North American Journal of Economics and Finance 58 (2021) 101540

Contents lists available at ScienceDirect

North American Journal of Economics and Finance

journal homepage: www.elsevier.com/locate/najef

Applications of machine learning for corporate bond yield spread forecasting

Jong-Min Kim [a], Dong H. Kim [b], Hojin Jung [c,*]

[a] Statistics Discipline, Division of Science and Mathematics, University of Minnesota at Morris, Morris, MN 56267, United States of America
[b] The James F. Dicke College of Business Administration, Ohio Northern University, 525 South Main Street, Ada, OH 45810, United States of America
[c] Department of Economics, Jeonbuk National University, Jeonju-si, Jeollabuk-do 54896, Republic of Korea

ARTICLE INFO

JEL classification:
C53
C58
G10

Keywords:
Equity volatility
Forecasting
Machine learning
Yield spread

ABSTRACT

This article considers nine different predictive techniques, including state-of-the-art machine learning methods for forecasting corporate bond yield spreads with other input variables. We examine each method's out-of-sample forecasting performance using two different forecast horizons: (1) the in-sample dataset over 2003–2007 is used for one-year-ahead and two-year-ahead forecasts of non-callable corporate bond yield spreads; and (2) the in-sample dataset over 2003–2008 is considered to forecast the yield spreads in 2009. Evaluations of forecasting accuracy have shown that neural network forecasts are superior to the other methods considered here in both the short and longer horizon. Furthermore, we visualize the determinants of yield spreads and find that a firm's equity volatility is a critical factor in yield spreads.

1. Introduction

Forecasting financial markets is a challenging task, as financial time series exhibit non-stationary characteristics. Obtaining an accurate prediction of financial asset returns and movements is important for portfolio hedging, which helps investors reduce the risk of their financial losses. For this reason, investors and policymakers hope to quantify risk and more accurately estimate future returns. To achieve this goal, machine learning methods have recently gained considerable attention for modeling financial time series because of their potential to reduce prediction errors (see, e.g., Culkin & Das, 2017; De Spiegeleer, Madan, Reyners, & Schoutens, 2018; Ghoddusi, Creamer, & Rafizadeh, 2019; Khandani, Kim, & Lo, 2010).

In addition to machine learning-based models, other models can be used for forecasting. However, such modeling techniques, which have been used in previous studies (Brooks, 1997; Heravi, Osborn, & Birchenhall, 2004; Lekkos, Milas, & Panagiotidis, 2007), rely on linear assumptions. In addition, non-linear regression techniques considered in the literature (Kanas, 2003; Lin & Granger, 1994; McMillan, 2007) require a model pre-specification before parameters are estimated. While machine learning models still require assumptions about input distribution, they are more flexible compared to traditional statistical techniques (see, e.g., Kim & Jung, 2019; Svensén & Bishop, 2007). A fitting model can be varied using training datasets, making machine learning methods more flexible.

The purpose of this study is to apply various machine learning methods to forecast corporate bond yield spreads. In particular, this study utilizes Ridge, multivariate adaptive regression splines (MARS), neural networks, random forest, and support vector machine (SVM) regression. We compare the predictive power of these models with competing forecasting models, such as ordinary least squares (OLS), principal component regression (PCR), partial least squares (PLS), and Gaussian copula marginal regression (GCMR).

* Corresponding author.
E-mail addresses: d-kim@onu.edu (D.H. Kim), hojin.jung@jbnu.ac.kr (H. Jung).

https://doi.org/10.1016/j.najef.2021.101540
Received 3 March 2021; Received in revised form 18 July 2021; Accepted 13 August 2021
Available online 22 August 2021

First, we find that the neural network method provides superior forecasting performance based on our forecasting evaluation criterion. We conduct out-of-sample validation and obtain prediction errors in order to compare the forecast accuracy of our competing models, using a measure of root mean square error (RMSE). The neural network model provides the best fit in our empirical comparisons. Second, our visualization exercise shows a strong positive relationship between a firm's equity volatility and yield spreads during the sample period.

Our first finding is consistent with those of previous studies, demonstrating evidence that the neural network approach is more suitable than traditional statistical approaches for making predictions in finance. On a practical level, our study is useful for corporate bond portfolio optimization because our results provide a more accurate yield spread estimation. As described above, we find that machine learning techniques are more effective than those used in most previous studies. Based on this improvement, our results suggest that a firm's equity volatility is the most critical factor that determines corporate bond yield spreads. This can help bond portfolio managers establish investment strategies to optimize bond portfolios. Our model can suggest better trades and thus help to hedge against the risks to which corporate bonds are exposed. Kim and Stock (2014) find that there is a positive relationship between interest rate volatility and yield spreads and argue that their findings should be used to enhance hedging performance. Similarly, our findings can be applied to hedge a bond portfolio; a bond portfolio manager can hedge more effectively by increasing their relative holdings of treasury bonds to corporate bonds for firms with greater equity volatility.

The remainder of the paper is organized as follows. In the next section, we offer a literature review. Section 3 introduces the basics of various machine learning models and provides a description of the data. Our empirical results are discussed in Section 4, and the last section offers concluding remarks.

2. Literature review

Henrique, Sobreiro, and Kimura (2019) offer a complete review of the most influential articles relevant to machine learning in financial market prediction published in the last two decades. Their literature review clearly shows that various machine learning methods, including artificial neural networks, SVM, and random forest, have been used for financial market prediction and that they outperform traditional linear models. In recent studies, Mishra and Padhy (2019) use the support vector regression (SVR) model to predict stock prices and show that the prices predicted by the method almost match actual prices. Thus, they argue that their proposed method can be applied to efficiently construct a portfolio. In a similar attempt, Ma, Han, and Wang (2021) consider random forest, SVR, and deep learning models for portfolio optimization with predicted stock returns. They show that random forest is superior to the other models as it offers better prediction accuracy. In addition, a hybrid regression model combining selection operator and least absolute shrinkage, learning- and teaching-based optimization, and SVR, is applied to select stocks for the hedged portfolio (Mishra, Padhy, Mishra, & Misra, 2021). Their comparison of portfolio returns reports that the return of the constructed portfolio is higher than the return of the market portfolio in India. Furthermore, machine learning approaches are widely used to forecast credit rating and are useful for evaluating the level of financial risk. In particular, Golbayani, Florescu, and Chatterjee (2020) apply four machine learning models, such as SVM, random forest, bagged decision tree, multiple layer perceptron, and SVM to forecast corporate credit ratings and compare their performance by introducing the notch distance concept. They find that random forest and bagged decision tree outperform SVM and multiple layer perceptron when using three different sectoral stock data: that of the healthcare sector, energy sector, and financial sector. Moscatelli, Parlapiano, Narizzano, and Viggiano (2020) also show that machine learning models generally provide better forecasting performance than standard statistical models in predicting corporate default risk, especially when limited information is available.

Other researchers have applied various machine learning methods to financial data for return forecasting and risk assessment. Patel, Shah, Thakkar, and Kotecha (2015) predict Indian stock index variation using SVM, random forest, Naive-Bayes classifier, and artificial neural networks. A similar attempt is made by Kara, Boyacioglu, and Baykan (2011) to investigate stock price change in the Istanbul Stock Exchange. An artificial neural network approach is used to forecast stock returns for the Taiwanese stock market (Chen, Leung, & Daouk, 2003) and for the Chinese stock market (Cao, Leggio, & Schniederjans, 2005). SVMs are also used to forecast the stock price index in different countries, such as the Korean stock market (Kim, 2003) and the Japanese stock market (Huang, Nakamori, & Wang, 2005). Predicting corporate bankruptcy using machine learning techniques is another area that has received intensive attention in the literature. A number of studies utilize machine learning tools for credit risk estimation and show that such techniques substantially improve prediction accuracy. For example, Min and Lee (2005) examine the performance of a machine learning method by considering SVM applied to bankruptcy prediction and provide strong evidence that the machine learning approach can achieve better performance than classical models like multiple discriminant analysis and logistic regression. In addition, Kruppa, Schwarz, Arminger, and Ziegler (2013) estimate consumer credit risks by using machine learning approaches including random forests, which outperform logistic regression and k-nearest neighbors. Further, SVM and random forest are considered to predict corporate bankruptcy in Barboza, Kimura, and Altman (2017). Consistent with the previous studies, they show that the machine learning methods provide a significant improvement in prediction accuracy, especially when data with outliers are used and multicollinearity is present, which is also observed in our sample. In particular, they report that machine learning models achieve about 10% higher accuracy than competing traditional models, including multivariate discriminant analysis and logistic regression. Pal, Kupka, Aneja, and Militky (2016) introduce a hybrid methodology that combines linear regression and SVM and apply their method to predict business health. Their hybrid method achieves a high prediction accuracy of about 90%. In addition, the techniques are also used in different disciplines such as medicine, chemistry, education, and other fields. For example, machine learning tools have been used to examine clinical and genomic data (Yoo, Ramirez, & Liuzzi, 2014), to predict pneumonia mortality (Cooper et al., 1997), to estimate properties of new molecules (Huang & von Lilienfeld, 2020), to predict student dropout rates in

e-learning courses (Lykourentzou, Giannoukos, Nikolopoulos, Mpardis, & Loumos, 2009), and to analyze student retention (Delen, 2010).

Barboza et al. (2017) show that machine learning models provide a significant improvement in forecast accuracy when comparing discriminant analysis and logistic regression. Kim and Jung (2019) find that machine learning models provide better forecasts of winning bids than the traditional least squares method. In addition, Rasekhschaffe and Jones (2019) show that machine learning algorithms are more effective than linear models as a practical investment tool for portfolio construction. Based on around 150 relevant papers in the finance field, Ryll and Seidens (2019) demonstrate that machine learning applications, in general, outperform stochastic models using meta-analysis.

Among advanced machine learning techniques, according to Li and Ma (2010), the neural network approach has been overwhelmingly considered to be a reliable tool in finance, especially when analyzing the stock market. Le and Viviani (2018) compare forecasting accuracy by using data from 3000 US banks. They consider five different statistical approaches to predict bank failures: artificial neural networks, SVM, k-nearest neighbors, discriminant analysis, and logistic regression. Their empirical results show that the neural network outperforms the two traditional statistical methods and SVM in prediction accuracy. K-nearest neighbors were found to perform as effectively as artificial neural networks in their study. While predicting fraudulent financial reporting in Malaysia, artificial neural networks provide a high prediction accuracy of 94.87% compared to 92.4% using linear regression (Omar, Johari, & Smith, 2017) . Furthermore, Du Jardin (2010) demonstrates that a neural network-based model performs well regarding prediction accuracy, when predicting bankruptcy in the French retail sector. The method achieves a prediction accuracy of 94.03%, while logistic regression and discriminant analysis obtain 90.00% and 84.39%, respectively. More empirical evidence of forecasting performance can be found in Sureshkumar and Elango (2012), which demonstrates that the neural network approach can predict future stock prices. The significant predictive accuracy of the application is also reported in short-term exchange rate prediction (Galeshchuk, 2016). We refer to Paliwal and Kumar (2009) for additional studies that consistently produced similar findings. They provide a comprehensive literature review of comparisons of neural networks and traditional regression approaches.

Another example of neural network applications in finance is forecasting the yield curve or credit spreads in bond markets. Nunes, Gerding, McGroarty, and Niranjan (2018) consider the neural network method, particularly the multilayer perceptron, to predict the European yield curve for government bonds. While the method achieves highly accurate forecasts in the empirical analysis due to its high flexibility, they point out that one particular model cannot perfectly fit all target variables and conditions, so custom-built models are necessarily used to forecast bond yields. A similar study is conducted by Rosadi, Nugraha, and Dewi (2011) to forecast the yield curve dynamics of Indonesian government bonds by using the same machine learning algorithm as in Nunes et al. (2018). Recently, the neural network method has been used for forecasting corporate bond credit spreads (Luo & Liu, 2021). In addition, Luo and Liu (2021) determine the significant macroeconomic factors that affect the underlying systemic risks for credit spreads in the Chinese bond market. These macroeconomic factors include outputs, inflation, stock market volatility and returns, and inter-bank funding.

Kumar, Jain, and Singh (2020) point out that "Stock market forecasting is highly demanding and most challenging task for investors, professional analyst and researchers in the financial market due to highly noisy, nonparametric, volatile, complex, non-linear, dynamic and chaotic nature of stock price time series." Further, Ganguli and Dunnmon (2017) note that bond traders face a more substantial lack of trading information than stock traders. Considering that such a lack of relevant information makes it difficult to predict future bond prices, they claim that machine learning can help researchers predict future bond prices in a fast and accurate manner.

3. Method and data

3.1. Forecasting models

3.1.1. Linear and non-linear regression models

We first consider three linear regression models: OLS, PCR, and PLS. PCR is a standard linear regression model based on principal component analysis (PCA) used to estimate the coefficients of the model. PCA is a dimensionality reduction tool that projects high-dimensional data onto fewer dimensional subspaces, called principal components (PCs), whilst containing most of the information (variance) in the original dataset. This method is particularly useful for datasets with variables that are highly correlated. The 1^{st} PC is the direction of largest variance in the data and the k^{th} PC is the orthogonal direction of largest variance to the previous $k-1$ components. Therefore, the 1^{st} PC explains the largest possible variance, the 2^{nd} PC retains the second largest variance, and so forth. It should also be noted that the eigenvalue is used to measure the amount of variance preserved by each PC. For more details, see Kneip and Utikal (2001) and Kim and Jung (2018).

We also consider GCMR as a non-linear regression model. The GCMR method was used to find the determinants of corporate bond yield spreads in Kim, Kim, and Jung (2020). This regression-based copula is frequently used in the literature for modeling non-normal time series. In particular, we implement this method for predicting bond yield spreads.

A. Principal component regression

PCR is useful for overcoming the multicollinearity problem by using a subset of the high-variance principal components for regression. In a standard linear regression model $Y = X\beta + \varepsilon$, the regression coefficients can be estimated using $\hat{\beta} = (X'X)^{-1}X'Y$. To perform PCR, we transform a full-rank matrix of independent variables, X, to its principal components. Let $R = XU$, where U is a matrix of the normalized eigenvectors of $X'X$ with the property $U'U = UU' = I$. The standard linear regression model can be written as:

$$Y = XUU'\beta + \varepsilon = R_\eta + \epsilon,$$

where $\eta = U'\beta$ and η can be defined by $\hat{\eta} = (R'R)^{-1}R'Y$. Finally, the principal component estimator of β is defined by $\tilde{\beta} = U(R'R)^{-1}R'Y$, in which a different set of independent variables are used. For more details about PCR, see Boneh and Mendieta (1992).

B. Partial least squares

PLS regression creates linear regression by using the orthogonal projection of both the predicted and observed variables onto a problem space. Like PCR, PLS is appropriate for data with strongly correlated predictors. The cross-product matrix in the regression is defined as $K = X'Y$. In order to obtain scores s and υ, weight vectors, r and q, are used as follows:

$$s = Xr = Gr \quad \text{and} \quad \upsilon = Yq = Jq,$$

where G and J are initialized and X and Y loadings are gained by regressing on the score s such as $w = G's$ and $q = J's$. Matrices R, S, W, and Q can be attained from the vectors r, s, w, and q after every iteration. By using scores $S = XD$, the regression coefficients can be obtained by

$$\beta = D(S'S)^{-1}S'Y = DS'Y = DQ',$$

where the optimal number of principal components are determined by cross-validation criteria. For more details about PLS regression, see Mevik, Wehrens, and San Michele all'Adige (2015).

C. Gaussian copula marginal regression

We also consider several non-linear regression models, including five different machine learning methods. Our analysis utilizes GCMR with a marginal cumulative distribution for $x_i, G(\cdot|x_i)$. According to Sklar's theorem (see Sklar, 1959), any multivariate distribution function, $G_{XY}(x, y)$, can be expressed by $G_X(x)$ and $G_Y(y)$ by using a copula C, which is determined in $[0, 1]^n$. The joint cumulative distribution function in GCMR is given by

$$\text{Prob}(Y_1 \le y_1, \ldots, Y_n \le y_n) = \Omega_n\Big(\epsilon_1, \ldots, \epsilon_n; \mathbf{W}\Big),$$

where $\epsilon_i = \Omega^{-1}\Big\{G(y_i|x_i)\Big\}$. In particular, the Gaussian copula model is defined by (Masarotto & Varin, 2012) as:

$$Y_i = h(\mathbf{x}_i, \epsilon_i),$$

where ϵ_i is stochastic and follows a normal distribution with the correlation matrix of the Gaussian copula. For more details about GCMR, see Kim and Jung (2018) and Kim et al. (2020).

3.1.2. Machine learning models

A. Ridge

We consider a regularized linear regression method, Ridge, in which the underlying normality assumption is not required. Similar to OLS, the estimate of γ is obtained by solving the optimization problem

$$\underset{\gamma \in R^n}{\text{argmin}}\,(Y - X\gamma)'(Y - X\gamma) \quad \text{subject to} \quad \sum_{j=1}^{q}\gamma_j^2 \le t.$$

Then, the penalized residual sum of squares (T) is given by

$$\text{T}(\gamma) = (Y - X\gamma)'(Y - X\gamma) + \psi\|\gamma\|_2^2,$$

where ψ is the shrinkage parameter. Finally, we can obtain the optimization solution by taking derivatives:

$$\frac{\partial \text{T}(\gamma)}{\partial \gamma} = -2X'(Y - X\gamma) + 2\psi\gamma$$
$$\hat{\gamma} = \left(X'X + \psi I_q\right)^{-1}X'Y.$$

See Kim and Jung (2019) for a further discussion and other applications of the Ridge regression.

B. Multivariate adaptive regression splines

MARS, proposed by Friedman (1991), is a non-parametric statistical method that models multiple nonlinearities in data. A main advantage of MARS is its flexibility, as it can handle both linear and nonlinear relationships. The model can be constructed by product spline basis functions, where data determine product degrees and knot locations. A spline function is a piecewise polynomial function that connects smoothly at joint points called knots. For a given set of knots, a spline function can be written as

$$f(x) = \beta_0 + \beta_1 x + \cdots + \beta_p x^p + \sum_{k=1}^{K} \beta_{p+k}(x - t_k)_+^p$$

where p indicates the order of the splines. $t_1, \ldots, t_K$ show the knots and $(x)_+ = x$ if $x > 0$ and 0 otherwise.

Let us consider a random sample $(X_1, Y_1), \ldots, (X_n, Y_n)$. Then, MARS takes models of the form

$$f(X) = f(X|\beta) = \sum_{j=1}^{J} \beta_j B_j(X).$$

The optimal MARS model is chosen by a two-step procedure. In the first step, MARS constructs a number of basis functions that overfit the data. Second, a worse-fitting basis function is continuously removed by using the generalized cross-validation criterion. For a given set of the selected basis functions, MARS estimates unknown parameters via least squares.

C. Support vector machine

Developed by Vapnik (1998) and his colleagues, the SVM is another machine learning algorithm. It is primarily used for classification and can also be used as a regression method, called SVM regression. To construct an optimal hyperplane that performs classification tasks, the SVM adopts an iterative tool in order to minimize the following error function:

$$\frac{1}{2} r'r + C \sum_{i=1}^{K} \zeta_i,$$

subject to the following constraints:

$$y_i(r'\varphi(x_i) + b) \geq 1 - \zeta_i, \quad i = 1, 2, \ldots, K \quad \text{and} \quad y \in \{-1, +1\},$$

where i indicates K training sets and b is a constant. C, r, and ζ represent the capacity constant, a coefficient vector, and parameters for nonseparable data, respectively. The C parameter controls the error penalty, so a larger C heavily penalizes the error.

The SVM is primarily used for classification and is extended to regression analysis. When applying the SVM algorithm for classification to a regression method, the support vector regression separates hyperplanes, maximizing the margin to minimize the generalization error. The optimization process for the regression is based on a decomposition method introduced by Hsu and Lin (2002). The SVM regression takes a deterministic functional form and additive noise:

$$y = g(x) + noise \quad \text{and} \quad g(x) = \langle r, x \rangle + b,$$

where r is a coefficient vector. An optimal functional form of g can be obtained by training SVM on a training set. There are various kernel functions, including a radial basis function, which we use in our analysis.

$$H(X_i, X_j) = \exp(-\gamma |x_i - x_j|^2),$$

where $H(X_i, X_j) = \varphi(X_i) \cdot \varphi(X_j)$. Depending on the selection of adjustable parameters of γ and C, an SVM with the radial basis function kernel can provide different accuracy levels.

D. Random forest

The random forest method introduced by Breiman (2001) is an ensemble classification algorithm in which a tree is a base classifier. Like the SVM approach, the random forest does not require the assumption of a specific underlying distribution. A random forest consists of a series of individual decision trees. A single tree is constructed by the large amounts of training data and a random vector (θ_l) and a classifier consists of an input vector and θ_l. We perform bootstrap-aggregating in Breiman's random forest algorithm by using the **randomForest** R package, and the *mtry* function is utilized to randomly select the number of predictors at each decision split. By producing a collection of individual decision trees on bootstrapped training datasets, the method can provide powerful prediction performance.

Given an ensemble of classifiers, the training data are randomly drawn from the joint distribution of X and Y. We then define the margin function for an individual tree as

$$N_f(\mathbf{X}, Y) = av_s \mathbf{I}(l_s(\mathbf{X}) = Y) \quad max \quad av_s \mathbf{I}(l_s(\mathbf{X}) = m),$$

where $\mathbf{I}(\cdot)$ represents the indicator function. Further, the average prediction error for the random forest model is given by $E^* = P_{XY}(N_f(\mathbf{X}, Y) < 0)$. Forecasting is obtained by averaging the individual tree predictions.

E. Neural network

The neural network is a data-driven, self-adaptive method. According to the universal approximation theorem, neural networks with one hidden layer can approximate any continuous multivariate function to arbitrary accuracy (see Hornik, Stinchcombe, & White, 1989; Irie & Miyake, 1988). In forecasting, the functional relationship between X_i and y can be expressed as $y = f(x_1, x_2, \dots, x_p)$, which is functionally equal to a standard nonlinear regression model. The basic unit of a neuron is logistic in artificial neural networks.

Multilayer feedforward networks are frequently used in finance applications (Qi, 1996). Particularly, in three-layer feedforward neural networks, individual units generally consist of the input layer, middle layer, and output layer, and a transfer function delivers input signals to each network. Suppose that X, K, and y are the input layer $(x_1, x_2, \dots, x_p)$, the middle layer $(k_1, k_2, \dots, k_q)$, and the output layer, respectively. Middle layers receive input signals with a bias term and produce signals for the output layer:

$$k_l = G\left(\sum \beta_{il} x_i\right) = G\left(X'\beta_l\right) \quad \text{for} \quad l = 1, 2, \dots, q \quad \text{and} \quad i = 0, 1, 2, \dots, p,$$

where x_i and β represent the signal by ith input and a weight matrix. In a similar manner, the middle layer units send output signals to the output unit layer that produces a signal:

$$y = F\left(\sum \gamma_l k_l\right) \quad \text{for} \quad l = 1, 2, \dots, q,$$

where γ is a weight vector. Thus, γ_l indicates the weight from the lth middle layer unit to y. Using the two equations above, we can obtain

$$y = F\left(\gamma_0 + \sum_{l=1}^{q} \gamma_l G\left(\sum \beta_{il} x_i\right)\right) = g(X, \xi),$$

where ξ is a vector of network weights. The functional forms of F and G can be sigmoid or logistic. See Qi (1996) for a further discussion of the neural network structure.

3.2. Data

Following Kim and Stock (2014), we use the Trade Reporting and Compliance Engine (TRACE) to collect transaction data between 2003 and 2009. Following Edwards, Harris, and Piwowar (2007), we exclude corrected, repeated, and canceled interdealer transactions from the sample. Because a volume-weighted approach is preferably to an equal-weighted one regarding specification and power (Bessembinder, Kahle, Maxwell, & Xu, 2009), we use the volume-weighted approach to compute yields. More specifically, the volume-weighted average of all transaction prices is calculated from the last trading day of the month. When we compute yields, we base them on the last trading day of the month. Some bonds are very illiquid and thus may not trade for the last five to ten business days of the month. It is then possible for one bond's last transaction to occur during the last week of the month while the other bond's last transaction occurs during the second week of the month. For the sake of consistency, we eliminate bonds that did not trade for the last five business days. Kim and Stock (2014) used the same criteria and eliminated those bonds.

The treasury constant maturity yields, measured as the difference between the yield of the corporate bond and its treasury constant maturity counterpart, are obtained from the H.15 release of the Federal Reserve System. As in Collin-Dufresne and Goldstein (2001) and Duffee (1998), a simple linear interpolation method is employed to estimate the entire treasury yield curve. Bond characteristics, including credit quality, maturity, and coupon rate, are obtained from the Fixed Income Securities Database (FISD). Elton, Gruber, Agrawal, and Mann (2001) note that bonds with special features are priced differently. Therefore, bonds that have special features, including putability, convertibility, and sinking fund or make-whole provisions, are excluded from our sample. We also exclude bonds for which the coupon payment is not fixed or its frequency is odd. Following Duffee (1999) and Eom, Helwege, and Huang (2004), we restrict our sample to bonds with a maturity of more than one year.

We consider the explanatory variables that the previous studies reported to affect corporate bond yield spreads. We first include the credit rating, which has been the most widely used variable to measure a bond's default risk. We assign each bond a cardinalized S&P rating with the strongest quality bond (AAA) equal to one and the weakest quality bond (D) equal to twenty-two. Given that a bond's rating mainly captures its credit quality (default risk), we expect bonds with higher rating values (weaker credit quality) to have higher yield spreads. As in Kim and Stock (2014), we measure interest rate volatility by calculating, during the period of the 12 months prior to the bond transaction date, the standard deviation of the daily one-month treasury constant maturity. Because interest rate volatility is positively correlated with the firm's asset volatility (Kim & Stock, 2014), we expect it to increase corporate bond yield spreads. As in Campbell and Taksler (2003), equity volatility is measured by calculating the standard deviation of the CRSP value-weighted index-adjusted daily excess returns. This is measured over the 12-month period preceding the bond transaction date. The effect of equity volatility on yield spreads should be similar to that of interest rate volatility on yield spreads. This is because both equity volatility and interest rate volatility increase a firm's asset volatility, thereby leading to a higher yield spread (Campbell & Taksler, 2003). Consequently, we expect equity volatility to be positively associated with yield spreads.

We define short-term interest rates as the one-month treasury constant maturity rate. Duffee (1998) and Longstaff and Schwartz (1995) find that an increase in the short-term interest rate increases the drift of firm value under the risk-neutral measure, thereby decreasing the default risk. Therefore, we expect to find a negative relationship between the level of the short-term interest rate and yield spreads. We compute the slope of the yield curve by subtracting the one-year treasury constant maturity rate from the 10-year rate. The theoretical and empirical literature highlight the role of the slope of the yield curve as a determinant of yield

Table 1
Estimation results.

	Case 1		Case 2	
	2008	2009	2009	
	OLS	GCMR	OLS	GCMR
Liquidity	−0.140***	−0.313***	−0.187***	−0.317***
	(0.013)	(0.007)	(0.022)	(0.007)
Equity volatility	60.563***	43.671***	134.042***	34.303***
	(1.119)	(0.693)	(0.916)	(0.359)
Coupon rate	−0.019***	0.022***	−0.031***	0.020***
	(0.003)	(0.002)	(0.005)	(0.002)
Slope	−0.015	0.002	−0.912***	−0.234***
	(0.020)	(0.011)	(0.027)	(0.009)
r	−0.029*	0.003	−0.793***	−0.218***
	(0.013)	(0.007)	(0.018)	(0.006)
Interest rate volatility	0.926***	0.742***	0.952***	0.842***
	(0.029)	(0.016)	(0.028)	(0.010)
Maturity	0.032***	0.020***	0.019***	0.011***
	(0.001)	(0.001)	(0.001)	(0.000)
Rating	0.190***	0.090***	0.225***	0.093***
	(0.002)	(0.001)	(0.002)	(0.001)
Constant	−0.916***	−1.31***	1.546***	−0.232***
	(0.067)	(0.039)	(0.092)	(0.029)

Standard errors are in parentheses. *** $p < 0.01$, ** $p < 0.05$, * $p < 0.1$.

spreads (Breeden, 2011; Ederington & Stock, 2002; Estrella & Hardouvelis, 1991; Estrella & Mishkin, 1996). The slope of the yield curve reflects the expected strength of the future economy, so it increases when the economy is expected to grow strongly, thereby lowering the default risk. Thus, we expect to find a negative relationship between the slope of the yield curve and yield spreads.

The measure of maturity is the number of years left until the bond's expiration date. The relationship between maturity and yield spreads is determined by the relative size of the slopes of the corporate bond yield curve and the government yield curve. A positive coefficient for maturity would indicate that the corporate bond yield curve has a steeper slope than its government counterpart. The coupon rate is measured in percentages. As in Elton et al. (2001) and Longstaff, Mithal, and Neis (2005), we include coupon rates to allow for tax effects. Whereas interest payments on corporate bonds are taxable at the state level, those on treasury bonds are tax-exempt. Since investors demand a higher return for corporate bonds with higher coupon rates, coupon rates are expected to be positively correlated with yield spreads. Following Kim and Stock (2014), we measure liquidity by dividing the number of days a bond was traded during the past 12 months by the number of business days during the same 12 months. Since investors demand a liquidity premium, we expect yield spreads to be negatively correlated with liquidity.

4. Empirical analysis

To forecast non-callable corporate bond yield spreads, we use the various forecasting techniques described in the previous section. We consider two different forecast horizons to ensure our findings are consistent with different horizons. In particular, our empirical analysis considers an out-of-sample forecast for the next two years of monthly corporate bond yield spreads. For example, let Y_t denote the observations at year t and F_t denote the forecasts of Y_t. We compute out-of-sample forecasts F_{2007+j} for $j = 1$ and 2 based on data from times $t = 2003, \ldots, 2007$. That is, monthly corporate bond yield spreads of 2008 (one-year-ahead) and 2009 (two-year-ahead) are forecasted from our in-sample data. Our goal is to apply various machine learning methods to forecast corporate bond yield spreads and suggest a method that provides practitioners with superior forecasting performance.

Each model is estimated using two cases: 1) the out-of-sample forecast horizon covers the period from 2008 to 2009 with in-sample data (2003 to 2007), which we call Case 1 in this article; 2) the in-sample dataset over 2003–2008 is used to forecast the non-callable corporate bond yield spreads in 2009 (Case 2). Table 1 presents the estimation results using OLS and GCMR. GCMR is a better-fitting model because its estimation results are more consistent with the corporate finance literature. For example, as suggested by Elton et al. (2001) and Longstaff et al. (2005), there should be a positive association between coupon rates and yield spreads because of the tax disadvantage of interest payments on corporate bonds. All other coefficient estimates are consistent with our prior expectations.

$$\text{RMSE} \equiv \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}$$

Then, we also conduct out-of-sample testing to measure predictive accuracy. In particular, we employ RMSE as a goodness-of-fit measure to evaluate forecast accuracy.[1] The *pcr* function in the **pls** R package is used to implement PCR analysis. By default, we

[1] Note that the number of observations for corporate bond yield spreads the values of which are less than 1 is 61,9370. This accounts for around 46% of the total observations in our sample dataset. Hyndman and Koehler (2006) warn against the use of MAPE (mean absolute percentage error) when the actual values are close to zero. Therefore, we use RMSE, instead of MAPE, to measure forecast accuracy in our empirical analysis.

Table 2
Forecasting the non-callable corporate bond yield spreads.

Model	RMSE		
	Case 1		Case 2
	2008	2009	2009
OLS	4.121	4.755	4.381
PCR	4.968	6.212	5.031
PLS	4.818	6.150	5.018
GCMR	5.149	5.899	5.957
Ridge	4.121	4.756	4.384
MARS	8.144	77.818	16.412
SVM	4.618	6.074	4.166
Random forest	4.371	5.396	3.650
Neural network	**3.841**	**4.523**	**3.485**

use 10 random segments for cross-validation, which determines 6 components as an optimal value in our empirical application. Similarly, we employ the *plsr* function in the same R package for PLS. In order to fit the PLS regression model with the kernel algorithm, we consider 6 components and 10 segments that are used to perform cross-validation. Further, the *gacmr* function in the **gcmr** R package is used to fit the GCMR model with the Weibull marginals. A likelihood approach is also employed to perform inference in the analysis. When implementing the Ridge regression, we use the *glmnet* function in the **glmnet** R package and the tuning parameter (ψ) value of 0.001. The regression performs the L^2 norm regularization algorithm to avoid over-fitting by penalizing large coefficients. The response type is set to be Gaussian in our Ridge analysis. In addition, the *earth* function in the **earth** R package is considered for the MARS model. Specifically, we use a 10-fold argument for cross-validation with a tuning parameter value of 2 involved in the maximum degree of interactions.

We utilize the R package **kernlab** to implement SVM. The package allows for cross-validation and grid search in the selection of hyper-parameters. By default, the package generates a Gaussian radial basis function kernel, which is frequently used in kernelized learning algorithms. Our analysis is performed with a value of 5 for the cost hyper-parameter using a 10-fold cross-validation. These hyper-parameter values are empirically determined. Furthermore, we fit the random forest model to our dataset by using the *randomForest()* in the **randomForest** R package, which implements Breiman's random forest algorithm. The number of variables and the growing number of trees that are randomly chosen at each split are 4 and 2000, respectively. In order to apply the neural network model to our dataset, we employ the *nnet* function in the **nnet** R package for a single layer feed-forward network. In our analysis, the number of neurons in the hidden layer for weight decay is set to 3 and 0.04, respectively.

As mentioned in the previous section, neural networks can approximate any functional form that characterizes a sample dataset, which is more beneficial when employing complex functional forms. We consider two different out-of-sample forecasts. It is worth noting that the forecasting ability of various models, including neural networks, is evaluated based on the RMSE criterion. The neural network model consistently provides the lowest RMSE values marked in bold across different forecast horizons, as shown in the last row of Table 2. Thus, the neural network model provides superior forecasts for financial time-series over our sample period.[2]

From a practical perspective, our results add additional evidence from the literature show that machine learning methods outperform traditional approaches that are widely used to predict future corporate bond yield spreads. Our findings should encourage practitioners to consider machine learning methods as reliable tools to implement quantitative analysis, especially when using non-normal yield spreads.

Fig. 1 exhibits a neural network fit in our analysis. Line widths in the figure represent the weights that are similar to coefficients, indicating a relationship between variables in a typical regression model. That is, it identifies the strength of the association of a particular input variable to a response variable. A thicker line indicates the larger weight of a specific explanatory variable. Note that a black (gray) line indicates a positive (negative) association between variables.

In Fig. 2, I, H, and O refer to input, hidden, and output, respectively. Bias ($\boldsymbol{B}$) represents an additional input included with the pre-output layer that has a unitary value. The bias unit is not tied to the previous layer, but it has outgoing connections. By shifting the activation function, a bias value allows us to obtain a better fitting to the data. This figure presents the importance of each independent variable in the neural network forecasting and emphasizes that a firm's equity volatility is the most important factor in the prediction of corporate bond yield spreads, as indicated by the largest absolute value in the figure.

Our visualization exercise provides evidence of a statistically positive relationship between equity volatility and yield spreads, which is consistent with the findings of Kim et al. (2020). In the study, they apply various copula functions to measure the dependence structure between yield spreads and our independent variables with the same dataset. The classic model of Merton (1974) attributes a firm's default risk to its asset volatility. As a firm's asset volatility results largely from its equity volatility, it

[2] As an anonymous reviewer suggested, we also consider a traditional time series approach, ARIMA(p, d, q) which is a generalized ARMA(p, q) model. The parameters of p, d, and q indicate p order autoregressive terms, d numbers of differencing to obtain stationarity, and q order moving-average terms, respectively. The values of p, d, and q are chosen based on model specifications, and we consider ARIMA(1,1,2) for Case 1 and ARIMA(4,1,1) for Case 2 in our analysis. The RMSE statistics for one-year-ahead and two-year-ahead forecasts in Case 1 are 4.87 and 6.11 while the statistics for Case 2 are 5.67. The neural network model still performs significantly better than the traditional time series model during the sample period.

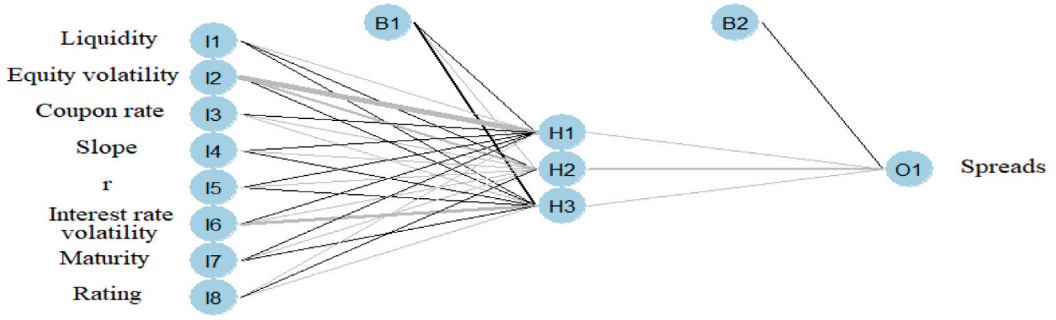

Fig. 1. Neural network for 2008 and 2009 forecasts. Note: This figure describes a single-hidden-layer neural network with 3 units in the hidden layer and a 0.04 parameter for weight decay.

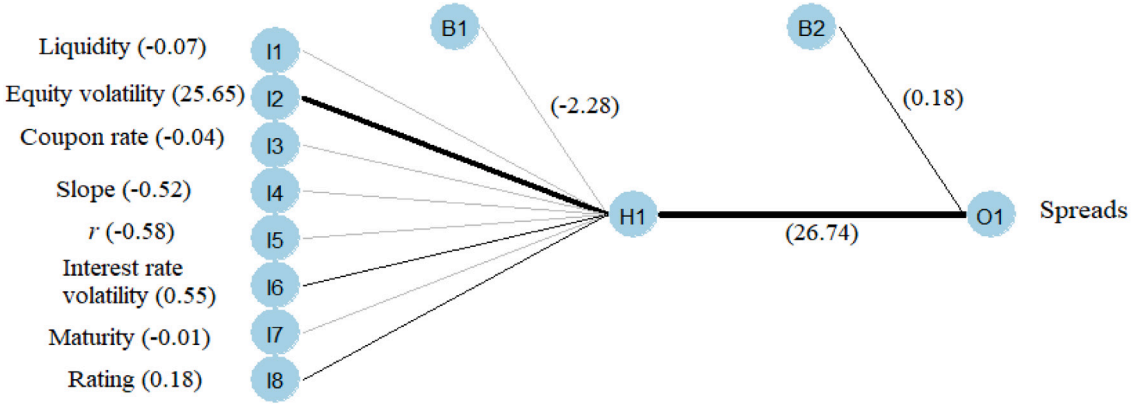

Fig. 2. Neural network for 2009 forecasts. Note: This figure describes a single-hidden-layer neural network with 1 unit in the hidden layer, a 12.0 parameter for weight decay, and a maximum iteration of 100.

makes sense that equity volatility is the most powerful determinant of yield spreads. Our result is also consistent with the empirical findings of Campbell and Taksler (2003), who show that the adjusted R^2 significantly increases with the equity volatility included in their regression models. Their findings suggest that the explanatory role of equity volatility in determining yield spreads is at least as significant as credit ratings.

As the recent COVID-19 pandemic has increased equity volatility, our findings have practical implications. Our results suggest that a firm's equity volatility is the most critical factor that determines corporate bond yield spreads. Therefore, during periods in which equity volatility increases, bond portfolio managers should hold more treasury bonds relative to corporate bonds, a flight-to-quality.

5. Conclusion

This study compares nine different predictive techniques, including five machine learning tools, for the purpose of yield spread forecasting. Forecast errors are generated and then compared across models based on RMSE. We find empirical evidence that the neural network approach clearly dominates competing models for forecasting purposes. As a preliminary analysis, we observe a positive association between a firm's equity volatility and yield spreads through the visualization of our sample data.

We then investigate the determinants of yield spreads provided by the neural network. Credit ratings have been shown to have the strongest effect on corporate bond yield spreads. However, Campbell and Taksler (2003) found that equity volatility was an equally important determinant of yield spreads. Kim and Stock (2014) also found that equity volatility and interest rate volatility played a key role in determining yield spreads. However, by relying on a more statistically sophisticated analysis, our study suggests that equity volatility has the strongest effect on yield spreads. In particular, the weight of a firm's equity volatility in the input vector turns out to be 23.65, while the second-most significant determinant's weight is just 0.58 in absolute value. From a practical point of view, our study helps achieve optimal portfolio management by offering more reliable corporate bond yield spread estimates. In addition, our analysis using a machine learning approach could allow an effective hedge for the default risk of corporate bonds in practice.

In this study, we apply machine learning methods to forecast corporate bond yield spreads. The accuracy of yield spread forecasting using machine learning could enable bond portfolio managers to enhance the quality of their portfolio strategies and the

performance of their portfolios. Our results suggest that equity volatility plays the most critical role in determining yield spreads. Corporate bond managers can use the greatest equity volatility effect to develop their hedging strategies and enhance their hedging performance. One limitation of our study is that we have not empirically tested our model's hedging implications; we leave these tests to future research.

CRediT authorship contribution statement

Jong-Min Kim: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation. **Dong H. Kim:** Data curation, Writing – original draft, Writing – review & editing. **Hojin Jung:** Investigation, Writing – original draft, Writing – review & editing, Supervision, Project administration.

References

Barboza, F., Kimura, H., & Altman, E. (2017). Machine learning models and bankruptcy prediction. *Expert Systems with Applications*, *83*, 405–417.

Bessembinder, H., Kahle, K. M., Maxwell, W. F., & Xu, D. (2009). Measuring abnormal bond performance. *Review of Financial Studies*, *22*(10), 4219–4258.

Boneh, S., & Mendieta, G. R. (1992). Regression Modeling Using Principal Components. In *4th Annual Conference on Applied Statistics in Agriculture Manhattan, KS, USA*.

Breeden, D. (2011). A Stocks, Bonds and Consumers Leading Index (SBCLI): Consumer Behavior as a Leading Indicator. In *World Finance Conference* Rhodes, Greece:.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32.

Brooks, C. (1997). Linear and non-linear (non-) forecastability of high-frequency exchange rates. *Journal of Forecasting*, *16*(2), 125–145.

Campbell, J. Y., & Taksler, G. B. (2003). Equity volatility and corporate bond yields. *The Journal of Finance*, *58*(6), 2321–2350.

Cao, Q., Leggio, K. B., & Schniederjans, M. J. (2005). A comparison between fama and french's model and artificial neural networks in predicting the Chinese stock market. *Computers & Operations Research*, *32*(10), 2499–2512.

Chen, A.-S., Leung, M. T., & Daouk, H. (2003). Application of neural networks to an emerging financial market: Forecasting and trading the Taiwan stock index. *Computers & Operations Research*, *30*(6), 901–923.

Collin-Dufresne, P., & Goldstein, R. S. (2001). Do credit spreads reflect stationary leverage ratios? *The Journal of Finance*, *56*(5), 1929–1957.

Cooper, G. F., Aliferis, C. F., Ambrosino, R., Aronis, J., Buchanan, B. G., Caruana, R., et al. (1997). An evaluation of machine-learning methods for predicting pneumonia mortality. *Artificial Intelligence in Medicine*, *9*(2), 107–138.

Culkin, R., & Das, S. R. (2017). Machine learning in finance: The case of deep learning for option pricing. *Journal of Investment Management*, *15*(4), 92–100.

De Spiegeleer, J., Madan, D. B., Reyners, S., & Schoutens, W. (2018). Machine learning for quantitative finance: Fast derivative pricing, hedging and fitting. *Quantitative Finance*, *18*(10), 1635–1643.

Delen, D. (2010). A comparative analysis of machine learning techniques for student retention management. *Decision Support Systems*, *49*(4), 498–506.

Du Jardin, P. (2010). Predicting bankruptcy using neural networks and other classification methods: The influence of variable selection techniques on model accuracy. *Neurocomputing*, *73*(10–12), 2047–2060.

Duffee, G. R. (1998). The relation between treasury yields and corporate bond yield spreads. *The Journal of Finance*, *53*(6), 2225–2241.

Duffee, G. R. (1999). Estimating the price of default risk. *Review of Financial Studies*, *12*(1), 197–226.

Ederington, L. H., & Stock, D. R. (2002). Impact of call features on corporate bond yields. *The Journal of Fixed Income*, *12*(2), 58–68.

Edwards, A. K., Harris, L. E., & Piwowar, M. S. (2007). Corporate bond market transaction costs and transparency. *The Journal of Finance*, *62*(3), 1421–1451.

Elton, E. J., Gruber, M. J., Agrawal, D., & Mann, C. (2001). Explaining the rate spread on corporate bonds. *The Journal of Finance*, *56*(1), 247–277.

Eom, Y. H., Helwege, J., & Huang, J.-z. (2004). Structural models of corporate bond pricing: An empirical analysis. *Review of Financial Studies*, *17*(2), 499–544.

Estrella, A., & Hardouvelis, G. A. (1991). The term structure as a predictor of real economic activity. *The Journal of Finance*, *46*(2), 555–576.

Estrella, A., & Mishkin, F. S. (1996). *The Yield Curve as a Predictor of US Recessions. Current Issues in Economics and Finance, Volume 2, No.7*. New York: Federal Reserve Bank.

Friedman, J. H. (1991). Multivariate adaptive regression splines. *The Annals of Statistics*, 1–67.

Galeshchuk, S. (2016). Neural networks performance in exchange rate prediction. *Neurocomputing*, *172*, 446–452.

Ganguli, S., & Dunnmon, J. (2017). Machine learning for better models for predicting bond prices. arXiv preprint arXiv:1705.01142.

Ghoddusi, H., Creamer, G. G., & Rafizadeh, N. (2019). Machine learning in energy economics and finance: A review. *Energy Economics*, *81*, 709–727.

Golbayani, P., Florescu, I., & Chatterjee, R. (2020). A comparative study of forecasting corporate credit ratings using neural networks, support vector machines, and decision trees. *The North American Journal of Economics and Finance*, *54*, Article 101251.

Henrique, B. M., Sobreiro, V. A., & Kimura, H. (2019). Literature review: Machine learning techniques applied to financial market prediction. *Expert Systems with Applications*, *124*, 226–251.

Heravi, S., Osborn, D. R., & Birchenhall, C. (2004). Linear versus neural network forecasts for European industrial production series. *International Journal of Forecasting*, *20*(3), 435–446.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359–366.

Hsu, C.-W., & Lin, C.-J. (2002). A simple decomposition method for support vector machines. *Machine Learning*, *46*(1–3), 291–314.

Huang, B., & von Lilienfeld, O. A. (2020). Effcient, accurate, scalable, and transferable quantum machine learning with am-ons. arXiv preprint arXiv:1707.04146.

Huang, W., Nakamori, Y., & Wang, S.-Y. (2005). Forecasting stock market movement direction with support vector machine. *Computers & Operations Research*, *32*(10), 2513–2522.

Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, *22*(4), 679–688.

Irie, B., & Miyake, S. (1988). Capabilities of three-layered perceptrons. *1*, In *IEEE International Conference on Neural Networks* (641–648), (p. 218).

Kanas, A. (2003). Non-linear forecasts of stock returns. *Journal of Forecasting*, *22*(4), 299–315.

Kara, Y., Boyacioglu, M. A., & Baykan, O. K. (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul stock exchange. *Expert Systems with Applications*, *38*(5), 5311–5319.

Khandani, A. E., Kim, A. J., & Lo, A. W. (2010). Consumer credit-risk models via machine-learning algorithms. *Journal of Banking & Finance*, *34*(11), 2767–2787.

Kim, K.-J. (2003). Financial time series forecasting using support vector machines. *Neurocomputing*, *55*(1–2), 307–319.

Kim, J.-M., & Jung, H. (2018). Relationship between oil price and exchange rate by fda and copula. *Applied Economics*, *50*(22), 2486–2499.

Kim, J.-M., & Jung, H. (2019). Predicting bid prices by using machine learning methods. *Applied Economics*, *51*(19), 2011–2018.

Kim, J.-M., Kim, D. H., & Jung, H. (2020). Modeling non-normal corporate bond yield spreads by copula. *North American Journal of Economics and Finance*, *53*, 1–15.

Kim, D. H., & Stock, D. (2014). The effect of interest rate volatility on corporate yield spreads on both noncallable and callable bonds. *Journal of Corporate Finance*, *26*, 20–35.

Kneip, A., & Utikal, K. J. (2001). Inference for density families using functional principal component analysis. *Journal of the American Statistical Association*, *96*(454), 519–542.
Kruppa, J., Schwarz, A., Arminger, G., & Ziegler, A. (2013). Consumer credit risk: Individual probability estimates using machine learning. *Expert Systems with Applications*, *40*(13), 5125–5131.
Kumar, G., Jain, S., & Singh, U. P. (2020). Stock market forecasting using computational intelligence: A survey. *Archives of Computational Methods in Engineering*, 1–33.
Le, H. H., & Viviani, J.-L. (2018). Predicting bank failure: An improvement by implementing a machine-learning approach to classical financial ratios. *Research in International Business and Finance*, *44*, 16–25.
Lekkos, I., Milas, C., & Panagiotidis, T. (2007). Forecasting interest rate swap spreads using domestic and international risk factors: Evidence from linear and non-linear models. *Journal of Forecasting*, *26*(8), 601–619.
Li, Y., & Ma, W. (2010). Applications of artificial neural networks in financial economics: a survey. In *2010 International Symposium on Computational Intelligence and Design, Vol. 1* (pp. 211–214). IEEE.
Lin, J.-L., & Granger, C. W. (1994). Forecasting from non-linear models in practice. *Journal of Forecasting*, *13*(1), 1–9.
Longstaff, F. A., Mithal, S., & Neis, E. (2005). Corporate yield spreads: Default risk or liquidity? New evidence from the credit default swap market. *The Journal of Finance*, *60*(5), 2213–2253.
Longstaff, F. A., & Schwartz, E. S. (1995). A simple approach to valuing risky fixed and floating rate debt. *The Journal of Finance*, *50*(3), 789–819.
Luo, Q., & Liu, X. (2021). Model of corporate bond spread based on improved neural network. *Journal of Intelligent & Fuzzy Systems*, (Preprint), 1–11.
Lykourentzou, I., Giannoukos, I., Nikolopoulos, V., Mpardis, G., & Loumos, V. (2009). Dropout prediction in e-learning courses through the combination of machine learning techniques. *Computers & Education*, *53*(3), 950–965.
Ma, Y., Han, R., & Wang, W. (2021). Portfolio optimization with return prediction using deep learning and machine learning. *Expert Systems with Applications*, *165*, Article 113973.
Masarotto, G., & Varin, C. (2012). Gaussian copula marginal regression. *Electronic Journal of Statistics*, *6*, 1517–1549.
McMillan, D. G. (2007). Non-linear forecasting of stock returns: Does volume help? *International Journal of Forecasting*, *23*(1), 115–126.
Merton, R. C. (1974). On the pricing of corporate debt: The risk structure of interest rates. *The Journal of Finance*, *29*(2), 449–470.
Mevik, B.-H., Wehrens, R., & San Michele all'Adige, T. (2015). Introduction to the pls package. *Help Section of the "Pls" Package of RStudio Software*, 1–23.
Min, J. H., & Lee, Y.-C. (2005). Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters. *Expert Systems with Applications*, *28*(4), 603–614.
Mishra, S., & Padhy, S. (2019). An efficient portfolio construction model using stock price predicted by support vector regression. *The North American Journal of Economics and Finance*, *50*(C).
Mishra, S., Padhy, S., Mishra, S. N., & Misra, S. N. (2021). A novel LASSO–TLBO–SVR hybrid model for an efficient portfolio construction. *The North American Journal of Economics and Finance*, *55*, Article 101350.
Moscatelli, M., Parlapiano, F., Narizzano, S., & Viggiano, G. (2020). Corporate default forecasting with machine learning. *Expert Systems with Applications*, *161*, Article 113567.
Nunes, M., Gerding, E., McGroarty, F., & Niranjan, M. (2018). Artificial neural networks in fixed income markets for yield curve forecasting. Available At SSRN 3144622.
Omar, N., Johari, Z. A., & Smith, M. (2017). Predicting fraudulent financial reporting using artificial neural network. *Journal of Financial Crime*.
Pal, R., Kupka, K., Aneja, A. P., & Militky, J. (2016). Business health characterization: A hybrid regression and support vector machine analysis. *Expert Systems with Applications*, *49*, 48–59.
Paliwal, M., & Kumar, U. A. (2009). Neural networks and statistical techniques: A review of applications. *Expert Systems with Applications*, *36*(1), 2–17.
Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert Systems with Applications*, *42*(1), 259–268.
Qi, M. (1996). Financial applications of artificial neural networks. *Handbook of Statistics*, *14*, 529–552.
Rasekhschaffe, K. C., & Jones, R. C. (2019). Machine learning for stock selection. *Financial Analysts Journal*, *75*(3), 70–88.
Rosadi, D., Nugraha, A., & Dewi, K. (2011). Forecasting the Indonesian government securities yield curve using neural networks and vector autoregressive model. *Bank for International Settlements*.
Ryll, L., & Seidens, S. (2019). Evaluating the performance of machine learning algorithms in financial market forecasting: A comprehensive survey. arXiv preprint arXiv:1906.07786.
Sklar, M. (1959). *Fonctions de Répartition À N Dimensions Et Leurs Marges*. Université Paris 8.
Sureshkumar, K., & Elango, N. (2012). Performance analysis of stock price prediction using artificial neural network. *Global Journal of Computer Science and Technology*, *12*(1), 18–25.
Svensén, M., & Bishop, C. M. (2007). *Pattern Recognition and Machine Learning*. Springer.
Vapnik, V. (1998). *Statistical Learning Theory, Vol. 3*. Wiley, New York.
Yoo, C., Ramirez, L., & Liuzzi, J. (2014). Big data analysis using modern statistical and machine learning methods in medicine. *International Neurourology Journal*, *18*(2), 50.