

Financial Asset Management Using Artificial Neural Networks

Roohollah Younes Sinaki, Ohio University, USA

 <https://orcid.org/0000-0001-5358-3711>

Azadeh Sadeghi, Ohio University, USA

Dustin S. Lynch, Ohio University, USA

William A. Young II, Ohio University, USA

Gary R. Weckman, Ohio University, USA

 <https://orcid.org/0000-0002-2445-4934>

ABSTRACT

Investors typically build portfolios for retirement. Investment portfolios are typically based on four asset classes that are commonly managed by large investment firms. The research presented in this article involves the development of an artificial neural network-based methodology that investors can use to support decisions related to determining how assets are allocated within an investment portfolio. The machine learning-based methodology was applied during a time period that included the stock market crash of 2008. Even though this time period was highly volatile, the methodology produced desirable results. Methodologies such as the one presented in this article should be considered by investors because they have produced promising results, especially within unstable markets.

KEYWORDS

Artificial Intelligence, Artificial Neural Networks, Asset Allocation, Decision Making, Financial, Machine Learning, Moving Average, Portfolio, Stocks

INTRODUCTION

A typical investment portfolio contains four main asset classes. The four main asset classes are U.S. stocks, bonds, international stocks, and hedge positions. Each of these asset classes has multiple stock indices. A successful investment portfolio is determined by the total return on investment over the course of the portfolio. The strategy of creating investment portfolios will vary depending on several factors. Likewise, the evaluation of the success of a portfolio can also vary by client. For example, a conservative client might be satisfied with a 5% return on investment, while an aggressive investor might not be satisfied with a 7% return on investment. The portfolio is an investment in the owner's future; therefore, the return over a long time period is a better determination of the portfolio's success. The time period that is often used to evaluate the performance of a portfolio is five years.

DOI: 10.4018/IJORIS.2020070104

This article, originally published under IGI Global's copyright on July 1, 2020 will proceed with publication as an Open Access article starting on February 1, 2021 in the gold Open Access journal, International Journal of Operations Research and Information Systems (converted to gold Open Access January 1, 2021), and will be distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

The diversification of a portfolio is an important part of deciding whether a portfolio is performing well. Although many strategies exist, in general, portfolios that are more diverse are considered more resilient to changes in the market and therefore, they are more desired. Another determination of the success of a portfolio is its variance. As was explained earlier, this will change depending on the owner of the portfolio. Investors range from being risk-averse to those who seek risk when it comes to their preferred investment strategy. If the variance in a portfolio is high, the risk is high, which means that there is an increased probability that the return on investment will be minimal or produces a loss. The amount of risk often depends on the age of the portfolio's owner. Given the risk associated with this type of portfolio, it is often difficult to satisfy the owner of an investment. For example, if the owner has a low-risk tolerance, the client might not be satisfied with the construction of a highly volatile portfolio even it is producing a positive return. Some investors are simply not satisfied with the drastic changes in the portfolio's performance over short periods of time even if the intent is to build a long-term positive gain. For these types of clients, a portfolio that shows a lower positive rate of return with a lower variance might be a better fit for the client's risk tolerance and age. Although most financial advisors would disagree with a risk-averse strategy and recommend a more risk-seeking strategy, ultimately, the final decision resides with the owner of the portfolio.

The traditional approach allocates assets for a diversified investment portfolio consists of a risk-tolerance questionnaire, a life stage assessment, a portfolio-objective-guidance-matrix, and finally, allocation judgments made by a financial advisor with the clients' approval. These questions help the client and advisor decide the best path for the client while keeping the risk at an acceptable level. It will help the advisor determine if the client is willing to accept a higher or lower risk/volatility of returns of their investment. The biggest limitation of the traditional approach is that stock indices are difficult to predict, due to volatile behavior and very complex interaction of multiple variables. The stock market is a complex system and has non-linear behavior making it exceedingly difficult to predict.

This research presented in this article has multiple objectives: first, to show that artificial neural networks (ANNs) are capable of predicting real-world occurrences, and secondly, to provide a decision support aid that financial advisors can consult when determining how assets are allowed when constructing a client's investment portfolio. Many financial firms currently do not use machine learning-based systems because they consider the market to be unpredictable due to its high volatility. The research presented in this article will demonstrate that investors should consider decision support systems built upon machine learning techniques because they are capable of producing high rates of returns even during times when the market is highly volatile.

LITERATURE REVIEW

The goal of creating a diversified portfolio is to perform well in bull and bear markets. A bull market is one in which prices are expected to rise, while in a bear market, prices are expected to decline at a rate of 20% or more (Encyclopedia Britannica, 2015). A market decline of 10-20% is considered a "correction" and investors have different strategies during these periods of time. In bull markets, it is common for investors to purchase securities and commodities in the hope that they can sell their investments for a profit in the future. In a bear market, on the other hand, the investor commonly sells their investments in the hope that they can buy them back later at a lower price. In both markets, there are always examples of clients looking to make quick profits, but diversification has proven to be the most consistent strategy that is used by financial advisors.

Technical trading and risk tolerance can take on multiple definitions, depending upon the source. Risk tolerance will vary from person to person and is an important measure when tied to an investment portfolio. An investment portfolio comprises a client's investments and is a crucial part of their future. Investments are money invested by the client, and therefore, quickly become important. Big swings or shifts in the portfolio's value can be hard for clients to accept; therefore, a measure of risk/variance is an important measure for a firm to consider for their clients.

Many different firms have used risk measurement strategies. Niko Canner in 1997 examined how investors' attitudes toward risk should influence the composition of their portfolios. His research offered a simple answer to the question by using the mutual-fund separation theorem, which states that all investors should hold the same composition of risky assets (Canner, Mankiw, & Weil, 1997). Most financial advisors recommend that the more conservative investor's portfolio should consist of a higher ratio of bonds-to-stocks, while a portfolio of a more aggressive investor should consist of a lower ratio of bonds-to-stocks.

Advisors are trying to help their clients optimize their portfolios, but their advice often contradicts economic theory and is very hard to explain between firms and advisors. With that said, more non-traditional methods are being developed and explored within the financial sector in an effort to help both clients and advisors. Although there are many brokerage firms that maintain traditional approaches, some are utilizing newer technologies and predictive methods. Charles Schwab is a brokerage and banking company that operates in four main divisions: investing, wealth management, banking, and trading (Charles Schwab, 2015). The company is based out of San Francisco, California, and has been in business since 1971. Charles Schwab Corporation added an automated investment portfolio service in 2015. Their technology aims to diversify the client's portfolio across many different asset classes while experts monitor the performance of their clients' portfolios. Their technology also rebalances portfolios so that they remain balanced and diversified appropriately in regard to the client's profile.

Artificial neural networks (ANNs) are primarily used only in academia because of their ability to model complex nonlinear systems, but some are beginning to use them for solving more problems in industry (Young & Weckman, 2009). As Young (2009) states, they are not commonly used in industry because they are hard to implement, usually not understood very well, and have a reputation of being "black-box" models. Although there are few mathematical models that exist that can outperform ANNs, most industries have been reluctant to accept them for solving practical problems in business (Young & Weckman, 2009).

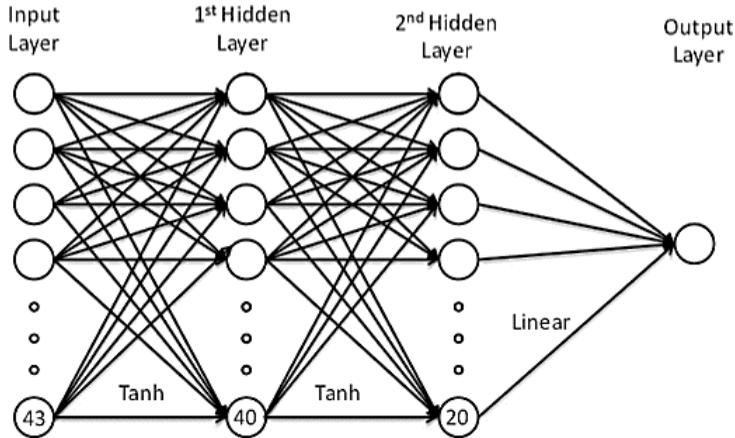
ANNs are nonlinear models consisting of various combinations of simple linear and nonlinear functions that are similar to biological neural networks in the human brain. The integration of mathematical functions enables these types of models to mimic the ability of a human brain in order to solve solving difficult, non-linear problems. ANNs are trained with historic data that is processed through various learning algorithms within the network structure. To many, ANNs are considered "universal approximators" (Neural Networks, 1989) that can learn any pattern hidden within the data being analyzed as long as the sample size of data is rather large and the structure of the ANN is reflecting the complexity of the data being modeled. One of the more significant advantages of using ANNs is that they do not require formal assumptions about data being modeled, which is unlike more traditional models derived from probability and statistics (Skapura, 1996).

The predictive quality and usability of ANNs depend on the quality of data being used to train the model. As with any business application, if the quality of information is high, the quality of the results produced by an ANN will also be higher. In other words, having subject domain experts help to decide what information is used to develop an ANN is always preferred. ANNs have a unique ability to overcome the limitations of noisy data (Burney, 1997). As noted, it is always preferred to have subject matter experts help determine what data is being used to train the model, however; it is not an absolute requirement although some caution practitioners that unwanted bias can be introduced if a model does not consider best practices within predictive analytics (Weckman, et al., 2016). The time required to process an ANN use to be a concern. However, in recent times, the time to develop a process is not a major limitation of their development. ANNs have almost always had the ability to model complex systems and in more recent times, they are capable of producing highly accurate results with a large dataset, which is why ANNs are becoming more popular for real-world business applications.

From a mathematical perspective, there are many ways that synapses can occur inside an ANN. Synapses are simply the ability to connect inputs to neurons, neurons to other neurons, and neurons to

the output layer of an ANN. Neurons act as linear or non-linear transfer functions that enable ANNs to model complex data. The most common ANN topology is the multi-layered perceptron (MLP) model. The MLP is considered feed-forward networks that are usually trained with backpropagation algorithms. The architecture of a typical MLP is shown in Figure 1. This ANN includes two hidden

Figure 1. MLP ANN (Akinwale et al., 2009)



layers, 40 neurons (i.e. processing elements) in the first hidden layer, 20 neurons in the second hidden layer, and one output neuron in the last layer. It should be noted that in this particular network, the first two hidden layers utilize a hyperbolic tangent function (i.e. Tanh) neuron while the output neuron utilizes a linear transfer function.

ANNs almost always outperform traditional linear and even other non-linear methods. They produce excellent results for pattern recognition problems through deep learning and preform particularly well with noisy data. ANNs typically perform better than linear methods in terms of predicting and classifying data because systems generally behave in non-linear ways. Principe stated that they provide excellent results for problems across many different fields and categories (Principe, Euliano, & Lefebvre, 1999). For these reasons, Qian (2007) suggested that ANNs are “ideal for stock market prediction.”

ANNs and other machine learning techniques are not new when it comes to being used to predict stock market outcomes. Researchers have used them to predict turns in individual stocks and bonds for a number of years. One such example is the work of Chiang (1996), who has used ANN methods to forecast the end-of-year net asset value (NAV) of mutual funds. The method uses ANNs and shows that they significantly outperform regression models in situations where data is limited due to its availability. Their ANN consists of one input layer of 15 inputs and one hidden layer of 20 neurons, with the output being the NAV of a mutual fund at the year’s end. The method utilized a total of 101 datasets and used stepwise regression in SAS to choose the most significant variables for each model. The datasets trained on a range from 1981 to 1985 and were tested on the actual data from 1986. The ANN performed 40% better than the linear regression models (Chiang, 1996).

Vaisla compared linear regression to an ANN and found that the regression models significantly underperformed with respect to the ANNs that were developed (Vaisla, 2010). Akinwale, Arogundade, and Adekoya (2009) also compared regression analysis to the predictive ability of an ANN when predicting Nigeria stock market prices and made similar conclusions.

Many articles and conference papers have been written about ANNs being used to predict portions of the stock market. The majority of articles found in literature attempt to predict the movement of

indices, which are composed of many individual stocks. Although this has been a popular use of ANNs, there are other methods that take a broader approach to utilizing ANNs to predict individual stock prices. Weckman et al. (2008) used an ANN model to predict individual stock prices. In addition, Lakshminarayanan (2005) and Hui (2000) also used ANNs to do the same.

The data for this project was a relatively small dataset, which means that additional effort was performed to better understand and improve the prediction performance of the model. Data sets with a limited sample size is a common challenge when developing time series models. This often presents a significant challenge to create useful models for this particular problem domain. This is especially true for ANNs because they typically require a substantial amount of data to “learn” the data properly. Simply put, ANNs normally require a large amount of data in order to discover the underlying patterns of the data that map inputs to the desired output through the machine learning paradigms that are used during the training process. However, methods have been established in order to overcome this limitation. For example, Chawla (2002) developed a popular method called the synthetic minority over-sampling technique (SMOTE). SMOTE was designed in a way that combined the use of random under-sampling of the majority class with the use of random over-sampling of the minority class (Chawla, 2002). This method oversamples the minority class by creating synthetic data examples along with the original data, rather than oversampling that minority class with replacement (Suvarna, 2013).

SMOTE is a method that generates synthetic data examples from the original data in order to improve predictive accuracies in minority classes. SMOTE commonly yields the best results when it comes to re-sampling and can create a larger dataset so that an ANN can learn patterns within datasets that are limited. SMOTE introduces synthetic data along with line segments by joining classes to its nearest neighbor (Chawla, 2002). The SMOTE technique essentially creates a new class sample by interpolating between several class examples that are close to one another. It does this in a “feature space” rather than a “data space.” Depending on the amount of synthetic data needed, neighbors from the k nearest neighbors are randomly chosen. Determining the way in which the nearest neighbors are selected is considered a limitation of SMOTE. However, Chawla stated that focusing on the nearest neighbor examples that were classified incorrectly could potentially improve the SMOTE method and overall performance of the technique (Chawla, 2002).

METHODOLOGY

Data Collection

Data is needed to train ANNs in order to predict the different asset classes’ performances and trends. The data that was used in the creation of the dataset used for this research consisted of economic data that was used to predict the percent variation of each of the four main asset classes for commonly used within investment portfolios. This data includes factors that affect the economy, such as the unemployment rate, gross domestic product (GDP), the political party in the house, senate, and the resident, just to name a few. A typical example of one of the indices that were examined in this methodology is the large-cap growth index and the percent return for the years 1995 to 2009. It is easy to see that the index varies significantly; during the market crash of 2008, the percent return was -38.44%.

The data associated with the financial market seems to be everywhere, but the data that actually affects how the market acts are more difficult to obtain. The performance of how well each index is performing is reported through the day, but mainstream news outlets often fail to report how well the performance of each of the four asset classes (i.e. U.S. stocks, international stocks, hedges, and bonds) perform throughout the day or even on a daily basis. Thus, one challenge that this research faced was aggregating the data of these four asset classes together so that it could be used to build a predictive model. In order to overcome this obstacle, Morningstar Principia software was used in order to retrieve the desired data for the four asset classes. Morningstar Principia has been a trusted resource for financial professionals for more than 15 years (Morningstar Principia, 2015) and the software

provided historical month-to-month performance data for each of the four different asset classes and their historic indices. Table 1 lists the major classes and indices used within the presented research.

The premise of this research is to provide an accurate forecast of an asset class performance so that financial advisors can use this information in order to construct an investment portfolio with higher

Table 1. Asset classes and indices

Bonds	Hedges	International Stocks	U.S. Stocks
Corporate Short Term	Cash	Emerging Markets	Large Cap Growth
Government Short Term	Commodities	Growth	Large Cap Value
Government Intermediate	Precious Metals	Value	Mid Cap Growth
Government Long Term	Real Estate		Mid Cap Value
High Yield Corporate			Small Cap Growth
International			Small Cap Value
Municipal Intermediate			
Municipal Long Term			
Municipal Short Term			

rates of return on investment for their clients. For this research, the output of the ANN was the percent return on a given investment. Thus, the output was numerical, and the variance of this prediction can help determine if the model produced reliable forecasts. This is because one determination of a good investment portfolio is low volatility. The inputs to the model included both economic factors and political parties. From a data pre-processing standpoint, the historic data was aggregated to the point where it would have been known when a forecast would need to be made. A summary of the inputs used in the ANN developed for this research is listed below:

- Political Party
- Effective Federal Funds Rate (%)
- Corporate Profits After Tax (\$B)
- Consumer Price Index-Urban Wage Earners and Clerical Workers
- Industrial Production Index 2007=100
- Consumer Sentiment Index 1966=100
- Unemployment Rate (%)
- Civilian Unemployment Rate
- Consumer Price Index for All Urban Consumers: All Items
- Crude Oil Prices: West Texas Intermediate (WTI)-Cushing, Oklahoma
- Gross Federal Debt
- Housing Starts: Total: New Privately Owned Housing Units Started
- ISM Manufacturing: PMI Composite Index
- M2 Money Stock
- 30-Year Conventional Mortgage Rate
- Producer Price Index: All Commodities
- Real Retail and Food Services Sales
- Total Construction Spending
- Total Vehicle Sales

- Trade Balance: Goods and Services, Balance of Payments Basis
- University of Michigan: Consumer Sentiment
- Real Gross Domestic Product
- S&P/Case-Shiller U.S. National Home Price Index
- Gross Domestic Product
- New Privately-Owned Housing Units Started
- Initial Claims
- Trade Weighted U.S. Dollar Index: Broad

The data found within the Morningstar Principia software was quite extensive, however; at times, the data was incomplete. In other words, missing data was present within the data collected from the software. Since completed records were needed for the ANN, additional resources that are commonly available from simple Internet searches were used in order to complete each and every record where missing data was present during the 1995 to 2009 time period used in this investigation. When the missing data could not be found from simple Internet searches, the Federal Reserve Bank of St. Louis' Economic Research site (2014) was utilized to complete data records. Ultimately, the information collected online, through the Morningstar Principia software, and the data from the Federal Reserve Bank of St. Louis' Economic Research were merged together into a single dataset that was later used to develop and validate the ANN presented in the following sections.

Forecast Time Period

With any predictive model, more data is generally preferred, but unfortunately, only 20 years of data were considered complete from the two primary resources that were used to construct the datasets for this research.

From the 20 years of data, forecasts utilized both a three-month and six-month moving average for each input attribute. Moving averages are a "succession of averages of data from a time series, where each average is calculated by successively shifting the interval by the same period of time" (Dictionary, 2015). In addition to the moving averages, the data was also lagged in order to predict a three-month (i.e., one quarter) window. For example, the data was trained so that the December 1991 output was used to predict March of 1992, then the January 1992 output was used for April 1992, and so on. These moving averages were then added to the database in order to predict the performance of the different asset allocations.

The first attempt in this research was to predict only the average of the four asset classes. ANNs were created and developed for each of the four asset classes and then examined based upon their predictive performances. An example of the variation of values for U.S. Stocks is shown in Figure 2. The figure is provided to simply highlight the complexity of the data being modeled. The data shows some very minor, yet inconsistent seasonality trends with shifting variations through the time series.

Since the variation of each asset's average values was rather large, the models generated for this data were poor. After this experiment was performed, the prediction of a three-month moving average of the output was examined. The three-month moving average was then calculated and added as a new column in the dataset. The ANNs were compared to predict the three-month moving average. The variation of the three-month moving averages for the U.S. Stocks is shown in Figure 3. In comparison to the individual averages found for each asset class, the three-month moving average reduced the overall variation, which results in a more accurate model to be created.

Even though the three-month moving average produced results that outperformed the overall average, the modification did not lead to a statistically significant change in predictive performance measures. Figure 4 shows an example of the six-month moving average for the U.S. Stock class. In comparison to the prior attempts, the six-month moving average showed less variation, which in return, allowed for a more accurate model of the performance of an asset to be created.

Figure 2. U.S. stock monthly variation in value

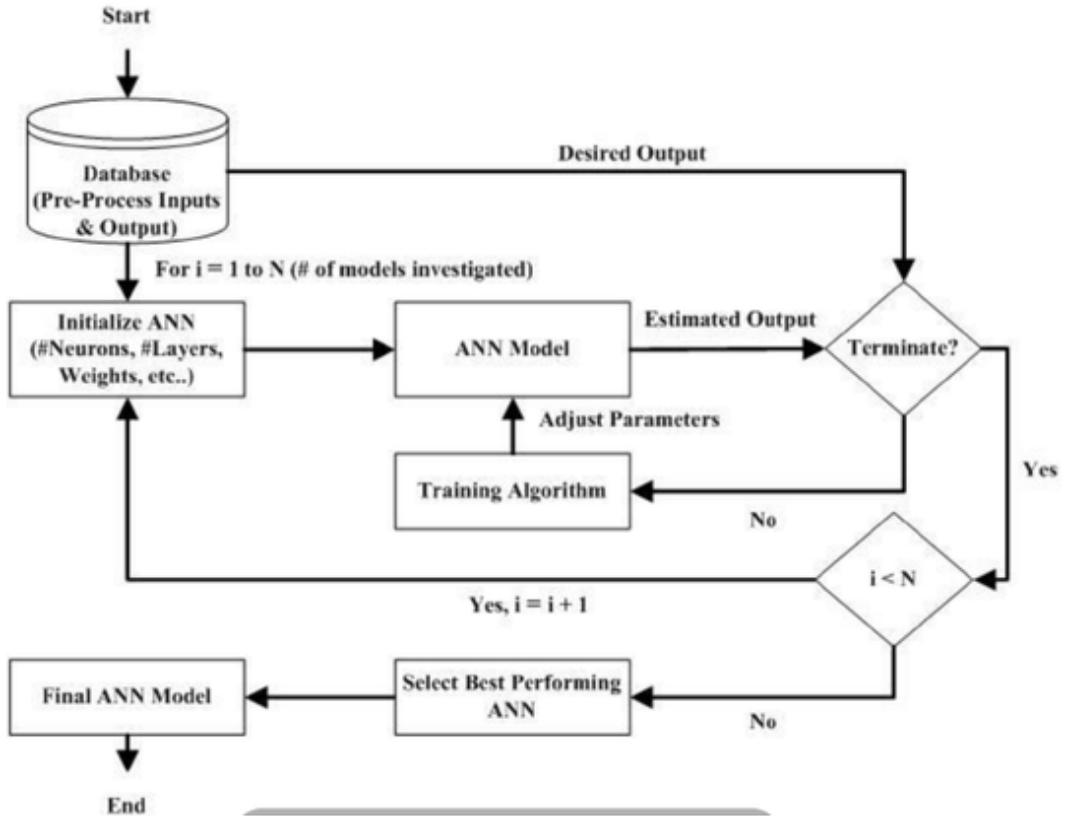
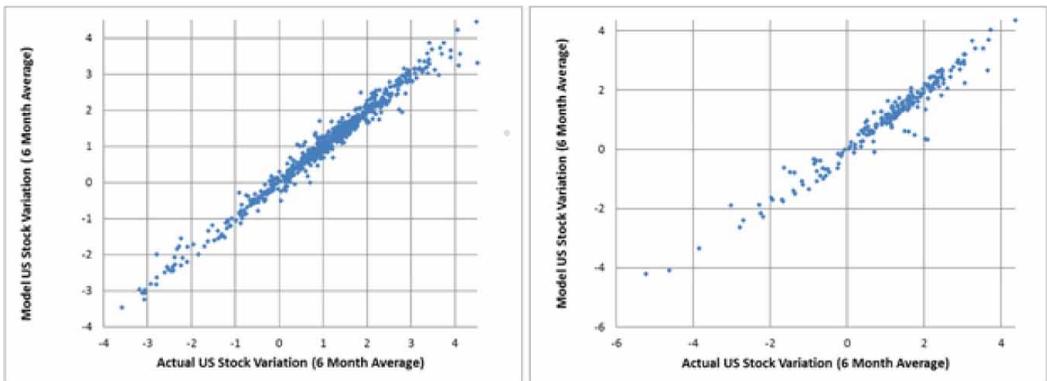
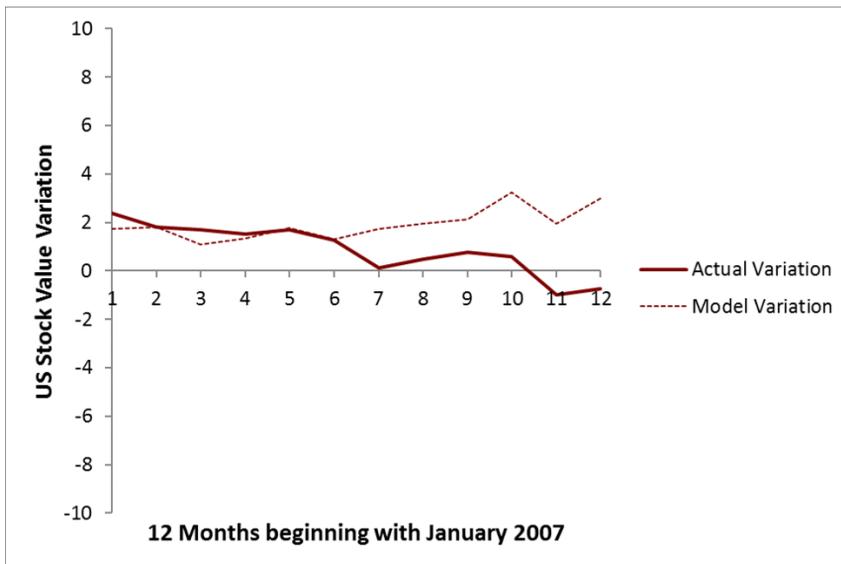


Figure 3. U.S. stock 3-month moving average variation in value



The time-series shown in Figure 4 demonstrates far less erratic behavior due to unexplainable variations within the data than the previous time-series that were analyzed. Thus, after analyzing the capabilities of an ANN’s ability to model the performance of each asset class, it was decided that the six-month moving average lagged output would be used for the remainder of the methodology. It should be noted that financial experts were consulted with the development of this model. The experts that were consulted favored the six-month time horizon given the improved quality of the

Figure 4. U.S. stock 6-month moving average variation in value



forecast. In addition, they did not react negatively that the time horizon of prediction was longer than three-months since long-term portfolio performance was their primary focus.

Determining SMOTE and Time Series Forecast

From the aggregated data, a baseline model was created. Other than the pre-processing that has been discussed previously, no other modifications were made to the aggregated data. This data was partitioned into a training, cross-validation, and testing datasets. The training dataset contained 178 records, which left 36 records for cross-validation and 12 records for independent testing. Again, this limited sample-size presents a challenge for ANNs to learn from the data it is presented within the training data. For example, an ANN was developed for Bonds and the results of the training session are shown in Figure 5. The figure for the training session does not show the desired characteristics. For example, the performance for both the training and cross-validation data do not slowly improve over epochs. The lowest average MSE occurs over just a few epochs, which often is an indication that the data is complex, or the architecture of the ANN is not adequate. However, after trying various MLP ANN structures, the accuracy did not improve, which indicates additional pre-processing techniques should be considered.

Figure 5. Bonds training and cross-validation results

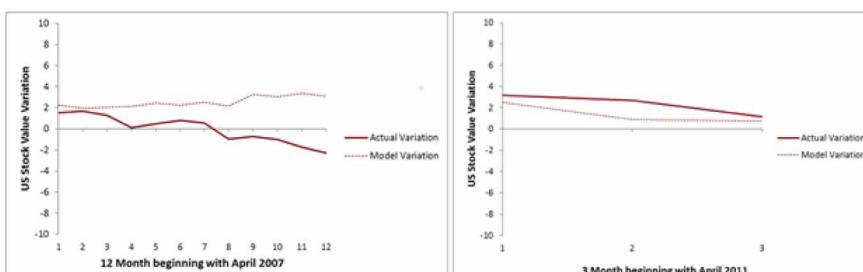
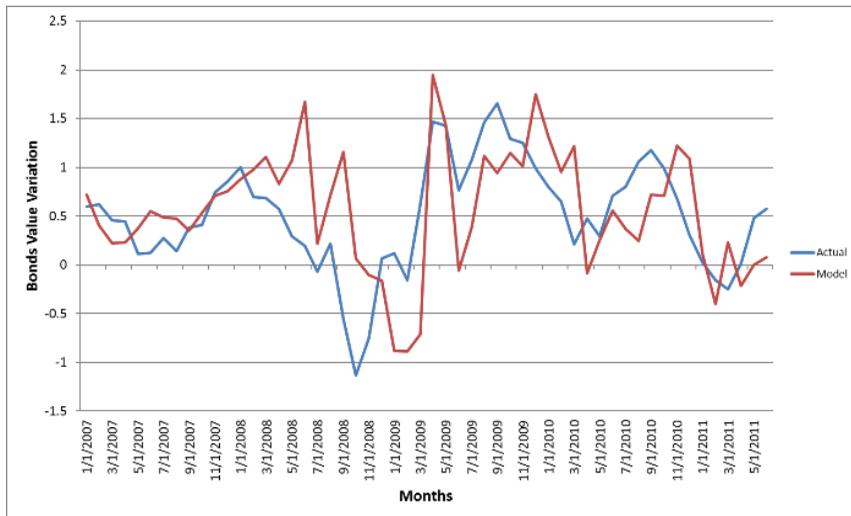


Figure 6 presents additional evidence of the poor fits that were obtained from the baseline ANN model that was constructed from the aggregated dataset. It is important to mention that the goal of developing a predictive model is to produce a model that generalizes the data well. In order to investigate whether the ANN model generalized the data well, the accuracy of the testing data

Figure 6. Bonds test results and model versus actual bond variation from test data

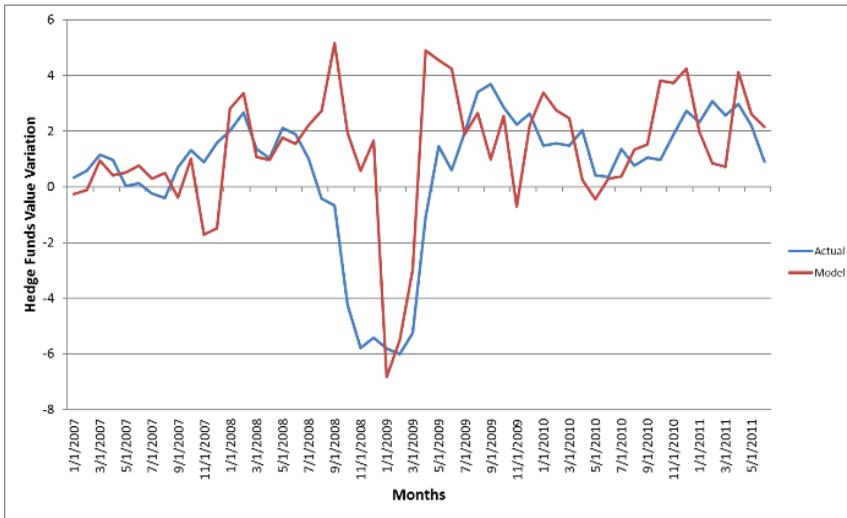


was explored. Each figure below shows two separate ways to highlight the unexplained variation of the model in terms of modeling the actual data points within the testing data. To investigate this, a 45-degree plot was created, and it is shown as the second figure below. The graph can be interpreted in the following manner. If a model was perfectly accurate, the actual values would fall along a 45-degree line. Visually it is clear that the trend fit and the location of the fit are both poor. Furthermore, the R-value for the testing set was found to be 0.0087, which is extremely poor.

The poor results shown in Figure 6 was not a surprise given the evidence shown in Figure 5. However, to be thorough with the analysis, the quality of fit was examined for both the training and cross-validation datasets. The 45-degree plots for the training and cross-validation datasets are shown below in Figure 7. These figures show a lot of variation, which is to be expected with complex datasets. The variation in the training and cross-validation data helps to explain why the predictive performances were poor for the testing data found in Figure 6. Simply put, if a model cannot produce a satisfactory quality of fit metrics for training and even more so, cross-validation, then the testing data will not perform well in practice. Thus, additional methods to pre-process the data were necessary in order to create a useful predictive model.

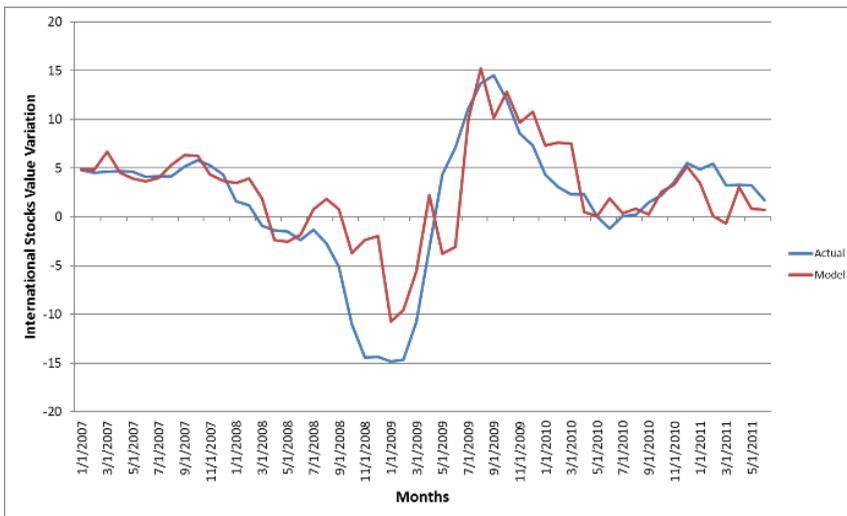
The dataset that was original compiled for this research did not produce reliable and useful results as demonstrated in the previous figures and paragraphs. As noted, the sample size of this dataset was not large, and the quality of fit was directly impacted by the limited sample size. In order to increase the number of records, which was assumed to improve the quality of fit for the model, a method called SMOTE was used to generate synthetic data samples based on the original data (Chawla, 2002). It is important to make the distinction that SMOTE was applied to the original dataset in order to increase the number of records available for training and cross-validation but was not used to generate artificial samples for testing. After creating additional synthetic data samples, the size of the dataset grew to 890 records, which would later be used to create new data partitions for both the training and cross-

Figure 7. Model versus actual bond variation from training data cross-validation data



validation. The impact of applying this pre-processing method is shown in Figure 8. This figure is much more desirable than the training session that did not use synthetic data points. The learning curve is smooth, and it requires several thousand iterations to arrive at a steady-state. For example, the ANN continued to improve in its predictive performance until the 9,460th epoch, versus the earlier

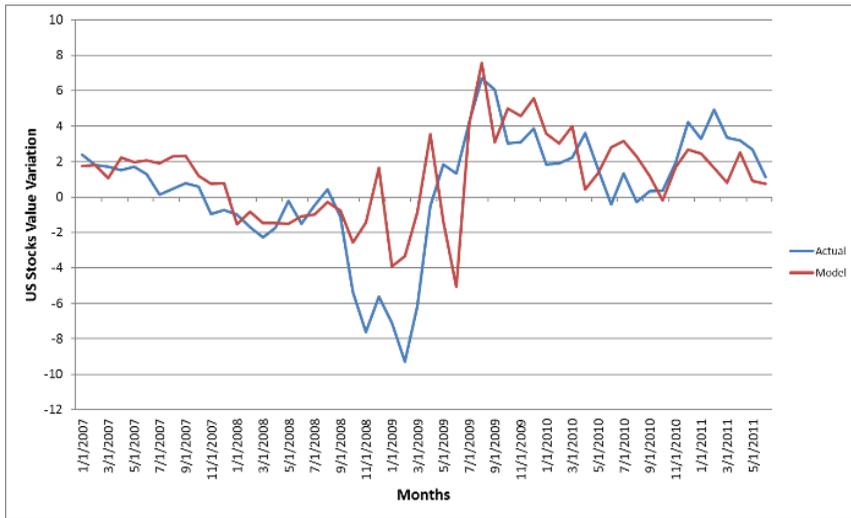
Figure 8. Bonds training and cross-validation results (synthetic data)



method with no synthetic data, which arrived at a steady-state point in only 19 epochs, which is not desirable given the complexity of the data being investigated.

Based on the increased sample size formulated by applying the SMOTE method, Figure 9 shows 45-degree plots for the new training and cross-validation datasets. The figures show a “tight” fit for

Figure 9. 45 degree plot of training data (synthetic) and cross-validation (synthetic)



both the training and cross-validation datasets, which is a desirable characteristic and implies the model should produce accurate fits to the testing data.

The synthetic data created by the SMOTE technique produced results that were very promising. Though the results looked promising after SMOTE was utilized, much more time and experimentation are necessary in order to develop the best performing ANN model. Therefore, the following section will document the process of creating the final ANN model that was used to aid in the process of allocating assets within an investment portfolio.

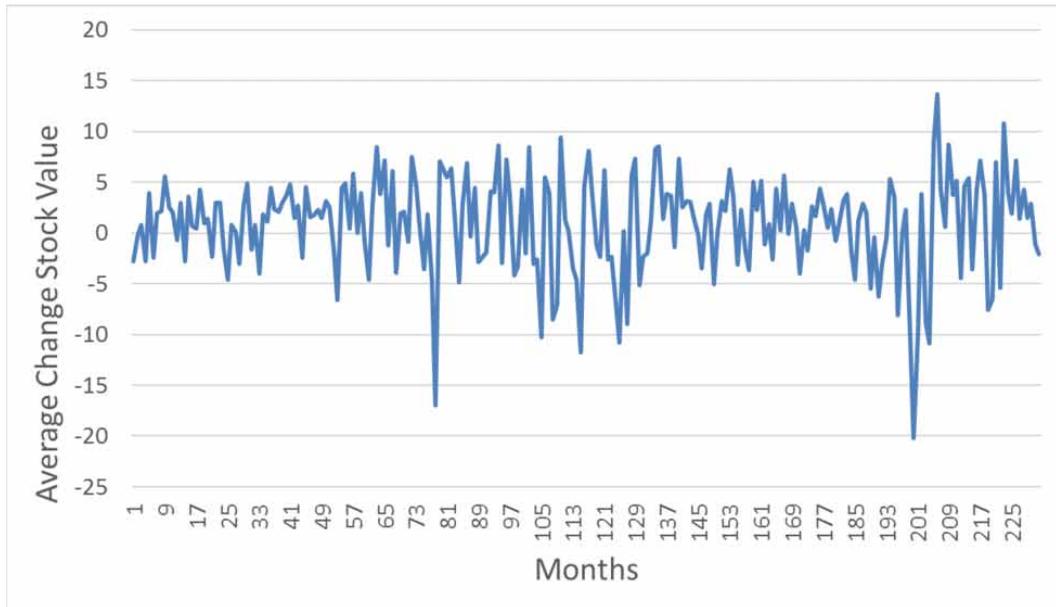
Final ANN Model Creation Based on SMOTE

Various ANN models were developed using many different types of architectures, learning algorithms, training percentages, cross-validation percentages, testing percentages, transfer functions, and hidden layers. Multiple models were developed for each of the four asset classes in order to find the best performing model for each asset class. Figure 10 shows a summary of the process flow utilized to create and find the “best” ANN. It shows three phases, pre-processing, training, and testing (Young, Holland, & Weckman, 2008). Utilizing the key characteristics and the flow chart, the ANNs were initialized and trained in order to make a final decision on which ANN would be used to determine a portfolio’s allocation of assets.

As noted, an ANN model was developed for each of the four asset classes in order to predict the performance of each. Using the dataset created by applying the SMOTE methodology, multiple ANNs were trained and evaluated based upon the six-month moving average, which would give ample time for investors to make decisions for their long-term investment performance goals. As previously described, the ANN models varied in architectures, randomization, learning algorithms, cross-validation and testing percentages, and hidden layers.

Example architectures that were varied for each class include generalized feed-forward (GFF) networks, typical multi-layered perceptron (MLP) networks, and modular networks. All of the ANNs that were developed were tested using different learning algorithms, which included momentum, delta bar delta, and conjugate gradient descent algorithms. The different architectures also utilized different transfer functions, such as Tanh, sigmoid, linear Tanh, and linear sigmoid. The number of hidden layers was also changed for each of the different combinations of architectures, learning algorithms, and transfer functions. The number of hidden layers ranged from one hidden layer to three

Figure 10. Procedural flow for ANN creation



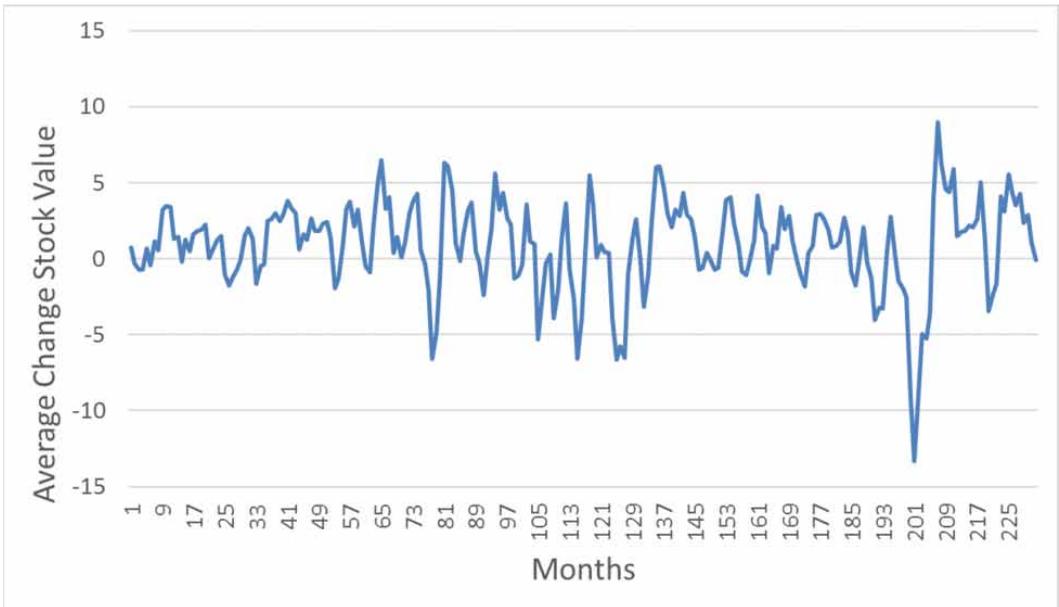
hidden layers for each of the different combinations. The search to find the best performing ANN for each asset was quite extensive and laborious. An example of the top-four performing models for U.S. Stocks is summarized in Table 2. Determining the top-performing model was determined from models producing the highest R^2 values.

Table 2. Four best performing models

Model #	Architecture	# of Layers	# of Processing Elements by Layer	Activation Function by Layer	Learning Algorithm
1	Multilayer Perceptron	3	100, 75, 50	Hyperbolic Tangent, Hyperbolic Tangent, Linear	Momentum
2	Multilayer Perceptron	3	100, 75, 50	Sigmoid, Sigmoid, Linear	Momentum
3	Multilayer Perceptron	3	30, 20, 10	Hyperbolic Tangent, Hyperbolic Tangent, Linear	Levenberg-Marquardt
4	Multilayer Perceptron	3	30, 20, 10	Sigmoid, Sigmoid, Linear	Levenberg-Marquardt

Training, cross-validation, and testing percentages were examined during the creation of the ANN's forecasting process. ANNs were created using Neurosolutions 6.5. Figure 11 shows that the model's output compared to the actual values for the desired output. The figures suggest that the estimates are fitting closely to the actual data points due to the "tightness" of the 45-degree plot. For

Figure 11. Model versus actual training and cross-validation



example, if a linear regression line was fitted to the cross-validation data appearing below, it would result in an R^2 value of 0.947, which is incredibly high.

Though the model seemed to fit the training and cross-validation well, it is not a true, unbiased evaluation of the model's ability to generalize the data well because of the influence of applying the SMOTE technique to the training and cross-validation data. However, in order to evaluate the generalizability of the model, the testing data, which has not been modified, can be assessed in an unbiased manner. The results of the testing data are shown in Figure 12. Figure 12 describes the output of the first forecasted values which is an evaluation of a 12-month time horizon that started on January 1st in 2007. To capture additional insight, a slightly different time horizon is shown in Figure 13. In this figure, the results of applying the best performing ANN model to U.S. Stocks over a 12-month time period that started on April 1st in 2007. Following this investigation, the last three months of the evaluation period were forecasted. The performance of the final three months of this study resulted in an R-value of 0.752.

RESULTS

This section presents the results for the ANN models that were developed for the four asset classes (i.e. Bonds, Hedge Funds, International Stocks, and U.S. Stocks) described throughout this article. Before the discussion of the results begins, it is important to remind readers that the U.S. stock market experienced a crash in 2008. Therefore, this time period is of particular interest and the time periods before the crash (BtC) and after the crash (AtC) will be discussed throughout this section.

The results for the Bond asset class are summarized in Figure 14. The overall performance of this asset class resulted in an R^2 of 0.231, which does not imply that the goodness-of-fit is strong. However, an argument could be made that the model is following the actual variation observed within the data well and would still have provided useful information in the process of planning investment portfolios. As noted, the U.S. stock market crashed in 2008. After this time period, the value of R^2 increased slightly to 0.297, which is about a 6% improvement AtC in 2008. Perhaps one could argue

Figure 12. Test results

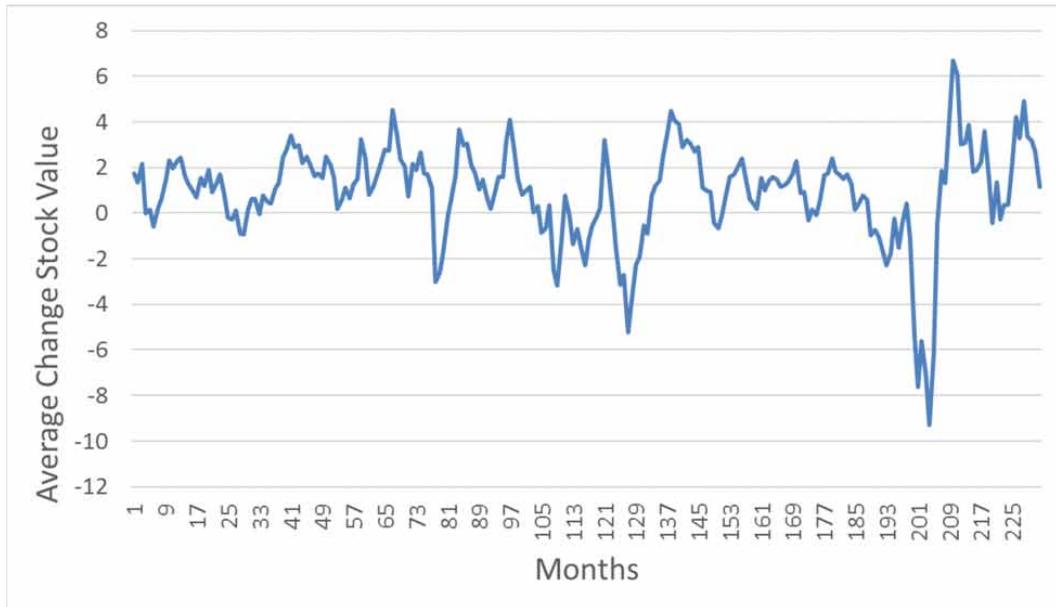


Figure 13. Desired output and actual network output

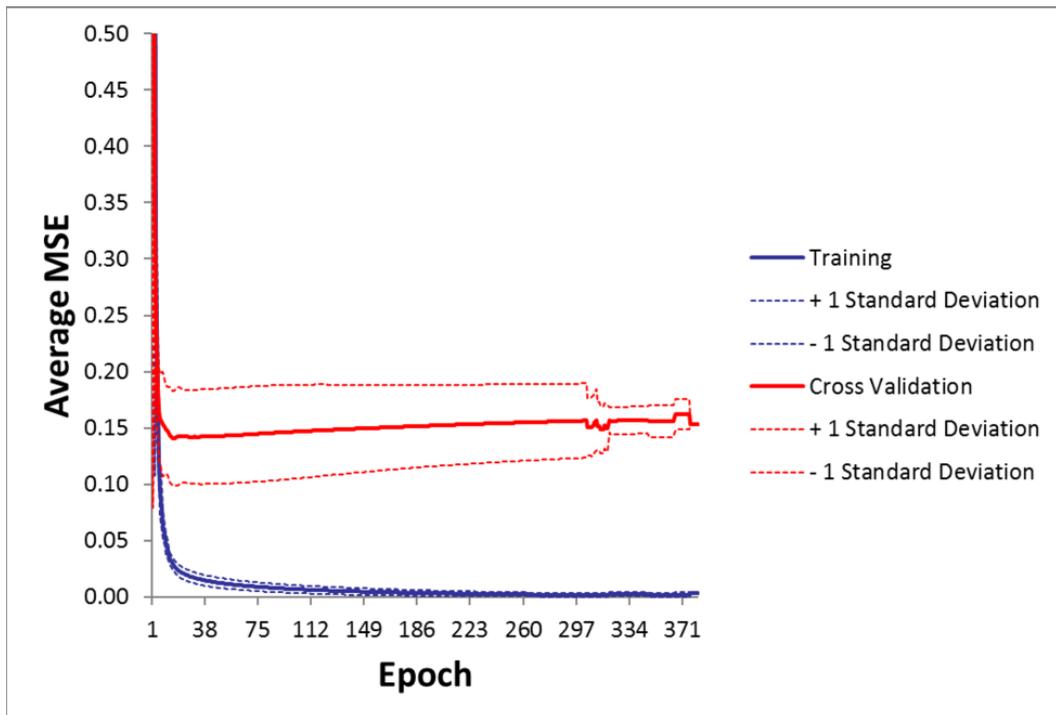
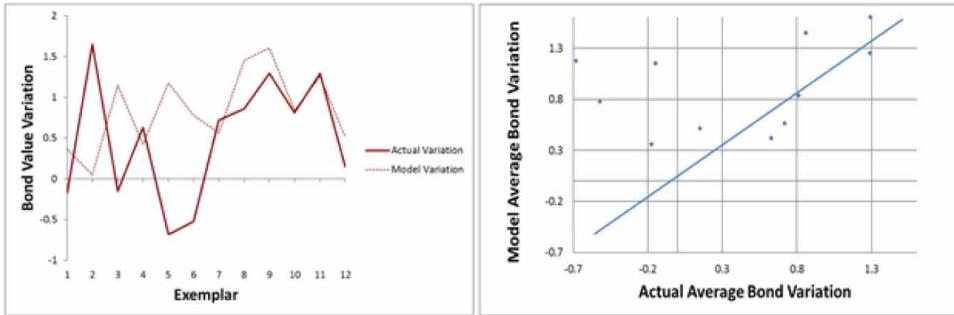


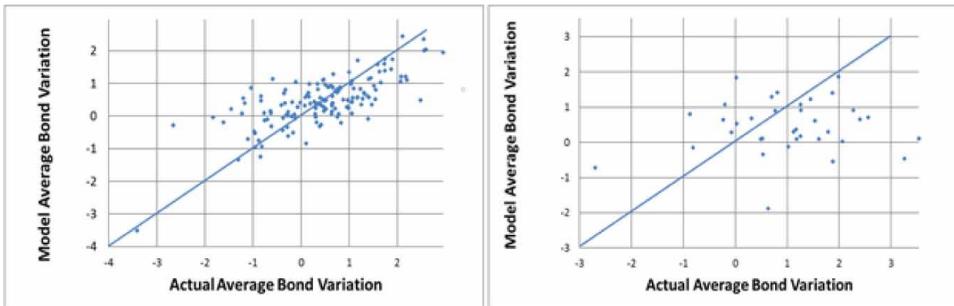
Figure 14. Final bond performance



that this is not a substantial or statistically significant improvement, but it does at least suggest that the ANN performed as equally well BtC as it did AtC. Thus, the ANNs ability to model the performance of Bonds was consistent and robust to high volatility.

The results of the Hedge class are summarized in Figure 15. The model’s predictive ability was similar to the Bond asset class. The overall performance of the model produced an R^2 value of

Figure 15. Final hedge performance



0.249. This R^2 value shows that the ANN only understood approximately 25% of the variation for the asset class’s performance. With respect to BtC and AtC, the ANN model seemed to be resilient to the highly fluctuating market. Likewise, the value of R^2 actually improved slightly AtC when the market was more stable.

The performance of the International Stock class is summarized in Figure 16. For this asset class, the ANN was able to produce an R^2 value of 0.651, which is a drastic improvement over the performance obtained for the Bond and Hedge asset classes. Unlike the previous asset classes that have been described, the ANN for the International Stock class did not improve AtC. In fact, the performance AtC decreased in value to 0.431. However, given the time frame in which decisions are made to allocate assets within an investment portfolio, the ANN is still able to provide useful information because the ANN is able to model the change in trend within the time-series.

The performance of the best performing ANN for the U.S. Stock asset class is shown in Figure 17. For this asset class, the model produced an R^2 value of 0.446, which is nearly the same value for the ANN model for International U.S. Stocks AtC. Thus, approximately 45% of the variation in the U.S. Stock class was explained by the ANN model.

Along with the performance of the ANN model regarding each of the asset classes, the model’s ability to model the trend of the data was also investigated. From an adviser’s perspective, it is extremely

Figure 16. Final international stock performance

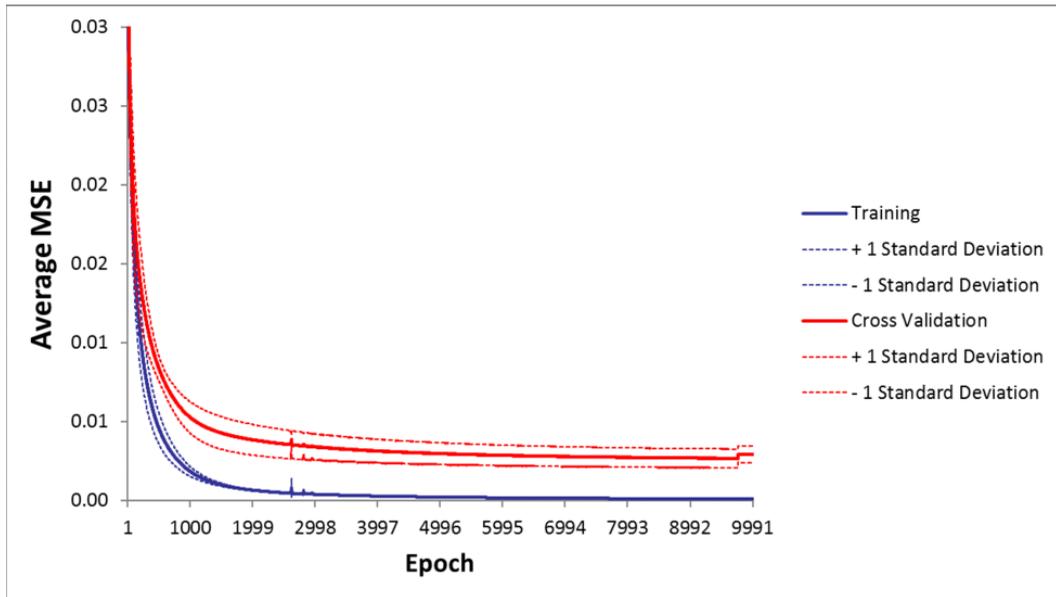
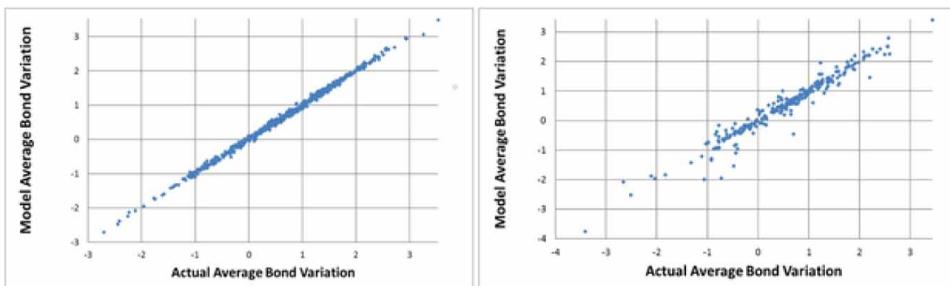


Figure 17. Final U.S. stock performance



important to know if the performance of a certain asset class is trending down (TD) or trending up (TU) in order to make decisions on how an investment portfolio is allocated. In other words, financial advisers need to know the anticipated change of trajectory of any given asset class to make informed decisions of how an investment portfolio should be constructed. Quarterly averages were calculated based upon the actual observed values over the last three months of a given time period. These values were compared against the last six-month average in order to establish whether the performance of an asset class was actually TU or TD. Thus, by comparing the actual values with the forecasted values for a given ANN, the accuracy of the trend could be determined. An example of the information that could be analyzed by financial advisers for the U.S. Stocks asset class is shown in Table 3.

In order to determine how well the trends were modeled by the proposed methodology, the concept shown in Table 3 was applied for all asset classes from 1/1/2007 through 6/1/20011. The results of this study are shown in Table 4. By comparing the actual trend trajectories with the forecasted trend trajectories, the methodology produced an overall accuracy between 64.7% to 70.6% for the four types of asset classes.

Table 3. U.S. stock trend prediction

Date	Actual 6MA	Actual Quarterly Average	Actual Trend	ANN 6MA	ANN Quarterly Average	ANN Trend
1/1/2007	2.388			1.743		
2/1/2007	1.820			1.808		
3/1/2007	1.707	1.972		1.079	1.543	
4/1/2007	1.522			2.224		
5/1/2007	1.696			1.978		
6/1/2007	1.289	1.502	D	2.052	2.085	U
7/1/2007	0.133			1.898		
8/1/2007	0.468			2.303		
9/1/2007	0.788	0.463	D	2.320	2.174	U
10/1/2007	0.575			1.192		
11/1/2007	-0.969			0.732		
12/1/2007	-0.721	-0.372	D	0.782	0.902	D

Table 4. Trend prediction accuracy for each asset class

	Bonds	Hedge	Intl Stocks	US Stocks
Total	64.7%	64.7%	70.6%	64.7%

DISCUSSION

This research methodology was explained approximately 25% of the variation seen in the Hedge asset class. The methodology did not perform as well as anticipated for this particular asset class. The reason why the models for the hedge asset class did not perform to a high degree of accuracy is potentially due to the fact that crash will certainly perform differently from real estate and gold in various markets. After further discussion with the financial advisors, one suggestion for future work was to improve the hedge prediction accuracy by splitting out the hedge asset class and using ANN models to compile an overall hedge asset class prediction.

The ANN models also did not perform as well as one would have wanted for the Bond asset class. At first, this was a concern because bonds typically do not fluctuate as much as stocks. It was assumed that the prediction and understanding of the variance would be much higher in terms of its accuracy. Unfortunately, the models were only capable of understanding approximately 23% of the variation within that asset class. After the crash of 2008, the model improved. The model had virtually no understanding before the crash based upon the inputs that were provided, while after the crash it was able to explain 29% of the variation for that asset class. However, when advisors were consulted about these results, they were concerned with this level of accuracy. They suggested that the prediction for a fixed income bond was less desirable with respect to the other asset classes that were investigated.

Machine learning methods, like the ones presented in this research, are slowly being adopted by financial firms. However, they have not been in practice long enough in order to determine how successful they will be in designing portfolios in the future. However, it does suggest that

methodologies, such as the one presented in this article should be at least considered when future methods are explored.

FUTURE RESEARCH

Although this research was considered a success, there are still areas that could be considered for future research in order to improve the performance and usability of the methodology. This section describes a few areas that should be considered by others as a result of the research findings presented in this article.

One avenue for future consideration would be to apply various deep learning models within the scope of financial performances related to investment portfolios. For example, deep learning approaches like convolutional neural networks (CNNs) or recurrent neural networks (RNN) could be one possible methodology to explore. In CNN, the feature extraction and prediction process occur simultaneously. This is not the case for traditional machine learning approaches like MLP ANN where the feature extraction process is implemented manually by an expert in the field. In CNNs, these types of transforms are made inherently and automatically within the framework of the methodology. However, one limitation of applying CNN models is, in contrast to other learning algorithms in which their performance decreases when the amount of data increases, that their performance is highly dependent upon the amount of data that is used. In addition to developing CNN models, recurrent neural network (RNN) models have been shown to be promising for time-series problems. In RNNs, each sample is assumed to be dependent on previous samples. In the context of portfolio management, this could be considered a reliable tool for investors making a decision on how a portfolio should be allocated.

Determining which set of input attributes, as well as the output attribute, of any model is vital to the application and success of a data-driven approach. Thus, additional research could explore a wider set of attributes that were not explored within the research presented. For example, emotional investing plays a critical role in the performance of the market. Thus, one area of future research could be including input attributes that reflect the emotions that influence market behavior. In addition to input attributes that could potentially be integrated into the methodology, the expert opinion could also be considered, which should improve the overall capabilities of a methodology developed for asset allocations within investment portfolios.

Methodologies can evolve with the integration of machine learning methods like CNN or RNNs with the other analytical capabilities such as optimization. Based on the forecasts that could be made with various types of ANNs, optimization could be employed in order to determine an optimal mix of assets to allocate within a client's investment portfolio. Prescriptive analytics could be leveraged in a way that minimizes the variance of the system in order to maximize the highest expected rate of return on the investment portfolio given the client's investment preferences, age, and other characteristics.

Methodologies designed to be used as a decision-making aid for portfolio design should not overlook the expertise of financial advisers. These tools should not be designed with the intent of replacing financial advisors, but they should be designed to help the advisors make more informed, and ultimately, better decisions for their clients. Therefore, with that being said, integrating expert knowledge into the design and implementation of a methodology should always be considered best practice.

REFERENCES

- Akinwale, A., Arogundade, O., & Adekoya, A. (2009). Translated Nigeria Stock Market Prices Using Artificial Neural Network for Effective Prediction. *Journal of Theoretical and Applied Information Technology*, 9, 36–43.
- Burney, K. (1997). *Introduction to Neural Networks*. UCL Press.
- Canner, N., Mankiw, N., & Weil, D. (1997). An Asset Allocation Puzzle. *The American Economic Review*, 87(1), 181–191.
- Charles Schwab. (2015). *Corporation*. Retrieved from https://en.wikipedia.org/wiki/Charles_Schwab_Corporation
- Chawla, N., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-Sampling Technique. *Artificial Intelligence Research*, 16, 321–357. doi:10.1613/jair.953
- Chiang, W. (1996). A Neural Network Approach to Mutual Fund Net Asset Value Forecasting. *Management Science*, 24(2), 205–215.
- Dictionary. (2015). *Moving Average*. Retrieved from <http://dictionary.reference.com/browse/moving+average>
- Economic Research Federal Reserve Bank of St. Louis. (2014). Retrieved from <https://research.stlouisfed.org/useraccount/datalists/126909>
- Encyclopedia Britannica. (2015). *Bear market*. Retrieved from <http://www.britannica.com/EBchecked/topic/57332/bear-market>
- Encyclopedia Britannica. (2015). *Bull market*. Retrieved from <http://www.britannica.com/EBchecked/topic/84336/bull-market>
- Hui, S., Yap, M., & Prakash, P. (2000). A Hybrid Time Lagged Network for Predicting Stock Prices. *International Journal of the Computer, the Internet, and Management*, 8(3).
- Jiang, Z., Xu, D., & Liang, J. (2017). *A deep reinforcement learning framework for the financial portfolio management problem*. arXiv preprint.
- Lakshminarayanan, S. (2005). *An Integrated Stock Market Forecasting Model Using Neural Networks* (M.S. Thesis). Ohio University.
- Morningstar Principia. (2015). *PNR General*. Retrieved from http://advisor.morningstar.com/Principia/pdf/PRN_General_040212.pdf
- Neural Networks. (1989). *Multilayer feedforward networks are universal approximators*. Retrieved from <http://portal.acm.org/citation.cfm?id=70408>
- Principe, J., Euliano, N., & Lefebvre, W. (1999). *Neural and adaptive systems: Fundamentals through simulations*. John Wiley & Sons Inc.
- Qian, B., & Rasheed, K. (2007). Stock market prediction with multiple classifiers. *Applied Intelligence*, 26(1), 25–33. doi:10.1007/s10489-006-0001-7
- Skapura, D. (1996). *Building Neural Networks*. Addison-Wesley.
- Suvarna, V. (2013). SMOTE Based Protein Fold Prediction Classification. *Advances in Computing & Inform. Technology*, 541–550.
- Vaisla, K. B. (2010). An Analysis of the Performance of Artificial Neural Network Technique for Stock Market Forecasting. *International Journal on Computer Science and Engineering*, 2, 2104–2109.
- Weckman, G., Dravenstott, R., Young, W. II, Ardjmand, E., Millie, D., & Snow, A. (2016). A Prescriptive Stock Market Investment Strategy for the Restaurant Industry using an Artificial Neural Network Methodology. *International Journal of Business Analytics*, 3(3), 1–21. doi:10.4018/IJBAN.2016010101
- Weckman, G., Lakshminarayanan, S., Snow, A., & Marvel, J. (2008). An Integrated Stock Market Forecasting Model Using Neural Networks. *International Journal of Business Forecasting and Marketing Intelligence*, 1(1), 30. doi:10.1504/IJBFMI.2008.020813

Young, W., Holland, W., & Weckman, G. (2008). Determining Hall of Fame Status for Major League Baseball Using an Artificial Neural Network. *Journal of Quantitative Analysis in Sports*, 4(4).

Young, W. II, & Weckman, G. (2009). Using a heuristic approach to derive a grey-box model through an artificial neural network knowledge extraction technique. *Neural Computing & Applications*, 19(3), 353–366. doi:10.1007/s00521-009-0270-2

Roohollah Younes Sinaki is a Ph.D. candidate in the Mechanical and Systems Engineering program at Ohio University. His primary research is focused on the application of mathematical programming to the problems from various domains including supply chain network design and manufacturing system design, validation of developed models with discrete simulation modeling. Roohollah is also highly interested in the application of machine learning and statistical modeling tools to solve complex problems related to social, environmental, and economic aspects of supply chain systems. Roohollah received his B.Sc. in Industrial Engineering at Azad University, Tehran, Iran in 2011 and his M.Sc. in Industrial and Systems Engineering at Ohio University in 2017.

Azadeh Sadeghi is a Ph.D. candidate in the Mechanical and Systems Engineering program at Ohio University. Her primary research is focused on humanitarian logistics, designing supply chain networks, and the application of discrete-event simulation models for determining operational parameters and performance measures. She has been publishing journal articles and conference proceedings in supply chain management and applied scheduling areas. Before joining Ohio University in 2015, she was a project control analyst at Moshanir Power Engineering Consultants Company, Tehran, Iran for five years. Azadeh received her B.Sc. in Industrial Engineering at Azad University, Tehran, Iran in 2011 and her M.Sc. in Industrial and Systems Engineering at Ohio University in 2017.

Dustin S. Lynch is a Master Black Belt Engineer. Lynch completed a master's degree in Industrial and Systems Engineering at the Russ College of Engineering and Technology at Ohio University in 2011. In addition to this degree, Lynch also completed a bachelor's degree in Plastics Engineering at Shawnee State University in 2007. Dustin now works for a major aerospace company where he specializes in cost estimation, operations management, and lean six sigma.

William A. Young II (PhD) is a Charles G. O'Brien Associate Professor of Business Analytics in the Department of Analytics and Information Systems at Ohio University. Young earned his doctorate degree in Mechanical and Systems Engineering from Ohio University in 2010. William also received a bachelor's and master's degree in Electrical Engineering at Ohio University in 2002 and 2005 respectively. William has collaborated with multidisciplinary teams of faculty, students, and professionals on projects and programs that have been funded by General Electric Aviation, the National Science Foundation, Sogeti Netherlands, and Ohio's Department of Labor. Young's primary research and teaching interests relate to operation management, healthcare services, and environmental systems, as well as specific interests in quantitative sports analysis, and educational technologies and techniques for innovative curriculum development and teaching instruction.

Gary Weckman (PhD) is an Associate Professor in the Department of Industrial and Systems Engineering in the Russ College of Engineering and Technology at Ohio University. Dr. Weckman has an extensive research background in data analysis, with specific training and expertise in key research areas of nonlinear modeling. He has authored or co-authored more than 100 peer-reviewed articles in journals and conferences. Currently, he has been researching multidisciplinary applications utilizing knowledge extraction techniques with artificial neural networks (ANN). Dr. Weckman has extensive experience in and possesses an internationally-renown reputation for neural network modeling of complex information systems, both traditional (e.g. applied applications of reliability analysis and telecommunications) and non-traditional (e.g. network modeling applications for financial and ecological monitoring). Prior to returning to university for an academic career, he has industrial engineering experience with over 12 years at such firms as General Electric Aircraft Engines.