SMU Data Science Review

Volume 1 | Number 4 Article 6

2018

Improving VIX Futures Forecasts using Machine Learning Methods

James Hosker
Southern Methodist University, jhosker@smu.edu

Slobodan Djurdjevic
Southern Methodist University, sdjurdjevic@smu.edu

Hieu Nguyen
Southern Methodist University, hdnguyen@smu.edu

Robert Slater
Southern Methodist University, rslater@smu.edu

Improving VIX Futures Forecasts using Machine Learning Methods

James J. Hosker[1], Slobodan Djurdjevic[2], Hieu Nguyen[3], Robert D. Slater[4]
[1] Master of Science in Data Science, Southern Methodist University,
Dallas, TX 75275 USA
{jhosker, sdjurdjevic, hdnguyen, rslater}@smu.edu

Abstract. The problem of forecasting market volatility is a difficult task for most fund managers. Volatility forecasts are used for risk management, alpha (risk) trading, and the reduction of trading friction. Improving the forecasts of future market volatility assists fund managers in adding or reducing risk in their portfolios as well as in increasing hedges to protect their portfolios in anticipation of a market sell-off event. Our analysis compares three existing financial models that forecast future market volatility using the Chicago Board Options Exchange Volatility Index (VIX) to six machine/deep learning supervised regression methods. This analysis determines which models provide best market volatility forecast. Using VIX futures and options data along with other technical indicators, our analysis compares multiple forecasting models for estimating the 1-month VIX futures contract (UX1) both 3 and 5-days forward. This analysis finds that machine/deep learning methods of Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) provide improved results over existing linear regression, principal components analysis (PCA) and ARIMA methods. Comparing estimated versus actual test data, both the RNN and LSTM methods show lower mean squared error (MSE), lower mean absolute error (MAE), higher explained variance, and higher correlation. Finally, an accuracy matrix was generated for each model, which showed RNN and LSTM had better overall accuracy due to high true positive and negative forecasts as well as much lower false positive forecasts.

1 Introduction

Investment managers are concerned about future market volatility. Fund managers want to reduce or hedge risk positions prior to a market sell-off event. This paper

[1] James Hosker is completing his MS in Data Science at SMU and has a BSEE and MSEE from Tufts University as well as an MBA from MIT Sloan. He has over 20 years of experience in financial engineering working in derivatives for investment banks.

[2] Slobodan Djurdjevic is completing his MS in Data Science at SMU. His academic background is in Mathematics and Physics and for the past 18 years he has worked in Information Technology.

[3] Hieu Nguyen is completing his MS in Data Science at SMU and has a BA in Mathematics/ Actuary from University of Texas at Austin. He has 5 years of experience in financial analysis with the Texas Health and Human Services Commission.

[4] Prof. Robert D. Slater is a professor in data science at SMU.

focuses on S&P 500 market risk. Investment managers actively create and refine models to assist in hedging market downside or Black Swan risks. Fund managers are always looking for improvement in their models to forecast market volatility. Nassim Taleb wrote about what causes and how to hedge market downside risk. Nassim Taleb coined the name Black Swan in his book 'The Black Swan: The Impact of the Highly Improbable' [1] in 2007. Taleb highlighted in his book how financial models can break down during highly improbable market events or market downturns.

For this paper, market volatility is represented by The Chicago Board Option Exchange (CBOE) Volatility Index[5] (VIX) for the S&P500. The VIX is essentially option volatility as an asset class or index. The VIX is forward looking, based on future market expectations since it uses the options market. It is not the historical or realized volatility of S&P500 (standard deviation of the S&P 500) but the 1-mth implied volatility from S&P 500 options. VIX is a measure of uncertainty, expectations or fear in the future; hence, it is also known as the "Fear" index for the S&P 500. For an introductory description of futures, options, calls, puts, and the VIX as well as how implied volatility is calculated for the VIX, see Appendix 1.

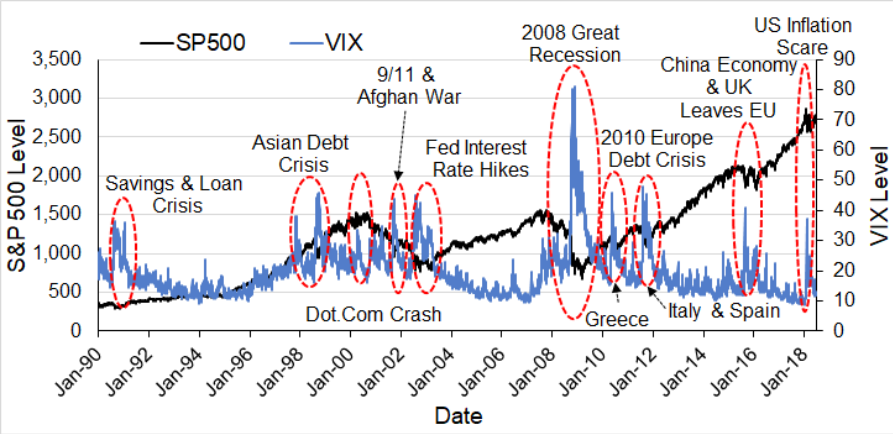


Fig. 1. S&P500 vs. VIX Level (Jan 1990 to Jun 2018)

The CBOE futures and options on the VIX are liquidly traded across different maturities, allowing investors to hedge potential market downside risk in the future. As shown in Figure 1, the VIX is inversely (negatively) correlated to the returns of the S&P 500, making it an attractive hedging instrument for fund managers to both use and forecast. As the S&P 500 index drops, the VIX (volatility) generally increases; and as the S&P 500 index rallies, the VIX generally moves lower or remains low. In the 2008 mortgage crisis (the Great Recession), the S&P 500 fell and the VIX spike to high levels. In the 2010 European debt crisis (Portugal, Italy,

[5] CBOE Volatility Index® (VIX® Index), futures and options are registered trademarks of Chicago Board Options Exchange.

Greece and Spain – the "PIGS"), the VIX actually moved higher before the S&P 500 sold-off.

Other assets exist that are negatively correlated to the S&P 500 market, such as precious metals (gold, silver, platinum) shown in Fig. 2, In addition, US Treasury Bonds sometimes are negatively correlated to S&P 500 returns (the flight to safety as investors globally buy US treasuries in a crisis). Finally, listed put and call options on the S&P 500 as well as other rate, FX and commodities instruments can be used as hedges to the S&P 500 risk. However, the VIX is one of the better hedges for investment fund managers for S&P 500 risk.

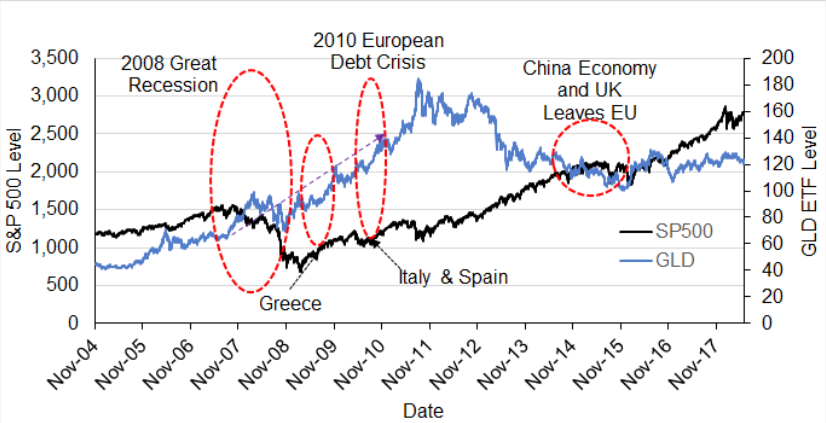


Fig. 2. S&P500 vs. Gold ETF (GLD) (Nov 2004 to Jun 2018)

This paper compares existing or common financial models to machine/deep learning supervised regression methods to improve the forecast of future market volatility using the VIX. Existing research has created individual machine learning models to forecast future market volatility or the VIX. However, few research papers compared different machine learning methods to existing or common models that are used to forecast market volatility (see Appendix 2 for more on background and prior research).

This paper assesses the quality of three existing or common market volatility forecasting models using linear regression, principal components analysis (PCA) and AutoRegressive Integrated Moving Average (ARIMA). These three common models are compared to six different machine learning supervised regression methods: Ensemble method, support vector regression (SVR), least absolute shrinkage and selection operator (LASSO), random forest (RF), recurrent neural networks (RNN) and long short-term memory (LSTM). The objective is to develop a higher quality model so that fund managers can utilized this analysis to assist in the hedging of their portfolios for volatility forecasts, while minimizing the cost of over-hedging if our forecast is for lower or reduced volatility. Our analysis uses similar evaluation metrics to assess the quality of the different models and methods.

The analysis finds that two methods provide improved results over Multivariate Linear Regression (MLR), PCA, and ARIMA: recurrent neural networks (RNN)

and long short-term memory (LSTM). RNN and LSTM have lower mean squared error (MSE), lower mean absolute error (MAE), higher explained variance, and higher correlation of test data actual versus estimated. In addition, an accuracy matrix was generated for each model, which showed RNN and LSTM had better overall accuracy due to higher true positive and negative forecasts as well as much lower false positive forecasts.

The paper is divided into seven sections. This section is the introduction that provides the motivation and basis for improving VIX futures forecast using machine learning methods. Section 2 describes the data set, the inputs (explanatory variables), the output (response variables), and our cross-validation technique. In addition, this section performs exploratory analysis of the dataset. Section 3 provides a roadmap of our methods and models used to analyze the data and to assess the quality of the results. It divides the models into two parts: three existing or common financial modeling methods and six machine/deep learning supervised regression methods. Section 4 provides the results that assess the quality of each of the methods and finds the optimal model for each method. Section 5 analyzes the results using a summary table of the best model for each method. The best method with the optimized model is selected. Section 6 addresses ethical issues surrounding our research. Finally, section 7 provides our conclusions. In addition, there are references and 17 appendices, including one for background research. UX1 in this paper will represent 1-mth VIX futures, which is our response variable, for 3 and 5-days forward.

2 Data Set and Data Exploration

Our data sources for this paper are Bloomberg and Option Metrics. Bloomberg was used for the VIX futures data and Option Metrics for the VIX options data. VIX futures were listed in March of 2004 but data on the VIX options started in July of 2006. Therefore, the data is from July 2006 to Jun 2018, which is the equivalent to 3009 business days or approximately 12 years of data, using market close to market close data. The size of the data set is approximately 8 GBs.

Table 1 groups our 71 input variables into the six factor types used in our analysis. There are 68 continuous time series variables and 3 categorical variables representing signals (1 or 0) based on their position in the time series. The following subsections of this paper provide a data description for some of these factor inputs in more detail. For the purpose of our analysis, the output or response variable is the 3 and 5-day forward front month (1-mth) VIX futures (UX1) level. However, our data set is robust enough that it could be used to forecast VIX futures for other maturities. Refer to Appendix 3 for a complete listing and description of all the 71 input (explanatory) and 2 output (response) variables.

Table 1. Breakout of the 71 Input Variables.

Factor	Number of Input Variables
Term Structure	21
Intraday Futures High-Low	7
Skew	30
Moving Average	9
Bollinger Bands	2
VVIX	2
Total	71

2.1 Data Cleaning and Validation

Data Cleaning. There is not much data cleaning for this data set from Bloomberg and Option Metrics since most of the data was continuous from July 2006 to June 2018. A few inputs had a small number of days without data that were forward filled using the prior days value.

Creation of Volatility Surface. The skew data was recreated from the Options Metrics data as inputs into the Black variance model (from Black-Scholes option model), using the QuantLib library in Python. As shown in Fig. 3, option metrics stores the normalized volatility surface data that the can be used to re-create the daily volatility surface. From this daily volatility surface for each maturity, all the implied volatility levels are extracted for the 80%, 90%, 100% (at-the-money or ATM), 110%, 120%, 150% and 200% OTM strikes. The volatility surface for each day is created for each maturity separately (1,2,3,6,9 and 12-mth option maturities). From this data, skew can be calculated. There was some noise in the early data (2006 – 2007) for far (out-of-the-money or OTM) strikes for the short-term maturities (1, 2, and 3-mth); therefore, the data from July 2006 to December 2007 for these strikes and these maturities was smoothed.

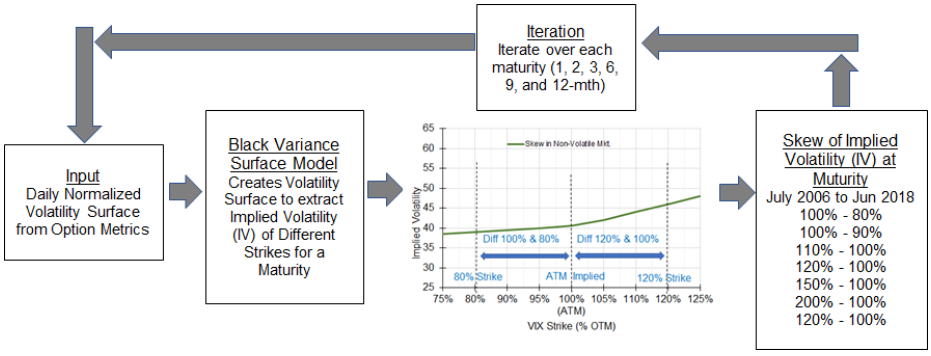


Fig. 3. Extraction of Skew Data from Normalized Volatility Data in Option Metrics

Traditional Time Series Split and K-Split Cross-Validation for Time Series. Our analysis cannot use the standard K-Fold cross-validation techniques of randomly

sampling data, since time series data is used. For time series data, cross-validation has to be continuous over consecutive days for both the training and test data sets. Two training and test data splits were performed. In the first split, we perform a traditional training and test split of first continuous 75% as the training data set and the remaining 25% as the test data set. However, without multiple test sets, the model could be overfitting the data with only one split of the data set. In the second split to adjust for potential overfitting, cross-validation is performed using K-Splits of the time series data for 5 and 10 splits. An average of our performance or assessment metrics (see section 2.5) are then taken using each of the splits. Fig. 4 shows an example of a 5-split customized time series (TS) for the different training and test data sets. The size of the training data set varies using different percentages of the data, but the test size is kept the same. Both training and test remain continuous. The best K-split cross validation results using this method is 10.

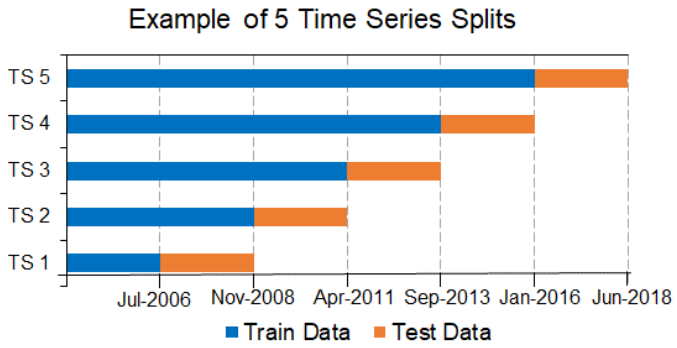


Fig. 4. Validation of Time Series Data Training and Test Datasets (July 2006 to June 2018)

2.2 Code Archive Description

The code for this analysis was performed in Python and the archive is submitted with this paper (see Appendix 4 for more details). The 'VIXproject.7z' code archive has 3 common financial models and 6 supervised regression methods. It will create a 'VixProject' directory with two iPython notebooks called 'Capstone_VIXProject.ipynb' that inputs the data from the file 'VIX_DataSkewFinal_New.csv' to run and output analysis for all our models; and 'CreateImpliedVolSurface.ipynb' that inputs the data file 'VolSurfaceVIX_2006to2010.xlsx', which creates our VIX skew data. The data files are located in the subdirectory called Data. The major Python libraries used in our analysis are Keras, Tensor Flow, Numpy, Scikit Learn, QuantLib, Pandas, Seaborn and Matplotlib as well as others. Keras and Tensor Flow are used for our neural network models, Scikit Learn for other models, and QuanLib for the extraction of the volatility surface.

2.3 Data Description and Exploration of Inputs and Output

Term Structure (28). Term structure of implied volatility represents the spread between future uncertainty from different maturities of the futures contract. The future contracts represent VIX 1-mth ATM implied volatility at different forward maturities. The VIX futures provides insight to which maturities have a higher amount of uncertainty perhaps due to market events yet to occur. Fig. 5 shows examples of different VIX future states and Table 2 defines different VIX futures states of contango, flattening and backwardation. The term structure spreads are between all combinations of 2, 3, 4, 5, 6, 7,and 8-mth futures (1-mth is removed since it is our response variable). The difference between the high and low intraday levels for each futures contract are included as input variables. There is a total of 21 term structure input variables and 7 intraday high minus low futures input variables.

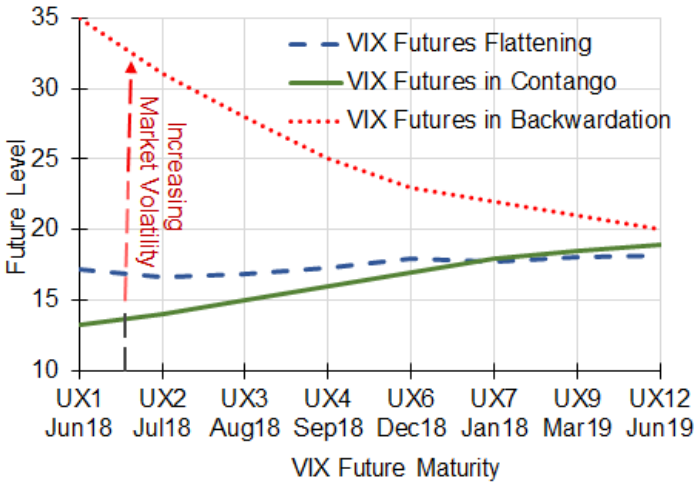


Fig. 5. Different VIX Term Structure Patterns for Flattening, Contango & Backwardation

Table 2. Description of Different Term Structure States

Term Structure State	Description
Cotango	This occurs during less volatile or normal market conditions. The volatility across maturities is upward sloping so with a longer maturity, there is generally more uncertainty. Longer-term futures are higher than shorter-term future contracts.
Flattening	Longer-term and shorter-term future levels are close, so short-term volatility moved higher but longer-term volatility remains sticky unless there has been a parallel shift.
Backwardation	Short-term volatility is much higher than longer-term volatility, which can make VIX hedging strategies very profitable. There is much uncertainty in the short-term but longer-term things could be better (e.g. 2008 mortgage crisis and other events).

Fig. 6 shows an example of data exploration for the term structure spread of 7-mth minus 2-mth VIX futures vs. the 1-mth VIX futures contract 3-days forward. There is evidence of all three term structure states. Contango constitutes a majority of the data points, fewer points in flattening and the fewest points in backwardation (since a downturn or market crisis is less frequent). Backwardation occurs at extreme levels, such as during the 2008 subprime mortgage crisis.

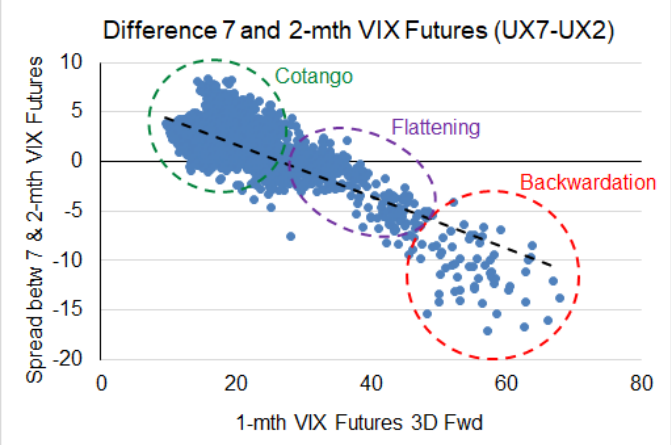


Fig. 6. For 7-Mth minus 2-Mth VIX Futures Terms Structure Spread, Evidence of Contango, Flattening & Backwardation (Jul 2006 to Jun 2018)

Skew (30 inputs). Skew represents the uncertainty or fear of a downside event at a particular maturity or time. The skew is the difference in implied volatility between the two strikes at a particular maturity. Unlike most stocks and indices where puts generally have high skew, calls generally have higher skew for the VIX, since the VIX is negatively correlated to the returns of the S&P 500. Typically, skew uses at-

the-money (ATM) strikes (current level) and several out-of-the money (OTM) strikes for the same maturity. In our analysis, the skew is calculated for multiple maturities. There is upside call skew and downside put skew for the VIX. In this paper, our data includes skew differences between 120% OTM and 80% OTM options, ATM (100%) and 80% OTM options, ATM (100%) and 120% OTM options, ATM (100%) and 150% OTM options, and ATM (100%) and 200% OTM options. The skew calculations are calculated for multiple maturities (1-mth, 2-mth, 3-mth, 6-mth, 9-mth, and 12-mth). There is a total of 30 skew input variables.

Fig. 7 shows the different skew pattern in different market environments. In a non-volatile or normal market, OTM calls have a slightly steep skew because in non-volatile times OTM protection is generally sold at a premium. In a market with some volatility, front month ATM implied volatility likely shifts higher and curve parallel shifts higher and so the need to charge more for OTM calls is reduced since volatility is already elevated. During a highly volatile market event, OTM calls are offered at a larger premium creating a much steeper skew.

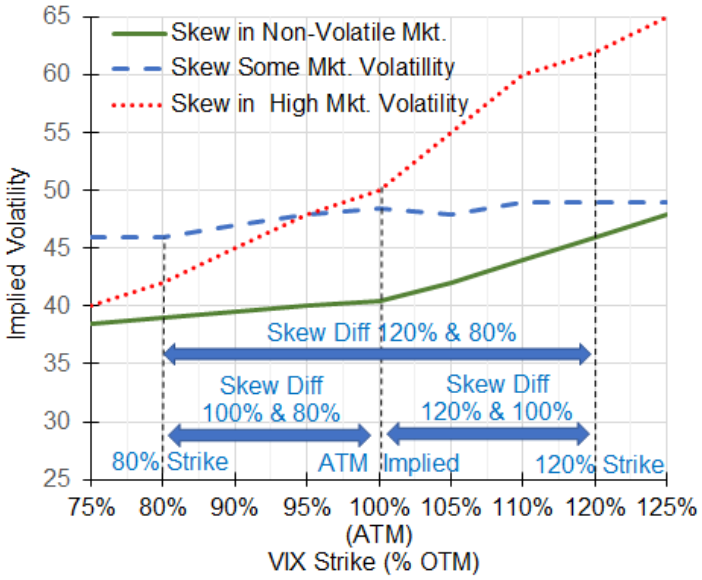


Fig. 7. Different skew patterns for less volatile to high volatile markets

Fig. 8 shows an example of data exploration for skew of 1-mth 150% OTM minus 100% ATM options vs. the 1-mth VIX futures contract 3-days forward. There is evidence of all three skew states. Less market volatility constitutes a majority of the data points for the S&P 500, fewer point in some market volatility and the fewest points in high market volatility.

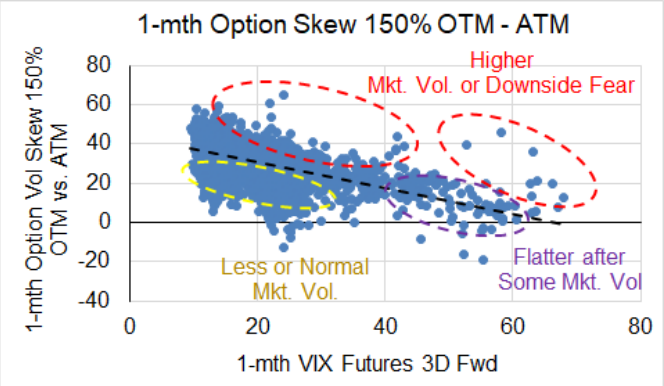


Fig. 8. Skew of 1-Mth 150% OTM minus ATM VIX Calls vs. 1-Mth VIX Futures 3D Fwd. (Jul 2006 to Jun 2018)

Technical Variables (11 inputs). There are 6 input variables for when the VIX level crosses above or below the prior 14, 50 and 100-day moving average (MA) using business days. An additional signal variable is calculated when the 14, 50 and 100-day moving average is exceeded for three days in a row creating 3 more input variables. In addition, Bollinger Bands are the two standard deviations (SD) levels away from a simple moving average. Typically, the price of the index is bracketed by an upper and lower 2-SD band using a 21-day simple moving average (1-mth in business days). Since standard deviation is a measure of volatility, when the markets become more volatile, the bands widen; during less volatile periods, the bands contract. When the VIX level cross the upper and lower Bollinger band based on the current VIX level, a signal is generated creating two more input variables.

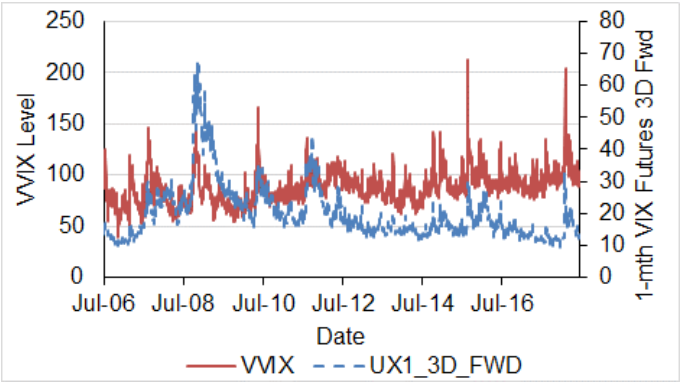


Fig. 9. VVIX vs. 1-Mth VIX Futures 3-Days Fwd. (Jul 2006 to Jun 2018)

The VVIX (2 inputs). The VVIX is 1-mth ATM option implied option volatility on the VIX itself. Fig. 8 shows the VVIX (left axis) vs. 1-mth VIX futures 3-days

forward (right axis) and they are very correlated. The series has history back to July 2006. VVIX is an input variable along with the intraday high minus low of the VVIX.

Response Variables: 1-Mth VIX Futures Levels 3 and 5 Days Forward (2 outputs). The outputs are forecasted separately by all the methods and methods. Fig. 10 shows 1-mth VIX futures contract (UX1) both 3 and 5 days forward historically from November 2006 to June 2018.

Autocorrelation in Response Variables: Autocorrelation is present in our two response variables UX1 3 and 5-days forward as show in Fig. 10. The maximum autocorrelation for both of our 3 and 5-days response variables occur at 1 lag as shown in Fig. 10. This will be useful when analyzing the ARIMA process.

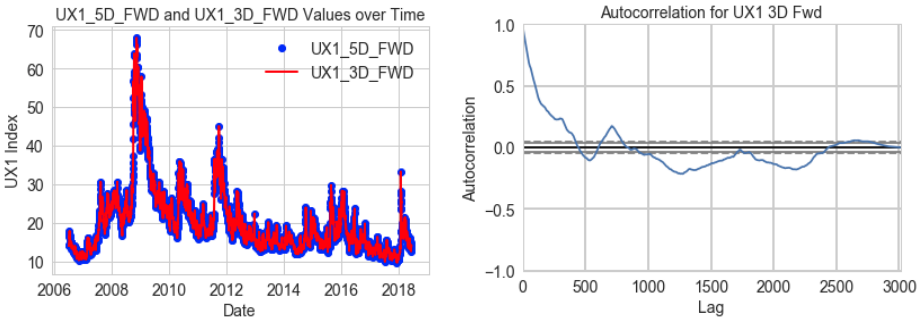


Fig. 10. 1-Mth VIX Futures Contract (UX1) 3 and 5-Days Forward and Autocorrelation Lag of 1 (Jul 2006 to Jun 2018)

2.4 Reduce Dimensionality or Feature Selection

The analysis in this paper has additional goals for both the common financial models and most machine learning models. With 71 input variables, there is multi-collinearity that inflates the variance explained by an R^2 from a simple linear regression or that inflates the assessed quality of the results. As shown in Appendix 5, the cross correlation of the term structure spreads and skews for many different combinations exceeds 66%. In addition, some models perform feature selection to select the input variables that explain most of the variance in data. Therefore, the first goal is to reduce dimensionality or perform feature selection.

2.5 Assessing Quality of Models: Metrics

The second goal to determine or assess the quality of the output using similar evaluation metrics. Accuracy or R^2 is our first metric that determines how well the model or methods is working overall. Our second set of metrics is based on estimated versus actual values of the test data and training data input. The test data

actual versus estimated is more important in this analysis. The metrics, using actual and estimated data sets, are mean squared error (MSE), mean absolute error (MAE), variance explained, and correlation. Finally, an accuracy matrix check is performed. This accuracy matrix is similar to a confusion matrix used for machine learning supervised classification problems. For our regression problem, the positive or up and negative or down moves of the estimated test data set are examined against the actual test data. True positive, true negative, false positive and false negative percentages are then calculated for our estimated versus actual test data. For further information on how the values of the matrix are calculated see Appendix 6.

3 Methods, Models and Workflow

The methods are separated into two sub-sections. The first section applies and assesses the quality of existing or common financial modeling methods of forecasting market volatility using MLR, PCA and ARIMA. The second section applies and assesses six machine or deep learning supervised based methods using SVR, Ensemble, LASSO, RF, RNN and LSTM.

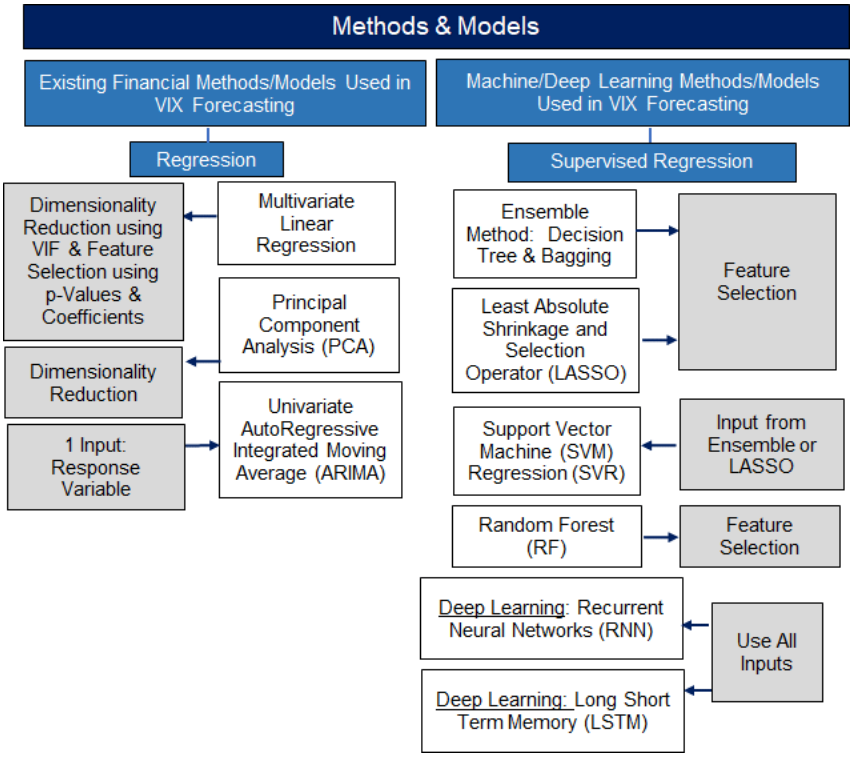


Fig. 11. Model/Methods used for Existing (Common) and Machine/Deep Learning Methods

Fig. 11 shows the methods and models applied to our data for both existing (common) and machine learning models and outlines whether the model performs feature selection or a reduction in dimensionality.

Fig. 12 shows the workflow of evaluating a total of nine models. The workflow includes creating and validating the training and test data sets; selecting the model; adjusting/optimizing hyper-parameters (input parameter to model); assessing the quality of the output for the method; and performing feature selection or dimensionality reduction on our inputs or explanatory variables. In Python, GridSearchCV was used to optimize hyper-parameters of most models. Once the best model is found for that method, all the best models for each method are compared to determine the best method and model for our training and test data sets.

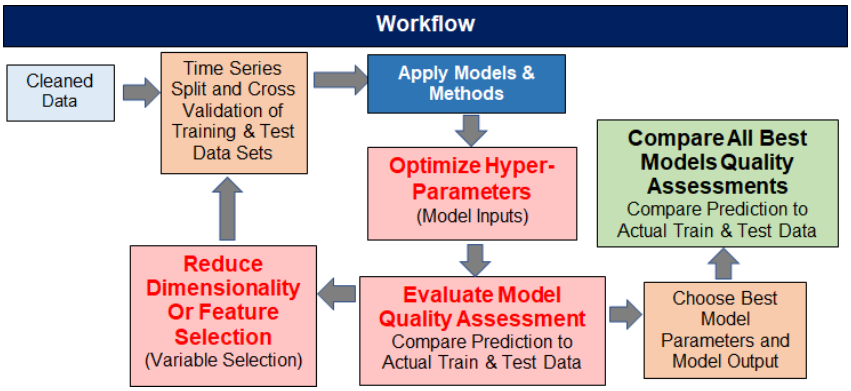


Fig. 12. Workflow used for All Models and Methods

3.1 Existing (Common) Financial Methods/Models for VIX Forecasting

Table 3 shows the common or existing financial methods with their inputs and quality assessment metrics.

Table 3. Common Financial Methods with their Inputs and Quality Assessments

Method	Dimensionality Reduction / Feature Selection	Input Selection	Quality Assessment
Multivariate Linear Regression (MLR)	Regression	Dimensionality reduction and feature selection by normalize data, Train & Test Data Variables using p-values, VIF and large coefficient values	Scatter Plot, R^2, Added R^2, MSE, Error histogram, Correl. Act. vs. Est.*, Accuracy Matrix
Principal Component Analysis	Dimensionality Reduction by creating Orthogonal Principal Components (PCs) followed by Regression	Dimensionality Reduction based on variance explained, coefficients and scores fed back into linear regression model	Explained variance, Scatter Plot, R^2,MSE, Error Histogram, Correl. Act. vs. Est.*, Accuracy Matrix
Autoregressive Integrated Moving Average (ARIMA)	ARIMA with lag, Response Variable is only variable	Determined autoregression lag.	Explained variance, R^2,MSE

*Note that 'Correl. Act. vs. Est' is the correlation of the actual training or test data set to the estimated or estimated training or test data set.

For Table 3, the common quality assessment metrics are detailed in section 2.5 of this paper. In addition to those metrics, MLR also used additional R^2, variance inflation factor, magnitude of coefficients (using normalized data) and p-value to reduce dimensionality and perform feature selection.

Multivariate Linear Regression (MLR). For multivariate linear regression, the data is first normalized, and the inputs can be reduced by ranking high to low coefficient values, p-values <0.05, and variance inflation factors (VIFs) < 10%. The best inputs for the regression model are found and the quality is assessed.

Principal Component Analysis (PCA). For all 71 inputs, PCA reduces the dimensionality of the data set by creating orthogonal factors. The eigenvalues and eigenvectors are used to create input variables for the linear regression model used to estimate our test and training data. The optimal number of principal components (PCs) is found using the explained variance and minimum MSE by testing the addition of another PC. The model quality is then assessed.

Univariate Autoregressive Integrated Moving Average (ARIMA). ARIMA fits the time series data to predict future points in the series (forecasting). This is applied for the univariate case in this paper. In the univariate case, the input variable is the response variable to forecast the response variable in the future.

3.2 Machine Learning Supervised Regression Methods

Table 4 shows the machine learning supervised regression methods, their inputs and their quality assessment metrics. The quality assessment (see section 2.5) is similar

to the existing financial models in section 3.1. The ensemble method provides a ranking of each input by their importance that is used to reduce the input features. The most important factors are inputs into the better models for ensemble (in our case, decision tree using bagging regression) that incorporates the prior error term. LASSO reduces dimensionality by a penalty factor and then uses the final features selected as inputs in a linear regression. For SVR, the most important factors from the ensemble and LASSO methods. The inputs using ensemble had the better results for our SVR model. RF optimized the most important features. For RNN and LSTM, all inputs are used. For more information on each of the machine learning models see Appendix 7.

Table 4. Machine Learning Supervised Regression Models/ Methods with their Inputs and Quality Assessments

Method	Machine Learning	Input Selection	Quality Assessment
Ensemble Method Output into Linear Regression with Prior Error Term	Supervised Regression	Feature Selection by selecting most important input variable or factors	Scatter Plot, R^2,MSE, MAE, Error Histogram, Correl. Act. vs. Est.*, Accuracy Matrix
Least Absolute Shrinkage & Selection Operator (LASSO)	Supervised Regression	Feature Selection using high alpha=0.95 to penalize and eliminate input variables to less than 15	Same as above
Support Vector Regression (SVR)	Supervised Regression	Input most important features selected by Ensemble and LASSO	Same as above
Random Forest (RF)	Supervised Regression	Input most important features selected by RF Method	Same as above
Recurrent Neural Networks (RNN)	Supervised Regression	Implementation using all 71 inputs where neural network has memory, iterates to reduce RMSE & loss	Performance, Scatter Plot Act. vs. Est., RMSE Plot, Error Histogram, MSE, MAE, Correl. Act. vs. Est.*, Accuracy Matrix
Long Short-Term Memory (LSTM)	Supervised Regression	Implementation using all 71 inputs where neural network has memory, iterates to reduce RMSE & loss	Performance, Scatter Plot Act. vs. Est., RMSE Plot, Error Histogram, MSE, MAE, Correl. Act. vs. Est.*, Accuracy Matrix

*Note that 'Correl. Act. vs. Est' is the correlation of the actual training or test data set to the estimated or estimated training or test data set.

4 Results

This section details the best model results with the optimized hyper-parameters for each method. For the plots and graphs in this this section, the traditional 75% training and 25% test data is used. However, the table of model quality assessment shows a summary of 10-split time series cross-validation results versus the traditional 75% train/25% test split. Section 5 of this paper analyzes the best model for each method and compares them to determine the overall best method using its best model.

4.1 Common Model: Multivariate Linear Regression (MLR)

Dimensionality Reduction for MLR. With all 71 input, the R^2 of a simple ordinary least squares (OLS) regression is 86.9% and with our reduced inputs of 13 variables the R^2 is 80.8% for 1-mth VIX futures 3-days forward. To reduce the dimensionality of our 71 inputs, the data was first normalized. For each regression, variables with p-values > 0.05 were removed. Second, the largest coefficients by absolute value for each input are kept. Third, the larger additional R^2 values for each input variable are kept because that input explains more of the overall variance. Fourth, the variance inflation factor (VIF) of each variable was calculated and those with VIFs > 10% were removed. The MLR was reduced to 13 inputs, all with VIFs below 7%, resulting in a model with an R^2 of 80.8% for 3-days forward. Appendix 8 shows the results using these metrics in the final run resulting in the reduction to 13 input variables

Inputs after Dimensionality Reduction. M3_200_100, M1_150_100, UX3_HILO, VVIX_HILO, BOLL_XUPPER, UX7MUX2, M2_120_80, SIGBUY14D3CD, M2_150_100, M2_200_100, UX6MUX4, UX6_HILO and M12_120_80. See Appendix 3 for descriptions of each variable. The same set of input variable using the same selection method were determined for forecasting the response of 1-mth VIX Futures both 3 and 5 days forward.

Quality Assessment of Results for MLR. Fig. 13 shows the MLR scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for UX1 3 days forward. The scatterplots show generally a linear relationship for both the test and training estimates for 3 days forward, but it has some variance. In addition, Fig. 13 shows the MLR error histogram of the actual versus estimated for the test data sets for UX1 3 days forward. The test data error histograms are left skewed due to the February 2018 inflation scare that caused volatility to jump. In addition, MLR shows variance in the error terms. Similar results exist for 5-days forward as shown in Fig. 14. Appendix 9 contains the complete test and training data graphs and tables for the MLR analysis for 1-mth VIX futures both 3 and 5 days forward.

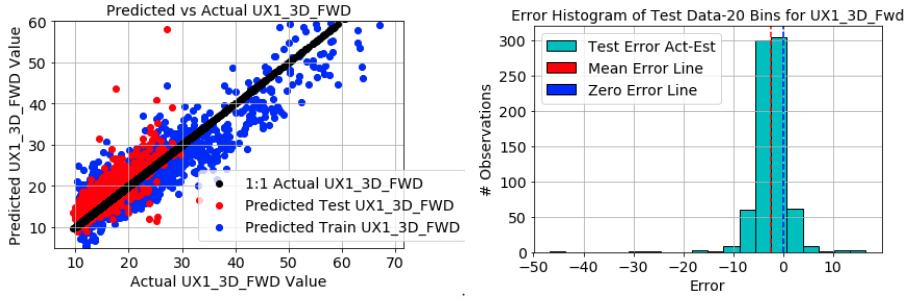


Fig. 13. MLR Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3-days Forward and Error Histogram of Estimated Test vs. Actual for UX1 3-days Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

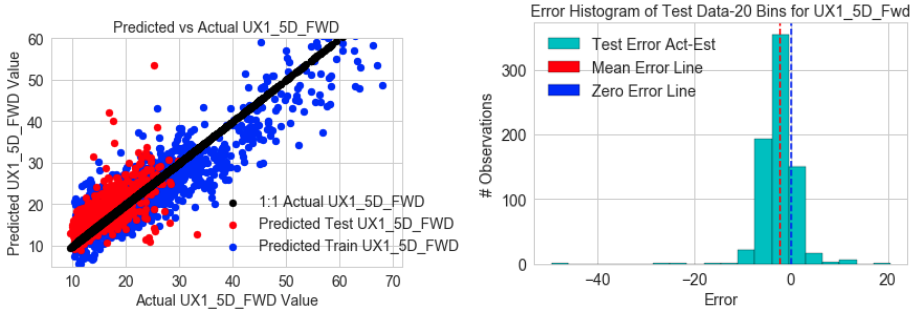


Fig. 14. MLR Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 5-days Forward and Error Histogram of Estimated Test vs. Actual for UX1 5-days Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test).

Table 5 shows a summary of results for both our 10-split cross validation and the traditional 75%/25% train/test split. Using 10-split cross validation, the MSE of the test data is higher and the variance explained (R^2) of the test is higher than the traditional split. For the output of our accuracy matrix, see Appendix 9.

Table 5. Some Quality Assessment Results of MLR Model

Output	Inputs	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	ρ(train)*	ρ(test)*	R^2_{test}	MSE_{test}
3D Fwd.	13	0.81	0.16	15.22	18.94	0.91	0.73	0.325	26.76
5D Fwd.	13	0.79	-0.05	17.25	22.09	0.89	0.63	0.315	29.34

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

4.2 Common Model: Principal Components Analysis (PCA)

Here, a PCA model is analyzed for the common or existing financial models. The data is first normalized prior to using PCA and the output is unnormalize for our graphs.

Dimensionality Reduction for PCA. Fig. 15 shows that the PCA model reduces the dimensionality from 71 inputs to 10 principal components (PCs) that explain over 90% of the variance of the model for both UX1 3 and 5-days forward. In the second graph, the number of PCs is chosen at the lowest MSE, which is 10. Similarly, in Appendix 10, maximum accuracy is shown to be optimized at 10 PCs.

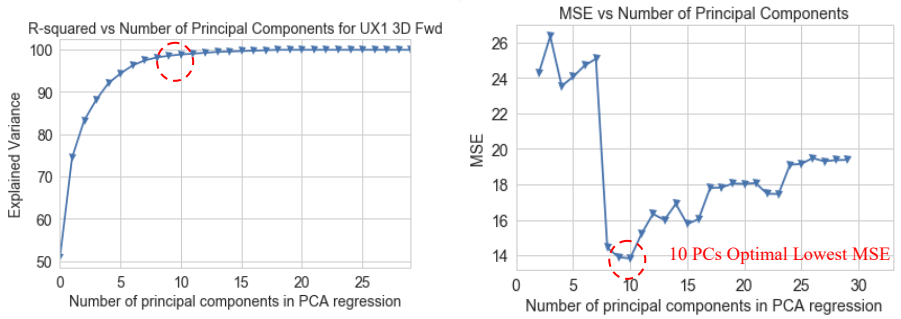


Fig. 15. PCA Reduction to 10 Principal Components (PCs) with Explained Variance over 90% for 1-mth VIX Futures (UX1) 3 and 5-days Fwd. In addition, the second graph shows that with 10 PCs the MSE is minimized for both 3 and 5-days Fwd. (Jul 2006 to Jun 2015)

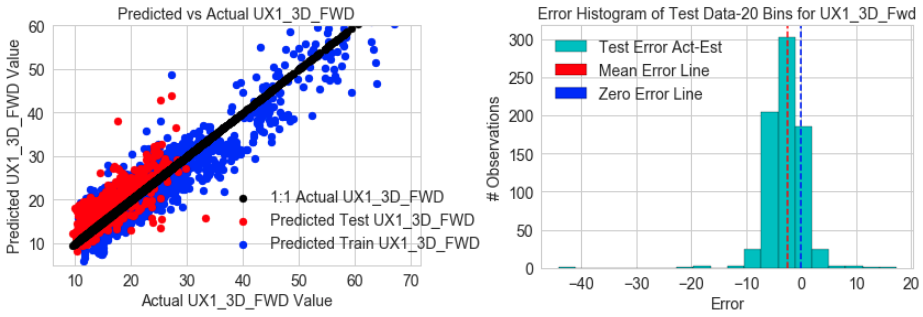


Fig. 16. PCA Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3-days Forward and Error Histogram of Estimated Test vs. Actual UX_3D_FWD (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Quality Assessment of Results for PCA. Fig. 16 shows the PCA scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for UX1 3-days forward. The scatterplots show generally a linear relationship for both the test and training

estimates with a slightly tighter variance in the test estimates. In addition, Fig. 15 shows the PCA error histogram of the actual versus estimated for the test data sets for UX1 3 days forward. The test data error histograms are still left skewed. Appendix 10 contains the complete test and training data graphs and tables for the PCA analysis for 1-mth VIX futures both 3 and 5 days forward.

Table 6 shows a summary of results for both our 10-split cross validation and the 75%/25% train/test split. Using 10-split cross validation, the MSE of the test data is slightly higher and the variance explained (R^2) of the test is higher than the traditional split. For the output of our accuracy matrix, see Appendix 10.

Table 6. Some Quality Assessment Results of PCA Model

Output	Inputs	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	ρ(train)*	ρ(test)*	R^2_{test}	MSE_{test}
3D Fwd.	10	0.86	0.22	11.80	19.38	0.93	0.70	0.339	29.10
5D Fwd.	10	0.84	0.03	13.77	21.93	0.92	0.61	0.334	30.39

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

4.3 Common Model: Univariate Auto-Regressive Integrated Moving Average (ARIMA)

Inputs: Univariate Autoregressive Integrated Moving Average (ARIMA) is a different model with only 1 input, the response variable. The response variable is used to forecast the future response.

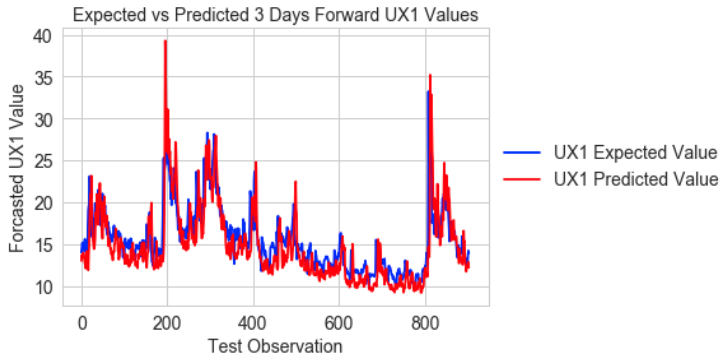


Fig. 17. ARIMA Scatter Plot of Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3-days Forward (Jun 2015 to Jun 2018 for Test)

For this to occur, there has to be autocorrelation in the variable as was shown in section 2.3 earlier in this paper. In section 2.3, the optimal lag for an ARIMA model was 1. Fig. 17 shows the actual versus the estimated 1-mth VIX 3-days forward for the ARIMA model. Fig. 18 shows the residuals which jump during high volatility moves; otherwise, variance is generally more consistent within a range for both UX1

3 and 5-days forward. Appendix 11 contains the complete test and training data graphs and tables for the ARIMA analysis for 1-mth VIX futures both 3 and 5 days forward.

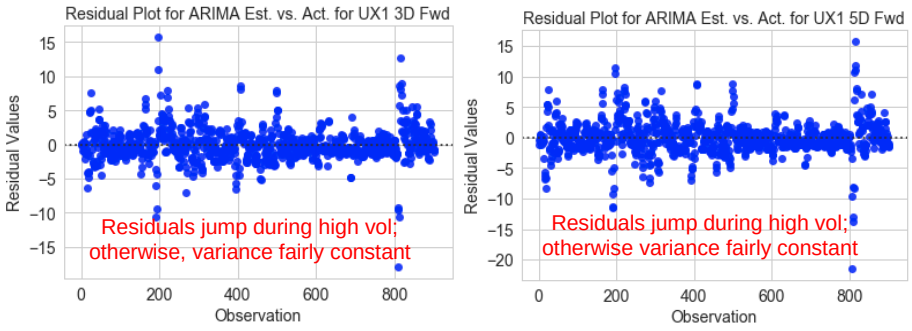


Fig. 18. ARIMA Residual Plot of Test Data for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jun 2015 to Jun 2018 for Test)

Table 7 is shows that the ARIMA model has a good explained variance and low MSE. However, it can be difficult to add more variables to the ARIMA model (multivariate ARIMA) compared to RNN and LSTM. In addition, ARIMA can have trouble forecasting inflection points based solely on the prior response level.

Table 7. Some Quality Assessment Results of ARIMA Model

Traditional 75%/25% Train/Test Split			
Output Forecasted	Inputs	R^2_{test}	MSE_{test}
3D Fwd.	1	0.52	6.44
5D Fwd.	1	0.36	8.63

4.4 Machine Learning: Ensemble Method

The ensemble method incorporates the error term from the forecast of the prior day. In our implementation, the data was first normalized, and then the ensemble method was used with a linear regression method, incorporating the prior error term into the forecast. In our case the error term cannot be known until 3 or 5 days from the closing price for each day in the dataset.

Feature Selection for Ensemble: Fig. 19 shows the top 15 predictors (input variables) plus 1 error term from our ensemble model for UX1 3 and 5 days forward. The top 15 predictors explain a majority of the variance and reduces the MSE to a minimum level.

Bootstrapping refers to any test or metric that relies on random sampling with replacement. It falls in to the broader class of resampling methods. It generates a new dataset for each ensemble member by bootstrapping, i.e. sample N items with

replacement from the original N. Bagging uses bootstrap sampling to obtain the data subsets for training the base learners. In addition, bagging uses averaging for regression.

In addition, ensemble usually adds an error term as an input to forecast the response variables after finding the optimal model. First, the error term for our dataset has to be moved forward 3 or 5 days because it is not known until the actual UX1 level 3 or 5-days forward is realized. Second, the error term is also predicted as a third response variable, which is not moved forward, since it is used as our training data response variable. The added error term improves the estimate. The predicted error term is added to the predicted UX1 levels 3 or 5-day forward using out data set with the error term as an input moved forward. In our case, ensemble chose decision trees as the best estimator.

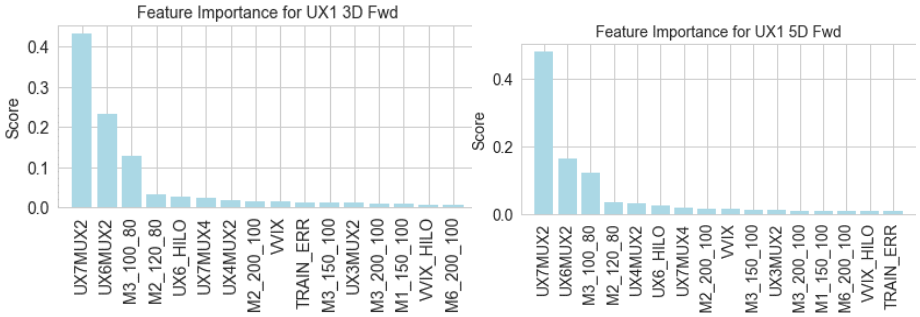


Fig. 19. Ensemble Top 15 Predictors plus 1 Error Term that Provide Optimal Results for UX1 3 and 5D Forward (Jul 2006 to Jun 2015)

Top Predictors (Inputs): UX6_HILO, VVIX, VVIX_HILO, UX4MUX2, UX3MUX2, UX7MUX2, UX7MUX4, UX6MUX2, M6_200_100, M3_150_100, M2_120_80, M3_100_80, M2_200_100, M3_200_100, M1_150_100 and TRAIN_ERR (training error term). See Appendix 3 for descriptions of each variable. The set of variables for 3 and 5-days forward is the same.

Optimization of Hyper-Parameters for BaggingRegressor Function in Python: The parameters are optimized by iterating using ParameterGrid for base estimator, maximum sample, maximum feature, and bootstrap (on or off) and bootstrap features (on or off). In addition, the base estimator iterates over estimators DecisionTree, DummyRegressor, DecisionTreeRegressor, KNeighborRegressor and SVR. The optimal hyper-parameters using the best estimator (DecisionTree) are all the samples (1.0), all the features (1.0), bootstrapping (True) and bootstrap features (False).

Quality Assessment of Results for Ensemble Incorporating Error Term: Fig. 20 shows the ensemble scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for UX1 both 3 and 5 days forward. The scatterplots show an estimate with increasing variance as volatility increases compared to the 1 to 1 plot line for the test estimate while the training estimates shows better results and a tighter variance versus the 1 to 1 plot. Appendix 12 contains the complete test and training data

graphs and tables for the ensemble analysis for 1-mth VIX futures both 3 and 5 days forward.

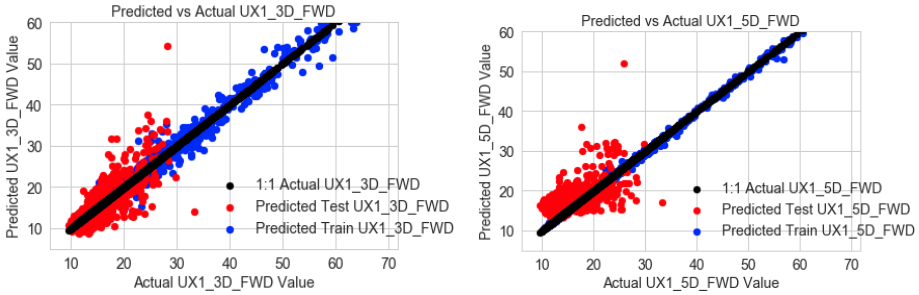


Fig. 20. Ensemble Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures 3 and 5 days Forward

Table 8 shows a summary of results for both our 10-split cross validation and the 75%/25% train/test split. The ensemble decision tree (DT) using bagging regression with a prior error term (DT with error term) shows great results for our traditional 75% train/25% test data split with a high explained variance (R^2) and low MSE but the 10-split time series cross validation shows a higher MSE and much lower explained variance. The higher MSE for the 10-split cross validation is due to much less accurate predictions of inflection points, such as the mortgage crisis of 2008 (the Great Recession) and the European debt crisis (the PIGS). Additionally, our model attempts to capture these inflection points. Similarly, for UX1 5D forward, the predictions or estimates also have good results for our 75% training /25% test data but worse results using our 10-split time series cross validation. For the output of our accuracy matrix, see Appendix 12. Once again, the accuracy matrix is good for the traditional split UX1 3D forward but less accurate for the traditional split of UX1 5D forward.

Table 8. Some Quality Assessment Results of Ensemble Decision Tree using Bagging Regression with Prior Error Term

Output	Inputs	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	ρ(train)*	ρ(test)*	R^2_{test}	MSE_{test}
3D Fwd.	16	0.98	0.40	1.58	9.11	0.99	0.80	0.05	43.49
5D Fwd.	16	0.99	0.26	0.14	15.57	0.99	0.59	-0.19	49.45

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

4.5 Machine Learning: Least Absolute Shrinkage and Selection Operator (LASSO)

For the Least Absolute Shrinkage and Selection Operator (LASSO) method, the data was first normalized and then then the linear model for LASSO was run in python

('linear_model.Lasso'). The LASSO performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the statistical model.

Dimensionality Reduction for LASSO: For UX1 3D forward, LASSO reduced the input dimensions from 71 to 16 and for 5D forward, from 71 to 15. LASSO reduces the number of predictors, identifies important predictors, selects among redundant predictors and produces shrinkage estimates with lower predictive errors than ordinary least squares. The selected input variables of LASSO are then used to select the final inputs of the linear regression model.

Top Predictors (Inputs): UX1 3D forward has 16 inputs and UX1 5D Forward has 15 inputs with a 94% overlap. LASSO for UX1 3D forward has the following inputs: UX7MUX2, UX8MUX2, VVIX, VVIX_HILO, M1_120_80, M1_150_100, M1_200_100, M2_120_80, M2_100_80, M2_200_100, M3_120_80, M3_100_80, M3_200_100, M6_120_80, M6_100_80, M12_200_100. LASSO for UX1 5D forward has all the same input excluding one, M2_200_100. See Appendix 3 for descriptions of each variable.

Optimization of Hyper-Parameters for LASSO: Alpha is the elasticity factor that controls the balance between lasso and ridge penalties. Our analysis uses a higher alpha of 0.95 (testing a range between 1.0 and 0) to reduce the MSE for both UX1 3 and 5-days forward shown in Fig. 21. The objective function is following:

$$\min_w [\ (1 / (2 * n_{samples})) * \|X\text{-}y\|^2_2 + \alpha * \|w\|_1\ ] \qquad (1)^6$$

The lasso estimate thus solves the minimization of the least-squares penalty with $\alpha*\|w\|_1$ added, where α is a constant and $\|w\|_1$ is the L1-norm of the parameter vector. The higher the alpha value, more restriction on the coefficients; while the lower the alpha, more generalization and coefficients are barely restricted (at zero, it becomes a simple linear regression). The maximum number of iterations does not seem to matter so we set it at 10k.

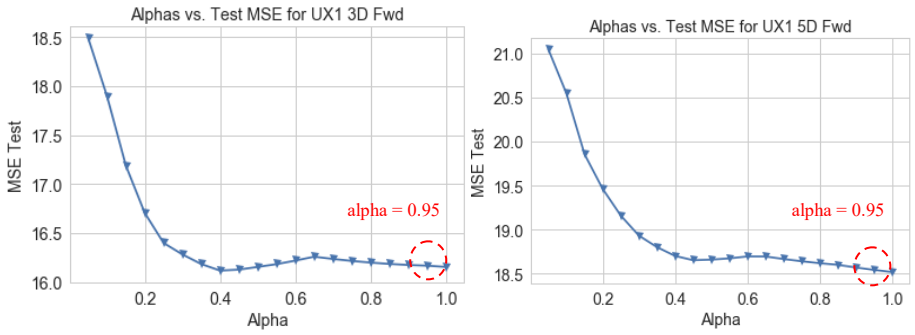


Fig. 21. LASSO Alphas versus MSE for test data for both UX1 3 and 5-days forward (Jun 2015 to Jun 2018)

[6] http://scikit-learn.org/stable/modules/linear_model.html

Quality Assessment of Results for LASSO: Fig. 22 shows the LASSO scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for both UX1 3 days forward. The scatterplots show generally a linear relationship for both the test and training estimates for 3 days forward. In addition, Fig. 22 shows the LASSO error histogram of the actual versus estimated for the test data sets for UX1 for 3 days forward. The test data error histograms are slightly right skewed but more normal than other models so far, indicating a slightly better fit using LASSO. Similar results exist for 5-days forward as shown in Fig. 23. Appendix 13 contains the complete test and training data graphs and tables for the LASSO analysis for 1-mth VIX futures both 3 and 5 days forward.

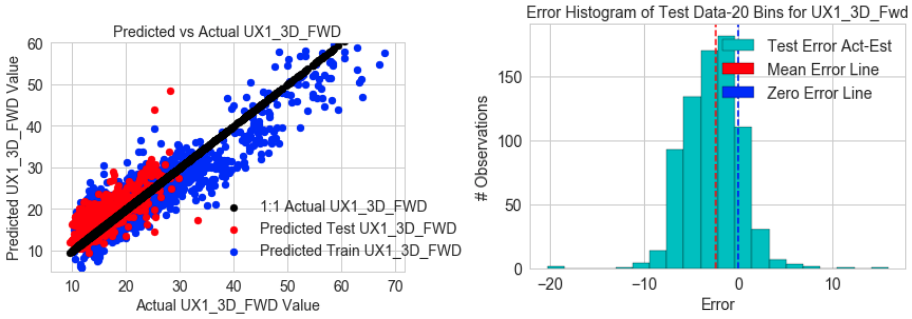


Fig. 22. LASSO Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures 3-days Forward and Error Histogram of Estimated Test vs. Actual UX_3D_FWD (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig. 23. LASSO Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures 5-days Forward and Error Histogram of Estimated Test vs. Actual UX_5D_FWD (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Table 9 shows a summary of results for both our 10-split cross validation and the 75%/25% train/test split. Using 10-split cross validation, the MSE of the test data is higher and the R^2 of the test is higher than the traditional split. The results so far

look very good compared to the models analyzed so far except the MSE for our 10-Split cross-validation is higher. For the output of our accuracy matrix, see Appendix 13.

Table 9. Some Quality Assessment Results of LASSO

Output	Inputs	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	ρ(train)*	ρ(test)*	R^2_{test}	MSE_{test}
3D Fwd.	16	0.83	0.39	14.21	16.16	0.91	0.72	0.33	42.75
5D Fwd.	15	0.81	0.22	16.09	18.54	0.90	0.62	0.32	53.64

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

4.6 Machine Learning: Support Vector Regression (SVR)

For the Support Vector Machine Regression (SVR) method, the data was first normalized.

Dimensionality Reduction for SVR: For SVR, the top features from the ensemble and LASSO model are used as optimized inputs. The inputs from ensemble worked the best and ensemble reduced dimensionality to 15 inputs.

Top Predictors (Inputs): UX6_HILO, VVIX, VVIX_HILO, UX4MUX2, UX3MUX2, UX7MUX2, UX7MUX4, UX6MUX2, M6_200_100, M3_150_100, M2_120_80, M3_100_80, M2_200_100, M3_200_100, M1_150_100. See Appendix 3 for descriptions of each variable. The input variables for 3 and 5-days forward are the same.

Optimization of Hyper-Parameters for SVR: The parameters optimized are the following: the better kernel is linear; penalty factor (c) is 0.1; max iterations = 10k; and tolerance is 0.0001. The better kernel is linear but the sigmoid, rbf, and poly kernels were tested as well. The penalty factor of the error term was moved to 0.1 with the better results, after testing a range from 1.0 to 0.01. For large values of (c), the optimization will choose a smaller-margin hyperplane if that hyperplane does a better job of getting all the training points classified correctly. Conversely, a very small value of (c) will cause the optimizer to look for a larger-margin separating hyperplane, even if that hyperplane misclassifies more points. A hard limit of 10K for number of iterations was set. The criteria of tolerance for stopping was made tighter from 0.001 to 0.0001 to achieve better results.

Quality Assessment of Results for SVR: Fig. 24 shows the SVR scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for both UX1 3 days forward. The scatterplots show generally a linear relationship for both the test and training estimates for 3 days forward; however, there are a few data points with large variances from the 1 to 1 line. In addition, Fig. 24 shows the SVR error histogram of the actual versus estimated for the test data sets for UX1 for 3 days forward. The test data error histograms are only slightly left skewed but still closer to normal, indicating a better fit. Similar results exist for 5-days forward as shown in Fig. 25. Appendix 13 contains the complete test and training data graphs and tables for the SVR analysis for 1-mth VIX futures both 3 and 5 days forward.

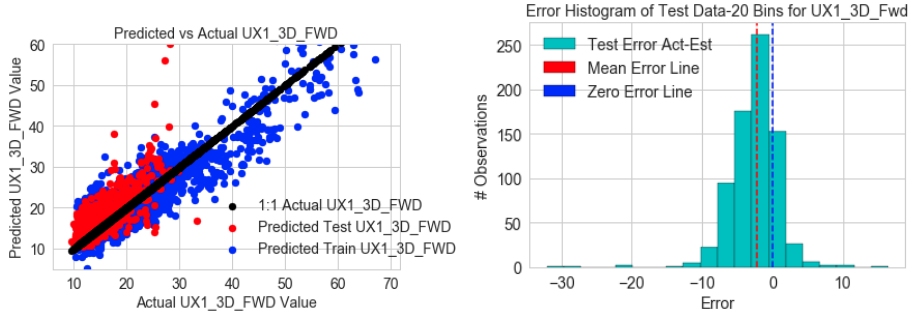


Fig. 24. SVR Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures 3-days Forward and Error Histogram of Estimated Test vs. Actual UX_3D_FWD (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig. 25. SVR Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures 5-days Forward and Error Histogram of Estimated Test vs. Actual UX_5D_FWD (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Table 10 shows a summary of results for both our 10-split cross validation and the 75%/25% train/test split. Using 10-split cross validation, the MSE of the test data is higher and the R^2 of the test is higher than the traditional split. The results so far look very good compared to the models analyzed so far except the MSE for our 10-Split cross-validation is high. For the output of our accuracy matrix, see Appendix 14.

Table 10. Some Quality Assessment Results of SVR

Output	Inputs	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	$\rho(train)^*$	$\rho(test)^*$	R^2_{test}	MSE_{test}
3D Fwd.	15	0.82	0.19	18.81	15.11	0.91	0.72	0.34	30.28
5D Fwd.	15	0.80	0.12	18.41	16.85	0.90	0.63	0.34	28.99

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

4.7 Machine Learning: Recurrent Neural Networks (RNN)

In traditional neural networks, all inputs and outputs are independent with no memory of prior levels. However, RNNs and LSTMs have "memory" to capture information about what is already calculated in the prior time series. Three of the many factors to optimize in neural networks (RNN and LSTM) are number of epochs, batch size and number of iterations.

Table 11 defines these inputs to the model. For batch size, 44 business days (2-mth) turns out to be optimal for RNN and 66 business days (3-mths), for LSTM. This makes sense since generally markets have shorter memories.

Table 11. Definition of Three Inputs in NN model for RNN and LSTM

Input Variable	Definition
1 Epoch	1 forward & 1 backward pass of all the training data
Batch Size	total number of data samples in a single batch for one forward and backward pass
Iterations	the number of batches or passes needed to complete 1 epoch
1 Pass	1 one forward and one backward pass

Inputs: All 71 inputs are utilized for both response variables
Optimization of Hyper-Parameters for RNN: The parameters optimized are the following using GridSearchCV in Python: optimizer is Adam; initialization mode is uniform; loss function is mean squared error; activation function is relu; number of neurons for each layer is 150; metric output is accuracy; epochs is 300; batch size is 44 (approximately two months of data); dropout rate is 0 and learning rate is 0.001. A smaller number of layers and neurons used due to our smaller data set of only 71 inputs of 3009 entries each. The number of hidden layers is 1 with 10 neurons with one output layer for our response variable. For the traditional 75% training / 25% test split, the training input size is 2256 by 71.
Quality Assessment of Results for RNN: Fig. 26 shows the validation accuracy versus loss per epoch for the training data, which shows that there is little improvement after 200 epochs for UX1 3 and 5-days forward. The lower the loss, the better a model (unless the model has over-fitted to the training data).

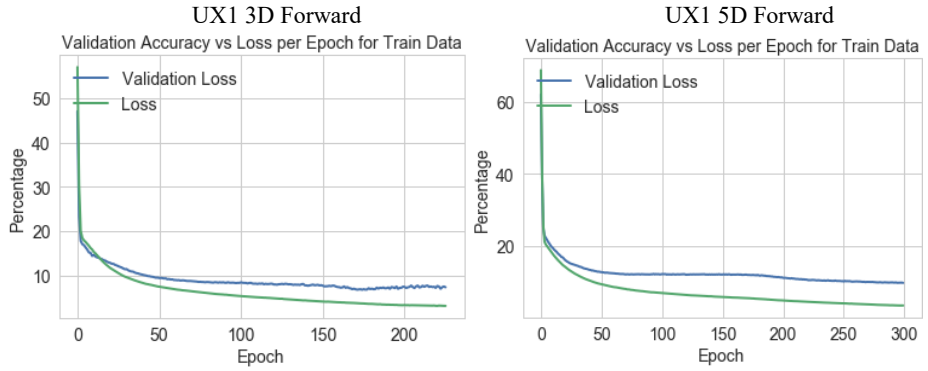


Fig. 26. Validation Accuracy versus Loss per Epoch for Training Data for both 1-mth VIX Futures 3 and 5-Days Forward

The loss is calculated on training and validation. The interpretation of the loss is how well the model is doing for these two sets. Unlike accuracy, loss is not a percentage. It is a summation of the errors made for each example in training or validation sets.

Fig. 27 shows the RNN scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for both UX1 3 and 5-days forward. The scatterplots show generally a linear relationship for both the test and training estimates for 3 days forward. In addition, Fig. 27 shows the RNN error histogram of the actual versus estimated for the test data sets for UX1 for 3 days forward. The test data error histograms are closer to a normal distribution, indicating a better fit and the variance of the test estimated are closer to the 1 to 1 line, indicating less variance. Similar results exist for 5-days forward as shown in Fig. 28. Appendix 15 contains the complete test and training data graphs and tables for the RNN analysis for 1-mth VIX futures both 3 and 5 days forward.

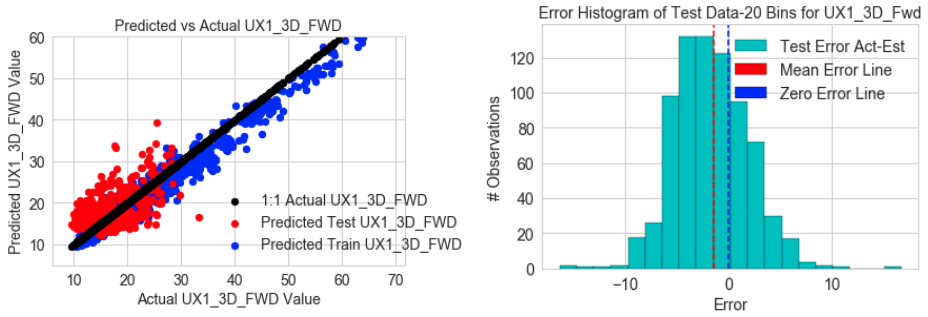


Fig. 27. RNN Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures 3-days Forward and Error Histogram of Estimated Test vs. Actual UX_3D_FWD (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

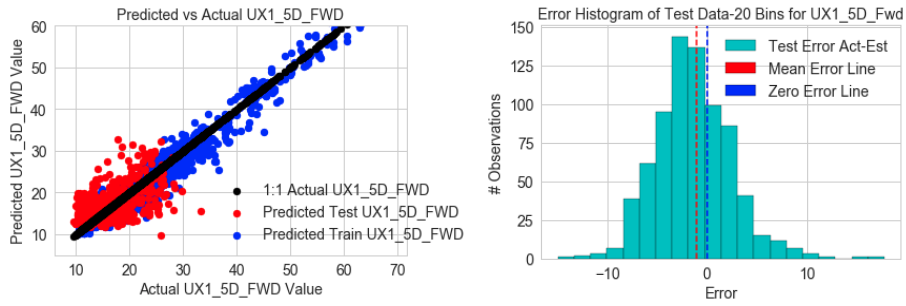


Fig. 28. RNN Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures 5-days Forward and Error Histogram of Estimated Test vs. Actual UX_5D_FWD (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Table 12 shows a summary of results for both our 10-split cross validation and the 75%/25% train/test split. Using 10-split cross validation, the MSE of the test data is higher and the R^2 of the test is about the same as the traditional split. Overall for both the traditional and 10-split cross validation, the results are very good compared to the models analyzed so far with higher variance explained (R^2) and lower MSE. For the output of our accuracy matrix, see Appendix 15.

Table 12. Some Quality Assessment Results of RNN

Output	Inputs	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	ρ(train)*	ρ(test)*	R^2_{test}	MSE_{test}
3D Fwd.	71	0.96	0.42	4.01	15.87	0.98	0.60	0.43	22.34
5D Fwd.	71	0.95	0.03	4.8	15.48	0.98	0.49	0.45	23.37

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

4.8 Machine Learning: Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) is similar to RNN but can have a longer memory of prior forecasts. Having multiple layers (a deeper network) makes your network more eager to recognize certain aspects of input data; however, our data is not as complex and only one hidden layer seems to improve performance over other models.
Inputs: All 71 inputs are utilized for both response variables.
Optimization of Hyper-Parameters for LSTM: The parameters optimized are the following using GridSearchCV in Python: optimizer is Adam; initialization mode is uniform; loss function is mean squared error; activation function is relu; number of neurons for each layer is 150; metric output is accuracy; epochs is 300, batch size is 66 (approximately three months of data); refit data is True; dropout rate is 0; and learning rate is 0.001. A smaller number of layers and neurons used due to our smaller data set of only 71 inputs of 3009 entries each. The number of hidden layers is 1 with 10 neurons with one output layer for our response variable. For the traditional 75% training / 25% test split, the input size is 2256 by 71.

Quality Assessment of Results for LSTM: Fig. 29 shows the validation accuracy versus loss per epoch for the training data, which shows that there is little improvement after 200 epochs for UX1 3 and 5-days forward. The lower the loss, the better a model (unless the model has over-fitted to the training data). The loss is calculated on training and validation and its interpretation is how well the model is doing for these two sets. Unlike accuracy, loss is not a percentage. It is a summation of the errors made for each example in training or validation sets.

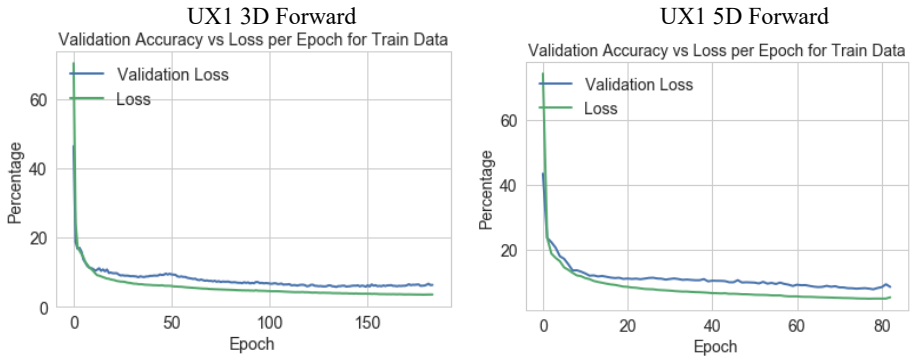


Fig. 29. Validation Accuracy versus Loss per Epoch for Training Data for both 1-mth VIX Futures 3 and 5-Days Forward

Fig. 30 shows the LSTM scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for both UX1 3 days forward. The scatterplots show generally a linear relationship for both the test and training estimates for 3 days forward. In addition, Fig. 30 shows the LSTM error histogram of the actual versus estimated for the test data sets for UX1 for 3 days forward. The test data error histogram has a left skew unlike RNN. Similar results exist for 5-days forward as shown in Fig. 31. Appendix 16 contains the complete test and training data graphs and tables for the LSTM analysis for 1-mth VIX futures both 3 and 5 days forward.

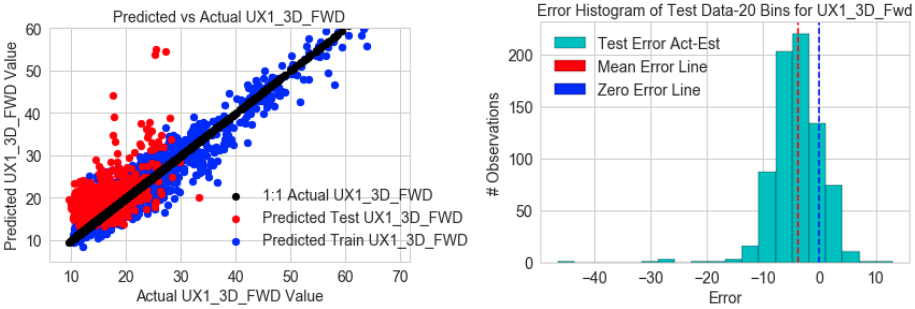


Fig. 30. LSTM Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures 3-days Forward and Error Histogram of Estimated Test vs. Actual UX_3D_FWD (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

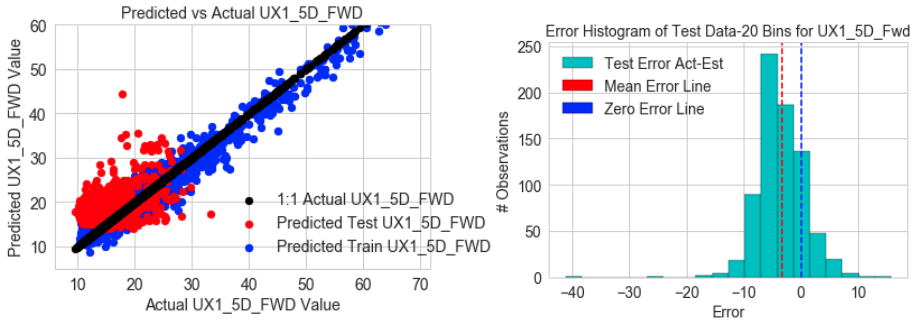


Fig. 31. LSTM Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures 5-days Forward and Error Histogram of Estimated Test vs. Actual UX_5D_FWD (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Table 13 shows a summary of results for both our 10-split cross validation and the 75%/25% train/test split. Using 10-split cross validation, the MSE of the test data is higher and the R^2 of the test is about the same as the traditional split. Overall for both the traditional and 10-split cross validation, the results are good compared to the models analyzed but still a slight left skew in the histogram and a bit more variance from the 1 to 1 line compared to RNN. The MSE is slightly higher for 10-split cross validation of the time series than for the traditional split. For the output of our accuracy matrix, see Appendix 16.

Table 13. Some Quality Assessment Results of LSTM

Output	Inputs	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	ρ(train)*	ρ(test)*	R^2_{test}	MSE_{test}
3D Fwd.	71	0.96	0.42	4.01	15.87	0.98	0.60	0.43	22.34
5D Fwd.	71	0.96	0.03	3.76	21.62	0.98	0.42	0.45	23.37

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

4.9 Machine Learning: Random Forest (RF)

Random Forest (RF) is an ensemble method that performs feature selection.
Top Features (Inputs): UX7MUX2, UX6MUX2, M3_100_80, UX5MUX2, UX6MUX3, M3_120_80, UX7MUX3, M2_120_80, M2_100_80, UX4MUX2, M2_200_100, UX7MUX4, UX2_HILO, and M12_200_100. See Appendix 3 for descriptions of each variable. See Appendix 3 for descriptions of each variable.

The top 14 input variables for 3 and 5-days forward are the same. And shown in Fig. 32.

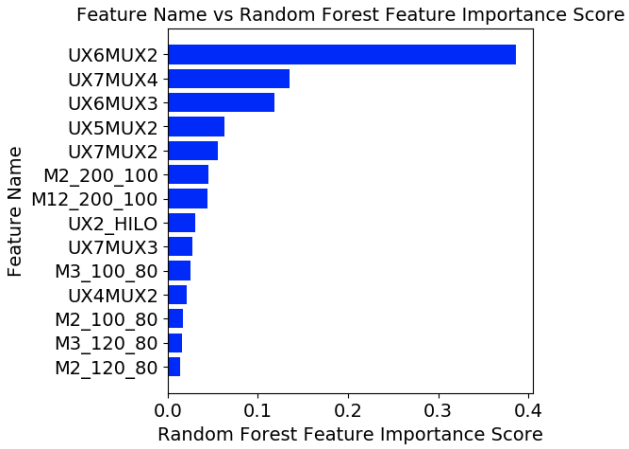


Fig. 32. Top 14 Features Selected for 1-mth VIX Futures 3 and 5-Days Forward

Optimization of Hyper-Parameters for RF: The parameters optimized are the following using GridSearchCV in Python: trees or estimators are 200, criterion is mean squared error, maximum depth has no limit, minimum leaf samples are 1, max features are auto, and bootstrap is True. Fig. 32 show the output of both 3 and 5-day feature selection using the top 15 factors to explain most of the variance.

Quality Assessment of Results for RF: Fig. 33 shows the RF scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for both UX1 3 days forward. The scatter plots show a bias to low ranges in the training data estimate which actually works for our test data estimate. Since the VIX generally stays at lower volatility levels, it makes sense a majority of the trees would have a lower range. Decision trees tend to have high variance when they utilize different training and test sets of the same data, since they tend to overfit on training data. This can lead to poor performance on forecasting inflection points. Unfortunately, this limits the usage of decision trees in predictive modeling as seen in our results. In addition, Fig. 33 shows the RF error histogram of the actual versus estimated for the test data sets for UX1 for 3 days forward. The test data error histogram has a right skew. Similar results exist for 5-days forward as shown in Fig. 34 but for 5-days the error histogram has more of a normal distribution. Appendix 17 contains the complete test and training data graphs and tables for the RF analysis for 1-mth VIX futures both 3 and 5 days forward.

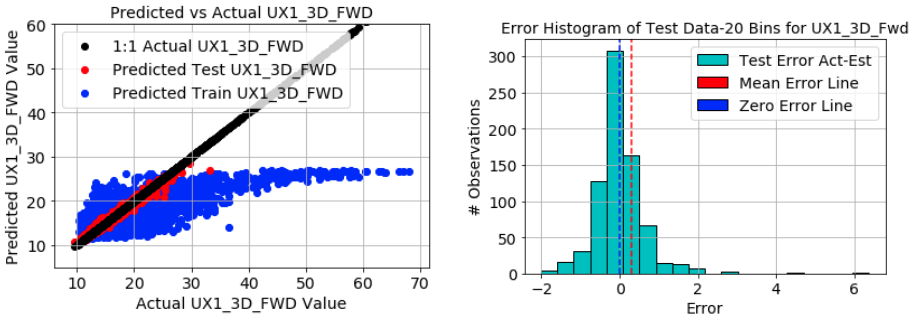


Fig. 33. RF Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures 3-days Forward and Error Histogram of Estimated Test vs. Actual UX_3D_FWD (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

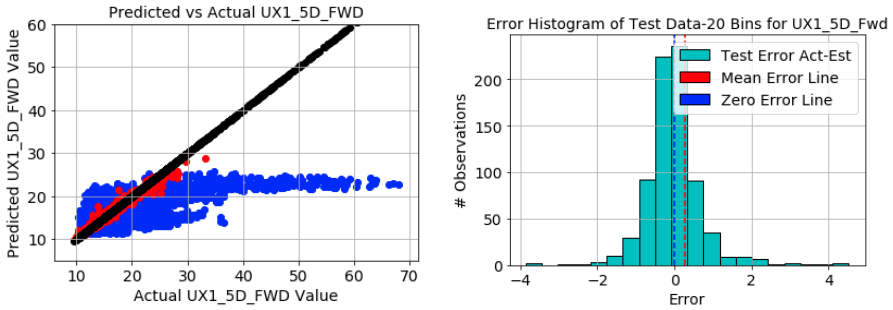


Fig. 34. RF Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures 5-days Forward and Error Histogram of Estimated Test vs. Actual UX_5D_FWD (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Table 14 shows a summary of results for both our 10-split cross validation and the 75%/25% train/test split. Using 10-split cross validation, the MSE of the test data is higher and the R^2 of the test is about the same as the traditional split. Overall for both the traditional and 10-split cross validation, the results are good compared to the models analyzed except for the bias toward a lower volatility forecast. The MSE is slightly higher for 10-split cross validation of the time series than for the traditional split. RF has some of the best quality metrics (high accuracy, low MSE, etc.); however similar to ensemble, predicting training data is biased to lower volatility forecasts due to the overfit even using the 10-split CV. For the output of our accuracy matrix, see Appendix 17.

Table 14. Some Quality Assessment Results of RF

Output	Inputs	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	$\rho(train)^*$	$\rho(test)^*$	R^2_{test}	MSE_{test}
3D Fwd.	14	0.43	0.97	62.93	0.41	0.71	0.98	0.37	45.52
5D Fwd.	14	0.33	0.96	74.55	0.54	0.61	0.98	0.35	50.34

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

5 Analysis

In this section, the results of choosing the best model for each method are compared for 1-mth VIX futures 3 and 5-days forward. In addition, the accuracy matrix calculations are presented and analyzed.

5.1 Analysis of Forecast Results for 1-Mth VIX Futures 3-Days Forward

Table 15 and 16 shows the result for the 1-mth VIX futures forecast 3 days forward across all models for both traditional 75% train/25% test split and cross-validation 10-split time series. The best first and second results for each column are highlighted in yellow. Across the multiple metrics, the machine/deep learning models RNN, LSTM, RF and the ensemble decision tree using bagging regressor with prior error term (Ensemble DT with Err. Term) have better quality assessment metrics compared to the other models. RNN has the best metrics for both the traditional 75% train/25% test split and the cross validation with 10 time series splits. Explained variance for the test data sets are generally low across most models. RF has great quality assessment, but it can be biased to lower volatility forecasts (see section 4.9). Similarly, the ensemble DT with error term (see section 4.4) shows great results for our traditional 75% train/25% test data split with a high explained variance (R^2) and low MSE but the 10-split time series cross validation shows a higher MSE and much lower explained variance, indicating potential overfitting using the traditional split. For RF and DT with error term, the higher MSE for the 10-split cross validation is due to much less accurate predictions of inflection points, such as the mortgage crisis of 2008 (the Great Recession) and the European debt crisis (the PIGS). Additionally, our model attempts to capture these inflection points.

Table 15. Quality Assessment Results of Best Models Using Cross Validation with 10 Time Series Splits for 1-mth VIX Futures 3-Days Forward

Method / Model	Input Reduced / Features Selected	Cross Validation with 10 Time Series Splits			
		MSE Test	MSE Train	R^2 / Var Explain Test	R^2 / Var Explain Train
RNN	71	22.34	13.00	0.429	0.870
RF	14	45.52	0.86	0.369	0.987
LSTM	71	79.16	19.37	0.289	0.665
Ensemble DT Err. Term	16	43.49	0.15	0.054	0.998
PCA	10	29.10	16.19	0.339	0.787
SVR	15	30.28	16.18	0.344	0.776
LASSO	16	42.76	13.41	0.326	0.811
MLR	13	26.76	14.37	0.325	0.814

Table 16. Quality Assessment Results of Best Models Using Traditional 75% Train / 25% Test Time Series Split for 1-mth VIX Futures 3-Days Forward

Method or Model	Input Reduced / Features Selected	Traditional 75% Train / 25% Test							
		MSE Test	MSE Train	MAE Test	MAE Train	R^2 / Var Expl Train	R^2 / Var Expl Test	Corr Train	Corr Test
RNN	71	15.87	4.01	3.24	1.58	0.959	0.421	0.98	0.60
RF	14	0.41	62.93	0.42	5.41	0.433	0.973	0.71	0.99
LSTM	71	22.69	3.81	3.66	1.50	0.956	-0.02	0.98	0.54
Ensemble	16	9.11	1.58	1.96	0.83	0.98	0.40	0.99	0.80
PCA	10	19.38	11.80	3.33	2.61	0.861	0.220	0.93	0.70
SVR	15	18.80	15.12	3.18	2.83	0.822	0.186	0.91	0.72
LASSO	16	16.17	14.21	3.20	2.84	0.832	0.390	0.91	0.72
MLR	13	18.94	15.22	3.09	2.98	0.820	0.155	0.91	0.73
ARIMA	13	6.44					0.521		

Our accuracy matrix compares the estimated and actual 1-mth VIX futures 3-days forward from the current level and determines if the forecast was actually higher or lower versus the estimated (see section 2.5 and Appendix 6). As shown in Table 17, the accuracy matrix shows that RNN, LSTM and RF are better predictors with high true positives and true negative rates, but also lower false positive rate compared to the other models. Most models have low false negative forecasts.

Table 17. Accuracy Matrix using Traditional 75%/35% of Data for 1-mth VIX Futures 3-Days Forward (Jun 2015 to Jun 2018)

Traditional 75% Train / 25% Test Split				
Model	True Positive Rate (%)	True Negative Rate (%)	False Positive Rate (%)	False Negative Rate (%)
RNN	90.8	36.7	63.3	9.2
RF	94.4	96.3	3.7	5.6
LSTM	93.1	34.6	65.4	6.9
PCA	95.2	10.5	89.5	4.8
SVR	94.2	16.4	83.6	5.8
LASSO	94.2	14.0	86.0	5.8
MLR	96.5	16.6	83.4	3.5
Ensemble DT Err.	83.1	75.4	24.6	16.9

RF and ensemble DT with error term for UX1 3D forward have great accuracy results for this 75% training /25% test data with high true negatives and positives as well as and low false negatives and positives. However, the accuracy results are worse than RNN and LSTM using our 10-split time series cross validation.

Fig. 35 shows the RNN actual versus the estimated UX1 3-days forward, which is our best model overall model and method. The estimated forecasts do well versus the actual test data.

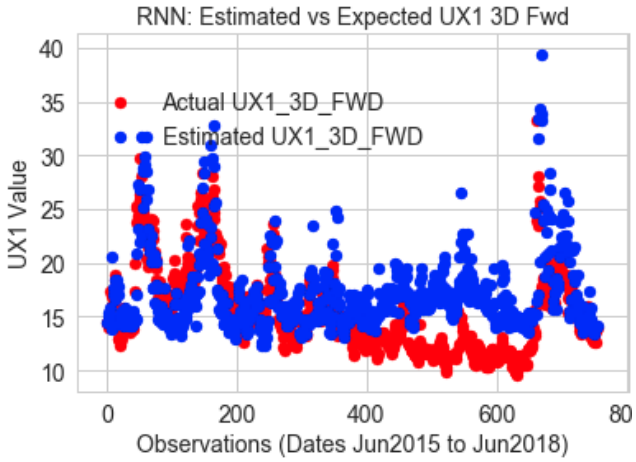


Fig. 35. RNN is Our Best Selected Model and Method. Plot of Actual vs. Estimated for UX1 3-Days Using RNN (Jun 2015 to Jun 2018).

5.2 Analysis of Forecast Results for 1-Mth VIX Futures 5-Days Forward

Similarly, Table 18 and 19 shows the result for the 1-Mth VIX futures forecast 5 days forward across all models. The best first and second results for each column are highlighted in yellow. Across the multiple metrics, the machine/deep learning models RNN, LSTM and RF have better quality assessment metrics compared to the other models. RNN has the best metrics for both the traditional 75% train/25% test split and the cross validation with 10 time series splits. Explained variance for the test data sets are generally low across most models. Again, RF has great quality assessment, but it can be biased to lower volatility forecasts (see section 4.9). Moreover, the quality assessment for ensemble DT with error term for 5-days forward had worse results for both our 75% training /25% test data split and the 10-split time series cross validation (see section 4.4).

Table 18. Quality Assessment Results of Best Models Using Cross Validation with 10 Time Series Splits for 1-mth VIX Futures 5-Days Forward

Method / Model	Input Reduced / Features Selected	Cross Validation with 10 Time Series Splits			
		MSE Test	MSE Train	R^2 / Var Explain Test	R^2 / Var Explain Train
RNN	71	23.37	8.09	0.425	0.890
RF	14	50.34	0.96	0.354	0.986
LSTM	71	74.39	21.12	0.282	0.624
Ensemble DT Err.	16	49.45	0.99	0.330	0.984
PCA	10	30.39	18.10	0.334	0.763
SVR	15	28.99	14.06	0.336	0.802
LASSO	15	43.64	15.37	0.321	0.791
MLR	13	29.35	16.55	0.315	0.786

Table 19. Quality Assessment Results of Best Models Using Traditional 75% Train / 25% Test Time Series Split for 1-mth VIX Futures 5-Days Forward

Method or Model	Input Reduced / Features Selected	Traditional 75% Train / 25% Test							
		MSE Test	MSE Train	MAE Test	MAE Train	R^2 / Var Expl Train	R^2 / Var Expl Test	Corr Train	Corr Test
RNN	71	15.47	4.08	3.01	1.61	0.959	0.029	0.98	0.49
RF	14	0.54	74.54	0.48	5.84	0.330	0.965	0.62	0.98
LSTM	71	21.62	3.76	3.70	1.47	0.955	-0.062	0.97	0.42
Ensemble	16	15.57	0.14	3.12	0.25	0.99	0.26	0.99	0.60
PCA	10	21.93	13.76	3.52	2.78	0.838	0.029	0.92	0.61
SVR	15	18.41	16.85	3.16	2.97	0.803	0.116	0.90	0.63
LASSO	15	18.54	16.09	3.42	2.99	0.810	0.218	0.90	0.62
MLR	13	22.09	17.25	3.33	3.12	0.796	-0.049	0.89	0.63
ARIMA	13	8.63					0.357		

As shown in Table 20, the accuracy matrix shows that RNN, LSTM and RF are better predictors with high true positives and higher true negative rates, but also a lower false positive rate compared to the other models. Most models have low false

negative rate. For ensemble DT with error term, the accuracy results degrade compared to the other models for 5-days forward and compared to the results for 3-days forward.

Table 20. Accuracy Matrix of Test Data using Traditional 75%/35% of Data for 1-mth VIX Futures 5-Days Forward (Jun 2015 to Jun 2018)

Traditional 75% Train / 25% Test Split				
Model	True Positive Rate (%)	True Negative Rate (%)	False Positive Rate (%)	False Negative Rate (%)
RNN	87.6	55.3	44.7	12.4
RF	87.1	89.1	10.9	12.9
LSTM	94.1	37.9	62.1	5.9
PCA	95.1	11.9	88.1	4.9
SVR	95.0	18.6	81.4	5.0
LASSO	93.9	13.6	86.4	6.1
MLR	96.9	16.7	83.3	3.1
Ensemble DT Err.	92.5	29.2	70.8	7.5

6 Ethics

Ethics are moral principles that govern a person's behavior. When it comes with investments in stocks and volatility, it is crucial to uphold customers privacy and data. Investment managers are always concerned about future market volatility. Employees should not provide non-disclosure information to anyone other than their team members. If employees were to disclose classified information, this would lead to a reputational decline of the company, vendor or fund manager. In addition to the reputation, consumers would have doubts. By having principles and ethics, this would maintain the integrity and trust of the data company, investment fund, and/or fund manager.

It is crucial to uphold customer's privacy around their data. For our analysis, two agreements for our data must be observed, one with Bloomberg and one with Option Metrics. First, Bloomberg users can download and analyze data, but cannot propagate it to individuals not associated with SMU, unless they have a Bloomberg license. The Bloomberg rules of data for data proliferation require that a close to close data license must be confirmed with the recipient prior to dissemination of the data. Option Metrics provides option implied volatility data. Similar to Bloomberg, the data cannot be propagated unless they have required license confirmation. Since our data set is combination of both data vendors, both licenses must be confirmed before dissemination of the data.

All the models used in this paper rely heavily on the financial data and their accuracy. From ethics perspective, the consumers and publishers of the data have equal responsibility to ensure accuracy of the information, since its use can have a significant impact on many. From publisher's perspective, correctness of the data is important since it is a starting point for conducting an analysis and determining a

course of action by the fund managers. Similarly, consumers of the data have an equal responsibility to have established and mature practices when creating models or using other methods to predict the volatility. In addition, the decisions and actions made as a result of these models should be used in the best interest of the client. Finally, this model should be used in conjunction with fundamental data and other models and methods for investment manager decisions.

Generally speaking, ethics concerns with this particular topic on data can be applied to other inputs and outputs of the model. All parties involved are expected to be responsible when it comes to handling privacy of the data and protect it from being used for unintended purposes that violates the agreements, privacy, and confidence of the true data owners. Similarly, the conclusions drawn using the methods and models outline in this paper should be used in conjunction with other methods. It is important to emphasis that all parties are responsible to ensure that unintended consequences of the data usage are prevented and eliminated.

7 Conclusions

Using the same training and test data set for the VIX, this paper built and compared three existing or common financial models to six machine learning regression model to determine if there is an improvement in volatility forecasting for the 1-mth VIX futures 3 and 5-day forward. Our analysis showed that RNN and LSTM are the better machine/deep learning models in forecasting 1-mth VIX Futures 3 and 5-days forward with RNN chosen as the better models. RNN has the best overall metrics and accuracy matrix for both the traditional 75% train/25% test split and the cross validation with 10 time series splits. Compared to all existing and machine learning methods, RNN had better overall accuracy and the better MSE, MAE, correlation of actual versus estimated, and explained variance for both our traditional training/test data split of 75%/25% and a 10-split cross-validation of our time series data. Finally, for RNN, LSTM, RF and ensemble DT with error term, our accuracy matrix showed higher true positive and negative rates than other methods but more importantly a lower false positive rate than other methods (false negative was low for most models).

There are some positive results individually for other models. For the existing models, univariate AutoRegressive Integrated Moving Average (ARIMA) model was the closest to RNN and LSTM. Random forest using feature selection also showed strong quality assessment results, but the forecast was generally bias toward lower volatility levels of 1-mth VIX futures 3 and 5-day forward, which occurs a majority of the time. Similarly, the ensemble DT with error term provided strong quality assessment quality for our traditional 75% train/25% test data split but only for UX1 3D forward . For 3D forward, DT with error term showed worse quality assessment results for 10-split time series cross validation, indicating that our traditional split may have overfit the data. In addition, for 5D forward, DT with error term showed worse quality assessment results than other models. Moreover, RF and ensemble DT with error term performed worse in prediction inflection points of higher 1-mth VIX future levels, such as the mortgage crisis of 2008 (the Great Recession) and the European

debt crisis (the PIGS). Additionally, our model attempts to capture these inflection points. In contrast, RNN and LSTM likely work better around inflection or regime shifts in volatility, since they incorporate "memory" to capture information about what is already calculated in the predicted time series.

Generally, ensemble methods such as RNN, LSTM, RF and DT with error term produced the better results, where RNN had the best overall result for our data set. Ensemble methods combined with feature selection techniques produce comparable result while reducing the complexity of the models. Finally, RNN and LSTM combined with our K-split time series cross-validation method allow variables to be added without dimensionality reduction or feature selection unlike MLR, PCA and ARIMA and other methods.

References

1. Taleb, Nassim Nicholas; *The Black Swan: The Impact of the Highly Improbable*, Random House New York (2007).
2. Yu, Michael and Seco, Luis (advisor). Predicting the Volatility Index Returns Using Machine Learning. ProQuest Dissertations and Theses Publishing, 2017.
3. Rosillo, Rafael; Giner, Javier; and Fuente, David. The effectiveness of the combined use of VIX and Support Vector Machines on the prediction of S&P 500, Neural Computing and Applications, August 2014, Vol.25(2), pp.321-332, Springer Publishing.
4. Hansson, Magnus and Prof. Nilsson, Birger: On sock return prediction with LSTM networks. Lund University. Department of Economics. Thesis. June 1, 2017.
5. Ahoniemi, Katja: Modeling and Forecasting the VIX Index. Imperial College Business School. Thesis, July 17, 2008.
6. Tak-chung Fu, Fu-lai Chung, Vincent Ng and Robert Luk, Pattern Discovery from Stock Time Series Using Self-Organizing Maps, Dept of Computing, Hong Kong Polutechnic University. Dec 2016.

Appendix 1: Description of Futures, Options, Calls, Puts, and the VIX as well as the Calculation of Implied Volatility of the VIX

Description of Futures Contract: A futures contract is a legal agreement to buy or sell a particular commodity or asset at a predetermined price at a specified time in the future. Futures contracts are standardized for quality and quantity to facilitate trading on a futures exchange. The buyer of a futures contract is taking on the obligation to buy the underlying asset when the futures contract expires. The seller of the futures contract is taking on the obligation to provide the underlying asset at the expiration date.[7]

What is an option? Options represent the right (but not the obligation) to take some sort of action by a predetermined date. That right is the buying or selling of shares of the underlying stock or index. There are two types of options, calls and puts. And there are two sides to every option transaction -- the party buying the option, and the party selling (also called writing) the option. Each side comes with its own risk/reward profile and may be entered into for different strategic reasons. The buyer of the option is said to have a long position, while the seller of the option (the writer) is said to have a short position.[8]

Description of Calls: A call is the option to buy the underlying stock at a predetermined price (the strike price) by a predetermined date (the expiry). The buyer of a call has the right to buy shares at the strike price until expiry. The seller of the call (also known as the call "writer") is the one with the obligation. If the call buyer decides to buy -- an act known as exercising the option -- the call writer is obliged to sell his/her shares to the call buyer at the strike price.[8]

So, say an investor bought a call option on Intel with a strike price at $20, expiring in two months. That call buyer has the right to exercise that option, paying $20 per share, and receiving the shares. The writer of the call would have the obligation to deliver those shares and be happy receiving $20 for them. We'll discuss the merits and motivations of each side of the trade momentarily.[8]

Description of Puts: If a call is the right to buy, then perhaps unsurprisingly, a put is the option to sell the underlying stock at a predetermined strike price until a fixed expiry date. The put buyer has the right to sell shares at the strike price, and if he/she decides to sell, the put writer is obliged to buy at that price.[8]

Investors who bought shares of Hewlett-Packard at the ouster of former CEO Carly Fiorina are sitting on some sweet gains over the past two years. And while they may believe that the company will continue to do well, perhaps, in the face of a potential economic slowdown, they're concerned about the company sliding with the rest of the market, and so buy a put option at the $40 strike to "protect" their gains. Buyers of the put have the right, until expiry, to sell their shares for $40. Sellers of the put have the obligation to purchase the shares for $40 (which could hurt, in the event that HP were to decline further).[8]

---

[7] https://www.investopedia.com/terms/f/futurescontract.asp

[8] https://www.fool.com/investing/options/options-the-basics.aspx

Description of the VIX and Calculation of Implied Volatility: The CBOE VIX is essentially one-month at-the-money (ATM) implied volatility on the S&P 500 (SPX) as of today. It uses an interpolation of SPX options that expire over the next 1 to 2 months to determine the current at-the-money (ATM) implied volatility. For example, if there are 20 calendar days left to the nearest option expiry, it uses 20 days of the current expiry and 11 days of the next expiry. In addition, the VIX methodology rolls a few days prior to the front month expiry to the next expiry for its interpolation. In March 2004, the CBOE listed the futures and options on the VIX, which became more liquidly traded by 2006. Therefore, the data set starts in July of 2006 and end in June 2018.

Before we begin, the Black-Scholes model is below for any option price based on Ito's Lemma:

$$C = S * N(d_1) - N(d_2) * K * e^{-r*t} \quad \textbf{(A1.1)}$$

$$P = K * e^{-r*t} * N(-d_2) - S * N(-d_1) \quad \textbf{(A1.2)}$$

$$d_1 = [\ \ln(S/K) + (r + \sigma^2/2)*t\ ] / (\sigma * \text{sqrt}(t)) \quad \textbf{(A1.3)}$$

$$d_2 = d_1 - (\sigma * \text{sqrt}(t)) \quad \textbf{(A1.4)}$$

where

C = call premium
P = put premium
S = current stock price or index
t = time to maturity left for the option
r = risk-free interest rate
K = option strike price
N = cumulative standard normal distribution
e = exponential term
σ = standard deviation
ln = NaturalLog

Note that all of the above inputs are known except one, σ. Implied volatility, σ, is calculated and represents the uncertainty associate with an asset. This is why the standard deviation becomes the implied volatility of the option and explains the variance to that maturity and strike of the stock or index plus any added uncertainty. ***The implied volatility, σ, provides unique insight into explaining uncertainty in an asset based on how the market is pricing it.***

Below are all the knowns in the Black-Scholes formula.

- C, P and S are determined by the market
- K is the strike chosen by the investor
- t is time to maturity of the option (which is known from today)
- r is the risk-free rate of the bank or credit entity from today to that expiry or maturity. Bootstrapping the yield curve (plus the credit funding of the entity) is used to determine r.
- N(), exp and ln are known mathematical terms

Here, we provide a basic explanation of some of the independent (explanatory) and dependent (response) variables used in our analysis:

The CBOE VIX is basically one-month at-the-money (ATM) implied volatility on the S&P 500 as of today. ATM mean current spot level of the VIX. It uses an interpolation of options that expire over the next 1 to 2 months to determine the current at-the-money (ATM) and out-of-the-money (OTM) implied volatilities. For

example, if there are 20 days left to the nearest option expiry, it uses 20 days of the current expiry and 11 days of the next expiry. In addition, the VIX methodology rolls a few days prior to the front month expiry to the next expiry for its interpolation because in the last few days of expiration of the front month option, both prices and volatility can become unstable/manipulated for many reasons.[9]

Volatility of Volatility: The concept of volatility of volatility[10] is very complex and beyond the scope of this research; however, it is part of the reason that we can forecast the VIX; therefore, we provide a link to some research on this topic.

[9] CBOE VIX white paper: https://www.cboe.com/micro/vix/vixwhite.pdf

[10] Concept of volatility of volatility using VIX: http://www.cboe.com/rmc/2014/Day2Session1BDeb-revised.pdf

Appendix 2: Background and Prior Research

Table A2 below provides a summary of prior research.

Table A2. Background & Prior Research

Reference	Research Description	Abstract Summary
[2]	Forecast VIX using Machine Learning	This paper probes how predictable the short-term future behavior of the VIX is given past market price data within the constraints of a simple classic machine learning framework.
[3]	Neural Network to Predict S&P 500 using VIX	The effectiveness of VIX is shown when used with Support Vector Machines (SVMs) to forecast the weekly change in the S&P 500 index. A trading simulation is implemented so that statistical efficiency is complemented by measures of economic performance. The SVM identifies the best situations in which to buy or sell in the market.
[4]	Predict S&P 500 Stock Returns using RNN	In this thesis, LSTM (long short-term memory) recurrent neural networks are used to perform financial time series forecasting on return data of three stock indices. The results show that the outputs of the LSTM networks are very similar to those of a conventional time series model, namely an ARMA(1,1) - GJRGARCH(1,1), when a regression approach is taken. However, they outperform the time series model with regards to directional change.
[5]	Predicting VIX using ARIMA	This paper models the implied volatility of the S&P 500 index, with the aim of producing useful forecasts for option traders. The results indicate that an ARIMA (1,1,1) model enhanced with exogenous regressors has predictive power regarding the directional change in the VIX index. Out-of-sample option trading over a period of fifteen months yields positive returns when the forecasts from the best models are used as the basis for investment decisions
[6]	Pattern Discovery from Stocks using SOMs	A clustering approach is proposed for pattern discovery from time series. In view of its popularity and superior clustering performance, the self-organizing map (SOM) was adopted for pattern discovery in temporal data sequences and applied to financial time series data.

Appendix 3: Description of 71 Input and Two Output Variables.

Table A3 below provides a summary of all 71 input variables and two output variables. IV stands for implied volatility in table. For the IN/OUT column, IN is input or explanatory variable and OUT is the output or response variable

Table A3. Description of 71 Input (Independent) and 2 Output (Response or Dependent) Variables

Input #	IN or OUT	Input (Explanatory) Variable Name	Description
1	IN	BOLL_XUPPER	=1 when VIX crosses upper Bollinger band
2	IN	BOLL_XLOWER	=1 when VIX crosses lower Bollinger band
3	IN	SIGBUY14D	=1 when VIX crosses upper 14-day MA
4	IN	SIGSELL14D	=1 when VIX crosses lower 14-day MA
5	IN	SIGBUY14D3CD	=1 when VIX crosses upper 14-day MA 3 consecutive days
6	IN	SIGBUY50D	=1 when VIX crosses upper 50-day MA
7	IN	SIGSELL50D	=1 when VIX crosses lower 50-day MA
8	IN	SIGBUY50D3CD	=1 when VIX crosses upper 50-day MA 3 consecutive days
9	IN	SIGBUY100D	=1 when VIX crosses upper 100-day MA
10	IN	SIGSELL100D	=1 when VIX crosses lower 100-day MA
11	IN	SIGBUY100D3CD	=1 when VIX crosses upper 100-day MA 3 consecutive days
12	IN	UX2_HILO	Intraday High – Low Spread of 2-mth VIX future
13	IN	UX3_HILO	Intraday High – Low Spread of 3-mth VIX future
14	IN	UX4_HILO	Intraday High – Low Spread of 4-mth VIX future
15	IN	UX5_HILO	Intraday High – Low Spread of 5-mth VIX future
16	IN	UX6_HILO	Intraday High – Low Spread of 6-mth VIX future
17	IN	UX7_HILO	Intraday High – Low Spread of 7-mth VIX future
18	IN	UX8_HILO	Intraday High – Low Spread of 8-mth VIX future
19	IN	UX3MUX2	Term Structure Sprd of 3-mth - 2-mth VIX future
20	IN	UX4MUX2	Term Structure Sprd of 4-mth - 2-mth VIX future
21	IN	UX5MUX2	Term Structure Sprd of 5-mth - 2-mth VIX future
22	IN	UX6MUX2	Term Structure Sprd of 6-mth - 2-mth VIX future
23	IN	UX7MUX2	Term Structure Sprd of 7-mth - 2-mth VIX future
24	IN	UX8MUX2	Term Structure Sprd of 8-mth - 2-mth VIX future
25	IN	UX4MUX3	Term Structure Sprd of 4-mth - 3-mth VIX future
26	IN	UX5MUX3	Term Structure Sprd of 5-mth - 3-mth VIX future
27	IN	UX6MUX3	Term Structure Sprd of 6-mth - 3-mth VIX future
28	IN	UX7MUX3	Term Structure Sprd of 7-mth - 3-mth VIX future
29	IN	UX8MUX3	Term Structure Sprd of 8-mth - 3-mth VIX future
30	IN	UX5MUX4	Term Structure Sprd of 5-mth - 4-mth VIX future
31	IN	UX6MUX4	Term Structure Sprd of 6-mth - 4-mth VIX future
32	IN	UX7MUX4	Term Structure Sprd of 7-mth - 4-mth VIX future
33	IN	UX8MUX4	Term Structure Sprd of 8-mth - 4-mth VIX future

Input #	IN or OUT	Input (Explanatory) Variable Name	Description
34	IN	UX6MUX5	Term Structure Sprd of 6-mth - 5-mth VIX future
35	IN	UX7MUX5	Term Structure Sprd of 7-mth - 5-mth VIX future
36	IN	UX8MUX5	Term Structure Sprd of 8-mth - 5-mth VIX future
37	IN	UX7MUX6	Term Structure Sprd of 7-mth - 6-mth VIX future
38	IN	UX8MUX6	Term Structure Sprd of 8-mth - 6-mth VIX future
39	IN	UX8MUX7	Term Structure Sprd of 8-mth - 7-mth VIX future
40	IN	VVIX	1-mth ATM Implied VIX Volatility
41	IN	VVIX_HILO	Intraday High – Low Spread of VVIX
42	IN	M1_120_80	Skew IV Sprd. 1-mth 120% OTM – 80%OTM
43	IN	M1_100_80	Skew IV Sprd. 1-mth 100% OTM – 80% OTM
44	IN	M1_120_100	Skew IV Sprd. 1-mth 120% OTM – 100% ATM
45	IN	M1_150_100	Skew IV Sprd. 1-mth 150% OTM – 100% ATM
46	IN	M1_200_100	Skew IV Sprd. 1-mth 200% OTM – 100% ATM
47	IN	M2_120_80	Skew IV Sprd. 2-mth 120% OTM – 80% OTM
48	IN	M2_100_80	Skew IV Sprd. 2-mth 100% OTM – 80% OTM
49	IN	M2_120_100	Skew IV Sprd. 2-mth 120% OTM – 100% ATM
50	IN	M2_150_100	Skew IV Sprd. 2-mth 150% OTM – 100% ATM
51	IN	M2_200_100	Skew IV Sprd. 2-mth 200% OTM – 100% ATM
52	IN	M3_120_80	Skew IV Sprd. 3-mth 120% OTM – 80% OTM
53	IN	M3_100_80	Skew IV Sprd. 3-mth 100% OTM – 80% OTM
54	IN	M3_120_100	Skew IV Sprd. 3-mth 120% OTM – 100% ATM
55	IN	M3_150_100	Skew IV Sprd. 3-mth 150% OTM – 100% ATM
56	IN	M3_200_100	Skew IV Sprd. 3-mth 200% OTM – 100% ATM
57	IN	M6_120_80	Skew IV Sprd. 6-mth 120% OTM – 80% OTM
58	IN	M6_100_80	Skew IV Sprd. 6-mth 100% OTM – 80% OTM
59	IN	M6_120_100	Skew IV Sprd. 6-mth 120% OTM – 100% ATM
60	IN	M6_150_100	Skew IV Sprd. 6-mth 150% OTM – 100% ATM
61	IN	M6_200_100	Skew IV Sprd. 6-mth 200% OTM – 100% ATM
62	IN	M9_120_80	Skew IV Sprd. 9-mth 120% OTM – 80% OTM
63	IN	M9_100_80	Skew IV Sprd. 9-mth 100% OTM – 80% OTM
64	IN	M9_120_100	Skew IV Sprd. 9-mth 120% OTM – 100% ATM
65	IN	M9_150_100	Skew IV Sprd. 9-mth 150% OTM – 100% ATM
66	IN	M9_200_100	Skew IV Sprd. 9-mth 200% OTM – 100% ATM
67	IN	M12_120_80	Skew IV Sprd. 12-mth 120% OTM – 80% OTM
68	IN	M12_100_80	Skew IV Sprd. 12-mth 100% OTM – 80% OTM
69	IN	M12_120_100	Skew IV Sprd. 12-mth 120% OTM – 100% ATM
70	IN	M12_150_100	Skew IV Sprd. 12-mth 150% OTM – 100% ATM
71	IN	M12_200_100	Skew IV Sprd. 12-mth 200% OTM – 100% ATM
1	OUT	UX1_3D_FWD	1-mth VIX Future Level 3D Forward
2	OUT	UX1_5D_FWD	1-mth VIX Future Level 5D Forward

Appendix 4: Breakout of Code Archive

The code for this analysis was performed in Python and the archive is submitted with this paper, detailed in Fig. A4. The 'VIXProject' code archive has 3 common financial models and 4 supervised regression methods. The coding archive used with this paper is called 'VixProjectCode.zip'. It will create a 'VIXProject' directory with two iPython notebooks called 'Capstone_VIXProject.ipynb' that inputs the data file 'VIX_DataSkewFinal_New.csv' and 'CreateImpliedVolSurface.ipynb' that inputs the data file 'VolSurfaceVIX_2006to2010.xlsx'. The data files are located in the subdirectory called Data.

Contact the authors of this paper for access to the Python code and data.

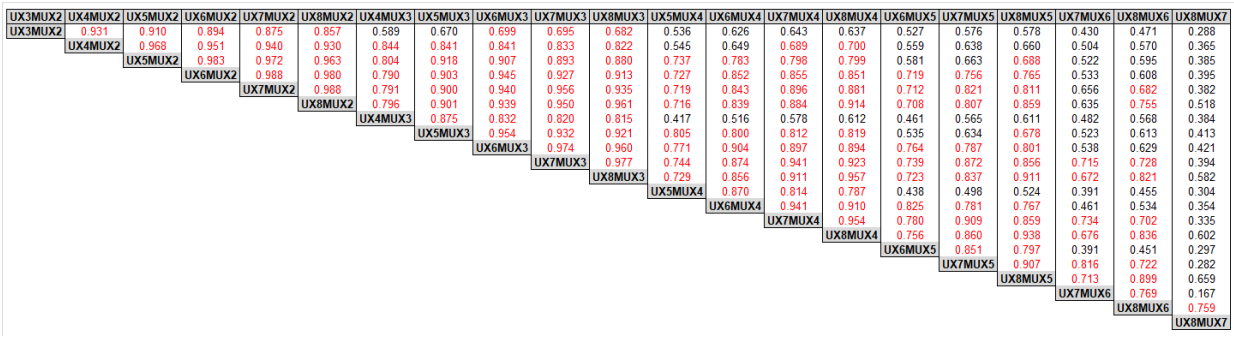
Table A4. Description of Python Code Archive and Data Files

Filename for Code	Coding Environment	Models
Capstone_VIXProject.ipynb	Python Notebook	Performs all analysis for 3 common financial model and 4 supervised machine learning models.
CreateImpliedVolSurface.ipynb	Python Notebook	Create Volatility Surface from input data from Option Metrics
VIX_DataSkewFinal_New.csv	csv or xlsx file	Input file of 71 independent and 2 possible responses variables
VolSurfaceVIX_2006to2010.xlsx	csv or xlsx file	Daily Normalized Volatility Surfaces for

Appendix 5: Cross Correlation of Term Structure and Skew Inputs

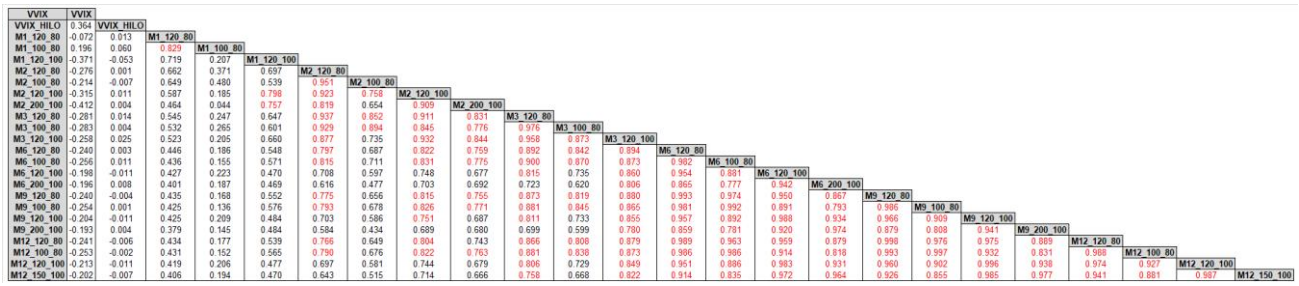
With 71 input variables, there is multi-collinearity that inflates the variance explained by an R^2 from a simple linear regression or that inflates the assessed quality of the results.

Table A5.1 Cross-Correlation of All 28 Term Structure Spread Input Variables
(Jul 2006 to Jun 2015)
Note: Red highlighted number indicates correlation is over 66%.



	UX3MUX2	UX4MUX2	UX5MUX2	UX6MUX2	UX7MUX2	UX8MUX2	UX4MUX3	UX5MUX3	UX6MUX3	UX7MUX3	UX8MUX3	UX5MUX4	UX6MUX4	UX7MUX4	UX8MUX4	UX6MUX5	UX7MUX5	UX8MUX5	UX7MUX6	UX8MUX6	UX8MUX7
UX3MUX2	0.931	0.910	0.894	0.875	0.857	0.589	0.670	0.699	0.695	0.682	0.536	0.626	0.643	0.637	0.527	0.576	0.578	0.430	0.471	0.288
UX4MUX2		0.968	0.951	0.940	0.930	0.844	0.841	0.841	0.833	0.822	0.545	0.649	0.689	0.700	0.559	0.638	0.660	0.504	0.570	0.365
UX5MUX2			0.983	0.972	0.963	0.804	0.918	0.907	0.893	0.880	0.737	0.783	0.798	0.799	0.581	0.663	0.688	0.522	0.595	0.385
UX6MUX2				0.988	0.980	0.790	0.903	0.945	0.927	0.913	0.727	0.852	0.855	0.851	0.719	0.756	0.765	0.533	0.608	0.395
UX7MUX2					0.988	0.791	0.900	0.940	0.956	0.935	0.719	0.843	0.896	0.881	0.712	0.821	0.811	0.656	0.682	0.382
UX8MUX2						0.796	0.901	0.939	0.950	0.961	0.716	0.839	0.884	0.914	0.708	0.807	0.859	0.635	0.755	0.518
UX4MUX3							0.875	0.832	0.820	0.815	0.417	0.516	0.578	0.612	0.461	0.565	0.611	0.482	0.568	0.384
UX5MUX3								0.954	0.932	0.921	0.805	0.800	0.812	0.819	0.535	0.634	0.678	0.523	0.613	0.413
UX6MUX3									0.974	0.960	0.771	0.904	0.897	0.894	0.764	0.787	0.801	0.538	0.629	0.421
UX7MUX3										0.977	0.744	0.874	0.941	0.923	0.739	0.872	0.856	0.715	0.728	0.394
UX8MUX3											0.729	0.856	0.911	0.957	0.723	0.837	0.911	0.672	0.821	0.582
UX5MUX4												0.870	0.814	0.787	0.438	0.498	0.524	0.391	0.455	0.304
UX6MUX4													0.941	0.910	0.825	0.781	0.767	0.461	0.534	0.354
UX7MUX4														0.954	0.780	0.909	0.859	0.734	0.702	0.335
UX8MUX4															0.756	0.860	0.938	0.676	0.836	0.602
UX6MUX5																0.851	0.797	0.391	0.451	0.297
UX7MUX5																	0.907	0.816	0.722	0.282
UX8MUX5																		0.713	0.899	0.659
UX7MUX6																			0.769	0.167
UX8MUX6																				0.759
UX8MUX7																				

Table A5.2 Cross-Correlation of All 30 Skew Plus Two VVIX Input Variables
(Jul 2006 to Jun 2015)
Note: Red highlighted number indicates correlation is over 66%.



	VVIX	VVIX																															
VVIX_HILO	0.364	VVIX_HILO																															
M1_120_80	-0.072	0.013	M1_120_80																														
M1_100_80	0.196	0.060	0.829	M1_100_80																													
M1_120_100	-0.371	-0.053	0.719	0.207	M1_120_100																												
M2_120_80	-0.276	0.001	0.662	0.371	0.697	M2_120_80																											
M2_100_80	-0.214	-0.007	0.649	0.480	0.539	0.951	M2_100_80																										
M2_120_100	-0.315	0.011	0.587	0.185	0.798	0.923	0.758	M2_120_100																									
M2_200_100	-0.412	0.004	0.464	0.044	0.757	0.819	0.654	0.909	M2_200_100																								
M3_120_80	-0.281	0.014	0.545	0.247	0.647	0.937	0.852	0.911	0.831	M3_120_80																							
M3_100_80	-0.283	0.004	0.532	0.265	0.601	0.929	0.894	0.846	0.776	0.976	M3_100_80																						
M3_120_100	-0.258	0.025	0.523	0.205	0.660	0.877	0.735	0.932	0.844	0.958	0.873	M3_120_100																					
M6_120_80	-0.240	0.003	0.446	0.186	0.548	0.797	0.687	0.822	0.759	0.892	0.842	0.894	M6_120_80																				
M6_100_80	-0.256	0.011	0.436	0.155	0.571	0.815	0.711	0.831	0.775	0.900	0.870	0.873	0.992	M6_100_80																			
M6_120_100	-0.198	-0.011	0.427	0.223	0.470	0.708	0.597	0.748	0.677	0.815	0.735	0.860	0.954	0.881	M6_120_100																		
M6_200_100	-0.196	0.008	0.401	0.187	0.469	0.616	0.477	0.703	0.692	0.723	0.620	0.806	0.865	0.777	0.942	M6_200_100																	
M9_120_80	-0.240	-0.004	0.435	0.168	0.552	0.775	0.656	0.815	0.755	0.873	0.819	0.880	0.993	0.974	0.950	0.867	M9_120_80																
M9_100_80	-0.254	0.001	0.425	0.136	0.576	0.793	0.678	0.826	0.771	0.881	0.845	0.866	0.981	0.992	0.891	0.793	0.986	M9_100_80															
M9_120_100	-0.204	-0.011	0.425	0.209	0.484	0.703	0.586	0.751	0.687	0.811	0.733	0.855	0.957	0.892	0.988	0.934	0.966	0.909	M9_120_100														
M9_200_100	-0.193	0.004	0.379	0.145	0.484	0.584	0.434	0.689	0.680	0.699	0.599	0.780	0.859	0.781	0.920	0.974	0.879	0.808	0.941	M9_200_100													
M12_120_80	-0.241	-0.006	0.434	0.177	0.539	0.766	0.649	0.804	0.743	0.866	0.808	0.879	0.989	0.963	0.959	0.879	0.998	0.976	0.975	0.889	M12_120_80												
M12_100_80	-0.253	-0.002	0.431	0.152	0.565	0.790	0.676	0.822	0.763	0.881	0.838	0.873	0.986	0.986	0.914	0.818	0.993	0.997	0.932	0.831	0.988	M12_100_80											
M12_120_100	-0.213	-0.011	0.419	0.206	0.477	0.697	0.581	0.744	0.679	0.806	0.729	0.849	0.951	0.896	0.983	0.931	0.960	0.902	0.996	0.938	0.974	0.927	M12_120_100										
M12_150_100	-0.202	-0.007	0.406	0.194	0.470	0.643	0.515	0.714	0.666	0.758	0.668	0.822	0.914	0.835	0.972	0.964	0.926	0.855	0.985	0.977	0.941	0.881	0.987	M12_150_100									

Table A5.1 shows the correlation between the 28 term structure input variables and Table A5.2 shows the correlation between the 30 skew input variables plus the two VVIX variables. The red highlight numbers indicate a correlation above 66% using our full training data set. Finally, UX1 in this paper will represent out response variable for 1-mth VIX Futures.

Goals of Models: Therefore, the analysis in this paper has two goals:

1. Reduce dimensionality or perform feature selections
2. Determine or assess the quality of the output using similar evaluation metrics for most of the models

Appendix 6: Calculation of Accuracy Matrix

Below are the calculations used in our accuracy matrix:

True Positive Rate (TPR) True Negative Rate (TNR)

$$TPR = TP/(TP+FN) \qquad TNR = TN/(TN+FP)$$

False Positive Rate (FPR) False Negative Rate (FNR)

$$FPR = FP/(FP+TN) \qquad FNR = FN/(FN+TP)$$

where:

- TP = # True Positives & TN = # True Negatives
- FP = # False Positives & FN = # False Negatives

Appendix 7: More Details on Machine Learning Methods

Ensemble Method Output using Regression with Prior Error Term. The Ensemble method is useful to determine the most important variables when there are large number of inputs in a machine learning model. Our analysis has over 71 inputs (explanatory) variables and 2 different output (response) variables. The Ensemble method uses bootstrap aggregating, also called bagging. Ensemble combine predictions from different models to generate a final prediction, and the more models we include the better it performs. Bootstrapping refers to any test or metric that relies on random sampling with replacement. For our time series sampling, our regression analysis uses voting (not averaging) since the different training data sets have similar quality assessments. Neural networks and decision trees models are suitable for the ensemble method because they are affected by bootstrapping since these are generally more less stable models. In addition, the ensemble output is fed into s linear regression model with the output of the prior error term.

Least Absolute Shrinkage & Selection Operator (LASSO). LASSO is a regression analysis method that performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the statistical model it produces. LASSO reduced the dimensionality using a penalty factor. LASSO reduces the number of predictors, identifies important predictors, selects among redundant predictors and produces shrinkage estimates with lower predictive errors than ordinary least squares. Alpha is the elasticity factor that controls the balance between lasso and ridge penalties. Our analysis uses a lower alpha of 0.35 to reduce more of the dimensionality of the 71 input factors in our data set. The selected input variables of LASSO are then used to select the final inputs of the linear regression model. All 71 inputs are used in LASSO and LASSO does the reduction.

Support Vector Regression (SVR). Classification and regression analysis can both use a supervised learning approach through support vectors (SVs), which are coordinates of observations. An SVM training algorithm builds a model that assigns sample to one category or the other, making it a non-probabilistic binary linear classifier. An SVM model is a representation of the samples as points in space, mapped so that the samples of the separate categories are divided by a clear gap that is as wide as possible. SVR uses the top 15 predictors from the ensemble method as its inputs.

Recurrent Neural Networks (RNN)-LSTM. RNN is a short-term memory method. In traditional neural networks (NN), all inputs (and outputs) are independent and have no memory but RNNs have a "memory" to capture information about what has been calculated so far in our time series (TS) forecast. RNNs use sequential information by utilizing connections between nodes from a graph, capturing dynamic temporal behavior of the time series. However, RNN results can be disappointing because the simplest RNN model has a major drawback, called vanishing gradient problem due to a lack of a long-term dependency, which prevents it from being accurate over the long-term.

Long Short-Term Memory (LSTM). LSTM are a special kind of RNN with a longer memory method that corrects the vanishing gradient problem due to a lack of a long-term dependency. LSTM can learn long-term dependencies. Remembering information for long periods of time is practically the default behavior of LSTM, not something they struggle to learn in a pure RNN.[11]

Random Forest (RF): Random forests, also known as random decision forests, are a popular ensemble method that can be used to build predictive models for both classification and regression problems. Ensemble methods use multiple learning models to gain better predictive results — in the case of a random forest, the model creates an entire forest of random uncorrelated decision trees to arrive at the best possible answer. Decision trees tend to have high variance when they utilize different training and test sets of the same data, since they tend to overfit on training data. This leads to poor performance on unseen data. Unfortunately, this limits the usage of decision trees in predictive modeling.

[11] http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Appendix 8: MLR Reduced Dimensionality Process

Dimensionality reduction. With all 71 input, the R^2 of a simple ordinary least squares (OLS) regression is 85.7% and with the reduced 13 inputs, R^2 is 80.8%. To reduce the dimensionality of our 71 inputs, the data was first normalized. First, for each regression, variables with p-values > 0.05 or < -0.05 were removed. Second, the largest coefficients by absolute value for each input are kept. Third, the larger additional R^2 value for each input variable are kept because that input explains more of the overall variance. Fourth, the variance inflation factor (VIF) of each variable was calculated and those with VIFs > 7% were removed. Fig. A8 shows the final results of our method.

Inputs after Dimensionality Reduction. M2_120_80, , UX7MUX2, UX3_HILO, M2_150_100, VVIX_HILO, M3_200_100, M12_120_80, BOLL_XUPPER, M2_200_100, UX6_HILO, SIGBUY14D3CD, UX6MUX4, M1_150_100

OLS Regression Results

Dep. Variable:	y	**R-squared:**	0.808
Model:	OLS	**Adj. R-squared:**	0.807
Method:	LeastSquares	**F-statistic:**	890.5
Date:	Thu	**28 Jul 2018**	Prob(F-statistic): 0
Time:	2:18:44PM	**Log-Likelihood:**	-1634.3
No. Observations:	2757	**AIC:**	3295
Df Residuals:	2744	**BIC:**	3372
Df Model:	13		
Covariance Type:	nonrobust		

						Confidence Interval				
Input Factor	**coef**	**Normalized ABS(coeff)**	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**	**VIF (%)**	**Individual R^2**	**Added R^2**
M2_120_80	-0.368	0.368	0.015	-25.079	0.000	-0.396	-0.339	3.075	0.4629	0.0439
UX7MUX2	-0.515	0.515	0.021	-24.402	0.000	-0.556	-0.474	6.379	0.6238	0.0416
UX3_HILO	0.160	0.160	0.017	9.529	0.000	0.127	0.193	4.025	0.3957	0.0063
M2_150_100	0.158	0.158	0.022	7.279	0.000	0.115	0.200	6.718	0.3918	0.0037
VVIX_HILO	-0.061	0.061	0.009	-6.675	0.000	-0.079	-0.043	1.188	0.0102	0.0031
M3_200_100	-0.093	0.093	0.014	-6.507	0.000	-0.121	-0.065	2.909	0.1935	0.0030
M12_120_80	-0.090	0.090	0.015	-6.092	0.000	-0.119	-0.061	3.122	0.3309	0.0026
BOLL_XUPPER	-0.043	0.043	0.010	-4.403	0.000	-0.062	-0.024	1.345	0.0005	0.0014
M2_200_100	-0.055	0.055	0.013	-4.205	0.000	-0.080	-0.029	2.436	0.0838	0.0012
UX6_HILO	0.058	0.058	0.014	4.024	0.000	0.030	0.086	2.965	0.1629	0.0011
SIGBUY14D3CD	-0.030	0.030	0.009	-3.226	0.001	-0.049	-0.012	1.253	0.0094	0.0007
UX6MUX4	-0.063	0.063	0.021	-2.944	0.003	-0.104	-0.021	6.460	0.5706	0.0006
M1_150_100	0.034	0.034	0.016	2.154	0.031	0.003	0.065	3.536	0.3242	0.0003

Omnibus:	112.78	**Durbin-Watson:**	0.518
Prob(Omnibus):	0	**Jarque-Bera (JB):**	301.03
Skew:	0.169	**Prob(JB):**	4.29E-66
Kurtosis:	4.583	**Cond. No.**	8.01

Fig. A8. Output of OLS Regression for UX 1 3D and 5D Forward with Columns for Abs(Coeff), VIF and Additional R^2.

Appendix 9: MLR Output

Quality Assessment of Results for MLR. Fig. A9.1 shows the MLR scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for UX1 3 and 5 days forward. The scatterplots show generally a linear relationship for both the test and training estimates for both 3 and 5 days forward.

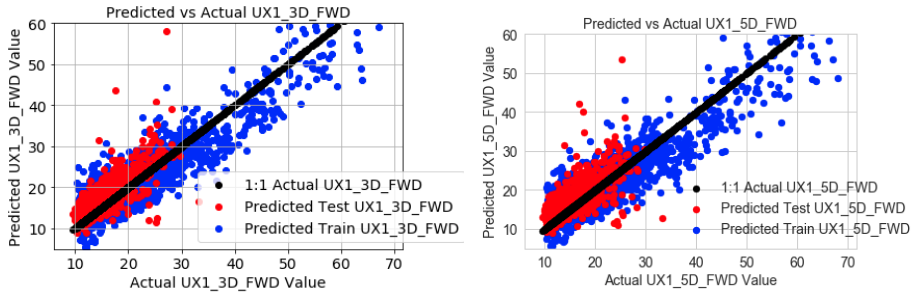


Fig. A9.1. MLR Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig, A9.2 shows the Actual versus the estimated test data only for UX1 both 3 and 5-days forward.

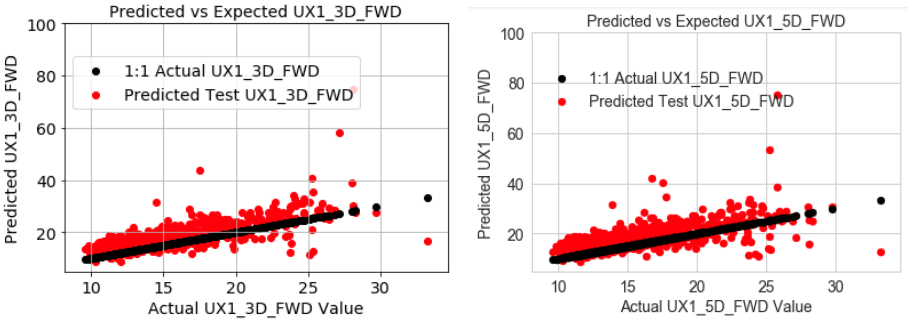


Fig. A9.2. MLR Scatter Plot of Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jun 2015 to Jun 2018)

Fig. A9.3 shows the MLR error histogram of the actual versus estimated for the test and training data sets for UX1 3 and 5-days forward. The test data error histograms are left skewed due to the February 2018 inflation scare that caused volatility to jump for UX1 both 3 and 5-day forward.

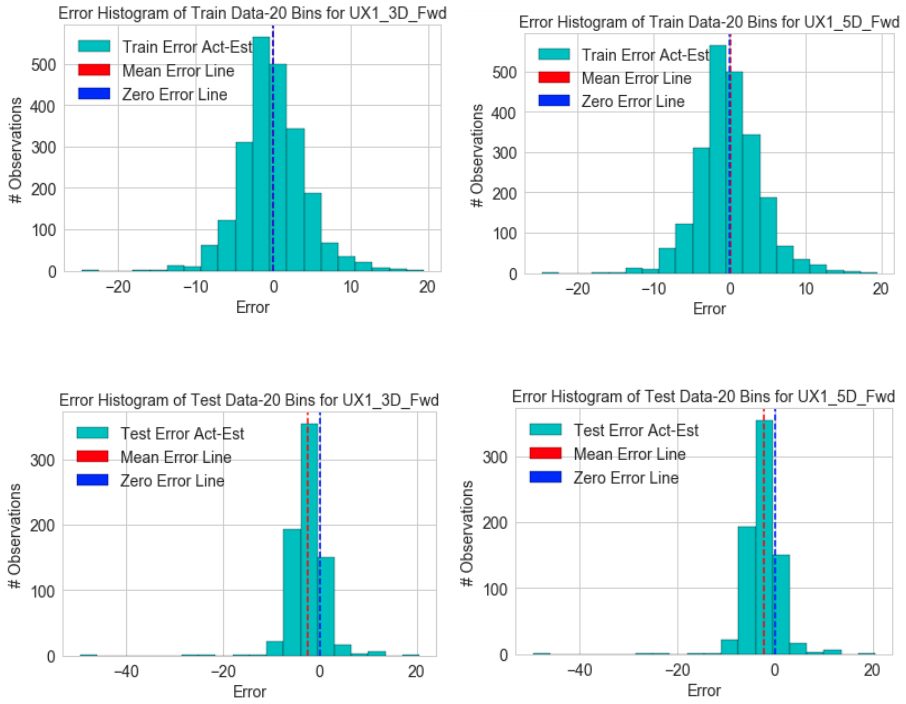


Fig. A9.3. MLR Error Histogram of Estimated Training vs. Actual Training and Test vs. Actual Test for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig. A9.4 shows the residual plot for UX1 3 and 5-days forward.

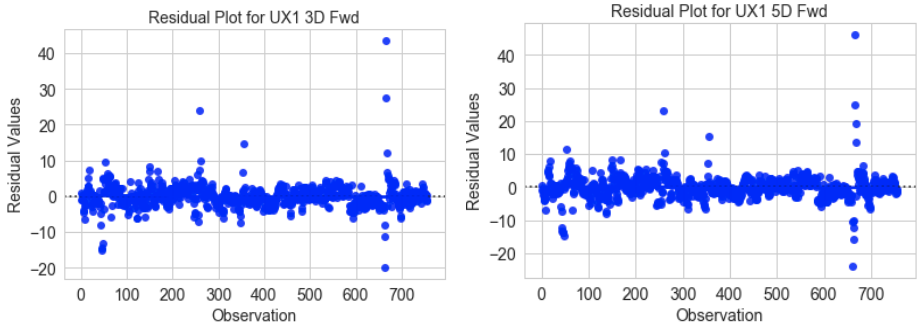


Fig. A9.4. MLR Residual Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A9.5 shows the QQ plots of for UX1 3 and 5-days forward.

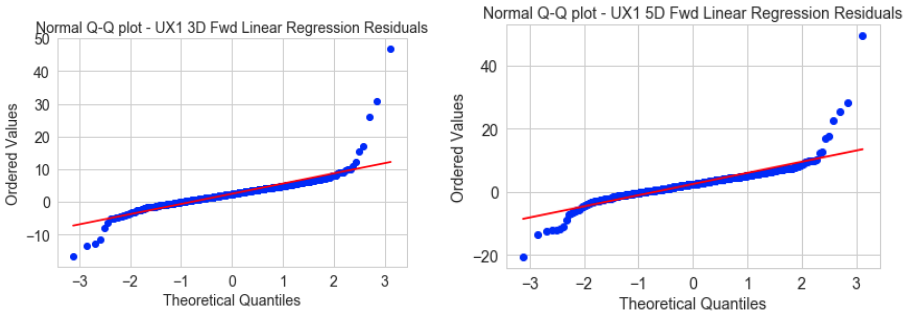


Fig. A9.5. MLR QQ Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A9.6 shows the test actual versus estimated line for UX1 for 3 and 5-days forward.

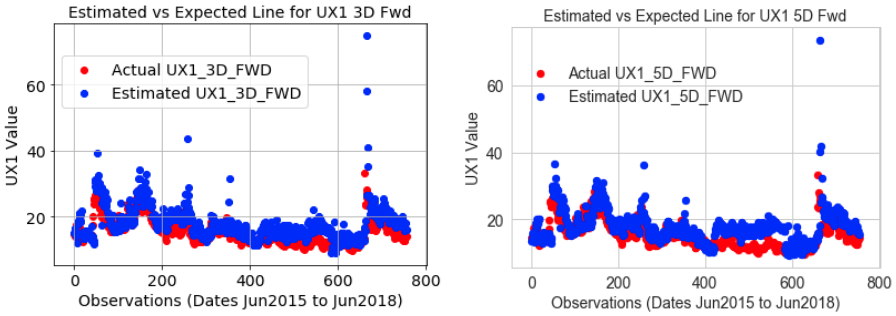


Fig. A9.6. MLR Test Actual versus Estimated Line for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Table A9.1 shows a summary of results for both our 10-split cross validation and the 75%/25% train/test split. Using 10-split cross validation, the MSE of the test data is higher and the R^2 of the test is higher than the traditional split.

Table A9.1 Some Quality Assessment Results of MLR Model

Output Forecasted	Inputs Reduced	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	$\rho(train)^*$	$\rho(test)^*$	R^2_{test}	MSE_{test}
UX1 3D Fwd.	13	0.81	0.16	15.22	18.94	0.91	0.73	0.325	26.76
UX1 5D Fwd.	13	0.79	-0.05	17.25	22.09	0.89	0.63	0.315	29.34

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

Table A9.2 contains the output of our accuracy matrix for true positives and negative as well as false positive and negatives for both 3 and 5 days forward.

Table A9.2 Accuracy Matrix of MLR (Jun 2015 to Jun 2018)

Response	True Positives	False Positives	True Negative	False Negative	TP Rate	TN Rate	FN Rate	FP Rate
3D Train	534	339	592	145	0.78	0.63	0.21	0.36
3D Test	274	226	45	10	0.96	0.17	0.83	0.04
5D Train	539	338	600	145	0.79	0.64	0.21	0.36
5D Test	277	230	46	9	0.96	0.17	0.03	0.83

Appendix 10: PCA Output

Here, a PCA model is analyzed for the existing or common financial models. The data is first normalized prior to using PCA and the output is unnormalize for our graphs.

Dimensionality Reduction for PCA. Fig. A10.1 shows that the PCA model reduces the dimensionality from 71 inputs to 10 principal components (PCs) that explain over 90% of the variance of the model for both UX1 3 and 5-days forward.

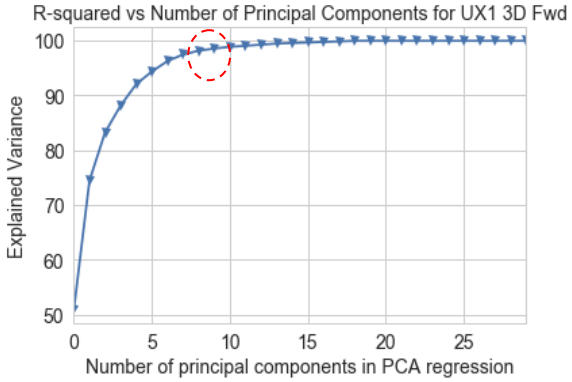


Fig. A10.1. PCA Reduction to 9 Principal Components (PCs) with Explained Variance over 90% for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jul 2006 to Jun 2015)

In Fig A10.2, the number of PCs is chosen at the lowest MSE, which is at 9 PCs.

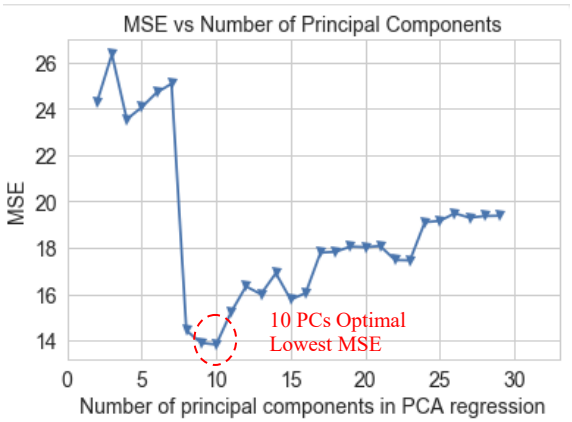


Fig. A10.2. PCA graph shows that with 9 PCs the MSE is minimized for both 3 and 5-days Fwd. (Jul 2006 to Jun 2015)

In Fig A10.3, the number of PCs is chosen at the highest accuracy, which is at 9 PCs.

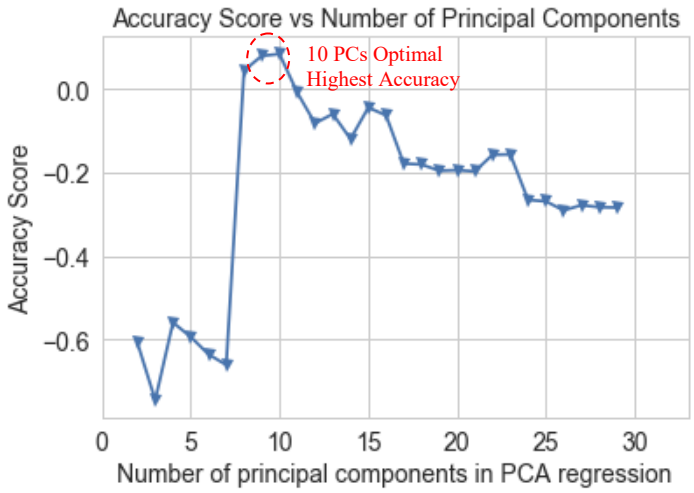


Fig. A10.3. PCA graph shows that with 9 PCs the accuracy is maximized for both 3 and 5-days Forward. (Jul 2006 to Jun 2015)

Quality Assessment of Results for PCA. Fig. A10.4 shows the PCA scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for UX1 3 and 5-days forward. The scatterplots show generally a linear relationship for both the test and training estimates.

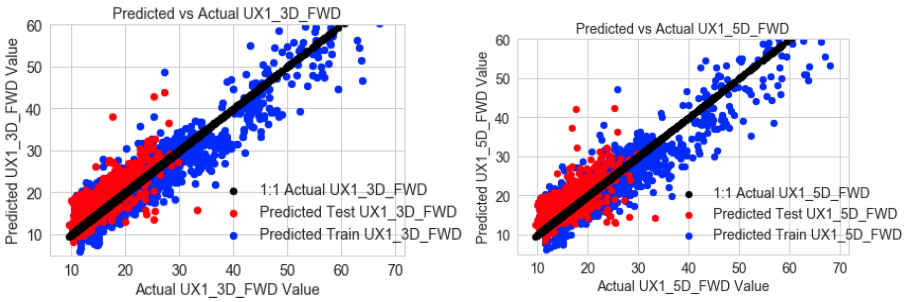


Fig. A10.4. PCA Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3-days Forward. (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig, A10.5 shows the Actual versus the estimated test data only for UX1 both 3 and 5-days forward.

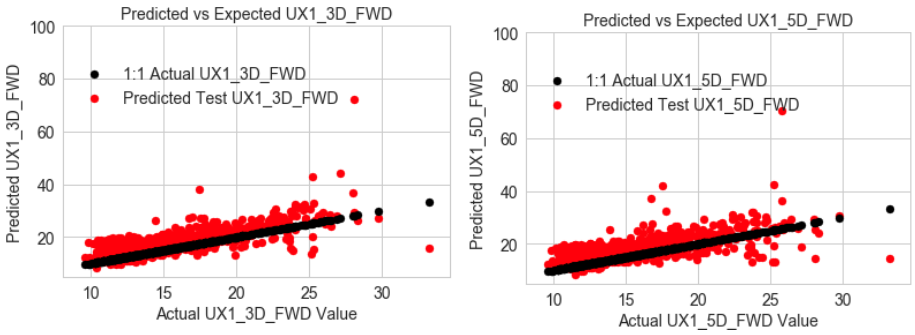


Fig. A10.5. PCA Scatter Plot of Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jun 2015 to Jun 2018)

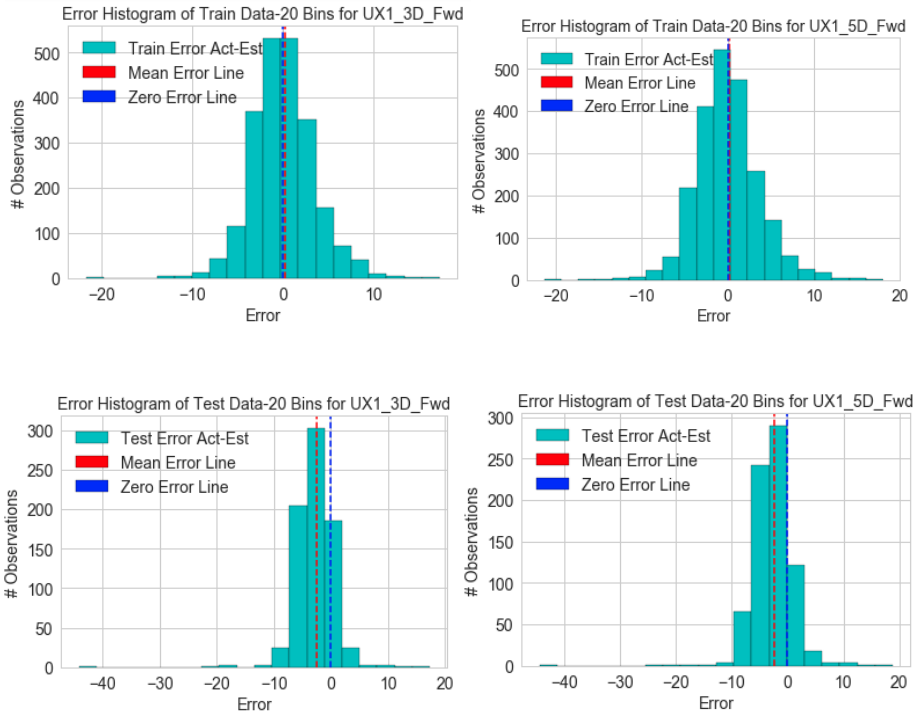


Fig. A10.6. PCA Error Histogram of Estimated Training vs. Actual Training and Test vs. Actual Test for 1-mth VIX Futures (UX1) 3 and 5-day Forward. (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig. A10.6 shows the PCA error histogram of the actual versus estimated for the test and training data sets for UX1 3 and 5-days forward. The test data error histograms are left skewed due to the February 2018 inflation scare that caused volatility to jump for UX1 both 3 and 5-day forward.

Fig. A10.7 shows the residual plot for UX1 3 and 5-days forward.

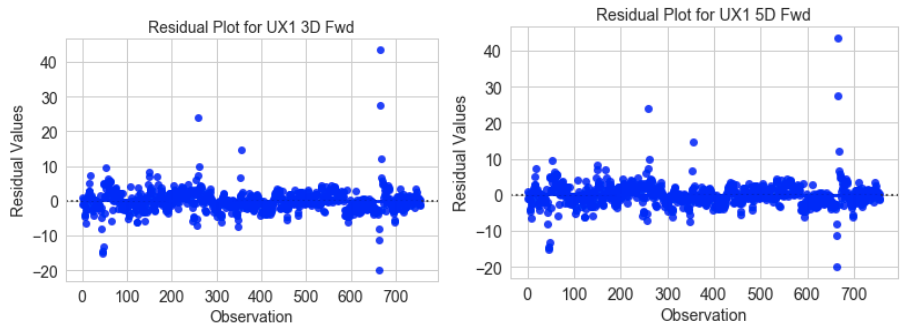


Fig. A10.7. PCA Residual Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A10.8 shows the QQ plots of for UX1 3 and 5-days forward.

Fig. A10.8. PCA QQ Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A10.7 shows the test actual versus estimated line for UX1 for 3 and 5-days forward.

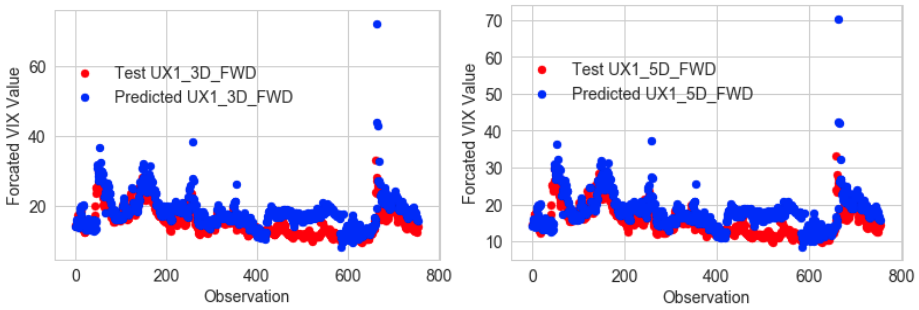


Fig. A10.7. PCA Test Actual versus Estimated Line for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Table A10.1 shows a summary of results for both our 10-split cross validation and the 75%/25% train/test split. Using 10-split cross validation, the MSE of the test data is higher and the R^2 of the test is higher than the traditional split.

Table A10.1 Some Quality Assessment Results of PCA Model

Output Forecasted	Inputs Reduced	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	$\rho(train)^*$	$\rho(test)^*$	R^2_{test}	MSE_{test}
UX1 3D Fwd.	10	0.86	0.22	11.80	19.38	0.93	0.70	0.339	29.10
UX1 5D Fwd.	10	0.84	0.03	13.77	21.93	0.92	0.61	0.334	30.39

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

Table A10.2 contains the output of our accuracy matrix for true positives and negative as well as false positive and negatives for both 3 and 5 days forward.

Table A10.2 Accuracy Matrix of PCA (Jun 2015 to Jun 2018)

Response	True Positives	False Positives	True Negative	False Negative	TP Rate	TN Rate	FN Rate	FP Rate
3D Train	534	337	595	161	0.77	0.64	0.23	0.36
3D Test	279	239	28	14	0.95	0.10	0.05	0.89
5D Train	546	346	588	168	0.76	0.63	0.24	0.37
5D Test	269	236	32	14	0.95	0.12	0.05	0.88

Appendix 11: Univariate ARIMA Output

Inputs: Univariate ARIMA is a different model with only 1 input, the response variable. The response variable is used to forecast the future response.

For this to occur, there has to be autocorrelation in the variable as was shown in section 2.3 earlier in this paper. In section 2.3, the optimal lag for an ARIMA model was 1. Fig. A11.1 shows the actual versus the estimated 1-mth VIX 3-days forward for the ARIMA model.

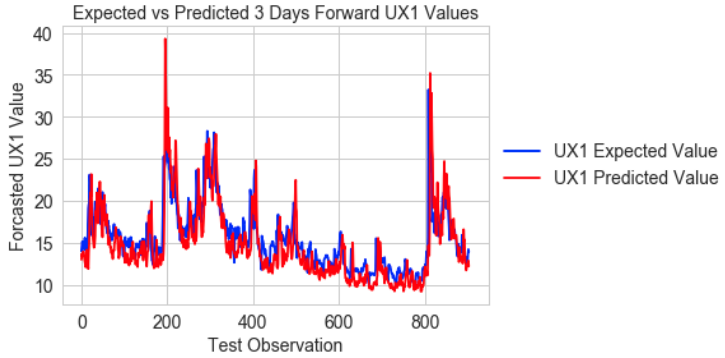


Fig. A11.1 ARIMA Scatter Plot of Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3-days Forward (Jun 2015 to Jun 2018 for Test)

Fig. A11.2 shows the residuals which jump during high volatility moves; otherwise, variance is generally more constant.

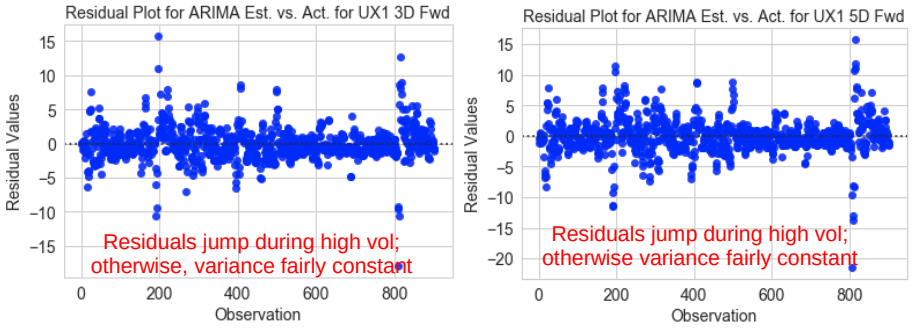


Fig. A11.2 ARIMA Residual Plot of Test Data for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jun 2015 to Jun 2018 for Test)

Fig. A11.3 shows a lag of 1 with the highest autocorrelation of 1 for both UX1 3 and 5-days forward.

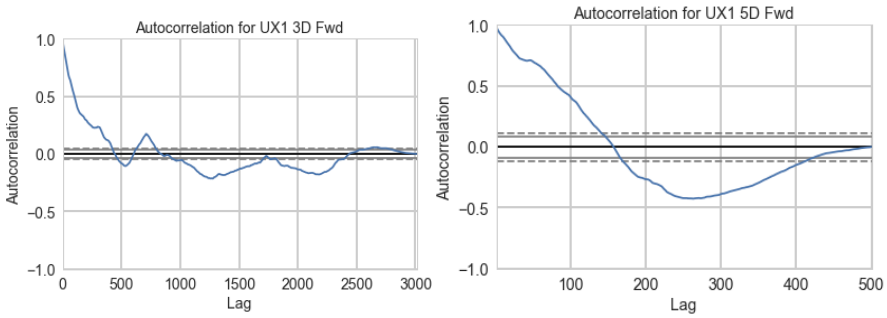


Fig. A11.3 ARIMA Optimal Autocorrelation Lag for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jun 2006 to Jun 2015 for Train)

Table A11.1 is shows that the ARIMA model has a good explained variance and low MSE. However, it can be difficult compared to RNN and LSTM to add more variables to the ARIMA model (multivariate ARIMA). In addition, ARIMA can have trouble forecasting inflection points based solely on the prior response level. An accuracy matrix analysis was not performed on the ARIMA model.

Table A11.1. Some Quality Assessment Results of ARIMA Model

Traditional 75%/25% Train/Test Split			
Output Forecasted	Inputs	R^2_{test}	MSE_{test}
UX1 3D Fwd.	1	0.52	6.44
UX1 5D Fwd.	1	0.36	8.63

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample).
ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

Appendix 12: Ensemble Model Output

The Ensemble method can incorporate an error term from the forecast. In our implementation, the data was first normalized, and then the Ensemble method was used with a linear regression method, incorporating the prior error term into the forecast. In our case the error term cannot be known until 3 or 5 days from the closing price for each day in the dataset.

Feature Selection for Ensemble: Fig. A12.1 shows the top 15 predictors (input variables) plus 1 error term from our ensemble model for UX1 3 and 5 days forward. The top 15 predictors explain a majority of the variance and reduces the MSE to a minimum level.

Bootstrapping refers to any test or metric that relies on random sampling with replacement. It falls in to the broader class of resampling methods. It generates a new dataset for each ensemble member by bootstrapping, i.e. sample N items with replacement from the original N. Bagging uses bootstrap sampling to obtain the data subsets for training the base learners. In addition, bagging uses averaging for regression.

In addition, ensemble usually adds an error term as an input to forecast the response variables after finding the optimal model. First, the error term for our dataset has to be moved forward 3 or 5 days because it is not known until the actual UX1 level 3 or 5-days forward is realized. Second, the error term is also predicted as a third response variable, which is not moved forward, since it is used as our training data response variable. The added error term improves the estimate. The predicted error term is added to the predicted UX1 levels 3 or 5-day forward using out data set with the error term as an input moved forward. In our case, ensemble chose decision trees as the best estimator.

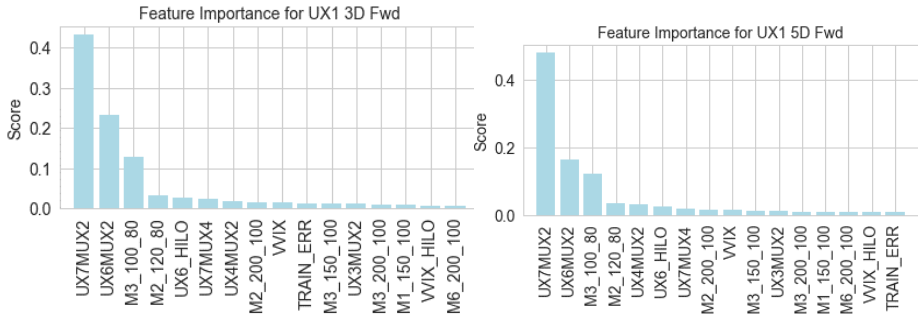


Fig. A12.1 Ensemble Top 15 Predictors plus 1 Error Term that Provide Optimal Results for UX1 3 and 5D Forward (Jul 2006 to Jun 2015)

Quality Assessment of Results for Ensemble Incorporating Error Term: Fig. A12.2 shows the ensemble scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset

as a benchmark for UX1 both 3 and 5 days forward. The scatterplots show an estimate with increasing variance as volatility increases compared to the 1 to 1 plot line for the test estimate while the training estimates shows better results and a tighter variance versus the 1 to 1 plot.

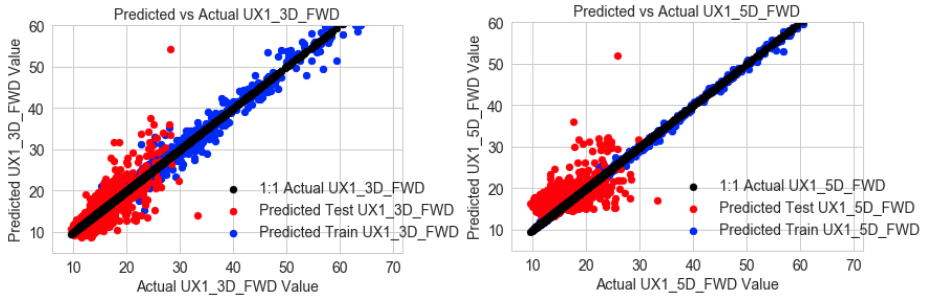


Fig. A12.2 Ensemble Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures 3 and 5 days Forward

Fig, A12.3 shows the Actual versus the estimated test data only for UX1 both 3 and 5-days forward.

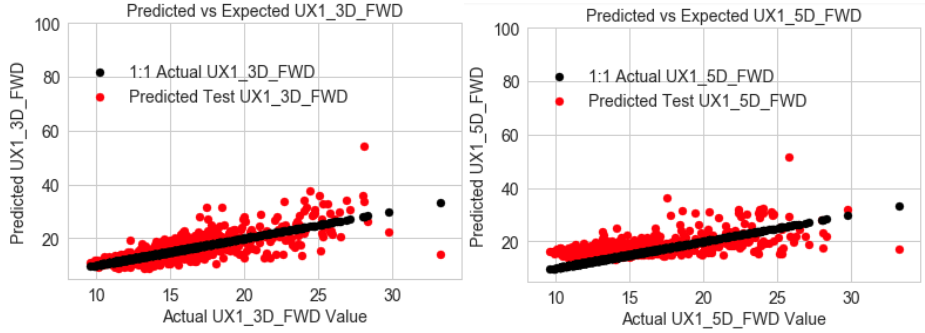


Fig. A12.3. Ensemble Scatter Plot of Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jun 2015 to Jun 2018)

Fig. A12.4 shows the ensemble error histogram of the actual versus estimated for the test and training data sets for UX1 3 and 5-days forward.

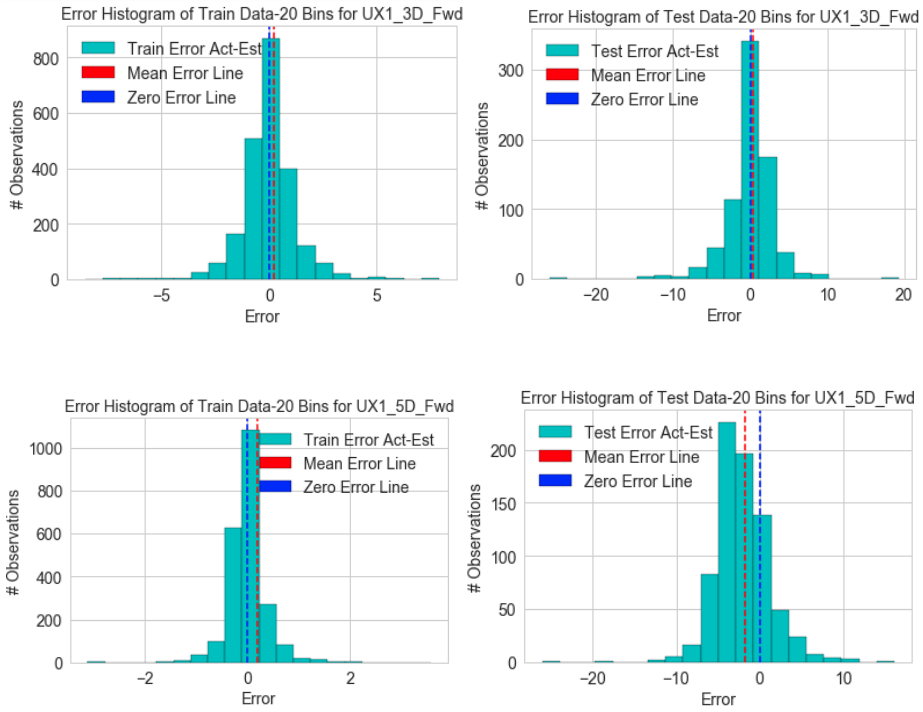


Fig. A12.4. Ensemble Error Histogram of Estimated Training vs. Actual Training and Test vs. Actual Test for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig. A12.5 shows the residual plot for UX1 3 and 5-days forward.

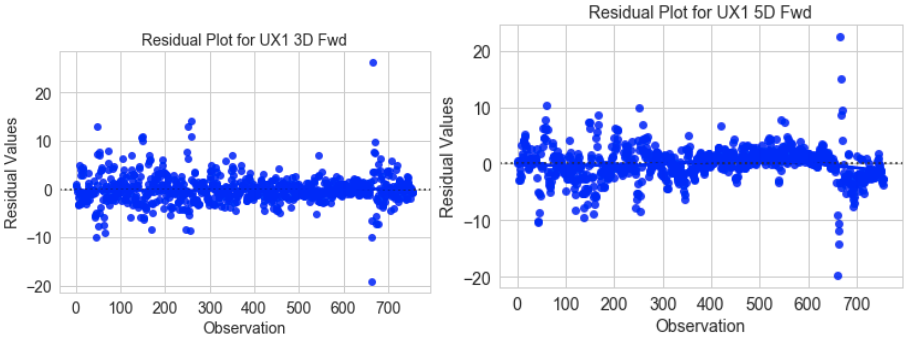


Fig. A12.5. Ensemble Residual Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A12.6 shows the QQ plots of for UX1 3 and 5-days forward.

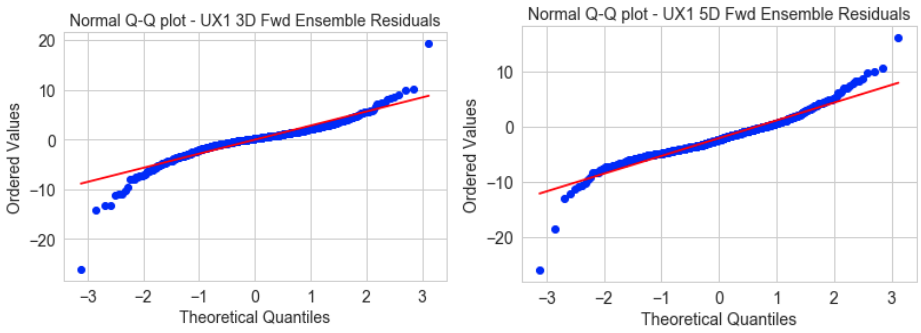


Fig. A12.6. Ensemble QQ Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A12.7 shows the test actual versus estimated line for UX1 for 3 and 5-days forward.

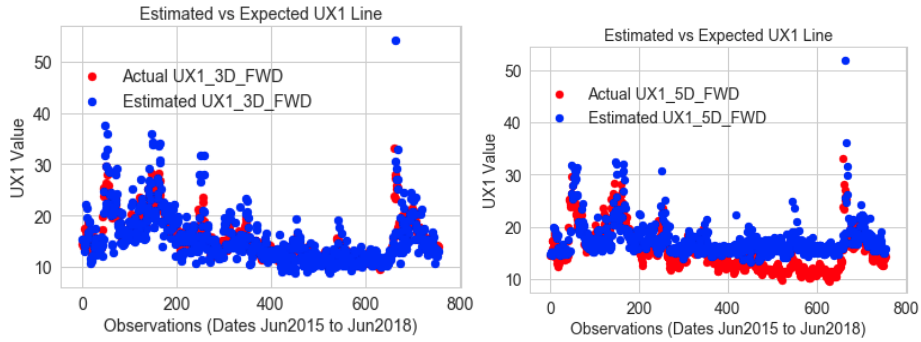


Fig. A12.7. Ensemble Test Actual versus Estimated Line for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Table A12.1 shows a summary of results for both our 10-split cross validation and the 75%/25% train/test split. .

Table A12.1 Some Quality Assessment Results of Ensemble Decision Tree using Bagging Regression with Prior Error Term

Output Forecasted	Inputs Reduced	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	$\rho(train)^*$	$\rho(test)^*$	R^2_{test}	MSE_{test}
3D Fwd.	16	0.98	0.40	1.58	9.11	0.99	0.80	0.05	43.49
5D Fwd.	16	0.99	0.26	0.14	15.57	0.99	0.59	-0.19	49.45

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

Table A12.2 contains the output of our accuracy matrix for true positives and negative as well as false positive and negatives for both 3 and 5 days forward.

Table A12.2 Accuracy Matrix of Ensemble (Jun 2015 to Jun 2018)

Response	True Positives	False Positives	True Negative	False Negative	TP Rate	TN Rate	FN Rate	FP Rate
3D Train	832	46	995	105	0.89	0.96	0.11	0.04
3D Test	177	84	258	36	0.83	0.75	0.17	0.25
5D Train	797	105	929	109	0.88	0.90	0.12	0.10
5D Test	260	209	86	21	0.93	0.29	0.07	0.71

Appendix 13: LASSO Output

Optimization of Hyper-Parameters for LASSO: Alpha is the elasticity factor that controls the balance between lasso and ridge penalties. Our analysis uses a higher alpha of 0.95 (testing a range between 1.0 and 0) to reduce the MSE for both UX1 3 and 5-days forward shown in Fig. A13.1.

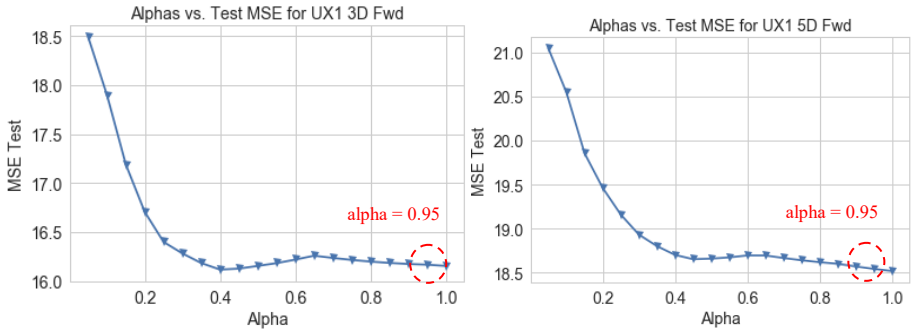


Fig. 13.1 LASSO Alphas versus MSE for test data for both UX1 3 and 5-days forward (Jun 2015 to Jun 2018)

Quality Assessment of Results for LASSO: Fig. A13.2 shows the LASSO scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for both UX1 3 and 5-days forward. The scatterplots show generally a linear relationship for both the test and training estimates for 3 and 5-days forward.

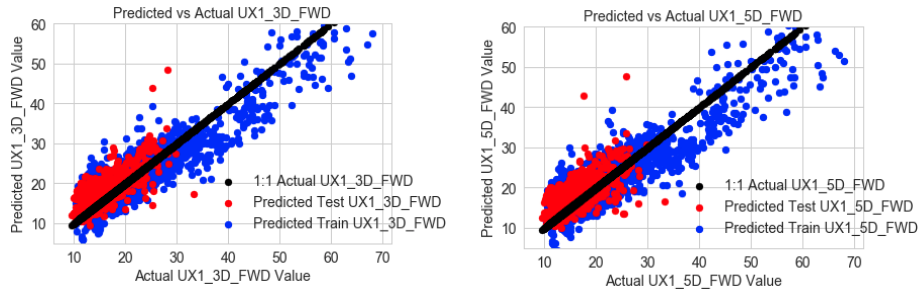


Fig. A13.2. LASSO Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig, A13.3 shows the Actual versus the estimated test data only for UX1 both 3 and 5-days forward for LASSO.

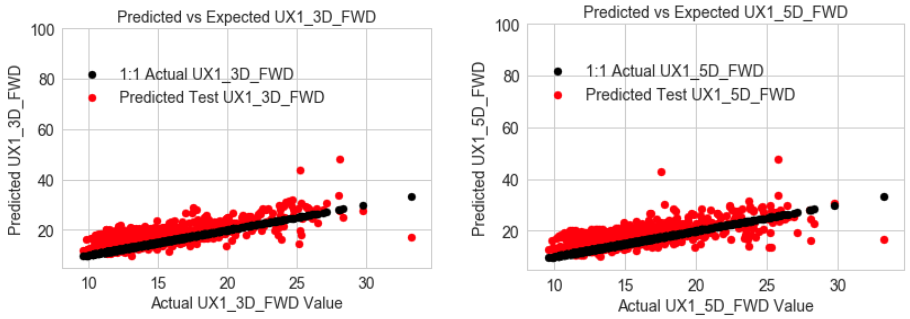


Fig. A13.3 LASSO Scatter Plot of Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jun 2015 to Jun 2018)

Fig. A13.4 shows the LASSO error histogram of the actual versus estimated for the test data sets for UX1 for 3 and 5-days forward.

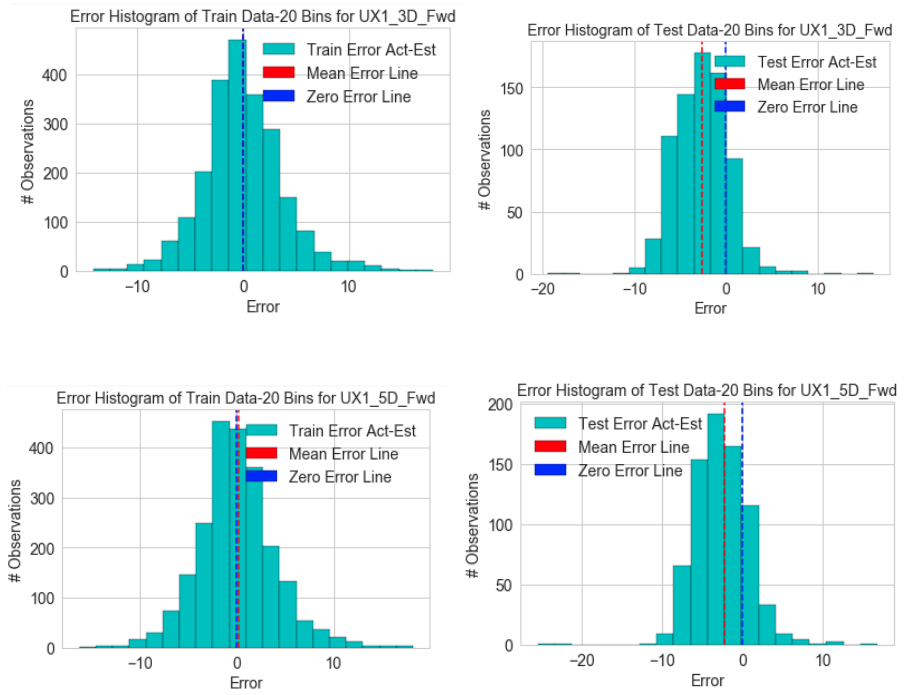


Fig. A13.4 LASSO Error Histogram of Estimated Training vs. Actual Training and Test vs. Actual Test for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig. A13.5 shows the residual plot for UX1 3 and 5-days forward for RF.

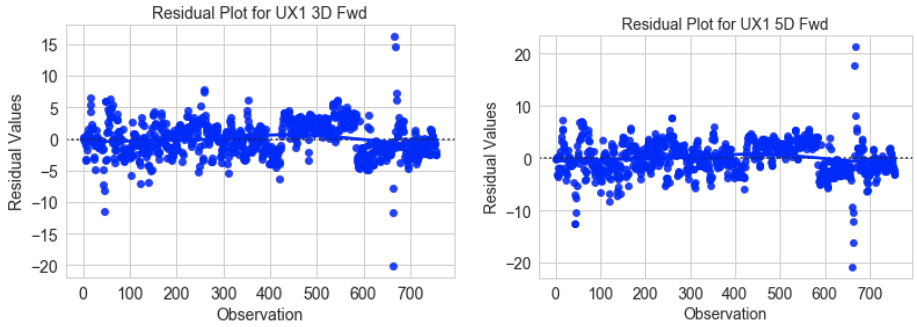


Fig. A13.5 LASSO Residual Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A13.6 shows the QQ plots of for UX1 3 and 5-days forward showing a mostly normal distribution.

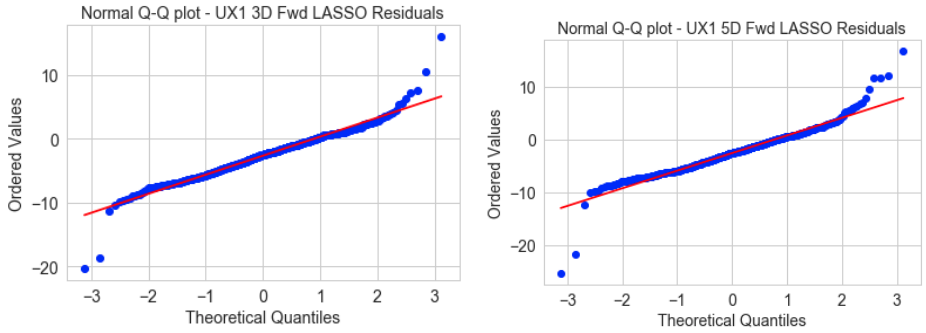


Fig. A13.6 LASSO QQ Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A13.7 shows the test actual versus estimated line for UX1 for 3 and 5-days forward.

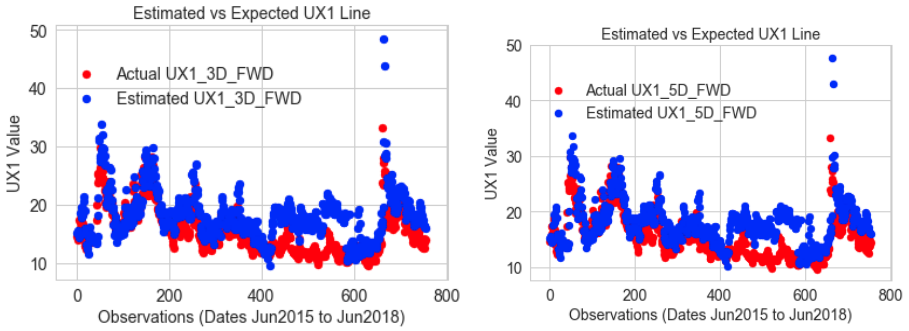


Fig. A13.7. LASSO Test Actual versus Estimated Line for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Table A13.1 shows a summary of results for both our 10-split cross validation and the 75%/25% train/test split. Using 10-split cross validation, the MSE of the test data is higher and the R^2 of the test is higher than the traditional split. The results so far look very good compared to the models analyzed so far except the MSE for our 10-Split cross-validation is high.

Table A13.1 Some Quality Assessment Results of LASSO

Output Forecasted	Inputs Reduced	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	$\rho(train)^*$	$\rho(test)^*$	R^2_{test}	MSE_{test}
3D Fwd.	16	0.83	0.39	14.21	16.16	0.91	0.72	0.33	42.75
5D Fwd.	15	0.81	0.22	16.09	18.54	0.90	0.62	0.32	53.64

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

Table A13.2 shows the accuracy matrix for the LASSO model for the 3 and 5-day training and test datasets.

Table A13.2 Accuracy Matrix of LASSO (Jun 2015 to Jun 2018)

Response	True Positives	False Positives	True Negative	False Negative	TP Rate	TN Rate	FN Rate	FP Rate
3D Train	547	329	605	139	0.79	0.64	0.20	0.35
3D Test	277	240	39	17	0.94	0.14	0.06	0.86
5D Train	552	339	602	137	0.80	0.64	0.20	0.36
5D Test	277	241	38	18	0.94	0.13	0.06	0.86

Appendix 14: SVR Output

Quality Assessment of Results for SVR: Fig. A14.1 shows the SVR scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for both UX1 3 and 5-days forward. The scatterplots show generally a linear relationship for both the test and training estimates for 3 and 5-days forward.

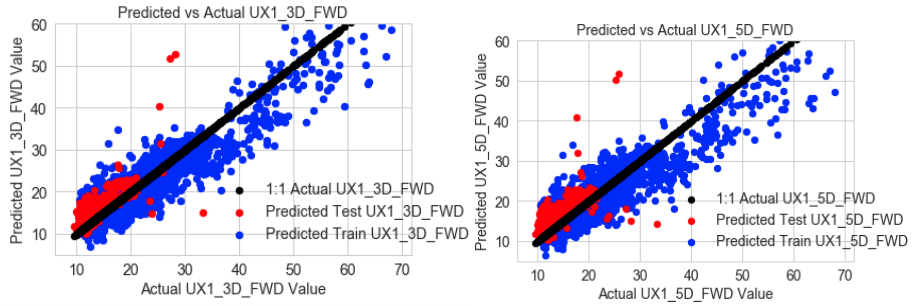


Fig. A14.1. SVR Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig, A14.2 shows the Actual versus the estimated test data only for UX1 both 3 and 5-days forward for SVR.

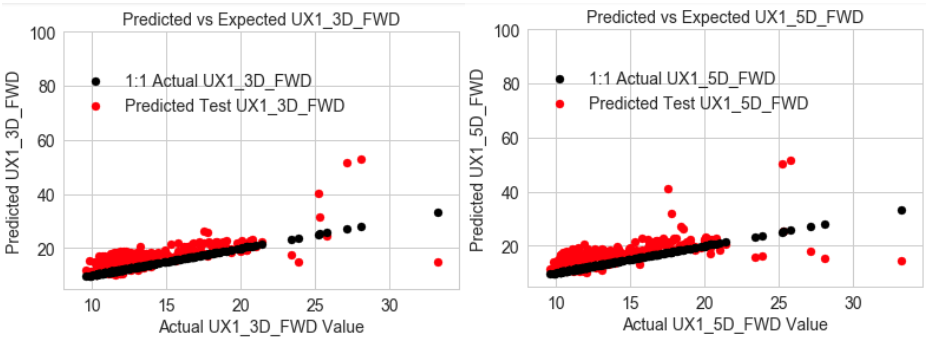


Fig. A14.2. SVR Scatter Plot of Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jun 2015 to Jun 2018)

Fig. A14.3 shows the SVR error histogram of the actual versus estimated for the test data sets for UX1 for 3 and 5-days forward.

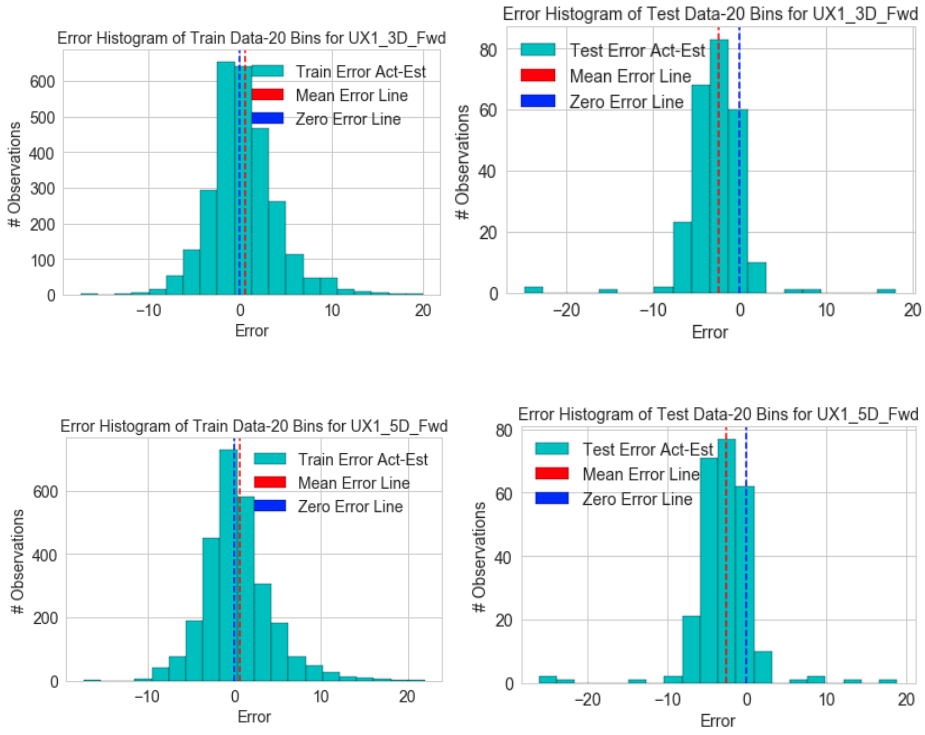


Fig. A14.3. SVR Error Histogram of Estimated Training vs. Actual Training and Test vs. Actual Test for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig. A14.4 shows the residual plot for UX1 3 and 5-days forward for SVR.

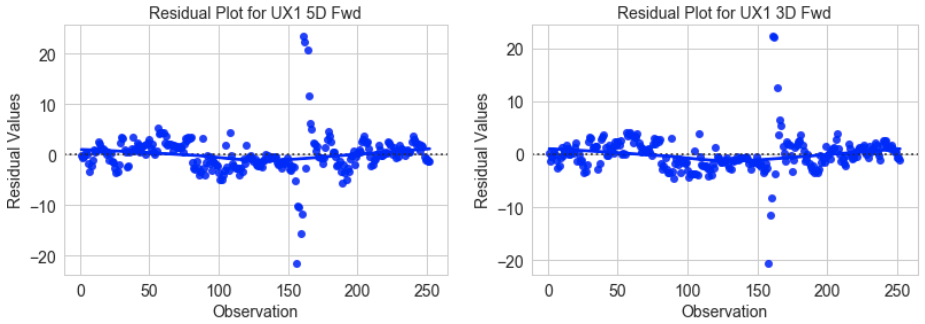


Fig. A14.4. SVR Residual Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A14.5 shows the QQ plots of for UX1 3 and 5-days forward.

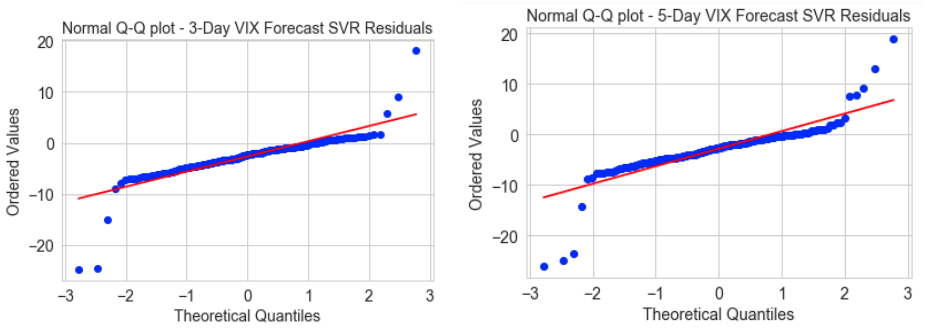


Fig. A14.5. SVR QQ Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A14.6 shows the test actual versus estimated line for UX1 for 3 and 5-days forward.

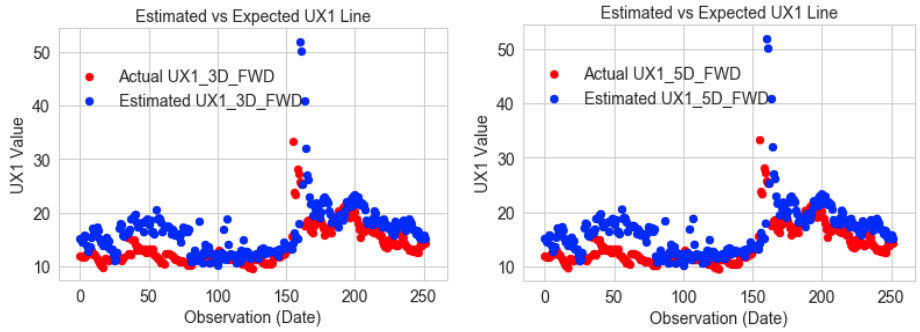


Fig. A14.6. SVR Test Actual versus Estimated Line for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Table A14.1 shows commentary for the SVR scatterplot and error histograms along with the MSE and correlation of the test and training actual versus estimated datasets for UX1 both 3 and 5 days forward. Table A14.2 shows the accuracy matrix for the SVR model for the 3 and 5-day training and test datasets.

Table A14.1. Some Quality Assessment Results of SVR

Output Forecasted	Inputs Reduced	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	$\rho(train)^*$	$\rho(test)^*$	R^2_{test}	MSE_{test}
3D Fwd.	15	0.82	0.19	18.81	15.11	0.91	0.72	0.34	30.28
5D Fwd.	15	0.80	0.12	18.41	16.85	0.90	0.63	0.34	28.99

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

Table A14.2. Accuracy Matrix of SVR (Jun 2015 to Jun 2018)

Response	True Positives	False Positives	True Negative	False Negative	TP Rate	TN Rate	FN Rate	FP Rate
3D Train	481	279	645	163	0.75	0.70	0.25	0.30
3D Test	262	224	44	16	0.94	0.16	0.06	0.84
5D Train	496	302	643	160	0.76	0.68	0.24	0.32
5D Test	266	215	49	14	0.95	0.19	0.05	0.81

Appendix 15: RNN Output

Quality Assessment of Results for RNN: Fig. A15.1 shows the validation accuracy versus loss per epoch for the training data, which shows that there is little improvement after 200 epochs for UX1 3 and 5-days forward. The lower the loss, the better a model (unless the model has over-fitted to the training data). The loss is calculated on training and validation and its interpretation is how well the model is doing for these two sets. Unlike accuracy, loss is not a percentage. It is a summation of the errors made for each example in training or validation sets.

UX1 3D Forward UX1 5D Forward

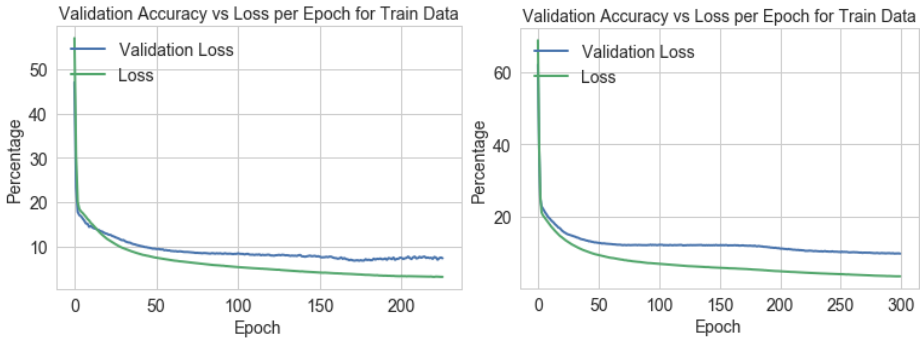


Fig. A15.1 Validation Accuracy versus Loss per Epoch for Training Data for both 1-mth VIX Futures 3 and 5-Days Forward

Quality Assessment of Results for RNN: Fig. A15.2 shows the RNN scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for both UX1 3 and 5-days forward. The scatterplots show generally a linear relationship for both the test and training estimates for 3 and 5-days forward.

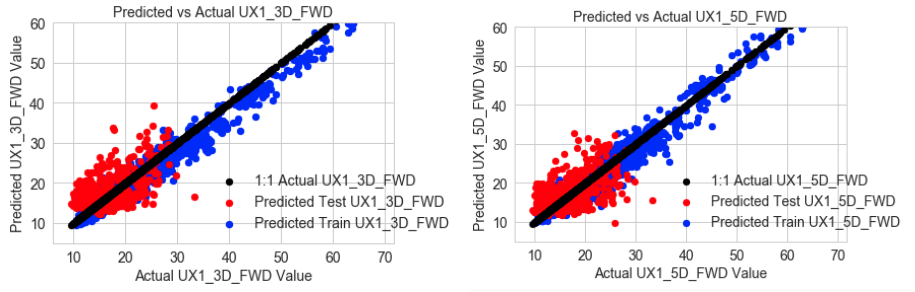


Fig. A15.2. RNN Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig, A15.3 shows the Actual versus the estimated test data only for UX1 both 3 and 5-days forward for RNN.

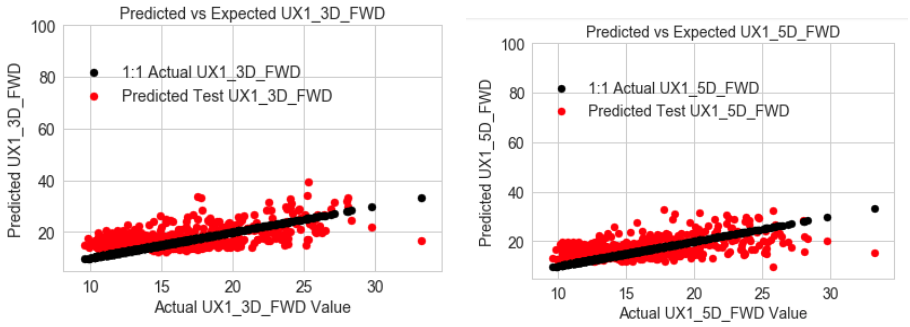


Fig. A15.3 RNN Scatter Plot of Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jun 2015 to Jun 2018)

Fig. A15.4 shows the RNN error histogram of the actual versus estimated for the test data sets for UX1 for 3 and 5-days forward.

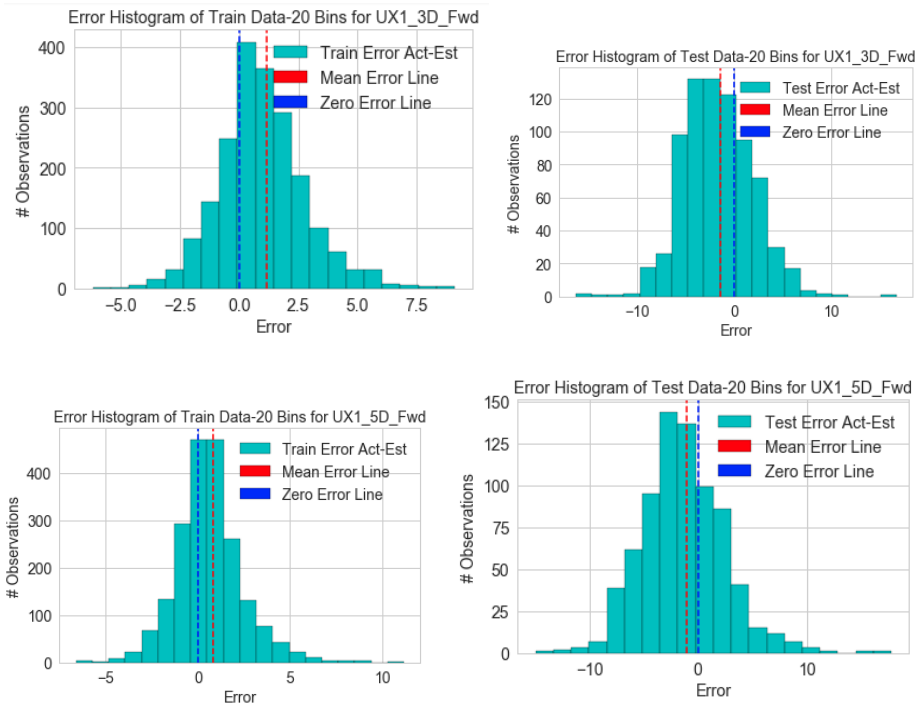


Fig. A15.4 RNN Error Histogram of Estimated Training vs. Actual Training and Test vs. Actual Test for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig. A15.5 shows the residual plot for UX1 3 and 5-days forward for RNN.

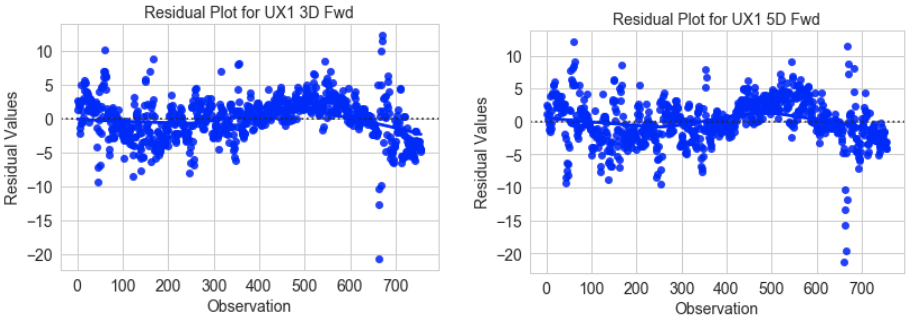


Fig. A15.5 RNN Residual Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A15.6 shows the QQ plots of for UX1 3 and 5-days forward showing a mostly normal distribution.

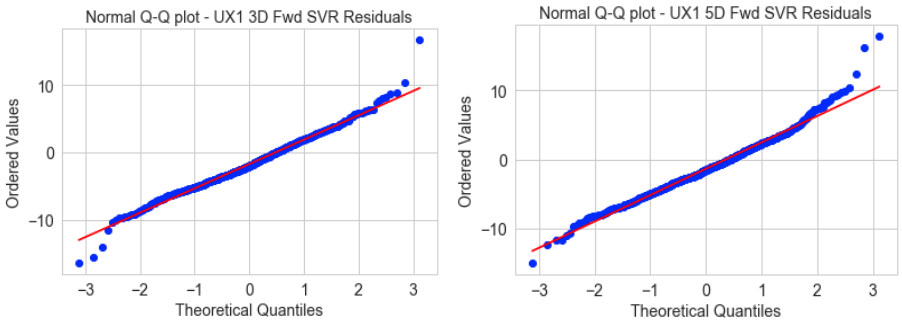


Fig. A15.6 RNN QQ Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A15.7 shows the test actual versus estimated line for UX1 for 3 and 5-days forward.

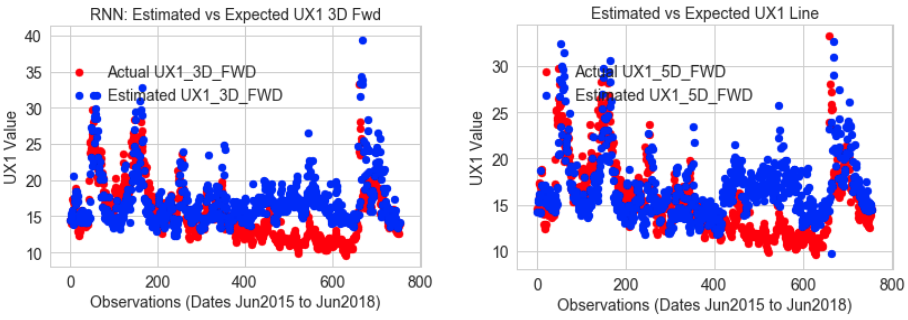


Fig. A15.7. RNN Test Actual versus Estimated Line for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Table A15.1 shows a summary of results for both our 10-split cross validation and the 75%/25% train/test split. Using 10-split cross validation, the MSE of the test data is higher and the R^2 of the test is about the same as the traditional split. Overall for both the traditional and 10-sokit cross validation, the results are very good compared to the models analyzed so far with higher variance explained (R^2) and lower MSE.

Table A15.1 Some Quality Assessment Results of RNN

Output Forecasted	Inputs Reduced	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	$\rho(train)^*$	$\rho(test)^*$	R^2_{test}	MSE_{test}
3D Fwd.	71	0.96	0.42	4.01	15.87	0.98	0.60	0.43	22.34
5D Fwd.	71	0.95	0.03	4.8	15.48	0.98	0.49	0.45	23.37

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

Table A15.2 shows the accuracy matrix for the RNN model for the 3 and 5-day training and test datasets.

Table A15.2 Accuracy Matrix of RNN (Jun 2015 to Jun 2018)

Response	True Positives	False Positives	True Negative	False Negative	TP Rate	TN Rate	FN Rate	FP Rate
3D Train	415	288	623	144	0.74	0.68	0.26	0.32
3D Test	228	195	113	23	0.91	0.37	0.09	0.63
5D Train	407	283	630	149	0.73	0.69	0.27	0.31
5D Test	198	148	183	28	0.88	0.55	0.12	0.45

Appendix 16: LSTM Output

Quality Assessment of Results for LSTM: Fig. A16.1 shows the validation accuracy versus loss per epoch for the training data, which shows that there is little improvement after 200 epochs for UX1 3 and 5-days forward. The lower the loss, the better a model (unless the model has over-fitted to the training data). The loss is calculated on training and validation and its interpretation is how well the model is doing for these two sets. Unlike accuracy, loss is not a percentage. It is a summation of the errors made for each example in training or validation sets.

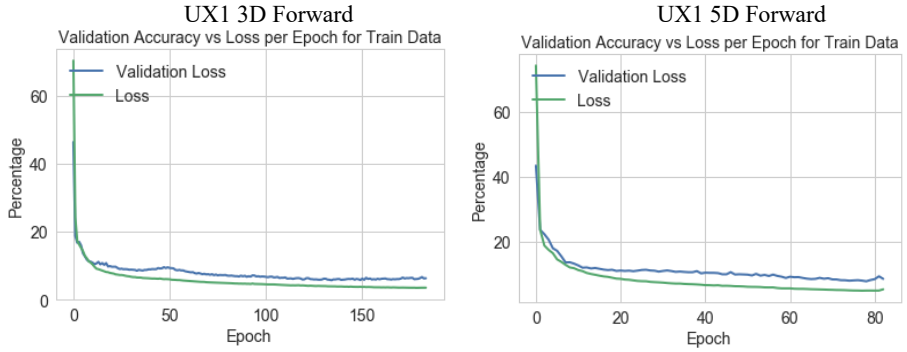


Fig. A16.1 Validation Accuracy versus Loss per Epoch for Training Data for both 1-mth VIX Futures 3 and 5-Days Forward

Fig. A16.2 shows the LSTM scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for both UX1 3 and 5-days forward. The scatterplots show generally a linear relationship for both the test and training estimates for 3 and 5-days forward.

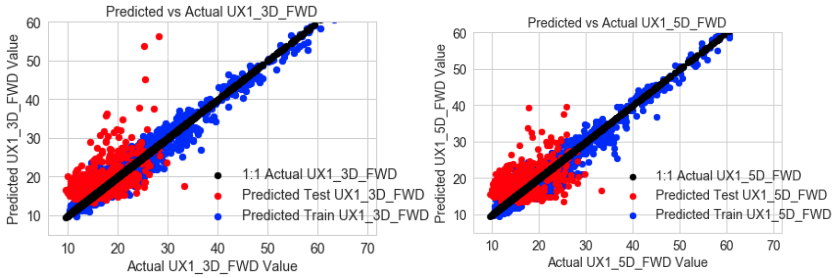


Fig. A16.2. LSTM Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig, A16.3 shows the Actual versus the estimated test data only for UX1 both 3 and 5-days forward for LSTM.

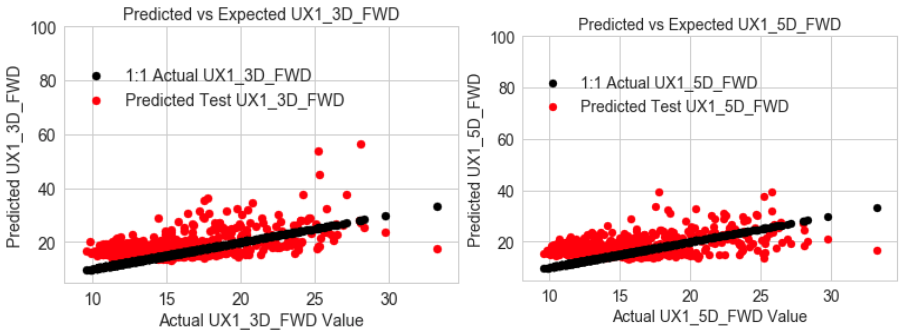


Fig. A16.3 LSTM Scatter Plot of Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jun 2015 to Jun 2018)

Fig. A16.4 shows the LSTM error histogram of the actual versus estimated for the test data sets for UX1 for 3 and 5-days forward. The test data error histograms are only slightly right skewed indicating a better fit.

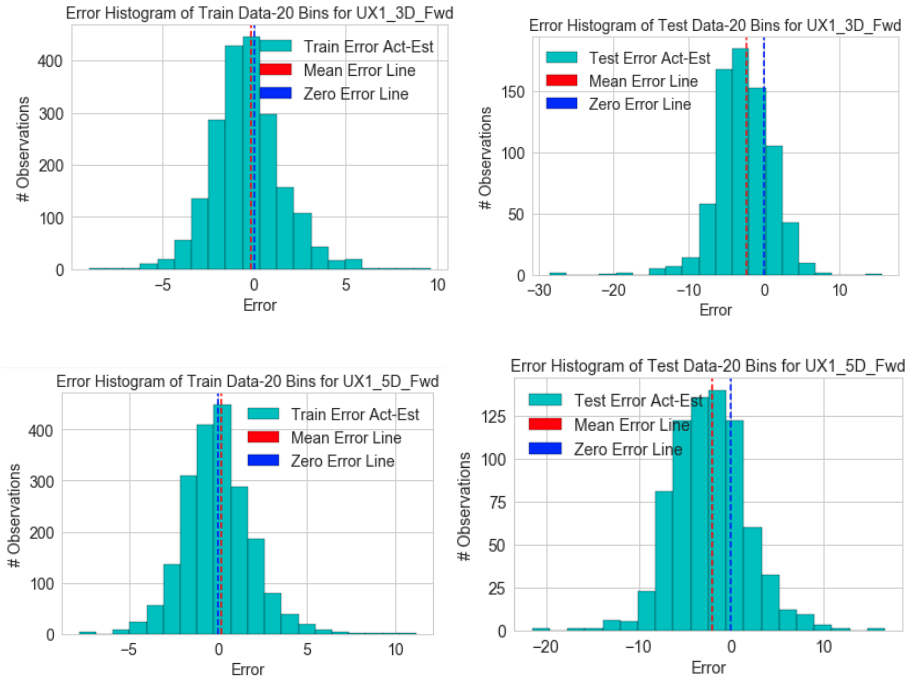


Fig. A16.4 LSTM Error Histogram of Estimated Training vs. Actual Training and Test vs. Actual Test for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig. A16.5 shows the residual plot for UX1 3 and 5-days forward for LSTM.

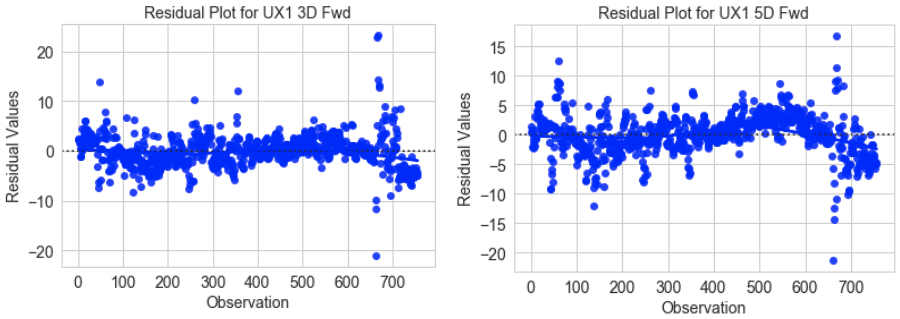


Fig. A16.5 LSTM Residual Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A16.6 shows the QQ plots of for UX1 3 and 5-days forward showing a mostly normal distribution.

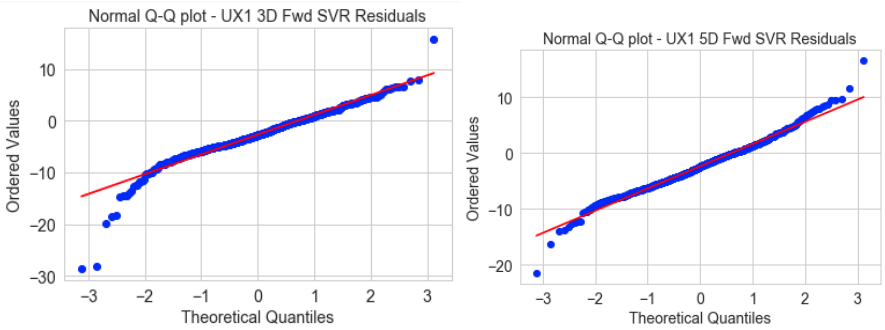


Fig. A16.6 LSTM QQ Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A16.7 shows the test actual versus estimated line for UX1 for 3 and 5-days forward.

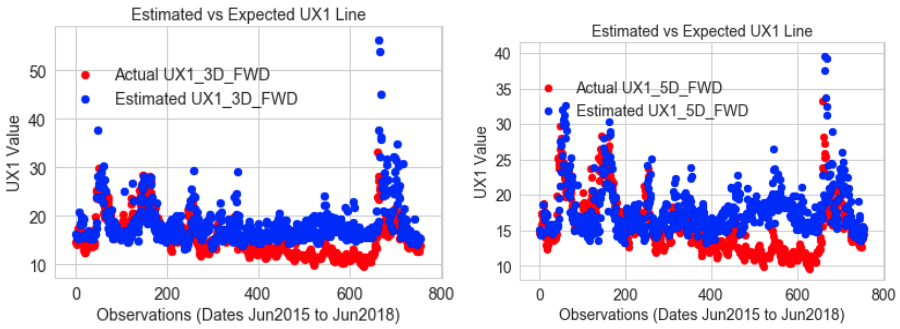


Fig. A16.7. LSTM Test Actual versus Estimated Line for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Table A16.1 shows a summary of results for both our 10-split cross validation and the 75%/25% train/test split. Using 10-split cross validation, the MSE of the test data is higher and the R^2 of the test is about the same as the traditional split. Overall for both the traditional and 10-sokit cross validation, the results are very good compared to the models analyzed so far with higher variance explained (R^2) and lower MSE.

Table A16.1 Some Quality Assessment Results of LSTM

Output Forecasted	Inputs Reduced	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	$\rho(train)^*$	$\rho(test)^*$	R^2_{test}	MSE_{test}
3D Fwd.	71	0.96	0.42	4.01	15.87	0.98	0.60	0.43	22.34
5D Fwd.	71	0.96	0.03	3.76	21.62	0.98	0.42	0.45	23.37

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

Table A16.2 shows the accuracy matrix for the LSTM model for the 3 and 5-day training and test datasets.

Table A16.2 Accuracy Matrix of LSTM (Jun 2015 to Jun 2018)

Response	True Positives	False Positives	True Negative	False Negative	TP Rate	TN Rate	FN Rate	FP Rate
3D Train	467	330	564	132	0.78	0.63	0.22	0.37
3D Test	228	195	113	23	0.91	0.37	0.09	0.63
5D Train	444	313	579	141	0.76	0.65	0.24	0.35
5D Test	238	193	118	15	0.94	0.38	0.06	0.62

Appendix 17: RF Output

Quality Assessment of Results for RF: The top 14 input variables for 3 and 5-days forward are the same. And shown in Fig. A17.1.

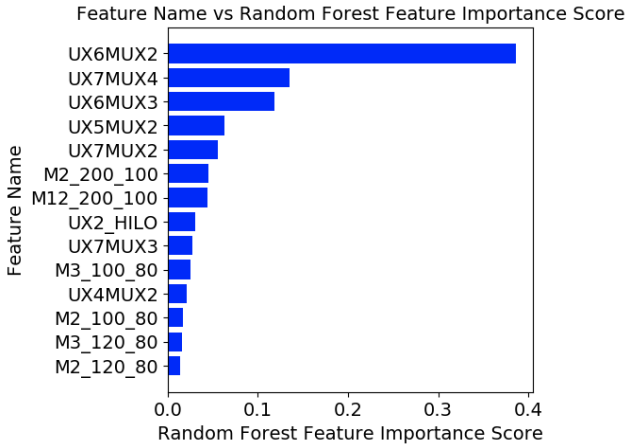


Fig. A17.1 Top 14 Features Selected for 1-mth VIX Futures 3 and 5-Days Forward

Fig. A17.2 shows the RF scatterplot of the output for the training versus test actual and estimated values as well as 1 to 1 plot of the perfect output for the training dataset as a benchmark for both UX1 3 and 5-days forward.

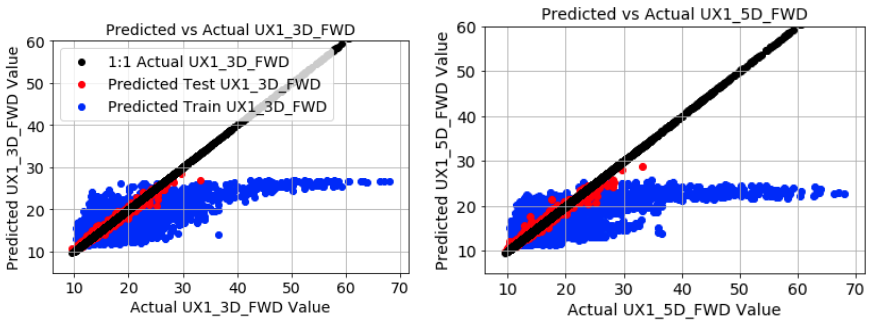


Fig. A17.2. RF Scatter Plot of Training & Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig, A17.3 shows the Actual versus the estimated test data only for UX1 both 3 and 5-days forward for RF.

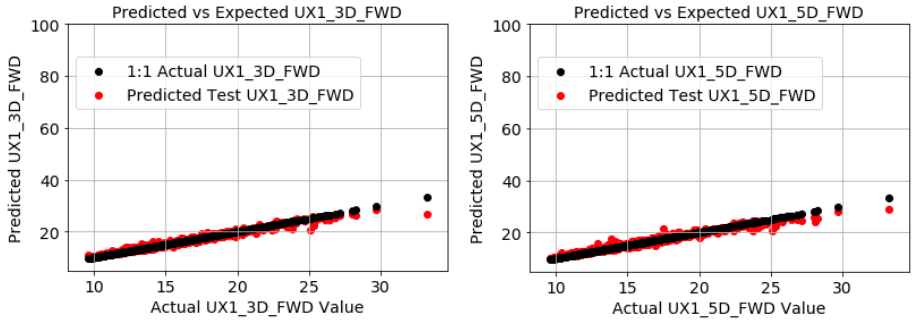


Fig. A17.3 RF Scatter Plot of Test Actual vs. Estimated for 1-mth VIX Futures (UX1) 3 and 5-days Forward (Jun 2015 to Jun 2018)

Fig. A17.4 shows the RF error histogram of the actual versus estimated for the test data sets for UX1 for 3 and 5-days forward. The test data error histograms are only slightly right skewed indicating a better fit.

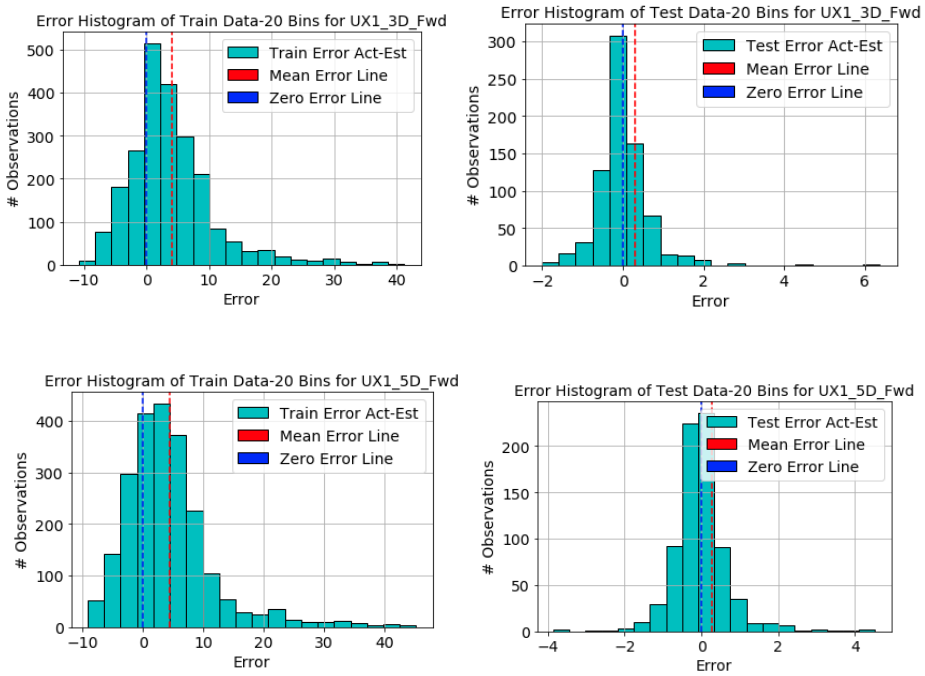


Fig. A17.4 RF Error Histogram of Estimated Training vs. Actual Training and Test vs. Actual Test for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jul 2006 to Jun 2015 for Train & Jun 2015 to Jun 2018 for Test)

Fig. A17.5 shows the residual plot for UX1 3 and 5-days forward for RF.

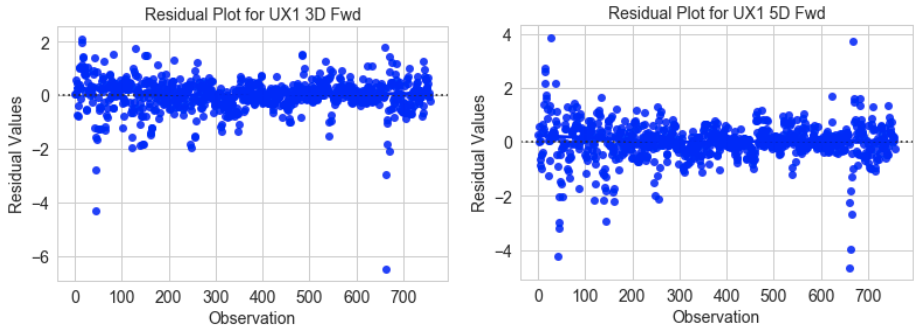


Fig. A17.5 RF Residual Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A17.6 shows the QQ plots of for UX1 3 and 5-days forward showing a mostly normal distribution.

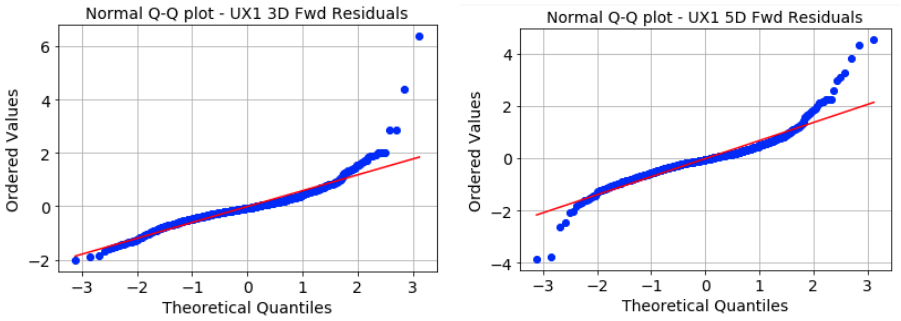


Fig. A17.6 RF QQ Plots for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Fig. A17.7 shows the test actual versus estimated line for UX1 for 3 and 5-days forward.

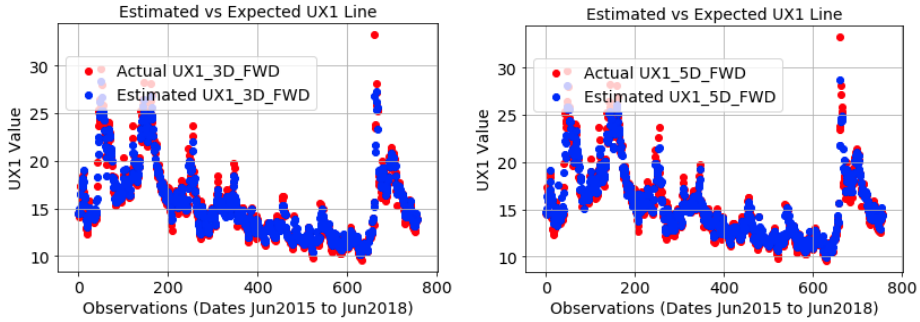


Fig. A17.7. RF Test Actual versus Estimated Line for 1-mth VIX Futures (UX1) 3 and 5-day Forward (Jun 2015 to Jun 2018)

Table A17.1 shows a summary of results for both our 10-split cross validation and the 75%/25% train/test split. Using 10-split cross validation, the MSE of the test data is higher and the R^2 of the test is about the same as the traditional split. Overall for both the traditional and 10-sokit cross validation, the results are very good compared to the models analyzed so far with higher variance explained (R^2) and lower MSE.

Table A17.1 Some Quality Assessment Results of RF

Output Forecasted	Inputs Reduced	Traditional 75%/25% Train/Test Split						10-Split CV	
		R^2_{train}	R^2_{test}	MSE_{train}	MSE_{test}	$\rho(train)^*$	$\rho(test)^*$	R^2_{test}	MSE_{test}
3D Fwd.	14	0.43	0.97	62.93	0.41	0.71	0.98	0.37	45.52
5D Fwd.	14	0.33	0.96	74.55	0.54	0.61	0.98	0.35	50.34

*ρ(train) is the correlation of the actual to the estimated training data set (in-sample). ρ(test) is the correlation of the actual to the estimated test data set (out-sample)

Table A17.2 shows the accuracy matrix for the RF model for the 3 and 5-day training and test datasets.

Table A17.2 Accuracy Matrix of RF (Jun 2015 to Jun 2018)

Response	True Positives	False Positives	True Negative	False Negative	TP Rate	TN Rate	FN Rate	FP Rate
3D Train	281	177	884	189	0.60	0.83	0.40	0.17
3D Test	284	14	365	17	0.94	0.96	0.06	0.04
5D Train	300	194	874	191	0.61	0.82	0.39	0.18
5D Test	236	40	328	35	0.87	0.89	0.13	0.11