


# Exploiting Visual Features in Financial Time Series Prediction

Adil Gürsel Karaçor, Atilim University, Ankara, Turkey

 <https://orcid.org/0000-0003-2179-2417>

Turan Erman Erkan, Atilim University, Ankara, Turkey

## ABSTRACT

The possibility to enhance prediction accuracy for foreign exchange rates was investigated in two ways: first applying an outside the box approach to modeling price graphs by exploiting their visual properties, and secondly employing the most efficient methods to detect patterns to classify the direction of movement. The approach that exploits the visual properties of price graphs which make use of density regions along with high and low values describing the shape; hence, the authors propose the name 'Finance Vision.' The data used in the predictive model consists of 1-hour past price values of 4 different currency pairs, between 2003 and 2016. Prediction performances of state-of-the-art methods; Extreme Gradient Boosting, Artificial Neural Network and Support Vector Machines are compared over the same data with the same sets of features. Results show that density based visual features contribute considerably to prediction performance.

## KEYWORDS

Artificial Neural Networks, Extreme Gradient Boosting, Forex, Machine Learning, Machine Vision, Predictability, Quantitative Analysis, Support Vector Machine

## INTRODUCTION

Huge amount of money flows into the markets all around the world for the trade of various financial instruments such as stocks, commodities, foreign exchange (FOREX), futures, and so on every day. For example, according to experts and professionals, average daily turnover in FOREX markets alone is in excess of 5 trillion US dollars (Chandler, 2013). Investment and trading decisions, whether they tend to be long term or short term, are mainly based on predicting the future movements of the financial instrument(s) in question. There has been an ongoing debate among researchers on whether financial markets are predictable or not for a long time. Some think that financial market movements are nothing but random walk, and some findings support that claim: for example, VIX futures prices were found to be unpredictable by Konstantinidi and Skiadopoulos (2011). Furthermore, some researchers even claimed that none of the conventional predictive models proposed in the literature on stock prediction seems capable of systematically predicting stock returns in long range of time horizons, and speculators do not earn significant profits in commodity and interest rate futures markets

DOI: 10.4018/IJCINI.2020040104

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

in aggregate (Hartzmark, 1987), (Bossaerts & Hillion, 1999), (Goyal & Welch, 2008). On the other hand, many researchers disagree with this random walk approach. Some of them claim speculators can gain profits on commodity and currency futures (Yoo & Maddala, 1991), (Taylor, 1992), (Kho, 1996), (Wang, 2004), (Strozzi & Zaldivar, 2005). Some others believe that financial instruments, commodities in particular, are predictable, at least to a certain extent (Campbell & Thompson, 2008), (Zunino et al., 2010). The debate has not been settled yet, however, it is fair to say that financial instruments are definitely hard to model or predict, if not totally unpredictable. Obviously, it would be valuable information for the investor if he or she could know in advance which way an instrument would go. In this paper an investigation is carried out in order to give the investor a potential advantage by trying to answer the following question: would a certain FOREX rate be likely to go up or down in the following hours? Technical (quantitative) analysis i.e. past price values of the instruments is solely used, and fundamental analysis is not in the scope of this study.

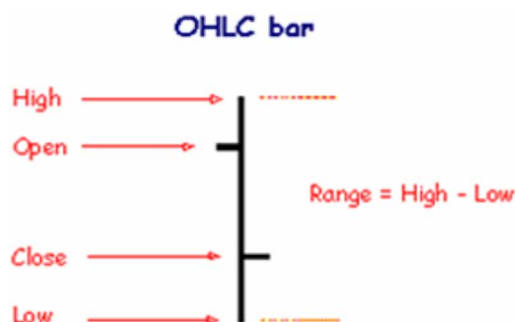
## A DIFFERENT APPROACH TO TIME SERIES

As stated at Introduction, the main and intriguing question here is: would a certain FOREX rate be likely to go up or down in the following hours?

Vast majority of time series prediction models are Moving Average (MA) based; such as Auto Regressive Moving Average (ARMA), Auto Regressive Integrated Moving Average (ARIMA), Auto Regressive Integrated Moving Average with eXogeneous input (ARIMAX) and Nonlinear Auto Regressive Moving Average with eXogeneous input (NARMAX). Since the intention is to use past price movements only, there is a need to come up with something creative to make a difference. Therefore, we do not include any MA in our prediction model. Instead, we propose the usage of visual features related to the shapes of FOREX price movements to classify future trends. We are inspired by the fact that one can somewhat determine the differences among the price movements by visually inspecting their graphs. The proposed approach; called herein Finance Vision method, is similar to Machine Vision which is the technology that employs image processing methods to recognize patterns, in the way that, the FOREX price movements are treated like images to get certain features, just as the saying goes: “a picture is worth a thousand words” (Duda et al., 1973), (Gonzales & Woods, 2007). These features are then used for training state-of-the-art classifiers to recognize future price trends. Experiments show that comparable recognition rates are obtained. Consequently, Finance Vision can be defined as financial time series recognition and prediction using visual properties of financial time series graphs.

As per visual properties of financial time series graphs; density levels are used together with peak and trough values of the financial instruments. Financial data are typically represented by bar charts with Open (O), High (H), Low (L), and Close (C and OHLC altogether) denoting corresponding price values, see Figure 1. Data graphs for financial time series are commonly made up of series of

Figure 1. Common financial data representation



OHLC bars. Here, a price value range is said to be dense if there are a large number of OHLC values in that particular range.

## DATA

As mentioned above there are many research papers on predictability of financial instruments; however, predictability comparison of these instruments in literature is very rare. In fact, the only study we came across was the one comparing real estate returns with stock returns by Serrano and Hoesli (2010). On the other hand, in our previous work we did an extensive predictability comparison, and the most predictable instrument was found to be FOREX (Karacor & Erkan, 2016). In the same work, the most predictable frequencies were turned out to be 1-minute and 5-minute. However, neither the predictability performance of 1-minute data nor the predictability performance of 5-minute data was much better than those of 1-hour or 4-hour data. Therefore, from the investor's point of view, trading decisions based on 1-hour or 4-hour data could even be more profitable, considering commissions and spread margins are usually more disadvantageous in higher frequencies like 5-minute or 1-minute. Hence as an optimum frequency, 1-hour data is chosen in this paper, covering the 14-year period between 2003 and 2016 with approximately 85000 data points per FOREX pair, before preprocessing. Four individual FOREX pairs are investigated; namely Australian Dollar against US Dollar (AUDUSD); Euro against Canadian Dollar (EURCAD); Euro against US Dollar (EURUSD); and US Dollar against Japanese Yen (USDJPY). The data are extracted from the authors' personal accounts, therefore comprise of real liquid tradable FOREX prices. The data quality is quite high; hence there are very few missing points and/or outliers which are replaced by 5 point moving averages before them.

## METHODOLOGY

### Feature and Target Selection

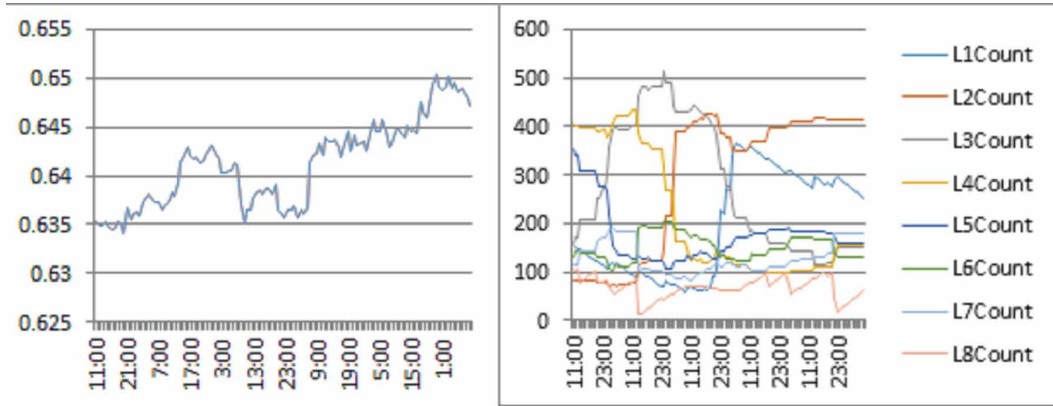
Twenty-three out of the twenty-four features used for our prediction model are related to visual attributes of price movements, and one is related to seasonality. Nine of the visual features are related to density. Density is visualized as distribution of total counts of OHLC values within 8 different price level bands over a certain period i.e. past 377 hours, in other words it is past OHLC count histograms. The remaining fourteen features comprise of various normalized high and low values to complete the overall shape of the financial instrument. Density levels are illustrated in Figure 2.

The output is the binary classification value denoting whether the FOREX instrument would go up a certain high threshold value (1) or go down another low threshold value (0) in the next five hours. However, this is not only a high or low value above threshold: when going up it should not go down the low threshold, and vice versa. For instance; if an instrument goes down 8 units before going up 3 units, successful prediction of 3 units of upward movement will not be useful information to enter a long position. The remaining in between values are tagged as no trend, and excluded from model training.

### State of the Art Classifiers

It should be pointed out that the aim of this study is to suggest a different approach (Finance Vision) in quantitative financial time series prediction; it is not to compare different classifiers. However, three most popular classifiers are chosen to get the best prediction results. The chosen state-of-the-art classifiers are: Extreme Gradient Boosting (XGB) method, Artificial Neural Network (ANN), and Support Vector Machine (SVM). Usage of ANNs is quite popular for modeling, prediction, and decision making over financial data and ANNs are regarded as an excellent tool for the purpose (Dutta & Shekhar, 1988), (Wedding & Cios, 1996), (Cheng et al., 1997), (Yu et al., 2008), (Dash

Figure 2. (a) Sample FOREX price movement; (b) Associated density levels (OHLC histograms)



et al., 2014), (Xi et al., 2014). Like ANNs, usage of SVMs in financial markets is quite common (Choudhury et al., 2014), (Fenghua et al., 2014), (Nayak et al., 2014), (Kumar et al., 2016). On the other hand, usage of XGB method is very rare (Zięba et al., 2016).

## XGB

XGB classifier is chosen as one of the prediction and classification methods. This is mainly because it has gained much popularity in the recent years, due to its overwhelming success in data science competitions (Chen & Guestrin, 2016). Despite its reputation, XGB method has not yet achieved popularity in the literature accordingly.

It is basically a decision tree and C++ based model, and has libraries and interfaces in R, Python, and Julia languages. The model can be described basically as a sum of probabilities:

$$\sum_{k=1}^K f_k \quad (1)$$

where each  $f_k$  is a prediction (probability) from a decision tree. Hence the model is a collection of decision tree probabilities. With all the decision trees, prediction is made by:

$$y_i = \sum_{k=1}^K f_k(x_i) \quad (2)$$

where  $x_i$  is the feature vector for the  $i^{\text{th}}$  data point.

In order to train the model, a loss function is needed. Since it is a binary classification in our case, the loss function is logarithmic loss (logss):

$$L = -\frac{1}{N} \sum_{i=1}^N (y_i \log(p_i) + (1 - y_i) \log(1 - p_i)) \quad (3)$$

where  $N$  denotes number of samples/instances,  $y$  binary variable, and  $p$  is the classification probability at instance  $i$ .

Another important part of the model is regularization. A good regularization term controls the complexity of the model and prevents overfitting:

$$\Omega = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2 \quad (4)$$

where  $T$  is the number of leaves,  $\gamma$  is the relaxation term (constant),  $\lambda$  is the L2 regularization term (constant), and  $w_j^2$  is the score on the  $j^{\text{th}}$  leaf.

With loss function and the regularization term together, the objective of the model is attained:

$$Obj = L + \Omega \quad (5)$$

In this study; following a parameter optimization, we employ an XGB model with 377 booster trees having maximum depth of 8, whilst default values are used for other parameters.

## ANN

ANNs are inspired from biological neural networks and are well known for their high approximation and modeling capabilities (Karacor & Denizhan, 2004), (Karacor et al., 2007). ANNs are used in a wide range of applications from job satisfaction performance evaluation (Aktepe & Ersoz, 2012), solving capacitated P-median problem (Shamsipoor et al., 2012) to prediction of aircraft accident occurrence (Yeoum & Lee, 2013) ANNs are trained in order that they learn a set of input-output data that represent usually a very complex or even undefined function. With sufficient number of hidden layers and neurons, they can model any given input-output relationship (Hornik et al., 1989). All nodes (artificial neurons) are interconnected, thus form a massive parallelism, and each connection has a weight that changes as the ANN is trained, and each node has an activation function. There are numerous activation functions, ranging from simple linear functions to various nonlinear ones. The nonlinearity of activation functions enables the ANN to learn even the most complex patterns.

In this study, again following a parameter optimization, an ANN model is constructed with a multi-layered feed forward network, having 2 hidden layers with 55 and 3 neurons, respectively. Hyperbolic tangent is chosen as activation function. The network is trained by back propagation gradient descent algorithm.

## SVM

SVM maps datasets to higher dimensional spaces in order to more easily (linearly) classify them. Therefore, it is a very strong classification tool. SVM is implemented using the kernel Adatron algorithm. The kernel Adatron maps inputs to a high-dimensional feature space, and then optimally separates data into their respective classes by isolating those inputs which fall close to the data boundaries. Therefore, the kernel Adatron is especially effective in separating sets of data which share complex boundaries. Gaussian kernel functions are used in this study. SVM is chosen as an alternative method to XGB and ANN for FOREX prediction in this paper.

## Classical ARMA Model

To compare the performance of proposed features to a classical method used in time series prediction, one of the most common time series prediction methods ARMA is chosen. An ARMA ( $p, q$ ) model is comprised of sums of  $p$  Auto Regressive terms and  $q$  Moving Average terms plus error terms of

a time series. An ARMA (1,1) model that uses past five-hour highs and lows to predict highest and lowest values of five hours ahead, is employed.

## TRAINING, TESTING, VALIDATION AND RESULTS

The data are randomized to achieve a fair distribution. Only 50% of the data is chosen to train the models, such that only data that go above certain fluctuation (up and down) threshold levels are kept and the rest are eliminated. 80% of data are used for training, 15% for testing, and the remaining 5% for cross validation. All models are trained until the results stop improving in the validation set. The model training and testing data cover the 13-year period between 2003 and 2015. The remaining, out of the range, 1-year data (2016) is used for validation. A statistical analysis between model building and validation data is made. The size of the validation data is approximately 1/12<sup>th</sup> of the model building data; hence, the validation data is repeated for this analysis to get equal data lengths. The ANOVA analysis for this comparison is given in Table 1. The correlation between the two data sets is 0.0249. Judging by the correlation and the analysis given in Table 1, it is fair to say that model building and validation periods have different characteristics. Neurosolutions software (1994 – 2015) is used for ANN and SVM training and Python software (Rossum, 1995) for XGB modeling.

To understand the effects of the density based visual features better, three different feature sets are used in the training and testing of the models: the first set comprises of full density and all other features, second set is the one containing fewer features with density features removed, and finally the third set contains additional features, such as additional high and low values and moving averages, in place of the density features, keeping the total number of features constant.

During the training process, how the mean squared training error changes for the ANN and SVM is illustrated in Figure 3 and Figure 4 respectively, as an example. A statistical R-value and classification performance analysis is made on the test data in order to compare the produced outputs with the actual values that indicated whether predictions succeed or not. Results of this analysis are given in Tables 2, 3, and 4 for the model with full density based visual features, with density features removed, and with density features removed but other extra features introduced instead respectively. Overall comparisons among feature sets are given in Figures 11 and 12 for the model building and out of range validation periods respectively. Also in Table 5, an R value analysis is given of all FOREX pairs on average for the validation set. Moreover, average training times in seconds of the three classifiers are given in Table 6. Finally results of the ARMA model are displayed in Table 7.

A statistically meaningful variable we used for predictability performance comparison is the correlation coefficient R. R is used to measure how well one variable fits on another, linear regression wise. In our case, these variables were predicted against desired, in other words, model outputs vs.

Table 1. ANOVA analysis between model building and validation data

Summary						
Groups	Count	Sum	Average	Variance		
Model building data	79151	67344.22	0.850832	0.014946		
Validation data	79151	59143.31	0.747221	0.000346		
ANOVA						
<i>Source of Variation</i>	<i>SS</i>	<i>df</i>	<i>MS</i>	<i>F</i>	<i>P-value</i>	<i>F crit</i>
Between Groups	424.8518	1	424.8518	55563.93	0	3.841518
Within Groups	1210.39	158300	0.007646			
Total	1635.242	158301				

Figure 3. MSE for ANN training

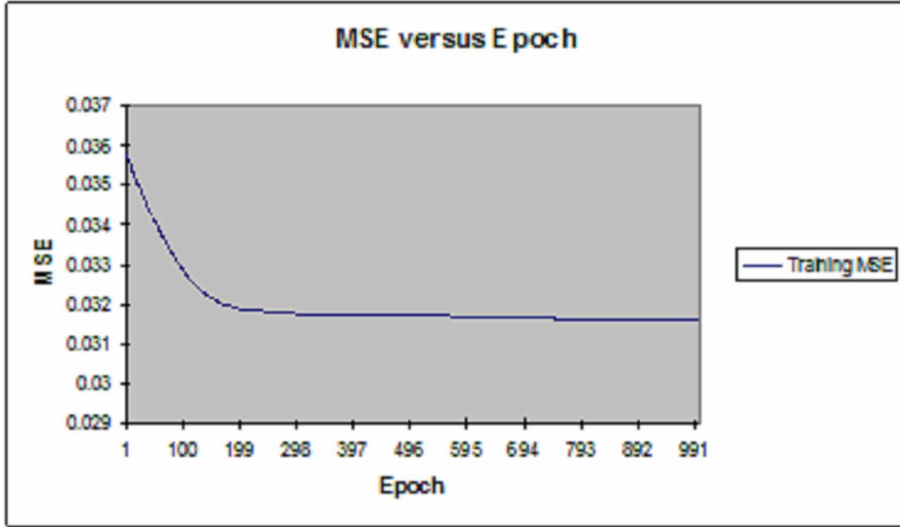
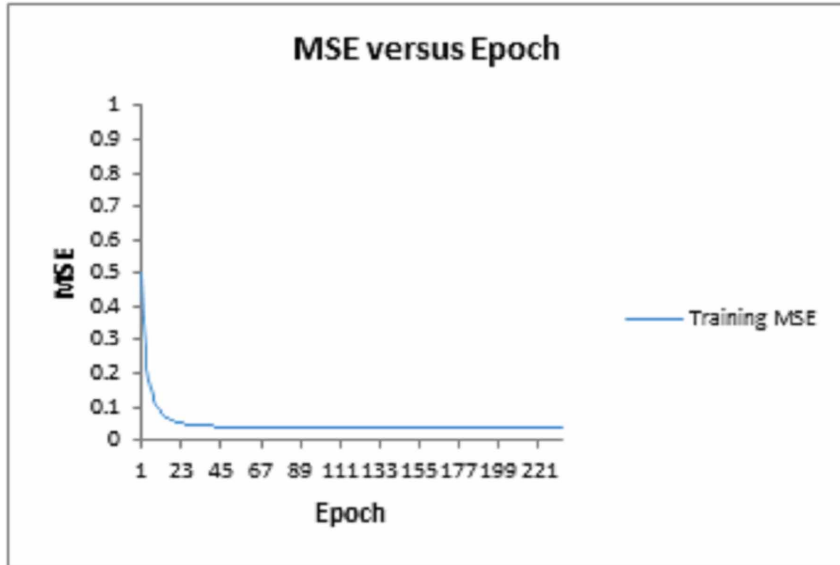


Figure 4. MSE for SVM training



actual classification values denoting movement directions of the FOREX pairs in the following 5 hours. R value is calculated by the formula below:

$$R = \frac{\sum_i (x_i - \bar{x})(d_i - \bar{d})}{\sqrt{\frac{\sum_i (d_i - \bar{d})^2}{N}} \sqrt{\frac{\sum_i (x_i - \bar{x})^2}{N}}} \quad (6)$$

**Table 2. Performance comparison for full visual density-based features**

	<b>XGB</b>				
	<b>Sensitivity</b>	<b>Specificity</b>	<b>Accuracy</b>	<b>Precision</b>	<b>R</b>
AUDUSD	0.78878505	0.802207	0.795819	0.7836583	0.590827
EURCAD	0.83746407	0.863375	0.838635	0.827683	0.698691
EURUSD	0.62103673	0.630012	0.623518	0.612748	0.458639
USDJPY	0.65473571	0.673684	0.664537	0.642535	0.493787
	<b>ANN</b>				
	<b>Sensitivity</b>	<b>Specificity</b>	<b>Accuracy</b>	<b>Precision</b>	<b>R</b>
AUDUSD	0.67291312	0.728733	0.699373	0.733519	0.415513
EURCAD	0.79364731	0.832647	0.803744	0.8547348	0.570436
EURUSD	0.53283632	0.593846	0.547354	0.6046932	0.294438
USDJPY	0.58354845	0.637453	0.604749	0.6544773	0.335256
	<b>SVM</b>				
	<b>Sensitivity</b>	<b>Specificity</b>	<b>Accuracy</b>	<b>Precision</b>	<b>R</b>
AUDUSD	0.80723982	0.835847	0.821685	0.8282266	0.683576
EURCAD	0.83746459	0.850938	0.846453	0.842834	0.749882
EURUSD	0.60999436	0.657484	0.635482	0.629374	0.512497
USDJPY	0.67458321	0.716359	0.708465	0.711286	0.578643

**Table 3. Performance comparison for no density features**

	<b>XGB</b>				
	<b>Sensitivity</b>	<b>Specificity</b>	<b>Accuracy</b>	<b>Precision</b>	<b>R</b>
AUDUSD	0.60186916	0.673175	0.639235	0.62585	0.275713
EURCAD	0.70027383	0.767734	0.723863	0.71989	0.404757
EURUSD	0.51836091	0.583741	0.553908	0.54552	0.189921
USDJPY	0.52539831	0.593896	0.553672	0.54355	0.210928
	<b>ANN</b>				
	<b>Sensitivity</b>	<b>Specificity</b>	<b>Accuracy</b>	<b>Precision</b>	<b>R</b>
AUDUSD	0.55712452	0.562285	0.560484	0.405607	0.187504
EURCAD	0.64983635	0.653709	0.643948	0.510039	0.273645
EURUSD	0.50938732	0.515673	0.515112	0.355638	0.137481
USDJPY	0.52536849	0.538292	0.529931	0.369984	0.157392
	<b>SVM</b>				
	<b>Sensitivity</b>	<b>Specificity</b>	<b>Accuracy</b>	<b>Precision</b>	<b>R</b>
AUDUSD	0.69130435	0.745841	0.717742	0.742991	0.477888
EURCAD	0.81536348	0.846376	0.828393	0.845362	0.632083
EURUSD	0.55029322	0.600846	0.585434	0.598693	0.348462
USDJPY	0.60667591	0.654438	0.619974	0.643946	0.380293



**Table 4. Performance comparison for no density but other extra features**

	<b>XGB</b>				
	<b>Sensitivity</b>	<b>Specificity</b>	<b>Accuracy</b>	<b>Precision</b>	<b>R</b>
AUDUSD	0.6728972	0.7249576	0.700178	0.689655	0.398406
EURCAD	0.7389458	0.812746	0.797838	0.787926	0.523949
EURUSD	0.6047511	0.574944	0.603922	0.556482	0.315367
USDJPY	0.6029374	0.640133	0.637492	0.638325	0.347458
	<b>ANN</b>				
	<b>Sensitivity</b>	<b>Specificity</b>	<b>Accuracy</b>	<b>Precision</b>	<b>R</b>
AUDUSD	0.5691861	0.6434231	0.600604	0.6850945	0.256393
EURCAD	0.6649503	0.749375	0.674835	0.7659401	0.345537
EURUSD	0.5294856	0.585942	0.544947	0.5839463	0.223944
USDJPY	0.5595754	0.628354	0.589407	0.6466588	0.213957
	<b>SVM</b>				
	<b>Sensitivity</b>	<b>Specificity</b>	<b>Accuracy</b>	<b>Precision</b>	<b>R</b>
AUDUSD	0.7852447	0.7805332	0.782696	0.752274	0.600139
EURCAD	0.8328467	0.846473	0.820125	0.817352	0.725844
EURUSD	0.5493644	0.648312	0.54837	0.600247	0.458353
USDJPY	0.6448465	0.696902	0.689432	0.595734	0.523949

**Table 5. R values on validation data**

	<b>XGB</b>	<b>ANN</b>	<b>SVM</b>
Full	0.157639	0.219886	0.055405
No density	0.089221	0.190187	0.019189
No density but other	0.147648	0.173114	-0.00287

**Table 6. Average training times in seconds**

<b>XGB</b>	4.16
<b>ANN</b>	720
<b>SVM</b>	2700

**Table 7. ARMA model performance**

	<b>Test</b>		<b>Validation</b>	
	<b>Accuracy</b>	<b>R</b>	<b>Accuracy</b>	<b>R</b>
AUDUSD	0.2543	0.1093	0.1795	0.0847
EURCAD	0.2884	0.1382	0.2142	0.1038
EURUSD	0.2147	0.0834	0.1483	0.0597
USDJPY	0.2223	0.0933	0.1508	0.0641

where:

$N$  = number of exemplars in the data set

$x$  = network output

$d_{ij}$  = desired output

The size of the mean square error (MSE) can be used to determine how well the model output fits the desired output; however, it does not necessarily reflect whether the two sets of data move in the same direction. For instance, by simply scaling the network output, we can change the MSE without changing the directionality of the data. The correlation coefficient  $R$  solves this problem. By definition, the correlation coefficient between a network output  $x$  and a desired output  $d$  is defined by Formula (6). The correlation coefficient is confined to the range  $[-1 \ 1]$ . When  $R = 1$  there is a perfect positive linear correlation between  $x$  and  $d$ , i.e. they vary accordingly, which means that they vary by the same amount. When  $R = -1$ , there is a perfect linear negative correlation between  $x$  and  $d$ , i.e. they vary in opposite ways (when  $x$  increases,  $d$  decreases by the same amount). When  $R = 0$  there is no correlation between  $x$  and  $d$ , i.e. the variables are called uncorrelated. Intermediate values describe partial correlations.

The performances of all three classifiers are evaluated in terms of sensitivity, specificity, accuracy, and precision. These parameters are statistical measures for classification. Values close or equal to 100% are desirable. They are related with true positive (TP), true negative (TN), false positive (FP) and false negative (FN) values, as explained below:

**TP:** Number of cases belonging to a certain class that are correctly classified.

**TN:** Number of cases not belonging to a certain class that are correctly classified.

**FP:** Number of cases belonging to a certain class that are incorrectly classified.

**FN:** Number of cases not belonging to a certain class that are incorrectly classified.

These parameters are calculated by the following equations:

$$Sensitivity = \frac{TP}{TP + FN} \quad (7)$$

$$Specificity = \frac{TN}{TN + FP} \quad (8)$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (9)$$

$$Precision = \frac{TP}{TP + FP} \quad (10)$$

Another popular method for measuring classification performance is Receiver Operating Characteristic (ROC) curve. In statistics, a ROC curve is a plot of the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. The true-positive rate is also known as sensitivity, recall or probability of detection in machine learning. The false-positive rate is also known as the fall-out or probability of false alarm and can be calculated as  $(1 - specificity)$ . The ROC curve

is thus the sensitivity as a function of fall-out. ROC analysis is used in many areas and is increasingly used in machine learning and data science research nowadays. As a score for model comparison; Area Under Curve (AUC) is used with ROC. AUC values closer to 1 indicate better classification (Hanley & McNeil, 1982). ROC curves and AUC scores are given for the AUDUSD models using full density based visual features in Figures 5, 6, and 7; as well as for the models without density based features in Figures 8, 9, and 10.

## DISCUSSION AND CONCLUSION

As can be seen in Tables 2, 3, 4, and Figures 5-12; there is a clear distinction between performances of density based and other features both R and classification criteria wise. In other words, density based visual features simply outperform the others by boosting the prediction and classification success. The performance of no density but extra features added instead is better than that of no density features. Of course, this is understandable as the former has more features. The best performing method is SVM, and XGB performs better than ANN. The most predictable FOREX pair turns out to be EURCAD, while AUDUSD comes second, and EURUSD is the least predictable. This is also interesting as EURUSD is the most traded FOREX currency pair all over the world. As per training times shown in Table 6.8, the performance of XGB classifier is outstanding, and the worst training time performance belongs to the SVM classifier. The performance of the well-known classical ARMA method is worse than all other methods in both test and validation sets.

XGB method is not a black box unlike ANN and SVM, the results achieved by XGB classifier, can be more easily explained. An analysis on feature importance of the XGB model is illustrated in Figure 13. As the illustration implies, 5 of the most important 8 features out of the total 24 are density variables. This also shows the superiority of the density based visual features.

On the other hand, as seen in Table 5, there is a sharp decrease in the performances of all methods over the validation data. This may be due to some overfitting because of intertwining past and future data belonging to the model building period. It could also stem from the ever-changing behavior of financial markets. However, density based visual features still perform better than the others. As per the performances of different methods, the list is reversed: ANN performs the best, XGB comes second, and SVM third over validation period.

For future work; ways to get better prediction results over out of range data could be investigated. Also, development of a trading strategy depending on the density based visual features could yield interesting results. This work should include profitability along with predictability, using buy, sell,

Figure 5. ROC curve for XGB model with full visual density-based features

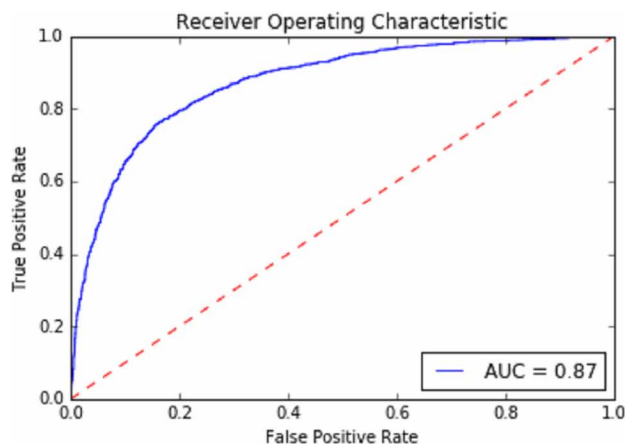


Figure 6. ROC curve for ANN model with full visual density-based features

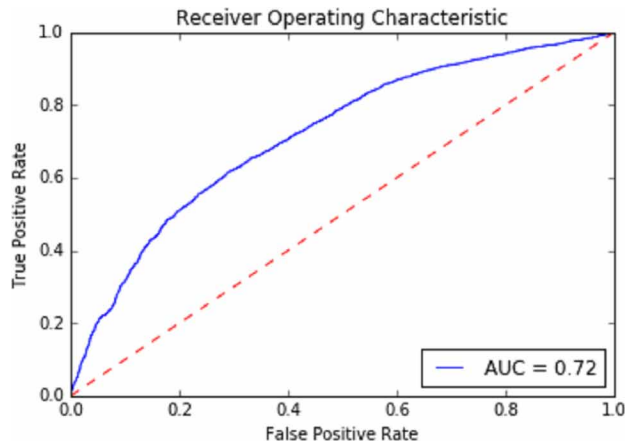


Figure 7. ROC curve for SVM model with full visual density-based features

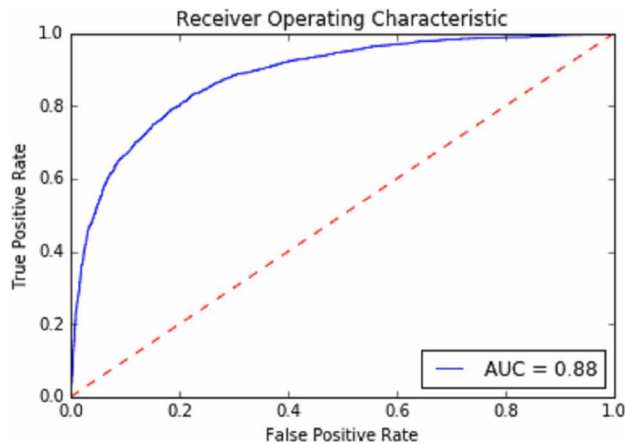


Figure 8. ROC curve for XGB model with no density features

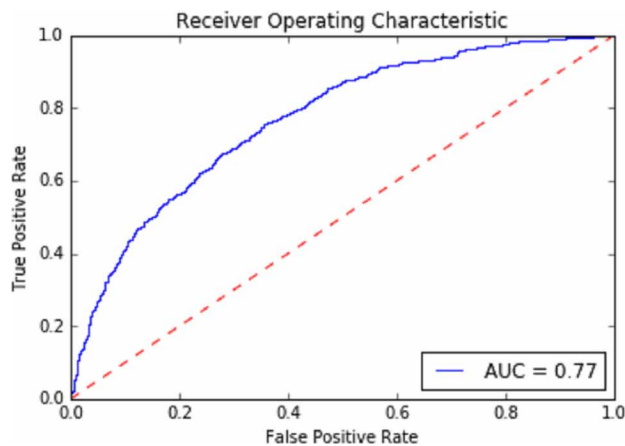


Figure 9. ROC curve for ANN model with no density features

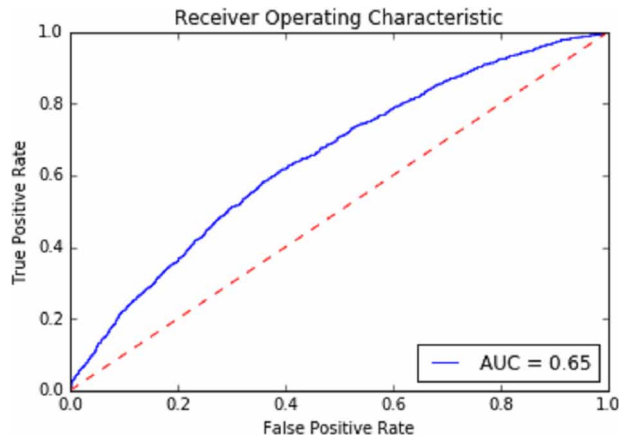


Figure 10. ROC curve for SVM model with no density features

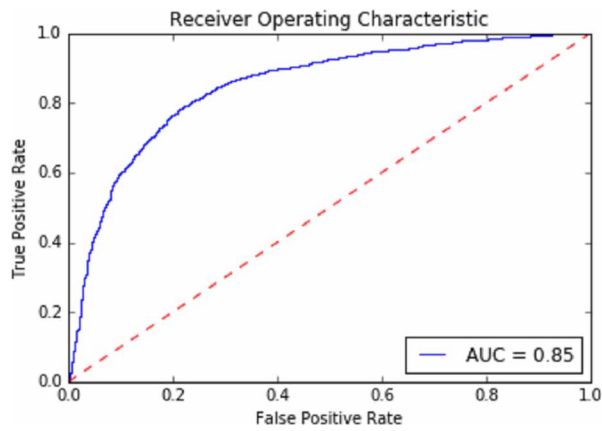


Figure 11. Overall comparison among feature sets

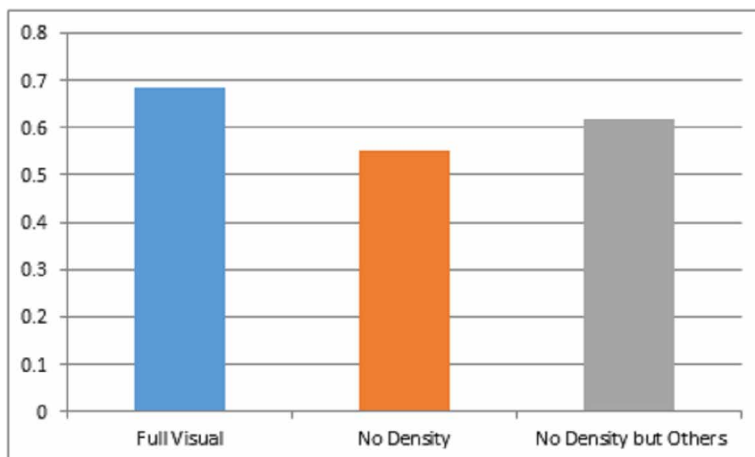


Figure 12. Overall comparison among feature sets on validation data

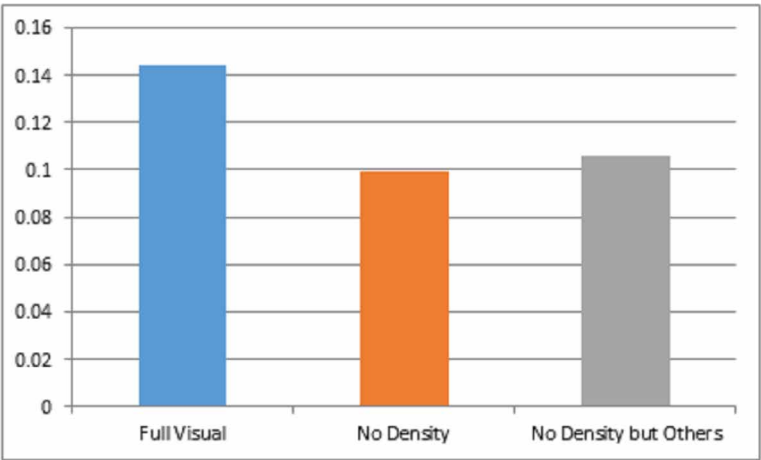
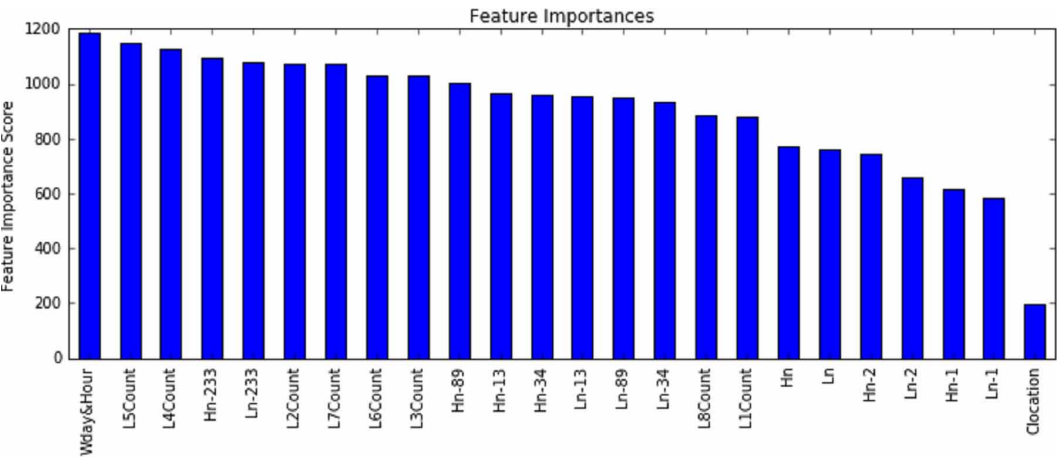


Figure 13. Feature importance of XGB classifier



take profit and stop loss conditions. Of course, good predictions may not necessarily lead to profitable trades, and of course whether returns outweigh costs and risks is a crucial question to be investigated; however, all these are would be in the scope of future work.

Consequently, by looking at prediction and classification performances of density based visual features, it is fair to say that they could open a window to Finance Vision.

## REFERENCES

- Aktepe, A., & Ersoz, S. (2012). A Quantitative Performance Evaluation Model Based on a Job Satisfaction - Performance Matrix and Application in a Manufacturing Company. *International Journal of Industrial Engineering*, 19(6), 264–277.
- Bossaerts, P., & Hillion, P. (1999). Implementing Statistical Criteria to Select Return Forecasting Models: What Do We Learn? *Review of Financial Studies*, 12(2), 405–428. doi:10.1093/rfs/12.2.405
- Campbell, J. Y., & Thompson, S. (2008). Predicting the Equity Premium Out of Sample: Can Anything Beat the Historical Average? *Review of Financial Studies*, 21(4), 1509–1531. doi:10.1093/rfs/hhm055
- Chandler, M. (2013). Turnover averages 5.3 trillion. Economonitor. Retrieved from <http://www.economonitor.com/blog/2013/09/bis-daily-fx-turnover-averages-5-3-trillion/>
- Chen, T., & Guestrin, C. (2016). *Xgboost: A scalable tree boosting system. Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). ACM; . doi:10.1145/2939672.2939785
- Cheng, W., McClain, B. W., & Kelly, C. (1997). Artificial Neural Networks Make Their Mark as a Powerful Tool for Investors. *Review of Business*, 4–9.
- Choudhury, S., Ghosh, S., Bhattacharya, A., Fernandes, K. J., & Tiwari, M. K. (2014). A real time clustering and SVM based price-volatility prediction for optimal trading strategy. *Neurocomputing*, 131, 419–426. doi:10.1016/j.neucom.2013.10.002
- Dash, R., Dash, P. K., & Bisoi, R. (2014). A self-adaptive differential harmony search based optimized extreme learning machine for financial time series prediction. *Swarm and Evolutionary Computation*, 19, 25–42. doi:10.1016/j.swevo.2014.07.003
- Duda, R. O., Hart, P. E., & Stork, D. G. (1973). *Pattern classification* (Vol. 2). New York: Wiley.
- Dutta, S., & Shekhar, S. (1988). Bond-rating: a Non-conservative Application of Neural Networks. *Proceedings of the IEEE International Conference on Neural Networks* (Vol. 2, pp 443–450). IEEE. 10.1109/ICNN.1988.23958
- Fenghua, W., Jihong, X., Zhifang, H., & Xu, G. (2014). Stock Price Prediction based on SSA and SVM. *Procedia Computer Science*, 31, 625–631. doi:10.1016/j.procs.2014.05.309
- Gonzalez, R. C., & Woods, R. E. (2007). *Image processing. Digital image processing*. Gatesmark Publishing.
- Goyal, A., & Welch, I. (2008). A Comprehensive Look at the Empirical Performance of Equity Premium Prediction. *Review of Financial Studies*, 21(4), 1455–1508. doi:10.1093/rfs/hhm014
- Hanley, J. A., & McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143(1), 29–36. doi:10.1148/radiology.143.1.7063747
- Hartzmark, M. L. (1987). Returns to individual traders of futures: Aggregate results. *Journal of Political Economy*, 95(6), 1292–1306. doi:10.1086/261516
- Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer Feedforward Networks are Universal Approximators. *Neural Networks*, 2(5), 359–366. doi:10.1016/0893-6080(89)90020-8
- Karaçor, A. G., Erkan, T. E. (2016). On the Comparison of Quantitative Predictabilities of Different Financial Instruments. *Intelligent Techniques for Data Analysis in Diverse Settings*, 282.
- Karacor, A. G., Sivri, N., & Ucan, O. N. (2007). Maximum Stream Temperature Estimation of Degirmendere River Using Artificial Neural Network. *Journal of Scientific and Industrial Research*, 66(5), 363–366.
- Karacor, G., & Denizhan, Y. (2004). Advantages of hierarchical organisation in neural networks. *International Journal of Computing Anticipatory Systems*, 16, 48–60.
- Kho, B. C. (1996). Time-varying Risk Premia, Volatility, and Technical Trading Rule Profits: Evidence from Foreign Currency Futures Markets. *Journal of Financial Economics*, 41(2), 249–290. doi:10.1016/0304-405X(95)00861-8
- Konstantinidi, E., & Skiadopoulos, G. (2011). Are VIX futures prices predictable? An empirical investigation. *International Journal of Forecasting*, 27(2), 543–560. doi:10.1016/j.ijforecast.2009.11.004

Kumar, D., Meghwani, S. S., & Thakur, M. (2016). Proximal support vector machine based hybrid prediction models for trend forecasting in financial markets. *Journal of Computational Science*, 17(Part 1), 1–13. doi:10.1016/j.jocs.2016.07.006

Lam, M. (2004). Neural Network Techniques for Financial Performance Prediction: Integrating Fundamental and Technical Analysis. *Decision Support Systems*, 37(4), 567–581. doi:10.1016/S0167-9236(03)00088-5

Nayak, R. K., Mishra, D., & Rath, A. K. (2014). A Naïve SVM-KNN based stock market trend reversal analysis for Indian benchmark indices. *Applied Soft Computing*, 35, 670–680.

NeuroSolutions. (n.d.). NeuroDimension Inc. Retrieved from [www.neurosolutions.com](http://www.neurosolutions.com)

Rossum, G. (1995). *Python tutorial*. Centrum voor Wiskunde en Informatica. CWI.

Serrano, J., & Hoesli, M. (2010). Are Securitized Real Estate Returns more Predictable than Stock Returns? *The Journal of Real Estate Finance and Economics*, 41(2), 170–192. doi:10.1007/s11146-008-9162-y

Shamsipoor, H., Sandidzadeh, M. A., & Yaghini, M. (2012). Solving Capacitated P - Median Problem by a New Structure of Neural Network. *International Journal of Industrial Engineering*, 19(8), 305–319.

Strozzi, F., & Zaldivar, J. M. (2005). Non-linear Forecasting in High-frequency Financial Time Series. *Physica A*, 353, 463–479. doi:10.1016/j.physa.2005.01.047

Taylor, S. J. (1992). Rewards Available to Currency Futures Speculators: Compensation for Risk or Evidence of Inefficient Pricing? *The Economic Record*, 68(Supplement), 105–116. doi:10.1111/j.1475-4932.1992.tb02298.x

Wang, C. (2004). Futures Trading Activity and Predictable Foreign Exchange Market Movements. *Journal of Banking & Finance*, 28(5), 1023–1041. doi:10.1016/S0378-4266(03)00047-5

31. Wedding, D. K. II, & Cios, K. J. (1996). Time series forecasting by combining RBF networks, certainty factors, and the Box-Jenkins model. *Neurocomputing*, 10(2), 149–168. doi:10.1016/0925-2312(95)00021-6

32. Xi, L., Muzhou, H., Lee, M. H., Li, J., Wei, D., Hai, H., & Wu, Y. (2014). A new constructive neural network method for noise processing and its application on stock market prediction. *Applied Soft Computing*, 15, 57–66. doi:10.1016/j.asoc.2013.10.013

Yeoum, S. J., & Lee, Y. H. (2013). A study on prediction modeling of Korea military aircraft accident occurrence. *International Journal of Industrial Engineering*, 20(9-10), 562–573.

Yoo, J., & Maddala, G. S. (1991). Risk Premia and Price Volatility in Futures Markets. *Journal of Futures Markets*, 11(2), 165–177. doi:10.1002/fut.3990110204

35. Yu, L., Lai, K. K., & Wang, S. (2008). Multistage RBF neural network ensemble learning for exchange rates forecasting. *Neurocomputing*, 71(16-18), 3295–3302. doi:10.1016/j.neucom.2008.04.029

Zięba, M., Tomczak, S. K., & Tomczak, J. M. (2016). Ensemble boosted trees with synthetic features generation in application to bankruptcy prediction. *Expert Systems with Applications*, 58, 93–101. doi:10.1016/j.eswa.2016.04.001

Zunino, L., Tabak, B. M., Serinaldi, F., Zanin, M., Perez, D. G., & Rosso, O. A. (2010). Commodity Predictability Analysis with a Permutation Information Theory Approach. *Physica A*, 390, 876–890.

*Adil Gursel Karacor graduated from Istanbul Technical University with a Bsc. in Computer and Control Engineering in 1995. He completed Msc. in Systems and Control Engineering at Bogazici University in 2002. He recently completed his PhD. at Atılım University in Modeling and Design of Engineering Systems. He worked as a tutor at Turkish Air Force Academy for 9 years. At present, he works as data scientist at a start-up company in the Netherlands. His research interest is mainly in artificial intelligence, machine learning, and predictive modeling.*

*Turan Erman Erkan graduated from Middle East Technical University with a double major BSc. in Physics and Economics in 1994. He completed his MA in Economics at the Middle East Technical University in 1996; he also completed his MBA at MIT in 2000. He obtained his PhD. in Economics at Gazi University in 2005. He is currently an associate professor and the head of Industrial Engineering Department at Atılım University.*