is triggered.
Trender_UP = EMAvg_MP + 0.5*EMAvg_TR + sensitivity*std_dev_TR;

% Computes the Trender_DOWN values for days 16..n.
%   If a price closes below the Trender Down value for the day,
%   then a SELL signal is triggered.
Trender_DOWN = EMAvg_MP - 0.5*EMAvg_TR - sensitivity*std_dev_TR;

% Sets both Trender_UP and Trender_DOWN for t < 30 to the
% average price from 2..t.  This ensures each day prior to t=30
% will have a BUY/SELL signal generated.
for t=2:29
    Trender_UP(t) = mean(close(1:t));
    Trender_DOWN(t) = mean(close(1:t));
end

% Constructs the output vectors buy and sell.
% Note: both buy(1) and sell(1) will be 0 since one day of price history
%       is not sufficient to predict a trend.
for t=2:n
    % a new buy signal - close at time t crosses above Trender_UP
    if(Trender_UP(t) < close(t))
        buy(t) = 1;
    % a new sell signal - close at time t crosses below Trender_DOWN
    elseif(Trender_DOWN(t) > close(t))
        sell(t) = 1;
    % no new signal - maintain previous buy/sell recommendation
    else
        buy(t) = buy(t-1);
        sell(t) = sell(t-1);
    end
end
```