Adaptive Neuro-Fuzzy Inference Systems(ANFIS): Using Fuzzy Logic and Neural Networks for Financial Modeling

William S. Ventura

Whiting School of Engineering, Johns Hopkins University

EN.605.745: Reasoning Under Uncertainty

Dr. Marc Johnson

January 23, 2023

Author Note

Correspondence concerning this article should be addressed to William Ventura, Whiting School of Engineering, Johns Hopkins University, 3400 N Charles St, Baltimore, MD 21218. E-mail: wventur1@jh.edu

Abstract

The stock market is essential within the global marketplace where trading and investments are the cornerstones of economic prosperity and job creation. It is one of the main bases of capitalism, allowing individuals to invest assets in stocks and in turn companies are able to raise funds from the stock market. However, retail and institutional investors have to analyze large amounts of financial information to understand the standing of a company, market behavior, and predict future market directions in hopes of formulating trading decision that achieve successful returns. Due to the complex nature of market dynamics and high stakes of trading decision especially among institutional investors, there has been increasing research into risk analysis and financial engineering. This paper aims to integrate fuzzy logic and neural networks for predictive market analysis. The hopes of this approach is to model the inter-dependent relationships among market factors, and real-world financial data to forecast future returns based on the results of fuzzy inference. This will be carried out using real world financial data of the Equity Index (S&P 500) from 2015 to 2020.

Keywords: Fuzzy-Logic, Neural Networks, GARCH, ANFIS, financial modeling, stock market, complex system, technical analysis

Introduction

The stock market is essential within the global marketplace where trading and investments are the cornerstones of economic prosperity and job creation. It is one of the main bases of capitalism, allowing individuals to invest assets in stocks and in turn companies are able to raise funds from the stock market. However, retail and institutional investors have to analyze large amounts of financial information to understand the standing of a company, market behavior, and predict future market directions in hopes of formulating trading decision that achieve successful returns. Due to the complex nature of market dynamics and high stakes of trading decision especially among institutional investors, there has been increasing research into risk analysis and financial engineering. This paper aims to integrate fuzzy logic and neural networks for both predictive market analysis. The hopes of this approach is to model the inter-dependent relationships among market factors, and real-world financial data to forecast future returns based on the results of fuzzy inference. This will be carried out using real world financial data of the Equity Index (S&P 500) from 2015 to 2020.

Problem Statement

Given the complex dynamics of the stock market and highs-stakes associated with trading decisions, this paper aims to integrate both neural networks and fuzzy logic principles to model the inter-dependent relationships among market factors and real-world financial data. Fuzzy logic and fuzzy set theory attempt to capture the vagueness of the world and can be used to model financial uncertainty.

Background

Most events experienced in real life are not attributed to a single point in time. They are described through multiple states of observations such that a complete final event is produced. This led to statisticians developing numerous methods for reasoning across temporal relationships, known as time-series analysis. Time-series is described by as a sample realization of a stochastic process that consists of a set of observations made sequentially over time. The stock market is a dynamical system such that inference on potential future events requires the knowledge of previous data in addition to the adoption of new incoming data as the system progresses along time.

Fuzzy logic

Lotfi Zadeh first introduced fuzzy sets in the mid 1960's at the University of California, Berkeley. It was the central concept of what would late be known as fuzzy theory. Unlike in classical set theory, where an element is completely contained by a set or not contained at all, Zadeh's idea proposed that elements can partially belong to sets. The degree of belonging to a set can even be determined using what is called a *membership function*, μ; which takes in any real value between and including 0 and 1 for each element in the universe of discourse (Zadeh, 1965). In contrast to traditional sets, where it is a binary Boolean logic of element $x \in X$ either being in the set $(\mu(x) = 1)$ or not $(\mu(x) = 0)$, each element in a fuzzy set can belong to the set to a certain degree i.e. "Completely Certain" - $\mu(x) = 1$, "Mostly Certain" - $\mu(x) = 0.75$, "Neutral" - $\mu(x) = 0.5$, "Not Really" - $\mu(x) = 0.25$, "Not At All" - $\mu(x) = 0.0$.

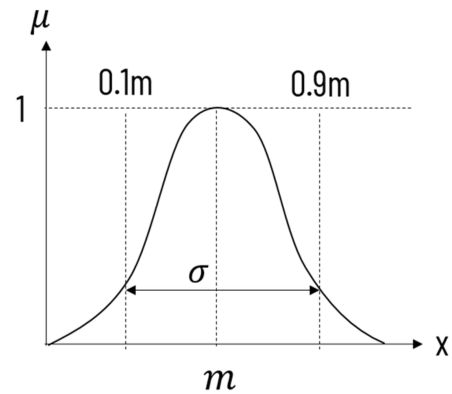
Membership Functions

The fuzzy membership functions are used to convert the crisp provided to the fuzzy inference system. Fuzzy logic itself is not fuzzy, but rather deals with the fuzziness in the data. This fuzziness is best described by the fuzzy membership function. The formal definition of a membership function for a set A on the universe of discourse X is defined as:

$$\mu_a : X -> [0, 1]$$

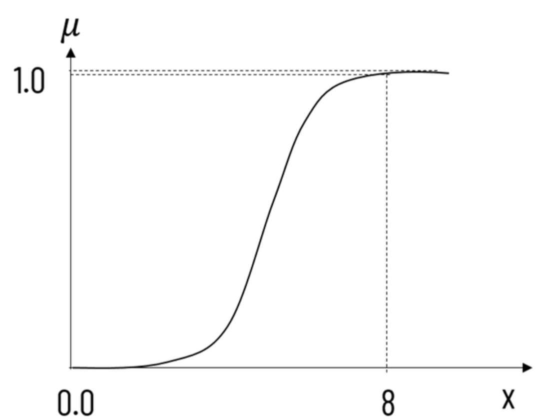
Where each element of X is mapped to a value between 0 and 1, called the *membership value*. There are multiple types of membership functions like Gaussian, Sigmoid, and Generalized Bell shapes defined as such(Talpur et al., 2017)(Kabir & Kabir, 2021):

Figure 1
Gaussian Membership Function



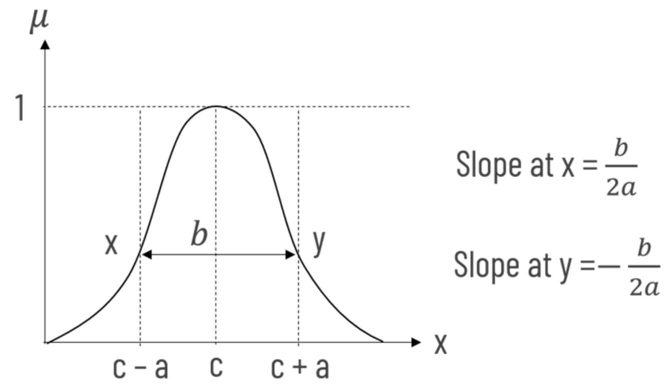
$$\mu_{gaussian}(x; m, \sigma) = e^{-1/2(\frac{x-m}{\sigma})^2}$$

Figure 2
Sigmoid Membership Function



$$\mu_{sigmoid}(x; a, c) = \frac{1}{1 + e^{-a(x-c)}}$$

Figure 3
Generalized Bell Shape Membership Function



$$\mu_{bell}(X; a, b, c) = \frac{1}{1 + |\frac{x-c}{a}|^{2b}}$$

Fuzzy logic has a wide range of applications and is frequently used in machine controllers and artificial intelligence. It allows to mimic human decision making, and is most useful for modeling complex problems with ambiguous or distorted inputs. However there are several drawbacks on fuzzy systems. Since the systems are designed for inaccurate data and inputs, requires broad validation and verification. Fuzzy control systems are also dependent on human expertise and knowledge.

Adaptive Neuro-Fuzzy Inference System (ANFIS)

ANFIS was introduced by Jang in 1993, it integrates both neural networks and fuzzy logic principles, encapsulating the benefits of both in a single framework. The inference system in ANFIS corresponds to a set of fuzzy IF-THEN rules capable of approximating nonlinear functions (Abraham, 2005) and is considered to be a universal estimator (Jang, 1993). It consists of five layers; the fuzzy layer (layer 1), the rule layer (layer 2), the normalization layer (layer 3), the defuzzification layer (layer 4), and the total output layer (layer 5). Layer 1 and 4 are the only layers that contain parameters that can be trained from the given input and output data. These parameters are called the premise and consequence parameters respectively (Karaboga & Kaya, 2019). Karaboga and Kaya, 2019 also states that layers 2, 3, and 5 are non-trainable(fixed) and provide a comprehensive review on ANFIS training. These layers are defined by Lenhard and Maringer, 2022; Talpur et al., 2017 as such:

Fuzzy Layer

The fuzzy layer performs fuzzification of the incoming crisp values, x_1, x_2 by applying the respective membership function to obtain the degree of membership to the respective fuzzy sets A_1, A_2, B_1, B_2. Every node in the fuzzy layer is adaptive and produces the output:

$$O_{1,i} = \begin{cases} \mu_{A_i}(x_1) & \text{for } i = 1, 2 \\ \mu_{B_{i-2}}(x_2) & \text{for } i = 3, 4 \end{cases}$$

x_n is the input to node i and A_m, B_m are the linguistic labels i.e. (Hot, Cold) defined respectively by an appropriate membership function $\mu_{A_i}, \mu B_i$.

Rule Layer

The second layer performs $\prod$ operation on the membership degrees in order to calculate the firing strength, w_i of each rule derived.

$$w_i = \mu_{A_m}(x_1) \cdot \mu_{B_l}(x_2) \text{ for } i = 1, .., n \text{ and } m, l = 1, 2$$

Normalization Layer

The third layer normalizes, N the firing strength of $(\bar{w_i})$ of each rule against all the rules.

$$\bar{w_i} = \frac{w_i}{\sum_{i=1}^{n} w_i} \text{ for } i = 1, .., n$$

Defuzzification Layer

This is another layer where every node is adaptive. This layer receives $\bar{w_i}$ and the inputs, x_1, x_2. It returns the weighted values of each rule's node defined by:

$$\bar{w_i} f_i = \bar{w_i} \cdot (\alpha_{i,0} + \alpha_{i,1} x_1 + \alpha_{i,2} x_2) \text{ for } i = 1, .., n$$

α_i would be the actual consequence parameters.

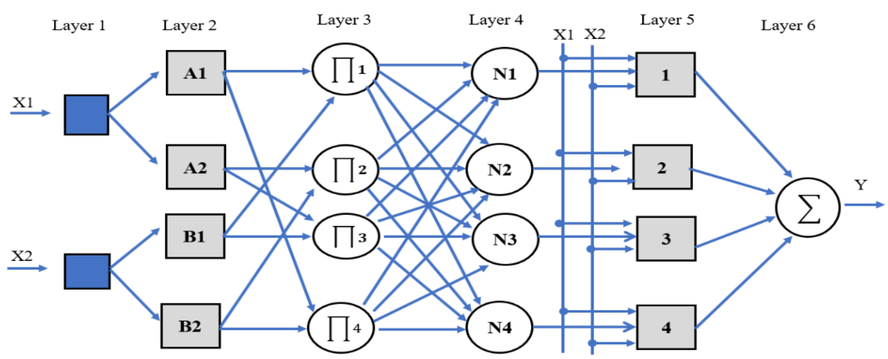
Total Output Layer

This layer produces the final output of ANFIS. It is simply a summation of the outputs of rules calculated in previous defuzzification layer, where i indicates each rule and $\bar{w}_i f_i$ is the output of the defuzzification layer.

$$f = \sum_i \bar{w}_i f_i$$

Figure 4

ANFIS Architecture of a two-input model with four rules

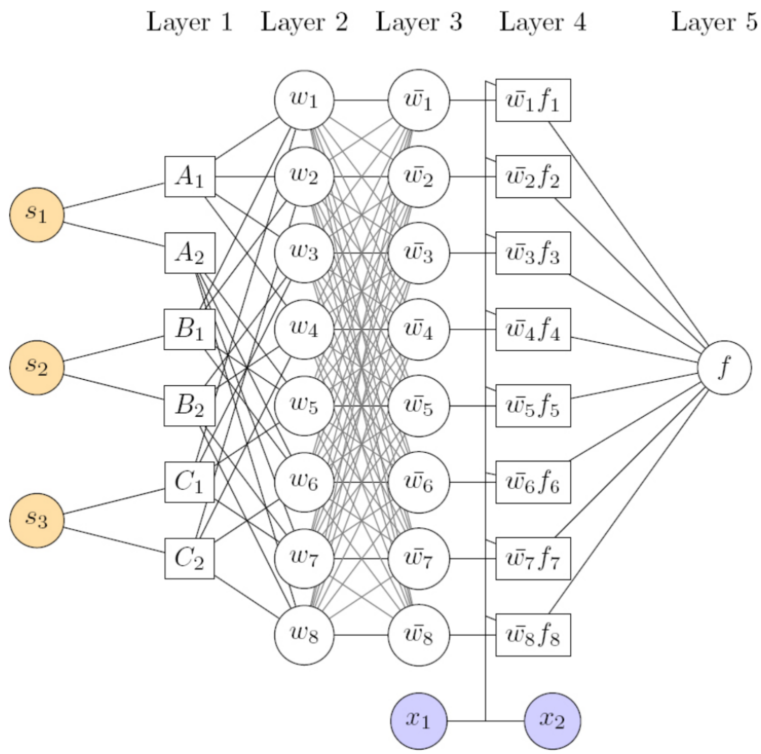


State-ANFIS

Lenhard and Maringer, 2022 proposed State-ANFIS (S-ANFIS), which is a variation of ANFIS. S-ANFIS works by distinguishing between two types of inputs: state variables, s, and explanatory variables x. This model uses a neuro-fuzzy modeling approach in two stages. In the first stage, the premise part, the membership functions of the external state variables are tuned and converted into fuzzy rules. The second part of the model, the parameters of the weighing of the explanatory variables are optimized for each case resulting in a weighted model combination. For more detail about S-ANFIS, see Lenhard and Maringer, 2022.

Figure 5

S-ANFIS Architecture for three state variables, S and two explanatory variables, X



Study Approach

This study will implement and compare ANFIS and S-ANFIS models in their respective ability to explain an individual stock's return, several technical and state factors will be calculated and performance of their ability to fit the data will be analyzed.

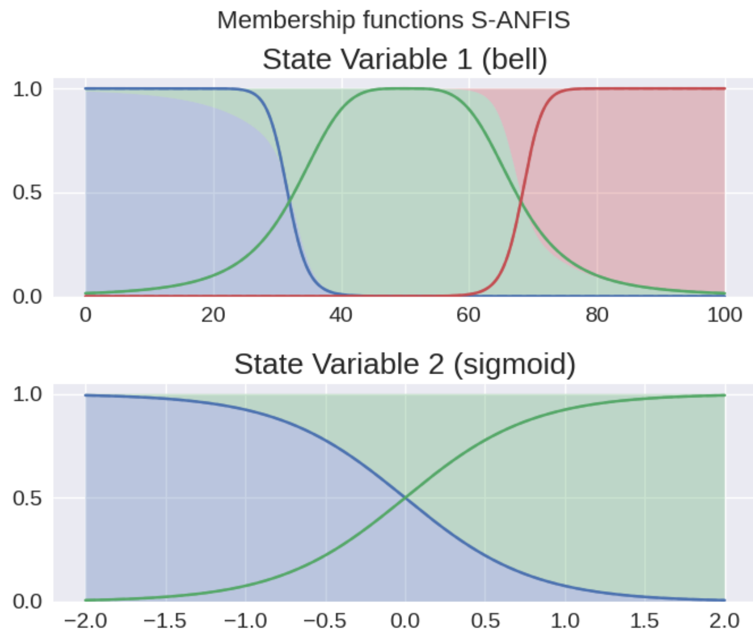
Data

Stock data of the S&P 500 will be retrieved from Yahoo Finance. The selected time frame for the data is a pre-Covid five-year window from January, 1st 2015 to January 1st 2020. The technical parameters calculated from the S&P 500 data will be Residual Strength Index (RSI), Momentum, short and long term Moving Average (MA), and log(Volume). The training data and testing data will be split into a four year training window and one year testing window.

Models

Both models will be composed of 2 membership functions. AFNIS will make use of an initialized bell curve to represent RSI at Oversold, Neutral, and Oversold conditions. The other remaining parameters will be fitted to the Sigmoid membership function respectively. When the models are fitted the hyper-parameters optimized through ADAM.

Figure 6
Membership Functions Initialized



AFNIS

Three ANFIS models are examined, each with two input variables. Model 1 will be made up of the RSI for a 14 day window and $Momentum$ for a 20 day window. Model 2 uses RSI for a 14 day window and $log(Vol)$. Model 3 is made up of $log(Vol)$ and $Momentum$.

RSI: is a momentum indicator that measures the magnitude of recent price changes to analyze overbought ($RSI >= 0.7$) or oversold ($RSI =< 0.3$). Where oversold refers to a condition where an asset has traded lower in price and has the potential for a price bounce and vice verse for overbought.

Momentum: Measures the velocity of price changes. It shows the rate of change in price movement over a period of time to help investors determine the strength of a trend.

Volume: Measures the number of shared traded in a stock. It can also indicate how much strength there is behind a move.

SANFIS

For State-ANFIS models, several state variables are calculated. Those calculated are the short and long term Exponential Weighted Moving Average (EWMA) and Volatility using GARCH(1,1). EWMA is a subset of GARCH(1,1) and is frequently used for estimating volatility of returns. This method of calculated conditional variance (volatility) gives more weightage to the current observations than past ones. It is defined as such:

$$\sigma^2_{ij/t} = \lambda\sigma_{ij/t-1} + (1+\lambda)r_{it-1}r_{jt-1}$$

where r represents the returns, λ is the smoothing constant and ensures that current variance is positively correlated with the previous volatility. It can be rewritten to adjust for window size as such:

$$\sigma^2_{i,t} = \frac{1}{(w-1)}\sum_{j=0}^{w}(r_{i,t-j} - \bar{r}_{i,t})^2$$

$$\bar{r}_{i,t} = \frac{1}{w}\sum_{j=0}^{w} r_{i,t-j}$$

Here w is the window size, set to 20 for the short term volatility and 120 for long term.

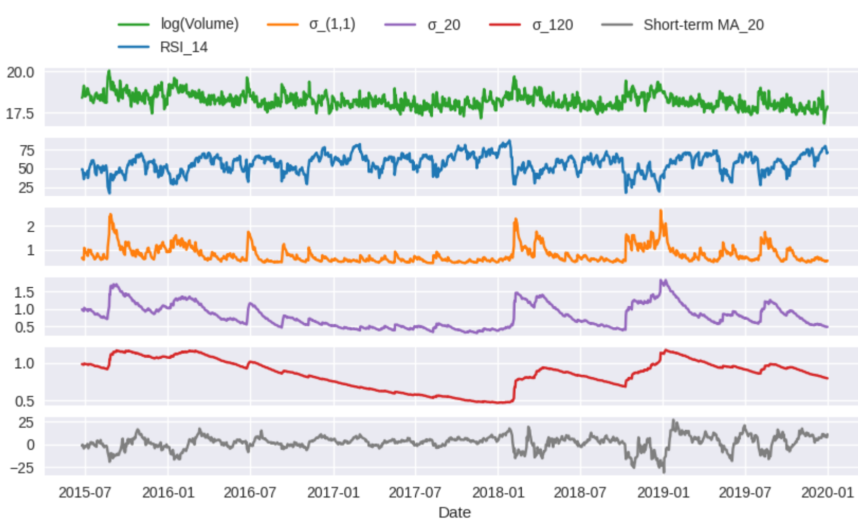
Generalized Autoregressive Conditional Heteroscedasticity (GARCH(1,1)) is another model for estimating volatility. It is derived from Autoregressive Conditional Heteroscedasticity (ARCH), where AR means that the models are auto regressive in squared returns. Conditional means that the next period's volatility is conditional on the information available in the current period. Heteroscedasticity means non-constant volatility which means the time series of a random variable has a time-varying variance. The generalized version GARCH can account for different factors in different markets, the most common for is GARCH (1,1) which is defined as:

$$\sigma_t^2 = \omega + \alpha\epsilon_{t-1}^2 + \beta\sigma_{t-1}^2$$

$$\omega > 0, \alpha, \beta >= 0$$

α is the weight for the lagged squared returns, β is the weight for the lagged variances, ω is a constant equal to $\gamma \cdot VL$, where VL is the long run variance rate and γ is its weight. The key component of GARCH(1,1) is that (1,1) indicates a -1 lag for each squared return and squared variance of the previous day. Model 4 is composed of $\sigma_{(1,1)}^2$ and σ_{20}^2, and Model 5 uses $\sigma_{(1,1)}^2$ and σ_{120}^2.

Figure 7
State Variables



Results

The performance results for models application on the S&P 500 are given in Table 1. The models that seem to best fit the S&P 500 data where Model 1 and Model 2 where the RMSE was equal to 0.78848267, and 0.79683 respectively. While there was no single best combination of variables that performed best, it does suggest that RSI is a good explanatory variable. The RSI membership function was initialized with 3 three members that would translate the conditions of an asset being Oversold, Neutral, and Overbought.

Table 1
Model's RMSE

Results	
Model	RMSE
Model 1 (RSI_{14}, $Momentum_{20}$)	0.78848267
Model 2 (RSI_{14}, $log(Vol)$)	0.79683
Model 3 ($log(Vol)$, $Momentum_{20}$)	0.81268775
Model 4 ($\sigma^2_{(1,1)}$, σ^2_{20})	1.2752211
Model 5 ($\sigma^2_{(1,1)}$, σ^2_{120})	0.8575762

The Figures section in the appendix has the fuzzy membership functions after the model has been fitted and also the performance for the models. One thing to notice is the that RSI membership curve for model 1 and 2 never reaches below 30 RSI which would mean that the asset is oversold and that there could be a potential bounce upwards. When looking at the predictions below, the model seems to fit the lower component which would be more representative of the higher values RSI, but fails to fit on the upwards movement.

Conclusion

A fuzzy logic approach allows for the inclusion of vague human assessments in computing problems. It involves intermediate possibilities between Boolean values and allows for flexible reasoning. In effect, fuzzy sets allow for elements to belong to a given set with a certain degree of membership to that set. ANFIS and S-ANFIS are methods of applying fuzzy systems with components of an artificial neural network to fit a time series data. The models here show that it is possible to fit linguistic values to variables of an asset. While performance showed that the models weren't able to fully fit the data, the complexity of the membership functions can be furthered increased by an outside expert to allow for a better fit.

References

Abraham, A. (2005). Adaptation of fuzzy inference system using neural learning. In N. Nedjah & L. d. Macedo Mourelle (Eds.), *Fuzzy systems engineering: Theory and practice* (pp. 53–83). Springer Berlin Heidelberg. https://doi.org/10.1007/11339366_3

Jang, J.-S. (1993). Anfis: Adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man, and Cybernetics*, *23*(3), 665–685. https://doi.org/10.1109/21.256541

Kabir, M., & Kabir, M. M. J. (2021). Fuzzy membership function design: An adaptive neuro-fuzzy inference system (anfis) based approach. *2021 International Conference on Computer Communication and Informatics (ICCCI)*, 1–5. https://doi.org/10.1109/ICCCI50826.2021.9402633

Karaboga, D., & Kaya, E. (2019). Adaptive network based fuzzy inference system (anfis) training approaches: A comprehensive survey. *Artif. Intell. Rev.*, *52*(4), 2263–2293. https://doi.org/10.1007/s10462-017-9610-2

Lenhard, G., & Maringer, D. (2022). State-ANFIS: A generalized regime-switching model for financial modeling. *2022 IEEE Symposium on Computational Intelligence for Financial Engineering and Economics (CIFEr)*, 1–8. https://doi.org/10.1109/CIFEr52523.2022.9776208

Talpur, N., Salleh, M. N. M., & Hussain, K. (2017). An investigation of membership functions on performance of anfis for solving classification problems. *IOP Conference Series: Materials Science and Engineering*, *226*(1), 012103. https://doi.org/10.1088/1757-899X/226/1/012103

Zadeh, L. (1965). Fuzzy sets. *Information and Control*, *8*(3), 338–353. https://doi.org/https://doi.org/10.1016/S0019-9958(65)90241-X

Appendix

Figures

Figure A1

Separation of Training and Testing

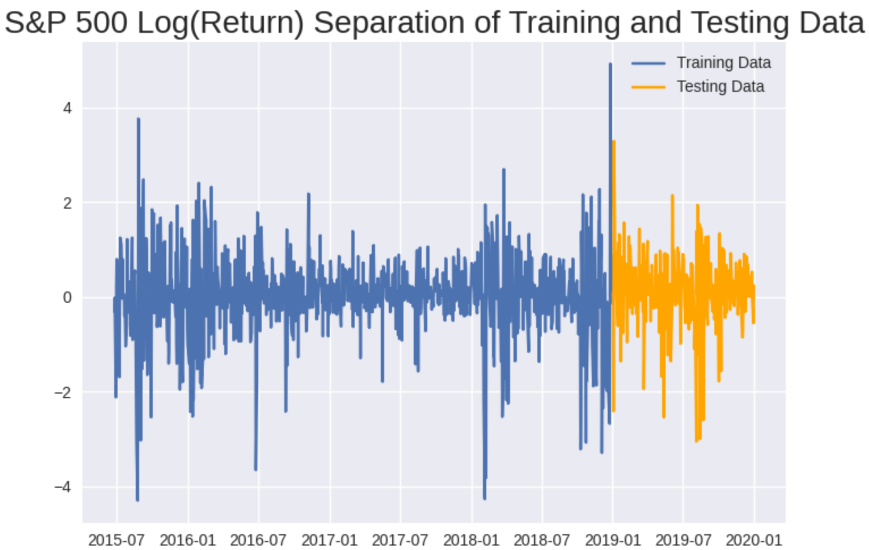


Figure A2

Membership Functions Initialized

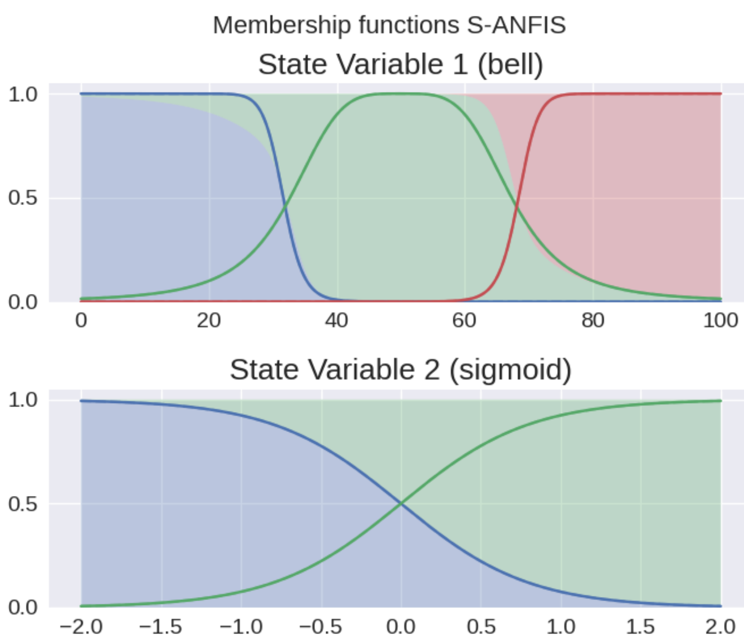


Figure A3

Model 1 Membership Function

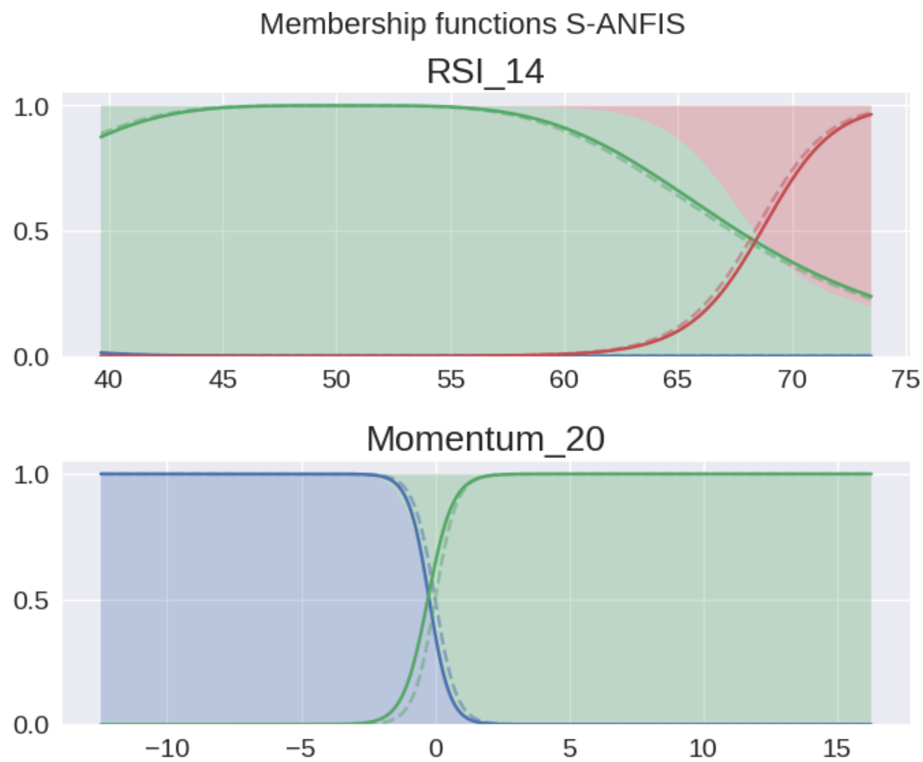


Figure A4
Model 1 Performance

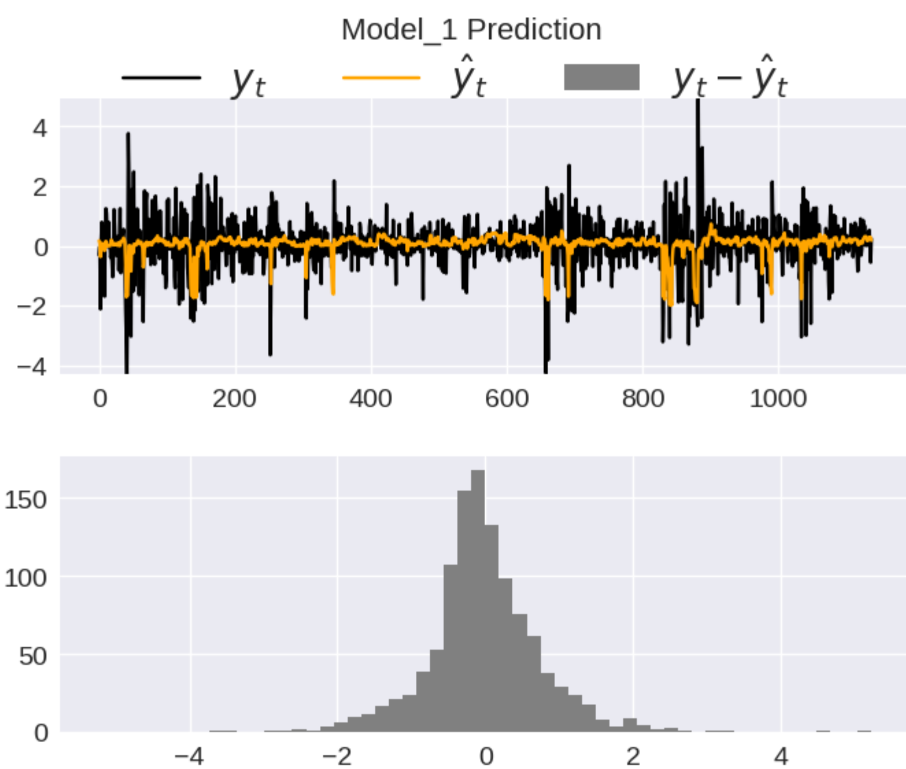


Figure A5
Model 2 Membership Function

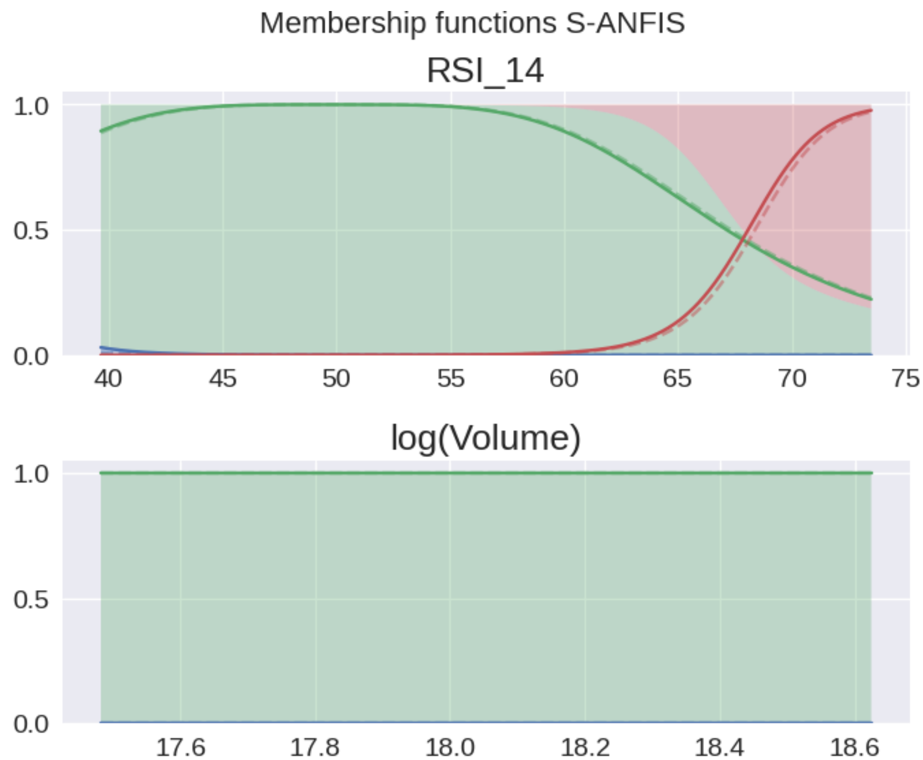


Figure A6
Model 2 Performance

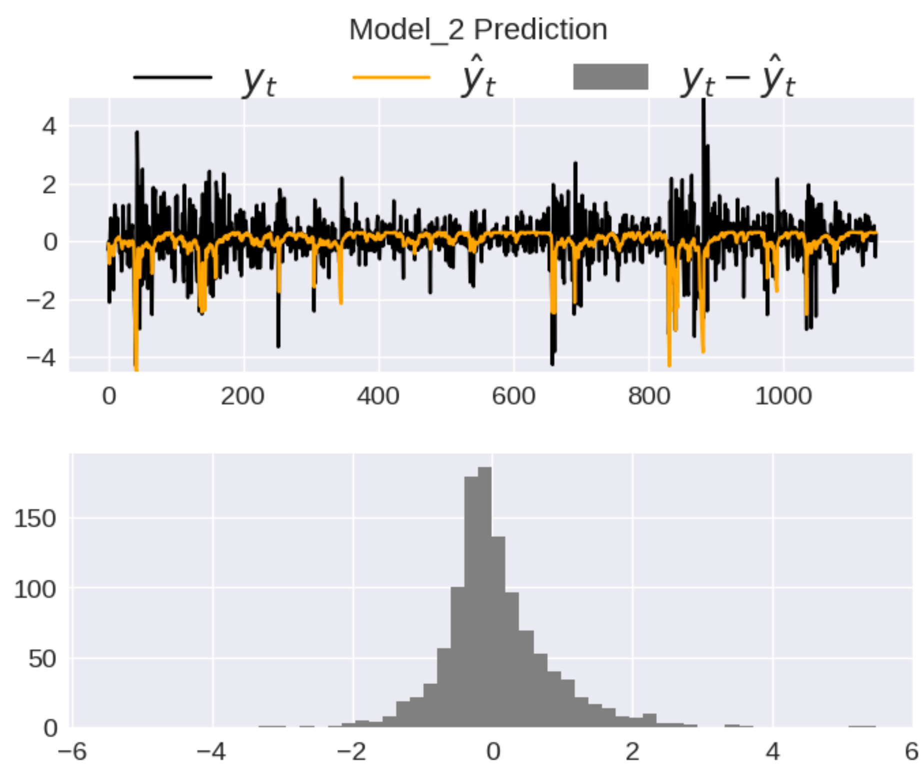


Figure A7

Model 3 Membership Function

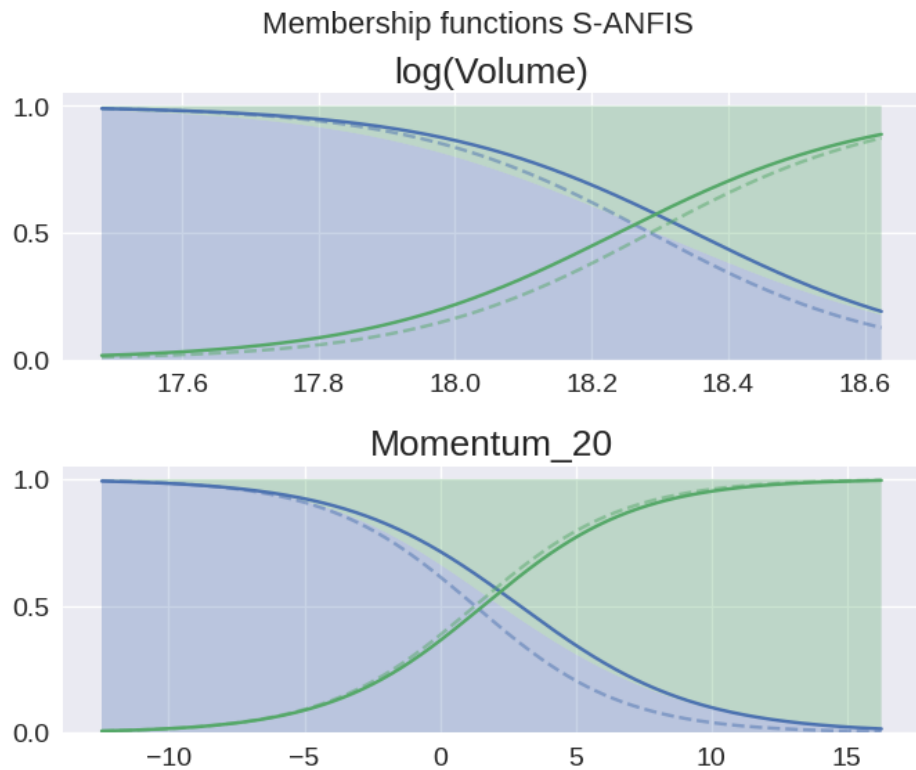


Figure A8

Model 3 Performance

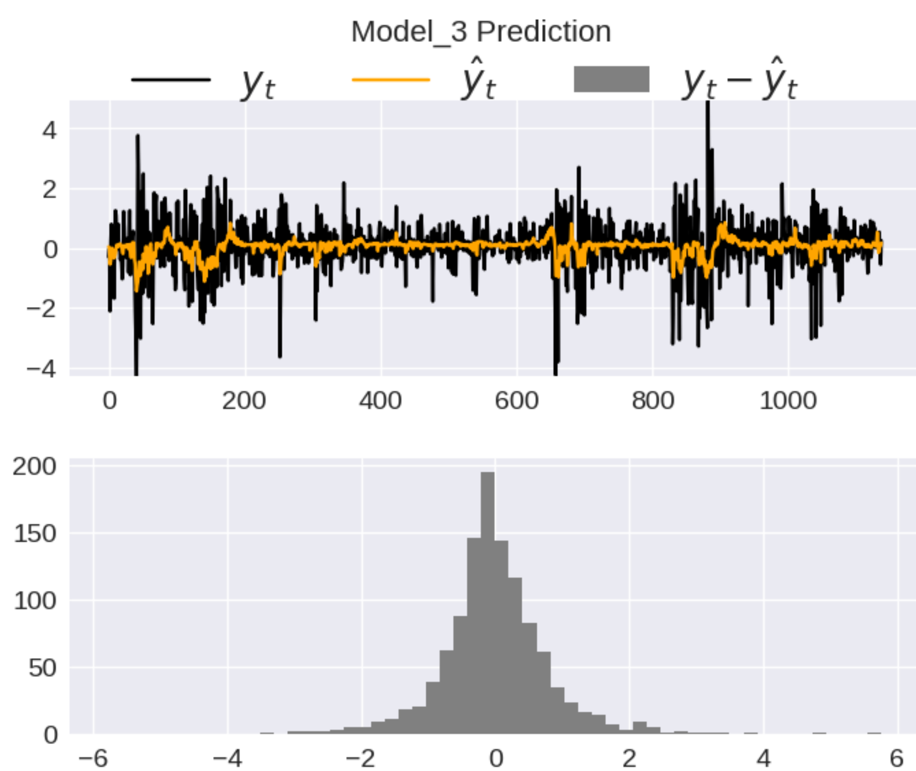


Figure A9

Model 4 Membership Function

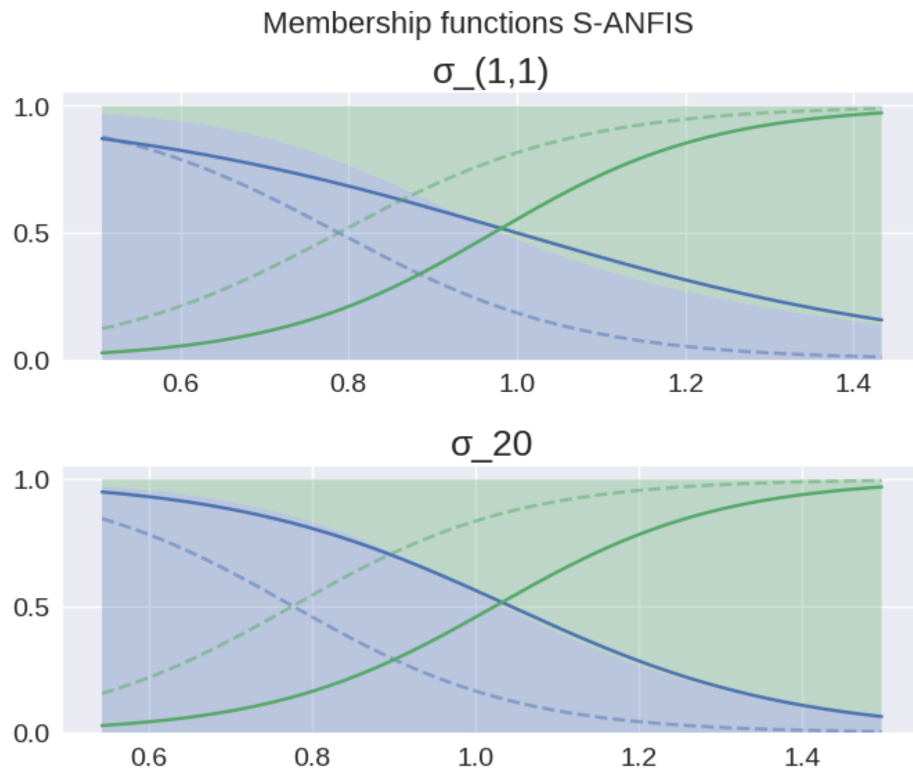


Figure A10

Model 4 Performance

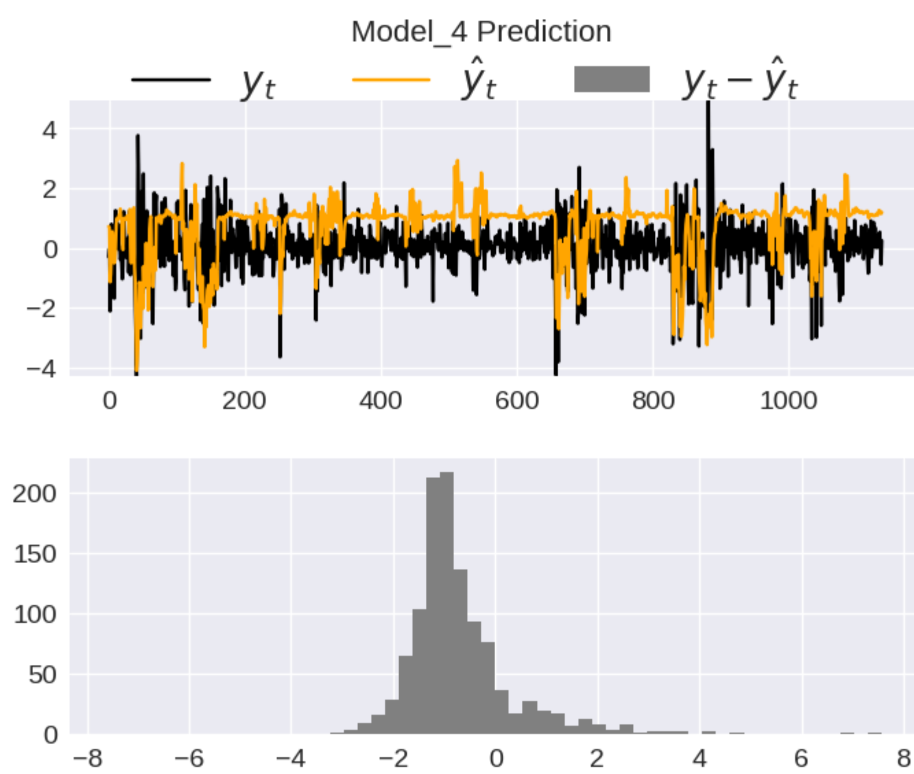


Figure A11

Model 5 Membership Function

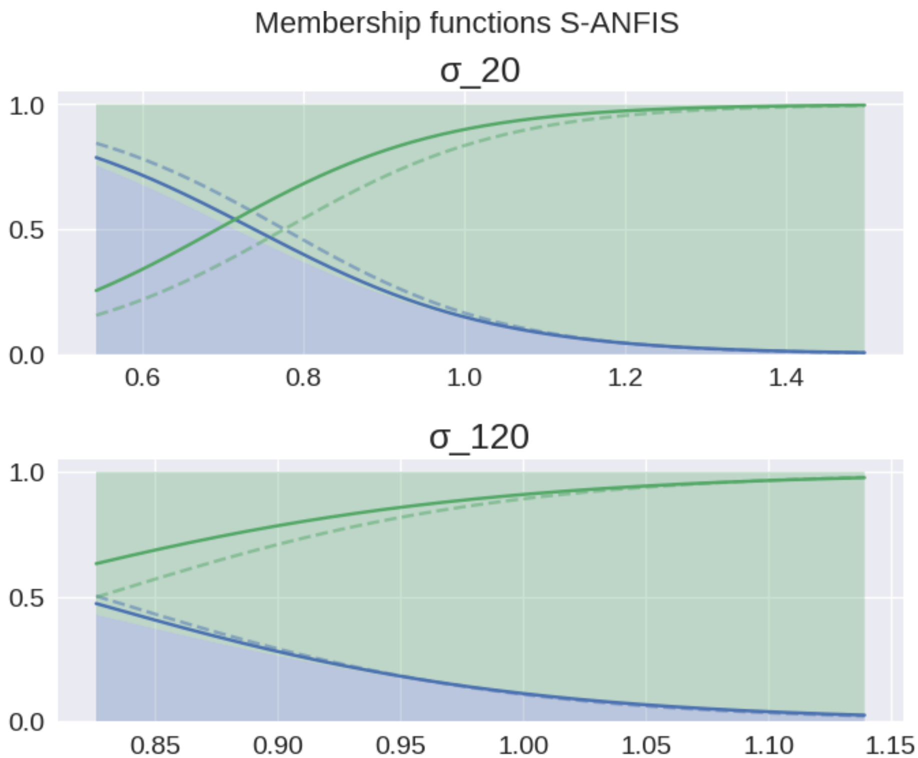


Figure A12

Model 5 Performance