# IAQF Annual Academic Competition 2022

# Title - TBD!

Team Quantivenom

## Abstract

## 1 Introduction

Financial markets are ever changing like the sea. While some changes may be transitory, the changed behavior often persists for many periods. Considering the recent turmoil spurred by the COVID-19 financial crisis, one may have the impression of a paradigm shift in investors' assessment of risky assets. The mean, volatility, and correlation patterns in stock returns, for instance, changed drastically at the start of, and continued through the global financial crisis of 2007-2009. Similar regime changes, some of which can be recurring like recessions vs expansions, and some of which can be permanent (structural breaks), are prevalent across a wide range of financial markets and in the behavior of many macro economic variables. The observed regimes in financial markets are closely related to the phases of business cycle as well as other factors. Regime changes present a big challenge to traditional trading strategies, demanding a more adaptive approach. Moreover, the time varying behavior of risk premiums, volatility, and correlation have important implications for risk-management.

It is generally accepted that one cannot predict the price of an individual security, but many believe that it may be possible to predict (or determine) the "hidden" state of the market. To that end, market regime detection models provide the ability to identify these hidden states to adjust one's portfolio efficiently. This study aims to develop a technique to determine the state of the market - bull, bear or static. Bull markets are characterized by expansions, while bear markets are a consequence of sudden market crashes. While Static markets lack a general consensus in terms of definition, in essence they are periods which although possibly turbulent, give an investor no incentive to stay in the market, as there is no long term expansion or contraction.

This study has been divided into several segments. We start with defining benchmarks for our predictive models, against which we measure the validity of our proposed model. Next, we briefly explain the theory behind the Hidden Markov Model and how Expectation-Maximization algorithm, and its variations can be effectively used to identify different states of the market (regimes). We also run through the various macro-economic indicators that have been used to identify the market state, both in-sample and out of sample. Finally, we propose our market state detection model, and examine how it measures up against the benchmarks we defined as well as the standard buy-and-hold strategy. We conclude the study by depicting the results of various models and discussing their future implications.

## 2 Defining the "True States"

We define a benchmark of market states to examine the accuracy of the state prediction algorithm that we will use. This will allow us to validate our model by comparing the PnLs generated by the classified market states to the PnLs generated by the benchmark market states. (Note that, the actual inherent market state is unknown, so throughout the study, whenever we refer to true market states, we mean the states that we will assign using the following algorithm.)

The steps that we undertake to define these "true states" of the market are as under:

- If the Russell 3000 index rises by at least 10% from the previous market low till the next market high, label it as true market state BULL.
- If the Russell 3000 index drops by at least 10% from the previous market high till the next market low, label it as true market state BEAR.
- For identification of static states, we look at the change in index levels during a 3 month period.

- If change > 20%, call it true market state BULL
- Else if change < -15%, call it true market state BEAR
- Else, call it true market state NEUTRAL

We also combine consecutive high change states into one state and check the change in index value in all resultant states as well.

Following are the identified true states according to the above algorithm.


Figure 1: True States

# 3 Hidden Markov Model

## 3.1 Motivation

The data set used in our analysis is the Russell 3000 index starting from 1987 to 2018, which covers a lot of phases over history. Such a time series data consists of many fluctuations of price movements over time. If we disregard short-term fluctuations (less than 3 month period), we can see that the price movements can be classified into regimes of bull, bear and neutral accompanied by periods of high and low volatility. One can leverage these market conditions and enter into trade positions to make profits. Therefore, classifying such regimes becomes important for investors to take trading positions in the market.

Modeling time series is used to forecast future events based on past observations. In reality, financial markets depend on various factors which contribute to the price changes, that are not explained using a single model. Instead, we attempt to identify market regimes. This is built on the intuition that for different time windows, we may get different market states.

Modeling and forecasting next-day returns is difficult for the reasons mentioned above, but for traders, identifying current regimes can be useful so that they can make informed decisions. Therefore, we need to consider several factors in a silo as well as in conjunction which essentially govern the changes in the market regime. For modeling the market states, we first consider the Hidden Markov Model (HMM).

HMM assumes that the time series consists of an unobservant (hidden) state which is a Markov process. This will help us capture the stochastic nature of the time series data using observable states. These hidden states will determine the behavior of the index prices which is hidden from the trader/investor. Therefore, we can model HMM on our historical data of the Russell 3000 index and identify the hidden state. This will classify the market regime although will not directly forecast values

## 3.2 Identifying market state using only Russell 3000 index price

Putting in simpler terms, Hidden Markov Model is a probabilistic model that gives an idea about the hidden truth, given the observed data of the model. These models are useful in finance when one wants to obtain information about the hidden truth like the market state (i.e. Bear, Bull, or Neutral) given the observed values like stock prices, index values, returns, etc. In technical terms, the hidden truth is referred to as the state variable whereas the observed data is referred to as the observed variable.

Here, the state variable is the hidden state of the market. Let's denote this state variable at time t by $S_t$. The observed variable at time t is the returns on Russel 3000 index at time t, denoted by $R_t$. (Note that the state of the market is not observable and hence is a latent variable for the model). Given a hidden state, there will be a fixed set of probabilities of transition to the next hidden state. This probability information is captured in state transition matrix A. Also, given the hidden state of the market, there is again a fixed set of probabilities of it corresponding to a specific value or returns. This probability information is captured in state emission matrix B. Finally, we have the initial state of the market, denoted by $\pi_0$. Together we refer to these parameters by $\theta$. The idea behind HMM model is to tune these three parameters (i.e. A, B and $\pi_0$) using a modified version of expectation maximization (which is generally used for parameter estimation of models with latent variables) and then use the tuned parameters to predict the hidden state of the market. So the process is naturally divided into two parts -

Learning and Decoding (Refer Appendix for formulae of variables in following subsections).

### 3.2.1 Learning

For this part, we use the Baum-Welch algorithm, which is a special case of expectation maximization (EM) algorithm because it builds on the basic EM algorithm and uses a forward backward algorithm as follows:

**Step 1:** Initialize the state transition matrix A, state emission matrix B and the initial state of the market as $\pi_0$ to some random values.

**Step 2:** Compute the following probabilities using forward - backward technique:

- $\alpha_i(t)$: In the forward pass, it calculates the forward probabilities $\alpha$ of Russell 3000 index price corresponding to a particular market state i.e. state i at time t and having the first t observed variables i.e. the returns upto time t.

- $\beta_i(t)$: In the backward pass, it calculates the backward probabilities $\beta$ of observing the remaining T-t observations i.e. the returns from time t onward, given a particular market state i.e. state i at any time t.

**Step 3:** Use these computed probabilities to find a better estimate for the parameters (A, B, $\pi_0$).

- Define temporary probability variables:
(i) $\gamma_i(t)$ is probability of being in market state i at time t given the observed sequence of returns and the parameters (A, B, $\pi_0$)
(ii) $\xi_{ij}(t)$ is the probability of being in market states i and j at times t and t+1 respectively given the observed sequence of returns and parameters (A, B, $\pi_0$) for Russell 3000 index.

- Update the parameters (A, B, $\pi_0$) using these temporary probability variables:
(i) $\pi_i$, which is the expected frequency spent in market state i at time 1
(ii) $a_{ij}$, which is the expected number of transitions from market state i to market state j compared to the expected total number of transitions away from market state i
(iii) $b_{ij}$, which is the expected number of times the returns have been equal to $R_i$ while in market state j over the expected total number of times in market state j.

**Step 4:** Iterate through steps 2 and 3 until parameters (A, B, $\pi_0$) converge.

### 3.2.2 Decoding

Once we have the trained parameters namely state transition matrix A, state emission matrix B and the initial state of the market as $\pi_0$ , we make inferences based on a trained model and the observed returns (Assume that the observation space has returns for the Russell index till time t). For making inferences, we use the Viterbi algorithm that divides the market states into one of the 3 states - Bull, Bear or Neutral) at time t, given the trained model parameters (A, B, $\pi_0$) and observed return upto time t. The algorithm works as follows:

**Step 1:** Define a matrix *trellis* of dimension (3 x t) to hold probability p of each market state upto time t, given each observation of returns upto time t

**Step 2:** Define another matrix *pointers* of dimension (3 x t) to hold backpointer to best prior state

**Step 3:** Determine each market state's probability at time 0

*for s in range(3):*
  *trellis[s, 0] ← pi[s] * B[s, 0]*

**Step 4:** Track each market state's most likely prior market state (call it state k)

*for r in range(1, length(R)):*
  *for s in range(3):*
    *k ← argmax(k in trellis[k,r-1]*A[k,s]*B[s,o])*
    *trellis[s,r] ← trellis[k,r-1]*A[k,s]*B[s,o]*
    *pointers[s,r] ← k*

**Step 5:** Collect the sequence of most likely hidden states in *bestPath*

*k ← argmax(k in trellis[k, length(R)-1])*
*for r in range(length(R)-1, -1, -1):*
  *bestPath.insert(0, S[k])*
  *k ← pointers[k, r]*

**Step 6:** Return sequence of states found in *bestPath*

Thus, we now have the hidden market states for each time instant upto time t given by the Hidden Markov Model.

# 4 Final Model

## 4.1 Motivation

The HMM model predicts the changes in states almost as similar as the "true states" but the changes are lagged which results in missing out on opportunities of entering into trading positions during dips and spikes (which would be the most profit-making time to enter into trading positions). Also, the HMM model results in some very small market regimes (of 5 days or less) which do not hold any economic significance.

We tried to deal with such issues by introducing new leading predictors to our model. Note however that, despite all of the shortcomings, we will treat the HMM model as the baseline model and try to add more information about the change in the market state using additional features.

## 4.2 Exploring additional features

Following are the new features that were explored to be added to the model, in addition to the Russell 3000 index price.

1. **Realized Volatility Signal:** The biweekly realized volatility is computed using the standard deviation of the Russell 3000 index over a 10 day period. Using scatter plots, we see a marked correlation between the biweekly realized volatility and the change of the market states. ——— The prediction of change of state of the market generated by the HMM and the realized volatility was a better predictor than the signal generated by the implied volatility and the HMM model. It was able to predict the "true state" of the market with better timing and hence using this signal as a trading strategy would generate better returns.

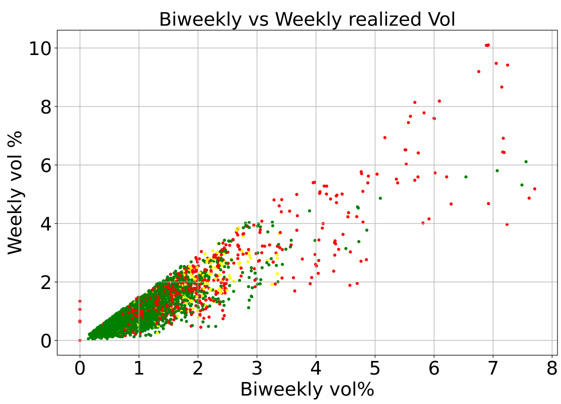


Figure 2: Realized Volatility

2. **Realized Volatility of VIX Index:** We will refer to this as vol of vol. The biweekly vol of vol is computed using the standard deviation of the VIX index over a 10 day period. Using scatter plots, we see a marked correlation between the biweekly realized vol of vol and the change of the market states. ——— The prediction of change of state of the market generated by the HMM and the realized vol of vol was a better predictor than the signal generated by the implied vol of vol (VVIX) and the HMM model. It was able to predict the "true state" of the market with much better timing and hence using this signal as a trading strategy would generate extremely good returns.

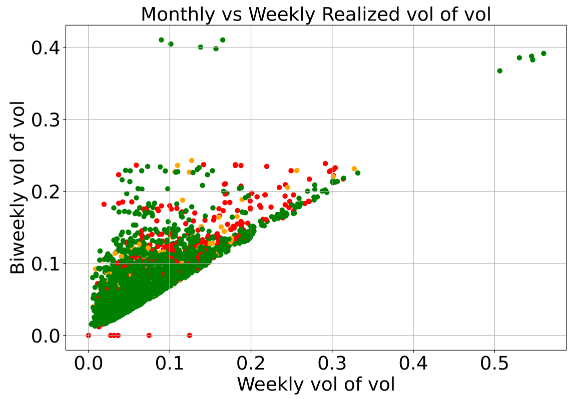


Figure 3: Realized Volatility of VIX

3. **Schiller Housing Prices Index:** Since housing bubbles are known to precede bear markets, we tried using the Schiller housing prices index as a feature of our model. The issue that was a hindrance to the model was the frequency. The Schiller housing index is published monthly whereas the other features we use are of daily frequency. Given the frequency mismatch, the Schiller housing price index was not able to provide any significant predictive power to our baseline HMM model. Hence, we eventually ended up dropping this feature.

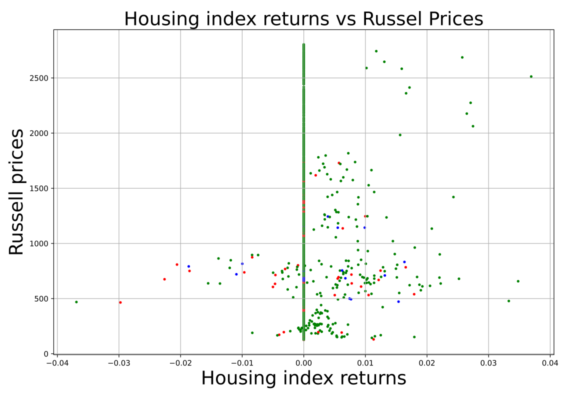


Figure 4: Schiller Housing Prices Index

4. **VIX:** VIX or the Fear Index was used with the motivation of being able to predict the bear regimes better. The VIX index is based on the

implied volatility of the near term S&P 500 index options. The implied volatility failed to give any promising results for state changes. The predictions of the change of state were lagging and failed to capture the "true state" of the market at the right time leading to a prediction model which was mistimed. The state changes using realized volatility were better in terms of timing and we eventually used them in our prediction model.

5. **VVIX:** Implied volatility of the VIX index (volatility of volatility) was also analyzed for the prediction model. As with the VIX feature, the state changes predicted using the VVIX index and the HMM model (which used the Russell 3000 index as input) also were lagging, resulting in not efficiently capturing the spikes and the plummets of the Russell 3000 index for efficient money-making trading strategy. The state changes using realized volatility of VIX were better predictors of change of state and hence captured the peaks and troughs better, leading to better cumulative returns on the training data. To capture sudden movements in VIX from persisting trends, we defined a feature as the deviation of the daily reading from the 15-day moving average of the VIX index.

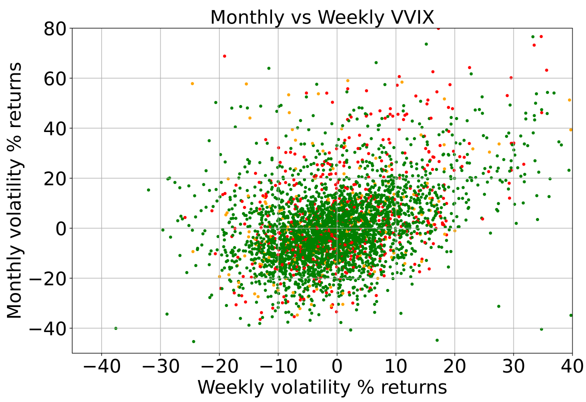


Figure 5: VVIX

6. **Treasury yields:** Treasury yields are typically a good indicator of investor confidence. Yields typically narrow during periods of low market confidence, indicating investors prefer moving out of risk assets and widen during periods of high confidence. Based on this theory, we expected Treasury yields to be a good leading indicator of the market state. However, on analysis, it was observed that this signal does not hold significant predictive power over and above the baseline HMM model. This is possibly due to the fact that there may be a variety of other reasons for the movement of Treasury yields, such as changing interest rate expectations, and yield movements are also regulated by Fed. Hence, we eventually ended up excluding this signal from our model.

7. **High Yield Spreads:** The spread between the yield-to-worst of High Yield corporate bonds and the 10-Y treasury rate was also analyzed as a potential indicator for the prediction model. We expect corporate bond spreads to widen preemptively as markets enter a bear phase, and tighten during a strong bull run this is also because of investor confidence. Due to the hybrid characteristics of High Yield bonds (credit products with high risk), this effect is expected to be more prominent for High Yield bonds as compared to Investment Grade bonds. This was verified in the analysis, as High Yield spreads proved to capture the market state better than Investment Grade spreads. We use the "BarCap US Corp HY YTW - 10 Year Spread" Index to proxy for High Yield spreads. We include this indicator to extract an alternative signal within our model.

8. **Daily Price Fluctuation:** Intra-day price fluctuations provide information regarding volatility and sudden market movements on a daily basis, which led us to believe such a signal may help capture the state of the market. We define the formula as follows:

$$\text{T-1 Daily Fluc} = \frac{high_{t-1} - low_{t-1}}{open_{t-1}}.$$

Sudden market crashes typically involve a large amount of fluctuation of the index price within a day, whereas Bull runs often have a more steady intra-day trading pattern. We thus, expected such a signal would be useful in capturing bear states accurately, which was verified through analysis as well. We include this feature to obtain a signal for our model.

## 4.3 Identifying market states using Signal Extraction

This section elaborates on the final indicators chosen to incorporate in our model to identify market states, and the methodology to extract signals from these indicators.

1. **Signal 1:** For this specific signal, we use the features T-1 daily returns of Russell 3000 index, T-1 daily fluctuations, T-1 High Yield spreads, T-1 Open - Close prices of Russell 3000, T-1 VIX deviations from smooth moving average. These features are trained on a Random Forest Classifier to classify the likely market state based on our "true states". Since machine learning models are unable to incorporate the concept of predicting "contiguous targets", there is expected to be some noise in identified states, i.e the odd daily state identified as bear/static within a broader bull run and vice versa. To gain more confidence from the signal extracted from our machine learning models, our predicted state for each data point is the most frequently occurring state in the set of states predicted for the current and trailing four data points. This smoothing ensures we reduce the noise in the predicted state sequence, which we feel is essential based on the spirit of the definition of a 'market state'.

2. **Signal 2:** For the second signal, we use the bi-weekly realized volatility of the Russell 3000 index as well as the daily fluctuation in intra-day prices as defined above. For defining the states using volatility model, we used an optimization approach to come up with the thresholds to classify bear, bull and static states of the market. The results of this signal have been presented in the tables in under Validity of Approach section as well as the Results section. It was observed that the states being identified by this model was able to predict the change in the states better that the baseline HMM model. Machine learning models are expected to result in some noise in the final state and to deal with this issue, so to reduce this noise we consider our predicted state for each data point as the most frequently occurring state in the set of states predicted for the current and trailing four data points. This smoothing ensures we reduce the noise in the predicted state sequence, which we feel is essential based on the spirit of the definition of a 'market state'.

3. **Signal 3:** After looking at the realized volatility results, we decided to identify the states of the market through realized vol of vol as well. For defining the states using vol of vol model, we used an optimization approach to come up with the thresholds to classify bear, bull and static states of the market. The results of this signal have been presented in the tables in under Validity of Approach section as well as the Results section. It was observed that the states being identified by the vol of vol model was able to predict the change in the states better than any of the other signals. The timing was the best for this vol of vol signal when compared to the other signals leading to highest PnL amounts. Machine learning models are expected to result in some noise in the final state and to deal with this issue, so to reduce this noise we consider our predicted state for each data point as the most frequently occurring state in the set of states predicted for the current and trailing four data points. This smoothing ensures we reduce the noise in the predicted state sequence, which we feel is essential based on the spirit of the definition of a 'market state'.

**Combining with HMM Baseline:** The 'pseudo signal state' extracted from each of the above signals is combined with the 'pseudo hmm state' from the HMM model to obtain the final market state prediction by switching the governing signal that predicts the state when either of the pseudo states indicates a change of state. This method ensures that there is no loss of information from the two signal sources, as we expect each signal to contribute separate crucial information for predicting the market state. To confirm the validity of this method, we run model validation as outlined below. While this method may increase the overall number of state changes within the predictions, validation experiments indicate the increase in state changes is not extreme as compared to the improvement in results.

## 4.4 Model Validation and Training

We consider the pseudo state predicted by HMM as a good starting point to predict the market state, but incorporating signals from the above identified indicators is expected to improve on the base case HMM prediction by using the above explained methodology. In order to validate our final model, we must impose a rolling window validation set, since maintaining the sequence of time series data is crucial. We take a rolling 9 year window for our train/validation sequence. The rationale behind selecting a 9 year window lies in the fact that typical business cycles last for 4.5-5 years on average, so it seems prudent to include a time period roughly equal to two business cycles in each train/validation sequence. Within these 9 years, we take a training period of 6 years, and a validation period of 3 years. As we roll this window across the entirety of the overall training set (1995 - 2017), the average performance of each model can be evaluated across validation sets.

**Validation of HMM Pseudo States:** HMM does not assign states based on bull/bear/static as we require, rather it merely identifies distinct states within the data, and does not assign any inherent characteristic to each state. In order to identify which states best coincide with our definition of bull/bear/static states, our validation methodology analyses across all possible combinations of state assignments of the identified pseudo states, and chooses the assignment that maximises the return from the training set based on our trading strategy. We then apply this assignment to the validation set predictions, and pass these assigned market states into our final model methodology for validation.

**Validation of Signal Model Pseudo States:** Using the same training/validation window, we train each signal model on the true labels from the training data, and use this model to obtain validation predictions of the market state. These predictions are passed into our final model methodology for validation.

**Validation of Market State Prediction Models:** In order to obtain predictions of market states, we combine 'pseudo states' obtained from the HMM model with each of the Signal models as outlined in the previous section. We then compare the returns of our trading strategy based on these predicted market states up against a buy-and-hold strategy to test whether our predictions in each validation set outperforms. Mean returns excess of the buy-and-hold strategy are computed, as well as standard deviations of these excess returns, and a Sharpe-Ratio-like metric is computed. These results are tabulated below.

Model 1 outperforms in the most number of validation sets, while Model 3 has the highest Sharpe-Ratio-like metric. Since these are returns in excess of the buy and hold strategy, and on observation it is seen that under-performance to this benchmark does not necessarily indicate poor returns, we believe the Sharpe-Ratio-like metric may be the most ideal validation metric to choose the best performing model.

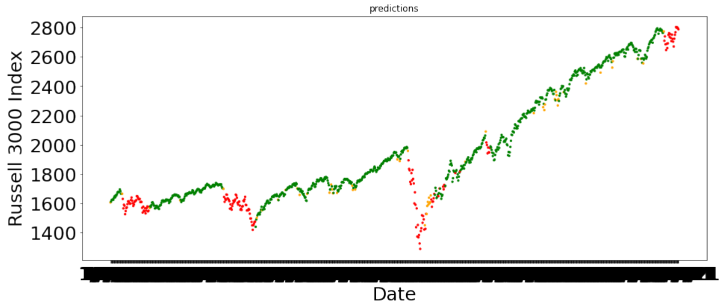


Figure 6: Signal 1

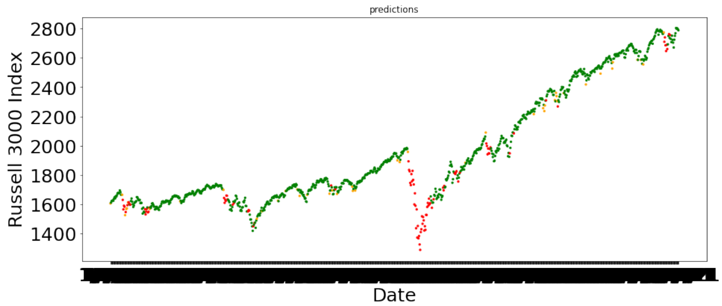


Figure 7: Signal 2

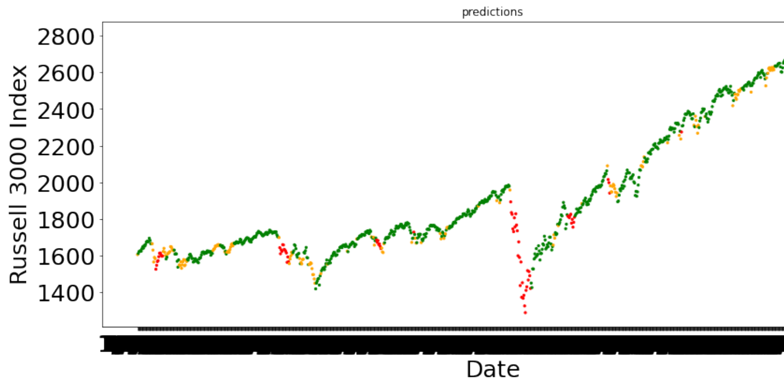


Figure 8: Signal 3

| Validation Results* | Model using Signal 1 | Model using Signal 2 | Model using Signal 3 |
|---|---|---|---|
| Outperformance in Validation Set | 9 | 8 | 7 |
| Mean Excess Return | 14.06% | 15.07% | 20.15% |
| Sharpe Ratio | 0.25 | 0.3 | 0.33 |

*excess returns signifies returns in excess of buy and hold strategy*

Figure 9: Validation Results Comparison

| Data set classification | Cumulative Returns | Buy and Hold | Model using Signal 1 | Trades in Model 1 |
|---|---|---|---|---|
| Train Dataset | 2/2/95 - 12/27/17 | 473.24% | 2655.26% | 429 |
| Test Dataset | 1/3/18 - 12/31/21 | 73.70% | 210.60% | 89 |

| Data set classification | Annualised Returns | Buy and Hold | Model using Signal 1 | Trades in Model 1 |
|---|---|---|---|---|
| Train Dataset | 2/2/95 - 12/27/17 | 8.04% | 15.82% | 0.1 |
| Test Dataset | 1/3/18 - 12/31/21 | 15.42% | 34.24% | 9.0 |

Figure 10: Result Comparison for Model 1

### 4.5 Results

## 5 Validity of Approach

We now test the validity of our model by using the market regimes that we detected to identify buy, sell or hold signals, and comparing the PnLs generated using this strategy with the lower and upper bound of achievable PnL.

The lower bound is obtained using a simple buy and hold strategy on the Russell 3000 index for the 2018

| Data set classification | Cumulative Returns | Buy and Hold | Model using Signal 2 | Trades in Model 2 |
|---|---|---|---|---|
| Train Dataset | 2/2/95 - 12/27/17 | 473.24% | 2655.26% | 429 |
| Test Dataset | 1/3/18 - 12/31/21 | 73.70% | 103.34% | 123 |

| Data set classification | Annualised Returns | Buy and Hold | Model using Signal 2 | Trades in Model 2 |
|---|---|---|---|---|
| Train Dataset | 2/2/95 - 12/27/17 | 8.04% | 15.82% | 0.1 |
| Test Dataset | 1/3/18 - 12/31/21 | 15.42% | 20.25% | 12.4 |

Figure 11: Result Comparison for Model 2

| Data set classification | Cumulative Returns | Buy and Hold | Model using Signal 3 | Trades in Model 3 |
|---|---|---|---|---|
| Train Dataset | 2/2/95 - 12/27/17 | 473.24% | 3086.32% | 504 |
| Test Dataset | 1/3/18 - 12/31/21 | 73.70% | 210.51% | 115 |

| Data set classification | Annualised Returns | Buy and Hold | Model using Signal 3 | Trades in Model 3 |
|---|---|---|---|---|
| Train Dataset | 2/2/95 - 12/27/17 | 8.04% | 16.56% | 0.1 |
| Test Dataset | 1/3/18 - 12/31/21 | 15.42% | 34.23% | 11.6 |

Figure 12: Result Comparison for Model 3

to 2021 period.

The upper bound is obtained by using a strategy that buys when we encounter a bull signal, invests in the T-bills when we encounter a neutral signal, and shorts when we encounter a bear signal.

The following table shows the comparative analysis of Train PnL, Validation PnL and Test PnL for all the different models.

## 6 Conclusion

## 7 Appendix

## References