



Pearson Correlation Coefficient-based performance enhancement of Vanilla Neural Network for stock trend prediction

Ankit Thakkar¹ · Dhaval Patel¹ · Preet Shah¹

Received: 22 July 2020 / Accepted: 1 July 2021 / Published online: 19 July 2021
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2021

Abstract

The prediction of a volatile stock market is a challenging task. While various neural networks are integrated to address stock trend prediction problems, the weight initialization of such networks plays a crucial role. In this article, we adopt feed-forward Vanilla Neural Network (VNN) and propose a novel application of Pearson Correlation Coefficient (PCC) for weight initialization of VNN model. VNN consists of an input layer, a single hidden layer, and an output layer; the edges connecting neurons in the input layer and the hidden layer are generally initialized with random weights. While PCC is primarily used to find the correlation between two variables, we propose to apply PCC for weight initialization instead of random initialization (RI) for a VNN model to enhance the prediction performance. We also introduce the application of Absolute PCC (APCC) for weight initialization and analyze the effects of RI, PCC, and APCC values as weights for a VNN model. We conduct an empirical study using these concepts to predict the stock trend and evaluate these three weight initialization techniques on ten years of stock trading archival data of Reliance Industries, Infosys Ltd, HDFC Bank, and Dr. Reddy's Laboratories for the duration of years 2008 to 2017 for continuous as well as discrete data representations. We further evaluate the applicability of these weight initialization techniques using an ablation study on the considered features and analyze the prediction performance. The results demonstrate that the proposed weight initialization techniques, PCC and APCC, provide higher or comparable results as compared to RI, and the statistical significance of the same is carried out.

Keywords Vanilla Neural Network · Artificial Neural Network · Pearson Correlation Coefficient · Stock Trend Prediction · Weight Initialization · Ablation Study

1 Introduction

Money investment in a stock market has been one of the favorable investment strategies among a large number of people. Such investment demands knowledge of the market behavior as well as appropriate planning before trading through one or more stocks. The volatile stock market

consists of nonlinear complex patterns; future stock price or trend prediction using such noisy data are a difficult task. The market experiences fluctuations due to several influential parameters such as political conditions, various events at national as well as international levels, global economy, to name a few [1]. Hence, having a considerably potential system to predict the stock market trend with good accuracy can be an important aspect. Various fundamental analysis and technical analysis are also performed before investing in a stock market. While fundamental analysis represents an analysis of the numeric data such as earnings, profits, and ratios to determine future predictions, technical analysis focuses on utilizing historical data and modeling techniques that help to identify trends [1–3].

To forecast the future stock price as well as price movement, various experiments and analyses have been

✉ Ankit Thakkar
ankit.thakkar@nirmauni.ac.in

Dhaval Patel
15bit031@nirmauni.ac.in

Preet Shah
15bit053@nirmauni.ac.in

¹ Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad, Gujarat 382 481, India

carried out [4]. Several methods have also been developed using Artificial Neural Network (ANN), Support Vector Machine (SVM), Genetic Algorithm (GA), Linear Regression (LR), and other computational intelligence techniques. The pattern identification capabilities of such techniques are explored to predict the market trend and various models are developed to classify the nonlinear time-series stock data. One of the powerful machine learning techniques is ANN; it is used to learn the complex stock data and present the associations between input and output variables [5]. It is a widely used stock prediction model [6–10]; its capability to adapt the change in the data patterns is useful for deriving important information on the stock trend [11]. The widespread applicability of ANNs includes diverse domains such as intrusion detection systems [12], human action recognition [13], sentiment analysis [14, 15], recommender systems [16], personality trait identification [17], emotion recognition [18], face recognition [19], texture analysis [20], to name a few. Among the wide variety of applications, it can be observed that the prediction performance of a model can be largely dependent on its hyperparameters tuning as well as input features; while a stock market dataset may contain irrelevant features, selection of appropriate features plays a vital role to attain good performance for the given prediction model. It helps to eliminate the irrelevant features and hence, reduce the number of inputs given to a classifier. Such feature selection can enhance the overall prediction efficiency of the model [1, 21].

The stock price and stock trend prediction have attracted a large number of researchers from financial as well as computational fields. Various studies have been carried out using ANN models to forecast the stock market; along with an appropriate feature selection method, the weight initialization of an ANN model is a crucial task to improve the overall model performance. Here, an ANN can be considered as a combination of an input layer, one or more hidden layers, and an output layer with neuron(s) in each of the layers; the edges connect neurons of one layer to the next layer, and weights are provided over these edges. Various survey articles have reviewed the significance of artificial intelligence as well as machine learning techniques, including ANN for a variety of stock market applications [22, 23]. On the other hand, among a large number of diverse ANN applications, the stock market has also been remarkably considered [24]. One of the earlier work proposed solutions to initialize ANN weights which influenced the convergence ability of the model [25]. On the other hand, a statistical method was adopted to generate the weights of an ANN's hidden layer neurons using vector quantization (VQ) [26]; the equivalence between VQ prototypes and circular backpropagation networks was also provided. Forecasting techniques presented in study [27]

used the relevance of input variables and computed the predictive relationships of numerous financial input variables. A sensitivity-based linear learning method was proposed in Ref. [28] for providing an initial set of weights for ANN. Also, various studies adopted GA for optimizing the edge weights of ANNs [29–31] to further predict the stock market; GA was helpful in removing irrelevant features and hence, reducing the space complexity of the input features [32]. Results showed that the proposed model gave better results over the linear transformation with conventional ANN. On the other hand, a neuro-fuzzy system and scaled conjugate algorithm were used along with ANN to forecast the stock values in Ref. [33]. ANN was combined with the Genetic Fuzzy Systems in Ref. [34] to predict stock prices, wherein a regression analysis was performed to find the optimal feature space. Hence, integration of appropriate weight initialization and feature selection methods can benefit for the performance enhancement of ANN as well as other prediction models.

In this paper, we adopt a feed-forward Vanilla Neural Network (VNN) consisting of an input layer, a single hidden layer, and an output layer for stock trend prediction. The use of VNN architecture is motivated by universal approximation theorem which states that “a feed-forward network with a single hidden layer containing a finite number of neurons can approximate continuous functions on compact subsets of R^n , under mild assumptions on the activation function” [35]. Generally, the initial weights of VNN are randomly assigned and hence, it may not give the desired performance within the specified number of iterations. Therefore, we propose a new weight initialization technique for VNN using Pearson Correlation Coefficient (PCC) and predict the open price trend of the stock market; we also consider ten technical indicators as provided in [36] for this experiment. While PCC is mainly used to find how well two variables are associated with each other, we propose this novel application of PCC to derive initialization weights of a VNN model; experimentations are performed using feed-forward VNN using ten years of stock trading archival data from years 2008 to 2017 for the stocks of Reliance Industries, Infosys Ltd, HDFC Bank, and Dr. Reddy's Laboratories. The considered stocks are traded in a good amount and belong to different sectors, viz. Technical, Banking, and Pharmaceutical.

The rest of the paper is organized as follows: an overview of the proposed approach along with the prediction model and proposed weight initialization techniques are provided in Sect. 2; Section 3 discusses the collection of data and computation of technical indicators; the experimental results, ablation study, and analysis are given in Sect. 4; concluding remarks and future scope are discussed in Sect. 5.

2 The proposed approach

In this study, we focus on predicting the direction of the stock price movement using VNN and by employing ten technical indicators as given by [36]. In VNN, initial weights are given on the edges of the input neurons that connect to the neurons of the hidden layer. As convergence tends to increase at each iteration, the weight on each of the edges becomes more accurate toward the prediction of the final output. Hence, it is safe to conclude that the efficiency of a VNN in terms of accuracy and f -measure depends on the edge weights.

To study the significance of weight initialization techniques on stock trend prediction using VNN, we initially consider the random initialization (RI) for the VNN model, wherein the edge weights of VNN are randomly initialized. As discussed earlier, we can improve the overall efficiency of the model by initializing edge weights with the appropriate initial values. Thus, we propose a PCC-based approach to initialize the edge weights connecting neurons between the input and the hidden layers of a VNN; the Pearson correlations are useful in providing the relationship of a particular parameter with the others. In our proposed approach, PCC of an input variable (feature) quantitatively shows its correlation with the output. The correlation values range from -1 to 1 ; while a negative correlation shows that the input variables vary inversely with the output, a positive correlation shows that the input variables vary linearly to the output value. Thus, adopting PCC to derive the edge weights helps the underlined VNN to improve performance. As mentioned in [37], edge weights having non-negative weights tend to give better results than the ones with negative weights. Therefore, we introduce another technique to initialize VNN with Absolute Pearson Correlation Coefficient (APCC) values as the edge weights wherein all edge weights range between 0 and 1 .

Using different weight initialization techniques, we demonstrate a VNN model to predict the stock open price movement; a structural overview of the proposed approach is given in Fig. 1. Here, the historical datasets are collected and ten technical indicators (features) are selected as given in Ref. [36]; also, the derived technical indicators are represented in continuous as well as discrete data. These features are combined into a feature set and Pearson correlations are derived; as it can be seen, three techniques, namely, RI (the standard approach), PCC values as weights, and APCC values as weights are considered for model training; the predicted trend is evaluated using accuracy and f -measure metrics. The formulas to compute the selected technical indicators are shown in Table 1;

these features serve as inputs to the VNN model along with the three weight initialization techniques.

2.1 Feed-forward VNN: an ANN model

Various machine learning models are used for financial predictive research. ANN is one of the most frequently used methods for the same; it works well with more number of input features and a larger dataset. It has a capability to analyze nonlinear relationship between input and output to derive unknown patterns [38]. The stock market prices are highly volatile in nature; also, various factors can affect these prices. In order to derive the historical data patterns, the stock market dataset of many years can be used for training such ANN models.

In this study, we adopt feed-forward VNN with an input layer, a single hidden layer, and an output layer. Motivated by the universal approximation theorem [35], we provide a feature set comprising of ten technical indicators as an input to the VNN model. For the hyperparameter tuning, neurons in the hidden layer are varied from 10 to 100 , whereas the output layer contains one neuron that provides the value, 0 or 1 , as an output, i.e., trend down or up, respectively. Every neuron of a particular layer is connected to every other neuron in the next layer via edges that create a dense neural network. These edges have some weights that are used to perform computation on a particular input feature. With the help of these weights, the features that contribute more to predicting the output as compared to others can be assigned with appropriate weights. Thus, finding a correlation between input parameters and the output and further, using this correlation as weight for respective input neuron can result in improvement in stock trend prediction accuracy.

2.2 PCC-based weight initialization of feed-forward VNN: a novel application of PCC

Correlation between the input and output data is a measure of how well they are related to each other. The most common measure of correlation is the Pearson Correlation Coefficient (PCC). PCC shows a linear relationship between the two sets of data.

In this article, we compute PCC between the input features and output feature, i.e., open price; we initialize weights on the edges connecting neurons from the input layer to the hidden layer with respective PCC values. Hence, all the edges emerging out from an input feature are initialized with the PCC value of that input feature and the open price. Here, PCC values range between -1 and 1 . If two parameters are positively correlated then the coefficient is positive, i.e., between 0 and 1 , otherwise, the coefficient is negative, i.e., between -1 and 0 which

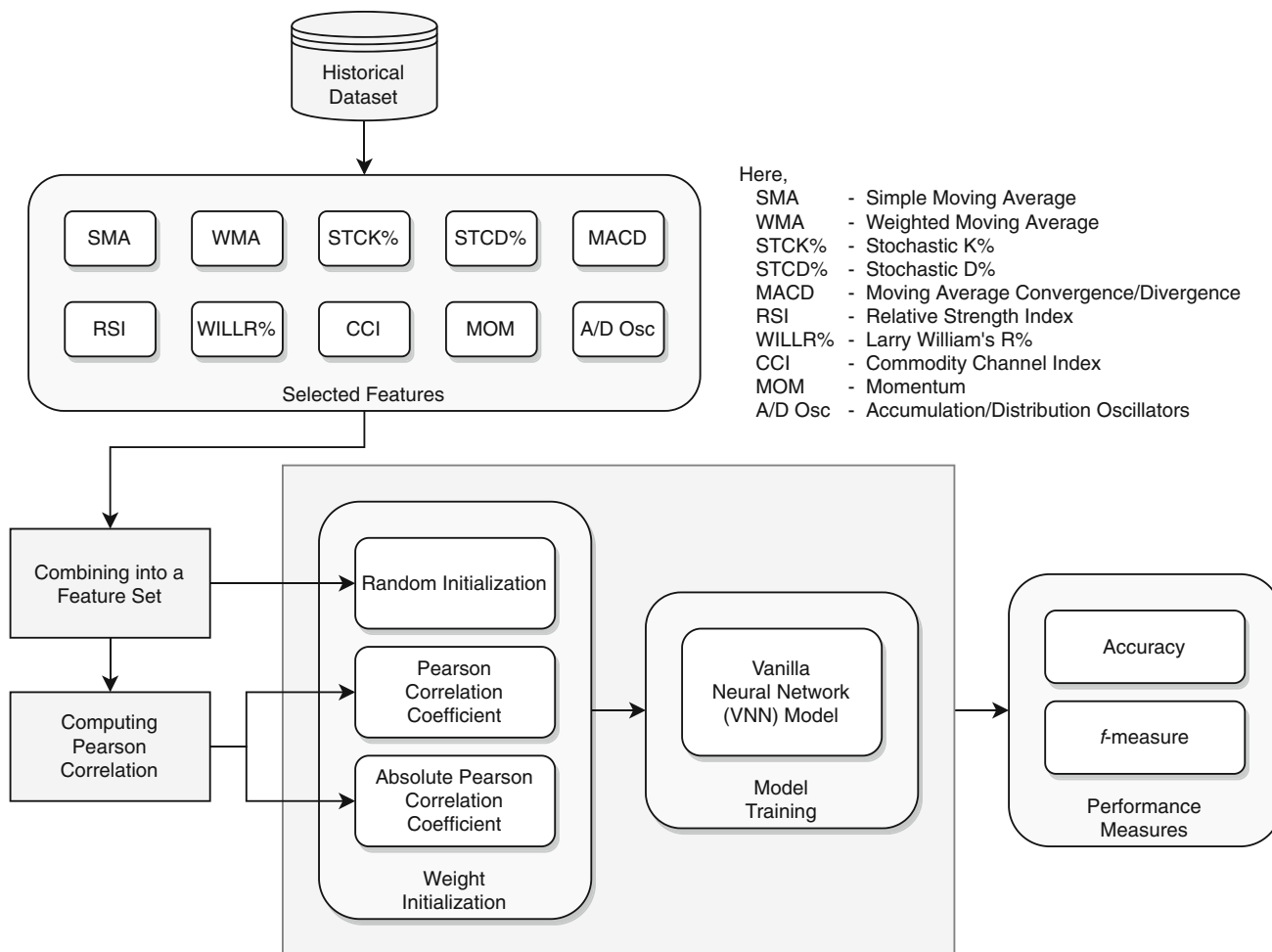


Fig. 1 A structural overview of the proposed approach

indicates a negative correlation. Using these coefficients as weights increases the accuracy because the highly correlated features receive positive weights and negatively affecting features receive negative weights as per their correlation with the output. The PCC (r) for variable x and y can be calculated using Eq. 1 [39].

$$r = \frac{N \sum xy - \sum x \sum y}{\sqrt{[N \sum x^2 - (\sum x)^2][N \sum y^2 - (\sum y)^2]}} \quad (1)$$

where, N denotes the number of pairs of scores, $\sum xy$ is the sum of products of paired scores, $\sum x$ and $\sum y$ indicate the sum of x scores and y scores, respectively; similarly, $\sum x^2$ and $\sum y^2$ give the sum of squared x scores and y scores, respectively.

After initializing the input neurons with respective PCC values, the weight update algorithm changes weights using Stochastic Gradient Descent (SGD) at every iteration during the training. The loss is measured in terms of binary cross-entropy. Here, the hidden layer has a Rectified Linear Unit (ReLU) activation function and the output layer has a

sigmoid activation function. Learning rate (lr), number of neurons in the hidden layer (n), number of iterations, i.e., epochs (ep), and momentum constant (mc) are the parameters of VNN that must be determined accurately to get good results. In this study, we consider continuous (as given by Ref. [36]) as well as discrete (as given by Ref. [40]) data representations of the considered technical indicators. For each representation, n is varied from 10 to 100 with an increment of 10, mc is varied from 0.1 to 0.9 with an increment of 0.1, ep is ranged from 1000 to 10,000 with an increment of 1000, and lr is set to 0.1; these parameter levels for feed-forward VNN are similar to Ref. [40]. The parameter combinations give $10 \times 10 \times 9 = 900$ different parameter settings for a specific company with a particular weight initialization technique using which VNN is trained. Thus, for a single company and for each data representation, we have performed 2700 iteration treatments in total which includes 900 training runs for each of the three weight initialization techniques, i.e., RI, PCC, and APCC.

Table 1 Selected technical indicators and their formulas [36]

Technical indicator	Formula
Simple $n(= 10)$ -day Moving Average (SMA)	$\frac{C_t + C_{t-1} + \dots + C_{t-10}}{10}$
Weighted $n(= 10)$ -day Moving Average (WMA)	$\frac{n \times C_t + (n-1) \times C_{t-1} + \dots + C_{t-10}}{n + (n-1) + \dots + 1}$
Stochastic K% (STCK%)	$\frac{C_t - LL_{t-(n-1)}}{HH_{t-(n-1)} - LL_{t-(n-1)}} \times 100$
Stochastic D% (STCD%)	$\frac{\sum_{i=0}^{n-1} K_{t-i}\%}{n}$
Moving Average Convergence/Divergence (MACD)	$MACD(n)_{t-1} + \frac{2}{n+1} \times (DIFF_t - MACD(n)_{t-1})$
Relative Strength Index (RSI)	$100 - \frac{100}{1 + (\sum_{i=0}^{n-1} Up_{t-i}/n) / (\sum_{i=0}^{n-1} Dw_{t-i}/n)}$
Larry William’s R% (WILLR%)	$\frac{H_n - C_t}{H_n - L_n} \times 100$
Commodity Channel Index (CCI)	$\frac{M_t - SM_t}{0.015D_t}$
Momentum (MOM)	$C_t - C_{t-n}$
Accumulation/Distribution Oscillator (A/D Osc)	$\frac{H_t - C_{t-1}}{H_t - L_t}$

Here, C_t , L_t , and H_t indicate closing price, low price, and high price at time t ; $DIFF = EMA(12)_t - EMA(26)_t$, where EMA is exponential moving average such that $EMA(k)_t = EMA(k)_{t-1} + \alpha \times (C_t - EMA(k)_{t-1})$ where α is a smoothing factor (i.e., $2/(1+k)$ wherein k is time period of k day EMA); LL_t and HH_t present the lowest low and highest high in the last t days, respectively; for $M_t = (H_t + L_t + C_t)/3$, $SM_t = \sum_{i=1}^n M_{t-i+1}/n$ and $D_t = (\sum_{i=1}^n |M_{t-i+1} - SM_t|)/n$; Up_t is the upward price change and Dw_t is the downward price change at time t

3 Research data collection

This section discusses the process to build the dataset and computation of technical indicators used for the experiments. Ten years of daily stock details for the years 2008 to 2017 are collected for Reliance Industries, Infosys Ltd, HDFC Bank, and Dr. Reddy’s Laboratories from the National Stock Exchange (NSE) of India [41]. The increase and decrease information of the datasets are important to find out whether the given data are balanced toward the ups and downs of the market. To provide an overview of the selected datasets, we elaborate the case study of Reliance Industries; the dataset covers a total of 2479 trading days and its increase and decrease cases of the distribution for each year are given in Table 2. It indicates that the total number of days with increase direction is 1253, while that for decrease direction is 1226, i.e., 50.54% cases for the increase direction and 49.46% cases for the decrease direction. For the experimentation, a similar distribution is obtained for other datasets.

VNN architecture contains various parameters such as connection weights, biases, learning rate, and a number of nodes in a hidden layer; each one of them affects the results based on the value assigned to them. Therefore, parameter tuning is required to select appropriate parameter values. Hence, we use a subset from the dataset which is 20% of the entire dataset. This subset is called the parameter selection dataset. This dataset is used in preliminary experiments to check for different parameter combinations

Table 2 The count of increase and decrease cases in the entire Reliance Industries dataset

Year	Increase	%Increase	Decrease	%Decrease	Total
2008	138	56.10	108	43.90	246
2009	112	46.09	131	53.91	243
2010	131	51.98	121	48.02	252
2011	136	54.40	114	45.60	250
2012	116	46.22	135	53.78	251
2013	129	51.60	121	48.40	250
2014	131	53.69	113	46.31	244
2015	122	49.19	126	50.81	248
2016	115	46.56	132	53.44	247
2017	123	49.60	125	50.40	248
Total	1253	50.54	1226	49.46	2479

to find a suitable parameter set that can be further used for the final experimentation using the entire dataset. For the preparation of the parameter selection dataset, we use two equally distributed 10% of the dataset for training and the rest 10% for the holdout. Here, an equal distribution indicates that 10% of increase cases, as well as decrease cases, are considered for training, as well as holdout, to ensure an unbiased distribution. For the final experimentation, the entire dataset is equally divided into training and holdout cases. From each year’s dataset, 50% of increase cases are assigned for training and the rest 50% of increase cases are

assigned to the holdout; the same procedure is carried out for decrease cases. Thus, equally distributed increase and decrease cases in training and holdout datasets are used to represent the entire dataset in equal parts. For each year, the count of increase and decrease cases for parameter selection of the Reliance Industries dataset is shown in Table 3 for parameter selection and for the final experimentation; these experimental settings are the same as [40].

The statistics of the selected indicators (Table 1) for the Reliance Industries dataset are given in Table 4; these technical indicators are used as a feature set. All indicators are further scaled down between $[-1, 1]$ so that each indicator is linearly normalized and thus, the large valued technical indicator is on the same scale with a small valued technical indicator. The output dataset consists of the value, 0 or 1; here, if the price of a stock at day T is higher than the price of that stock at day $T - 1$, then the output is assigned with 1, otherwise 0. In this study, the feature set of ten technical indicators is provided as the input to VNN which is further trained with three different weight initialization techniques, namely, RI, PCC, and APCC.

4 Experimental results and analysis

As explained in Sect. 3, VNN model is trained and tested on each of the four datasets using all the three weight initialization techniques, RI, PCC, and APCC. The simulation parameters considered for the experiments are given in Table 5. Here, we carry out the experimentation on two different data representations as given by Refs. [36] and [40]; while the former approach considers continuous data and further provides the same to VNN model, the latter approach introduces an opinion layer, wherein each

technical indicator undergoes the trend identifying calculation and the trading data of each technical indicator is further represented using $+1$ or -1 . Such data representation has displayed performance improvement in Ref. [40]. To extend the applicability of our proposed approach, we consider the continuous (Ref. [36]), as well as discrete (Ref. [40]), data representations and compare results of companies, Reliance Industries, Infosys Ltd, HDFC Bank, and Dr. Reddy's Laboratories, with respect to all the three weight initialization techniques as given in Tables 6, 7, 8 and 9, respectively. These experiments have been performed on Python using Keras framework [42]. For the analysis purpose, we consider accuracy and f -measure as the performance metrics. As we have carried out a series of operations with a different number of neurons, epochs, as well as mc values, the comparison tables present the top five results derived among the given combinations for each weight initialization technique and both of the considered data representations.

As shown in Table 6, the top five results for Reliance Industries dataset indicate that the proposed weight initialization techniques, i.e., PCC and APCC, outperform the conventional weight initialization technique for the VNN model in terms of accuracy for a continuous data representation. On the other hand, for discrete data representation, PCC and APCC have attained near-comparable results with respect to the standard weight initialization technique (RI). Also, it can be observed that the discrete data representation has provided higher results as compared to the continuous data for each of the three weight initialization techniques. Thus, the results show comparable or higher performance improvement for the Reliance Industries dataset.

Table 3 The count of increase and decrease cases for training and holdout in the Reliance Industries dataset

Year	Parameter selection						Final experimentation					
	Training			Holdout			Training			Holdout		
	Inc.	Dec.	Total	Inc.	Dec.	Total	Inc.	Dec.	Total	Inc.	Dec.	Total
2008	14	11	25	13	10	23	69	54	123	69	54	123
2009	11	13	24	11	13	24	56	66	122	56	65	121
2010	13	12	25	13	12	25	66	61	127	65	60	125
2011	14	11	25	13	11	24	68	57	125	68	57	125
2012	12	14	26	11	13	24	58	68	126	58	67	125
2013	13	12	25	12	12	24	65	61	126	64	60	124
2014	13	11	24	13	11	24	66	57	123	65	56	121
2015	12	13	25	12	12	24	61	63	124	61	63	124
2016	12	13	25	11	13	24	58	66	124	57	66	123
2017	12	13	25	12	12	24	62	63	125	61	62	123
Total	126	123	249	121	119	240	629	616	1245	624	610	1234

Here, Inc. and Dec. indicate number of increase and decrease cases, respectively

Table 4 Statistics for the selected indicators for the Reliance Industries dataset

Technical indicator	Max	Min	Mean	Standard deviation
SMA	3073.355	687.270	1148.230	468.322
WMA	3088.008	692.827	1148.193	466.739
STCK%	93.509	4.661	49.298	23.777
STCD%	90.961	6.310	49.241	21.048
MACD	140.296	– 276.968	– 4.886	43.567
RSI	90.075	10.438	49.743	14.4167
WILLR%	99.630	0.991	50.641	30.158
CCI	333.333	– 333.333	0.8341	103.829
MOM	483.900	– 1122.200	– 7.848	120.897
A/D Osc	12733655	– 10634668	– 537471	2653823.465

Table 5 Simulation parameters

Parameters	Values
Datasets	i) Reliance Industries ii) Infosys Ltd iii) HDFC Bank iv) Dr. Reddy’s Laboratories
Duration	2008–2017
Features	SMA, WMA, STCK%, STCD%, MACD, RSI, WILLR%, CCI, MOM, A/D Osc
Data representation	i) Continuous (as per [36]) ii) Discrete (as per [40])
Prediction model	Vanilla neural network (VNN)
Epochs (<i>ep</i>)	$ep \in \{1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000, 10000\}$
Number of neurons (<i>n</i>)	$n \in \{10, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$
Momentum constant (<i>mc</i>)	$mc \in \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$
Learning rate (<i>lr</i>)	0.1
Activation	Rectified Linear Unit (ReLU), Sigmoid
Optimizer	Stochastic Gradient Descent (SGD)
Weight initialization	i) Random initialization (RI) ii) Pearson Correlation Coefficient (PCC) iii) Absolute Pearson Correlation Coefficient (APCC)

In a similar way, Table 7 presents the top five results of Infosys Ltd dataset. Here, VNN model with APCC as weights outperforms while comparing the prediction accuracies for continuous data, whereas VNN model with PCC-based weights provides better or comparable results. Subsequently, VNN model with PCC-based weights outperforms for the discrete data representation, whereas VNN model with APCC-based weights indicates comparable results or an overall performance improvement. As in the case of Reliance Industries dataset, each of the weight initialization techniques for the discrete data representation approach has considerably higher prediction performance than that of the continuous data.

Table 8 depicts the top five results of the HDFC bank wherein the VNN model with PCC as weights perform

slightly lower than the VNN model with RI for continuous data; on the other hand, VNN model with APCC values as weights outperforms the conventional technique. As it can be observed, the average accuracy of using RI with VNN for discrete data is slightly higher than the proposed weight initialization techniques here, however, the top most accuracy of PCC-based weights for VNN is equivalent to that of RI and *f*-measure indicates higher value. These results are significant improvements to that of the continuous data for HDFC Bank dataset as well.

On the other hand, Table 9 describes the top five results of Dr. Reddy’s Laboratories. While both the proposed weight initialization techniques, i.e., PCC and APCC, outperform the standard weight initialization technique, RI, for continuous data, a notable performance improvement is

Table 6 Comparison of VNN model with three weight initialization techniques for Reliance Industries dataset

Data Representation	VNN model with RI as weights					VNN model with PCC as weights					VNN model with APCC as weights				
	Neurons	Epochs	mc	Accuracy	f-measure	Neurons	Epochs	mc	Accuracy	f-measure	Neurons	Epochs	mc	Accuracy	f-measure
Continuous (as per [36])	10	: 9000	: 0.8	70.29	68.16	10	: 2000	: 0.5	73.64	70.97	10	: 10000	: 0.2	71.55	72.36
	40	: 4000	: 0.4	69.46	69.46	20	: 9000	: 0.4	72.38	73.39	20	: 3000	: 0.7	70.29	68.44
	100	: 5000	: 0.6	69.46	70.92	70	: 1000	: 0.5	71.97	70.74	20	: 4000	: 0.4	70.29	71.49
	80	: 4000	: 0.6	69.46	68.94	90	: 5000	: 0.2	71.55	73.44	50	: 1000	: 0.3	69.87	69.23
	60	: 3000	: 0.9	69.04	71.32	40	: 2000	: 0.2	71.13	72.29	90	: 7000	: 0.4	69.46	69.46
	Mean			69.542	69.76	Mean			72.134	72.166	Mean			70.292	70.196
Discrete (as per [40])	Median			69.46	69.46	Median			71.97	72.29	Median			70.29	69.46
	30	: 2000	: 0.8	75.73	77.17	100	: 2000	: 0.7	76.57	78.63	30	: 2000	: 0.3	74.9	76.19
	90	: 3000	: 0.2	75.31	75.52	50	: 4000	: 0.9	76.15	75.74	100	: 7000	: 0.2	74.9	74.36
	20	: 1000	: 0.7	75.31	74.46	90	: 9000	: 0.7	75.73	77.34	100	: 7000	: 0.1	74.48	75.7
	20	: 8000	: 0.5	75.31	76.11	90	: 5000	: 0.2	75.73	76.61	50	: 5000	: 0.8	74.48	75.5
	70	: 10000	: 0.7	75.31	74.01	100	: 10000	: 0.5	75.31	75.31	20	: 4000	: 0.6	74.48	75.3
Mean			75.394	75.454	Mean			75.898	76.726	Mean			74.648	75.41	
Median			75.31	75.52	Median			75.73	76.61	Median			74.48	75.5	

Table 7 Comparison of VNN model with three weight initialization techniques for Infosys Ltd dataset

Data Representation	VNN model with RI as weights				VNN model with PCC as weights				VNN model with APCC as weights				
	Neurons	Epochs	mc	Accuracy <i>f</i> -measure	Neurons	Epochs	mc	Accuracy <i>f</i> -measure	Neurons	Epochs	mc	Accuracy <i>f</i> -measure	
Continuous (as per [36])	70	: 8000	: 0.9	75.73	20	: 4000	: 0.4	71.97	50	: 1000	: 0.9	73.64	74.29
	70	: 4000	: 0.8	71.55	30	: 1000	: 0.9	70.71	20	: 1000	: 0.3	73.22	71.93
	20	: 5000	: 0.1	71.55	80	: 4000	: 0.8	70.71	10	: 1000	: 0.6	71.55	71.43
	20	: 1000	: 0.5	71.13	90	: 3000	: 0.7	70.29	40	: 1000	: 0.6	70.29	65.7
	20	: 1000	: 0.3	71.13	10	: 2000	: 0.2	70.29	50	: 9000	: 0.4	70.29	67.87
	Mean			72.218	Mean			70.794	Mean			71.798	70.244
Discrete (as per [40])	Median			71.55	Median			70.71	Median			71.55	71.43
	90	: 6000	: 0.3	75.31	20	: 2000	: 0.2	76.15	50	: 10000	: 0.3	76.15	75.11
	100	: 8000	: 0.8	74.9	40	: 6000	: 0.2	75.31	50	: 3000	: 0.8	75.31	74.68
	80	: 10000	: 0.2	74.48	60	: 5000	: 0.4	75.31	80	: 9000	: 0.6	74.48	73.36
	90	: 8000	: 0.2	74.48	60	: 8000	: 0.8	74.9	20	: 5000	: 0.7	74.06	72.57
	50	: 2000	: 0.7	74.06	10	: 4000	: 0.5	74.9	60	: 7000	: 0.8	73.22	70.91
Mean			74.646	Mean			75.314	Mean			74.644	73.326	
Median			74.48	Median			75.31	Median			74.48	73.36	

Table 8 Comparison of VNN model with three weight initialization techniques for HDFC Bank dataset

Data Representation	VNN model with RI as weights				VNN model with PCC as weights				VNN model with APCC as weights						
	Neurons	Epochs	mc	Accuracy <i>f</i> -measure	Neurons	Epochs	mc	Accuracy <i>f</i> -measure	Neurons	Epochs	mc	Accuracy <i>f</i> -measure			
Continuous (as per [36])	30	2000	0.9	72.15	65.62	70	3000	0.9	72.57	75.1	50	2000	0.9	71.31	65.31
	30	4000	0.9	71.73	65.28	40	6000	0.4	72.15	72.03	10	1000	0.5	70.89	68.78
	40	3000	0.6	71.73	69.12	80	1000	0.1	71.73	71.97	50	6000	0.4	70.89	69.33
	10	1000	0.6	70.89	69.06	90	6000	0.9	71.31	68.81	70	10000	0.6	70.89	71.13
	20	9000	0.2	70.46	66.02	70	7000	0.4	71.31	67.62	10	8000	0.8	70.04	66.03
	Mean			71.392	67.02	Mean			71.814	71.106	Mean			70.804	68.116
Discrete (as per [40])	Median			71.73	66.02	Median			71.73	71.97	Median			70.89	68.78
	60	8000	0.9	76.37	75.22	70	2000	0.4	78.48	77.53	40	7000	0.1	76.79	74.18
	80	2000	0.8	75.11	71.77	60	1000	0.8	76.37	74.07	90	8000	0.3	75.95	76.35
	70	6000	0.7	75.11	75.11	30	5000	0.6	75.95	75.11	50	1000	0.3	75.11	73.06
	20	1000	0.8	75.11	73.78	10	6000	0.8	75.11	73.54	60	1000	0.2	74.68	73.21
	10	9000	0.4	74.68	73.68	50	7000	0.3	75.11	74.68	10	8000	0.1	74.26	73.59
Mean			75.276	73.912	Mean			76.204	74.986	Mean			75.358	74.078	
Median			75.11	73.78	Median			75.95	74.68	Median			75.11	73.59	

Table 9 Comparison of VNN model with three weight initialization techniques for Dr. Reddy’s Laboratories dataset

Data Representation	VNN model with RI as weights					VNN model with PCC as weights					VNN model with APCC as weights				
	Neurons	Epochs	mc	Accuracy	f_{-} measure	Neurons	Epochs	mc	Accuracy	f_{-} Measure	Neurons	Epochs	mc	Accuracy	f_{-} Measure
Continuous (as per [36])	50	: 5000	: 0.9	70.42	62.43	10	: 1000	: 0.9	72.92	72.1	10	: 8000	: 0.1	71.25	71.84
	60	: 10000	: 0.8	70	69.23	10	: 6000	: 0.9	72.08	73.73	100	: 2000	: 0.1	69.17	67.26
	80	: 10000	: 0.9	69.58	69.71	90	: 3000	: 0.5	69.58	67.26	80	: 7000	: 0.8	69.17	68.1
	40	: 7000	: 0.5	69.58	66.36	80	: 6000	: 0.3	69.58	68.4	60	: 8000	: 0.5	68.75	67.53
	80	: 10000	: 0.2	69.58	70.45	20	: 2000	: 0.4	69.17	69.42	60	: 10000	: 0.7	67.92	67.23
	Mean			69.832	67.636	Mean			70.666	70.182	Mean			69.252	68.392
	Median			69.58	69.23	Median			69.58	69.42	Median			69.17	67.53
	40	: 8000	: 0.9	75	73.91	80	: 1000	: 0.2	75.83	74.11	70	: 2000	: 0.1	76.25	75.54
	30	: 6000	: 0.2	74.17	72.32	70	: 7000	: 0.5	74.58	73.36	100	: 6000	: 0.5	75.83	74.11
	80	: 4000	: 0.5	73.75	70.42	80	: 6000	: 0.4	74.17	74.38	80	: 5000	: 0.2	75.83	74.11
Discrete (as per [40])	50	: 4000	: 0.4	73.75	72	80	: 6000	: 0.8	74.17	72.81	70	: 10000	: 0.8	75.42	74.01
	40	: 8000	: 0.7	73.75	73.19	10	: 8000	: 0.2	74.17	72.57	40	: 3000	: 0.8	75.42	72.04
	Mean			74.084	72.368	Mean			74.584	73.446	Mean			75.75	73.962
	Median			73.75	72.32	Median			74.17	73.36	Median			75.83	74.11

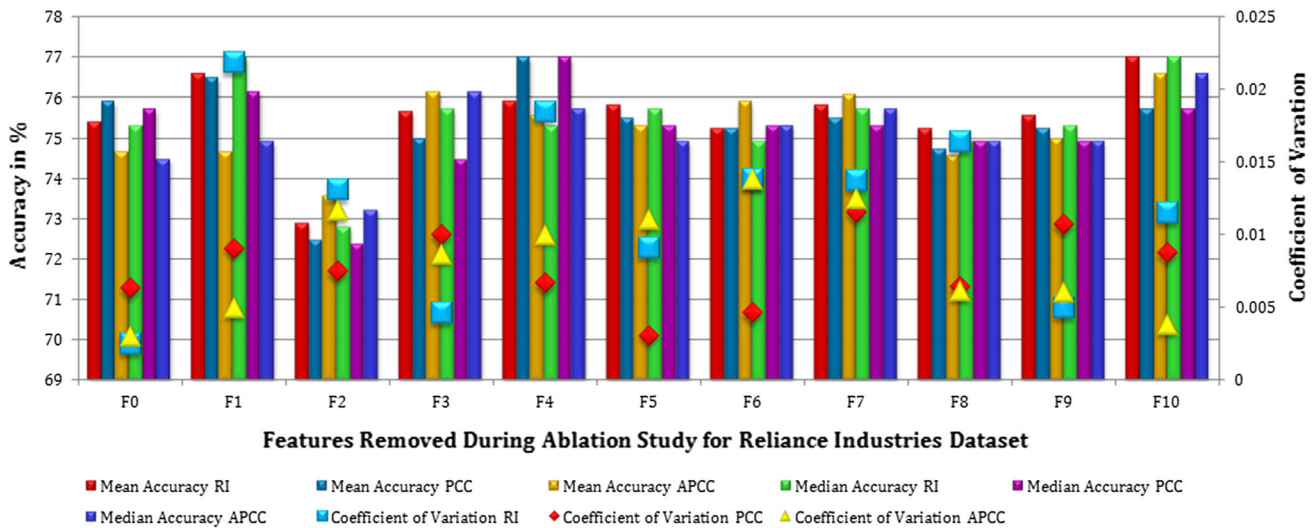


Fig. 2 Comparative analysis using ablation study for Reliance Industries dataset

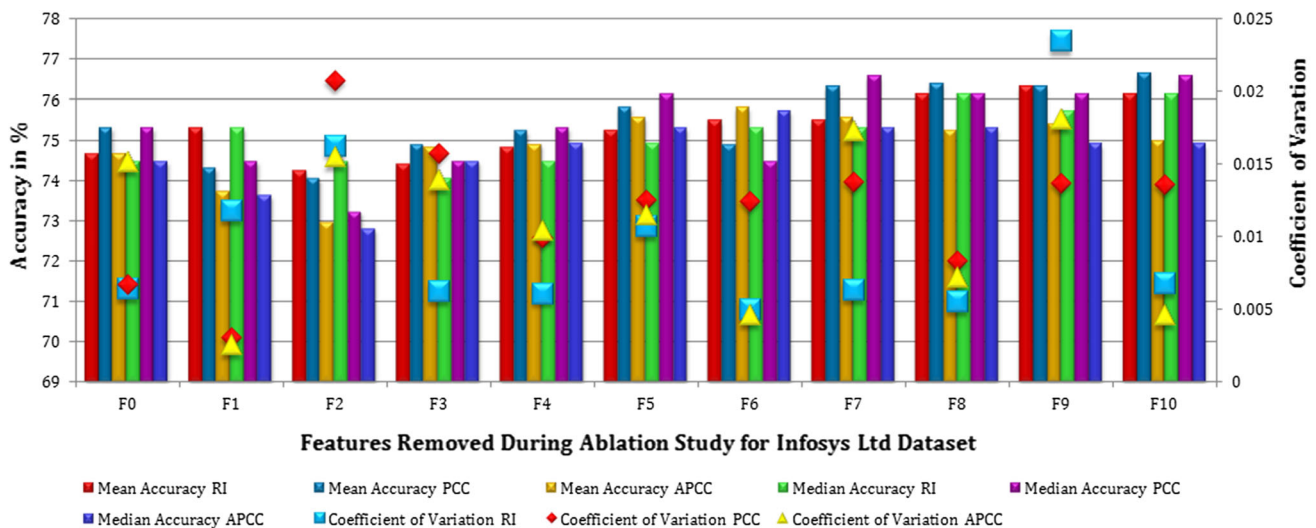


Fig. 3 Comparative analysis using ablation study for Infosys Ltd dataset

noticed for the discrete data for PCC-based VNN weights. Also, the higher results with discrete data representation are observed here. The above analysis is empirically carried out based on the accuracy metric.

In order to verify the statistical significance of the proposed approaches, PCC and APCC as weight initialization techniques for the VNN model, we compare the results with the RI-based VNN using the Wilcoxon signed-rank test. The results show that the proposed approach, APCC, as a weight initialization technique for VNN outperforms randomly initialized VNN for all the four companies for the continuous data wherein the obtained p -value is less than .05 [43]. On the other hand, PCC-based weight initialization for VNN model using discrete data representation indicate comparable or higher performance accuracy;

for Dr. Reddy's Laboratories dataset, it attains p -value less than .05 using Wilcoxon signed-rank test.

In this article, we select the ten technical indicators for a fair comparison based on Refs. [36] and [40]; the effectiveness of different weight initialization techniques can be further analyzed using the selected set of features, i.e., technical indicators, through an ablation study. An ablation study can be an important approach to understand the significance of individual indicators as well as a group of the selected indicators; such a study can be carried out to eliminate the least correlating input features from the considered indicators. In this article, we evaluate the impact of dropping a particular feature through an ablation study; the experiments are performed on discrete data representations due to their significant improvements observed through Tables 6, 7, 8 and 9. Here, for each

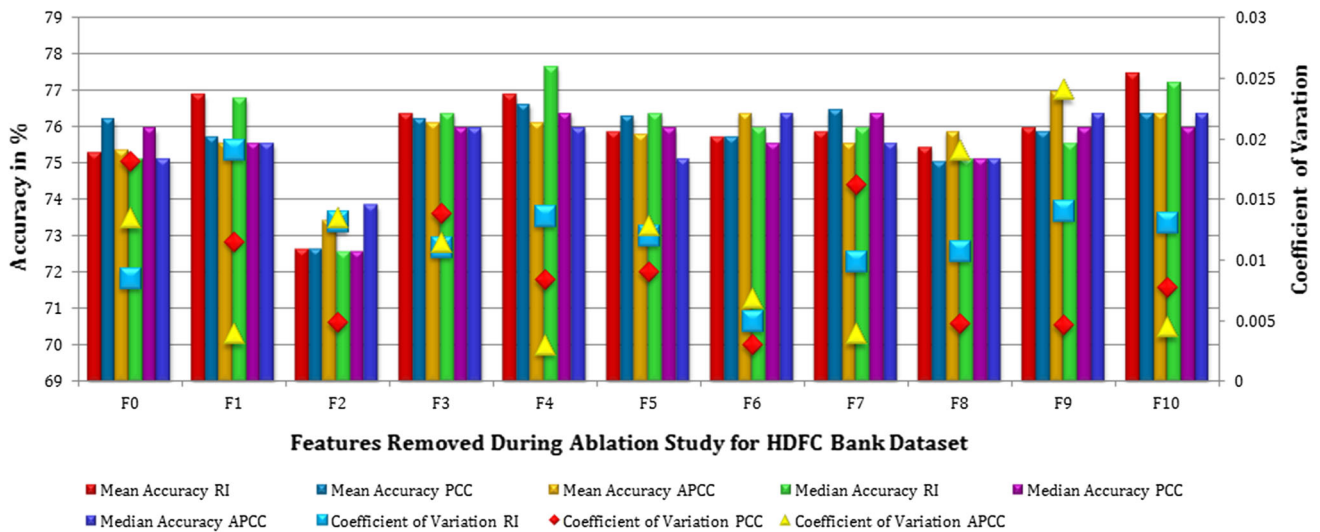


Fig. 4 Comparative analysis using ablation study for HDFC Bank dataset

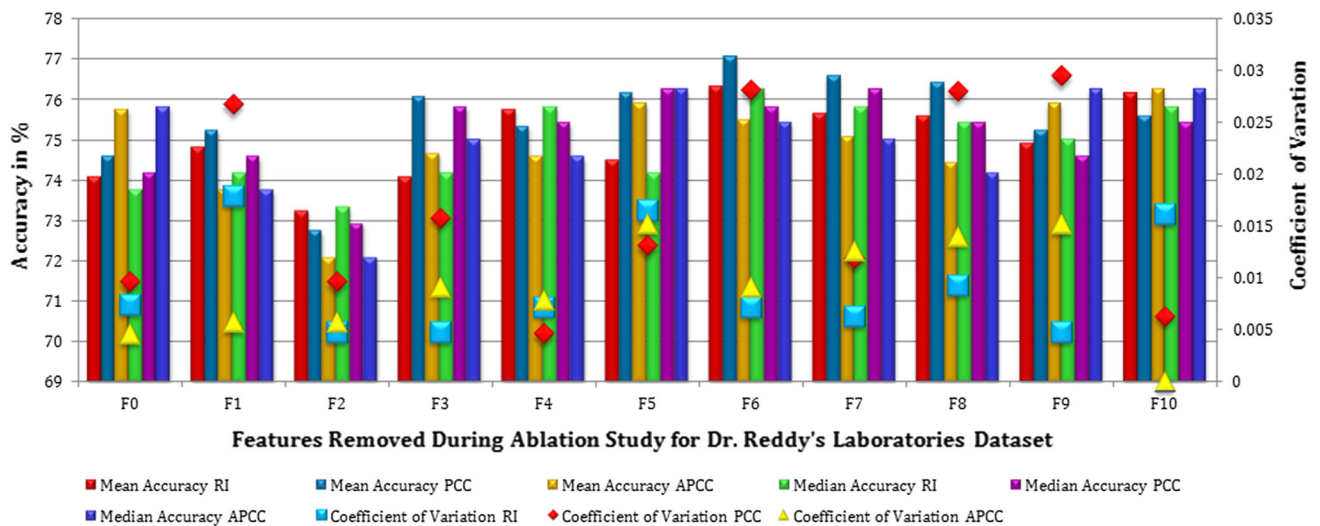


Fig. 5 Comparative analysis using ablation study for Dr. Reddy's Laboratories dataset

dropped feature under the ablation study, there are a total of 2700 combinations of the number of neurons, epochs, *mc* values, and weight initialization techniques for each dataset. Hence, by dropping one of the ten features for all four datasets, a total of $2700 \times 4 \times 10 = 108,000$ experimentations are carried out for the ablation study. As expressed earlier, comparisons of the mean and median accuracies of the top five results attained for each dataset are represented for three weight initialization techniques in Figs. 2, 3, 4 and 5. The graphical comparisons are further enhanced with coefficient of variation (CV) metric to indicate dispersion of the results [44]. Here, *F0* indicates that all ten features are considered, whereas *F1* to *F10* denote the removal of SMA, WMA, STCK%, STCD%, MACD, RSI, WILLR%, CCI, MOM, and A/D Osc feature,

respectively, and the rest nine features are considered for the specific set of experimentations.

It can be observed that PCC and APCC-based weight initialization techniques attain higher accuracies as compared to RI-based weight initialization in a large number of cases for an individual set of considered features. As compared to *F0* with all ten features, the removal of one of the features (*F1* – *F10*) has considerable impacts on the stock trend prediction; while dropping *F2*, i.e., WMA, significantly reduces the prediction accuracy, removing *F10*, i.e., A/D Osc denotes performance improvement; on the other hand, removal of certain features have a minimum impact on the resultant accuracies. The effectiveness of our proposed PCC and APCC-based weight initialization techniques can be viewed through individual ablation studies. We further analyze the same using CV metric; the

smaller values indicate smaller variability and thus, higher stability [44]. It is shown in Figs. 2 and 4 that CV values based on PCC, as well as APCC, are smaller than that of RI in the majority of the cases for Reliance Industries and HDFC Bank datasets; while close CV values are attained for Infosys Ltd dataset using all three weight initialization techniques (Fig. 3), overall smaller CV values can be observed using APCC for Dr. Reddy's Laboratories dataset (Fig. 5). Thus, the ablation study indicates the potential effectiveness of PCC and APCC-based weight initialization techniques over RI for a different set of features; it also helps to evaluate the impact of individual technical indicators in deriving the stock trend prediction. The same can be further utilized to engineer network weights related to the feature importance.

5 Concluding remarks and future scope

The paper aimed to predict the direction of the stock movement. We have accomplished the task by using the VNN model based on ten years (2008–2017) of stock trading archival data of Reliance Industries, Infosys Ltd, HDFC Bank, and Dr. Reddy's Laboratories. For this, we used ten technical indicators that were given as inputs to the VNN to predict the direction of the stock price.

In this paper, we have proposed a potential solution that computes the correlation coefficient between each input variable and the output variable and further uses these correlation values to initialize their respective edge weights that connect input neurons to the hidden layer neurons in the given VNN model. In the prior study, the aforementioned ten technical indicators were directly used to predict the stock movement; in contrast to that, we propose to initially compute the PCC values of these indicators and assign them as weights to their respective VNN edges. Subsequently, we also propose to extend the same approach using APCC-based weights for the VNN model. For a detailed understanding of the proposed approach, we carry out a series of experimentations using four datasets and three weight initialization techniques, i.e., RI, PCC, and APCC. The hyperparameter tuning can be a critical challenge and therefore, we consider to evaluate our proposed approaches with different combinations of the number of neurons, number of epochs, as well as variable mc values. Hence, the results obtained using various combinations of these hyperparameters are compared using accuracy as well as f -measure metrics. We further extend the applicability of our proposed approach and adopt an opinion layer as proposed in Ref. [40] which presents the technical indicator data using $+1$ and -1 , i.e., discrete representation. We further carry out the experimentations with the discrete data representation and evaluate the

performance for each of the four companies using each of the three weight initialization techniques. We also compared the prediction performance between the continuous and the discrete data representations as given in Tables 6, 7, 8 and 9. The results indicate that the proposed approaches deliver better or comparable results in predicting the movement of stock prices than that of RI. Subsequently, it has been observed that a significant performance enhancement is achieved with the discrete data representation. We further evaluate the statistical significance of the results that demonstrate that the proposed weight initialization techniques, PCC and APCC, achieve better or comparable results than the conventional weight initialization technique, RI.

The effectiveness of the proposed approach can be evaluated using an ablation study wherein one feature is dropped and the experiments are carried out with the remaining set of features to study the impact of the dropped feature. We considered to apply an ablation study on our set of features such that one of the ten technical indicators was dropped for each experiment. The removal of a feature can indicate the impact of that feature on the prediction performance. The exhaustive experimentations indicated how certain features could influence the prediction; while removal of WMA technical indicator showed to reduce the prediction accuracy, removal of A/D Osc technical indicator resulted in accuracy improvement. Thus, a thorough analysis can be beneficial for deriving the correlation, as well as significance, of individual technical indicators. We extended our study by incorporating CV metric to analyze the variability in derived prediction results. The smaller CV values indicated higher stability; among a different set of experimentations, an overall smaller or comparable CV value could be attained using PCC, as well as APCC, as compared to RI-based weight initialization. Such consistencies were observed for individual ablation studies as well. The results indicated the importance of PCC and APCC-based weight initialization techniques and their potential for further acceptance as weight initialization techniques for other models.

An empirical analysis of the proposed approach has been presented using accuracy and f -measure. However, along with accuracy and f -measure, several other metrics can be incorporated for further analysis. While the technical indicators have been derived using 10-day period, finding a suitable period for technical indicators is worth investigating. The proposed approaches can be extended for other domains that use VNN or ANN models for the given application. Initializing VNN with proper initial weights using PCC or APCC suggests that the model gets good initial weight values for the input parameters according to their contribution toward the output variable.

Thus, we can state that these weights play an important role to determine the performance of VNN. As a part of the future work, other macroeconomic variables such as inflation, currency exchange rates, or an average stock volume, can also be used along with the technical indicators to understand the dominance of such factors on the stock market as well as to enhance the prediction performance; the variety of data can be further fused to enhance the forecasting capability of the model [45]. On the other hand, the impact of such indicators on the prediction performance can be studied through ablation study for possible improvement in accuracy and reduction in training time; also, customized weights can be initialized based on the feature importance. Some of the recent works have indicated that companies having been listed on different stock exchanges can provide useful information [3, 46]; such factors can be potentially explored with improved weight initialization such as that proposed in our approach. The effectiveness of using PCC and APCC can also be evaluated for a variety of deep learning-based models for stock trend prediction [47].

The concept of transfer learning can be useful to forecast the trend of the stock movement for indices with incomplete data. The neural network model can be trained on the dataset of any other company and further, the knowledge gained by the model can be used to forecast the trend of the stock movement for the identified company. While we present PCC and APCC-based weight initialization techniques, other feature weight techniques may be compared and/or potentially combined to evaluate different ways to derive useful features [48]. The focus of this study is to predict the stock direction for short terms; this work can also be extended for the long-term prediction which may require other parameters such as analysis of stock's overall profit returns, the impact of the company throughout the financial year, revenue, to name a few. One of the considerable aspects is hyperparameter tuning; while, we have considered a series of experimentations to derive the top five combinations, a metaheuristic approach such as particle swarm optimization [1] can be further utilized. Other domains can also be explored using the proposed methods that may help in improving the performance of the model. The proposed approaches can also be integrated with deep learning techniques and can be considered as another future redirection.

Declarations

Conflict of interest The authors declare no potential conflict of interests.

References

1. Thakkar A, Chaudhari K (2020) A comprehensive survey on portfolio optimization, stock price and trend prediction using particle swarm optimization. *Archiv Comput Methods Eng* 28:1–32
2. Schumaker RP, Chen H (2009) Textual analysis of stock market prediction using breaking financial news: the AZFin text system. *ACM Trans Inf Syst (TOIS)* 27(2):12
3. Thakkar A, Chaudhari K (2020) CREST: cross-reference to exchange-based stock trend prediction using long short-term memory. *Procedia Comput Sci* 167:616
4. Ni LP, Ni ZW, Gao YZ (2011) Stock trend prediction based on fractal feature selection and support vector machine. *Expert Syst Appl* 38(5):5569
5. Zhang G, Patuwo BE, Hu MY (1998) Forecasting with artificial neural networks: the state of the art. *Int J Forecast* 14(1):35
6. Yoo PD, Kim MH, Jan T (2005) Financial forecasting: advanced machine learning techniques in stock market analysis. In: 2005 Pakistan Section Multitopic Conference, 2005, pp 1–7. <https://doi.org/10.1109/INMIC.2005.334420>
7. Chang PC, Liu CH, Lin JL, Fan CY, Ng CS (2009) A neural network with a case based dynamic window for stock trading prediction. *Expert Syst Appl* 36(3):6889
8. Tsai CF, Chiou YJ (2009) Earnings management prediction: a pilot study of combining neural networks and decision trees. *Expert Syst Appl* 36(3):7183
9. de Oliveira FA, Zárate LE, de Azevedo Reis M, Nobre CN (2011) The use of artificial neural networks in the analysis and prediction of stock prices. In: 2011 IEEE international conference on (IEEE) systems, man, and cybernetics (SMC), pp 2151–2155
10. Cai X, Lai G, Lin X (2013) Forecasting large scale conditional volatility and covariance using neural network on GPU. *J Supercomput* 63(2):490
11. Guresen E, Kayakutlu G, Daim TU (2011) Using artificial neural network models in stock market index prediction. *Expert Syst Appl* 38(8):10389
12. Thakkar A, Lohiya R (2020) A review on machine learning and deep learning perspectives of IDS for IoT: recent updates, security issues, and challenges. *Archiv Comput Methods Eng* 28:1–33
13. Pareek P, Thakkar A (2020) A survey on video-based human action recognition: recent updates, datasets, challenges, and applications. *Artif Intell Rev* 54:1–64
14. Thakkar A, Mungra D, Agrawal A (2020) Sentiment analysis: an empirical comparison between various training algorithms for artificial neural network. *Int J Innov Comput Appl* 11(1):9
15. Mungra D, Agrawal A, Thakkar A (2020) A voting-based sentiment classification model. In: *Intelligent communication, control and devices*. Springer, pp 551–558
16. Chaudhari K, Thakkar A (2020) A comprehensive survey on travel recommender systems. *Archiv Comput Methods Eng* 27:1545–1571
17. Chaudhari K, Thakkar A (2019) Survey on handwriting-based personality trait identification. *Expert Syst Appl* 124:282
18. Sharma R, Rajvaidya H, Pareek P, Thakkar A (2019) A comparative study of machine learning techniques for emotion recognition. In: *Emerging research in computing, information, communication and applications*. Springer, 906, pp 459–464
19. Thakkar A, Jivani N, Padasumbiya J, Patel CI (2013) A new hybrid method for face recognition. In: 2013 Nirma University international conference on engineering (NUICONE). IEEE, pp 1–9
20. Patel R, Patel CI, Thakkar A (2012) Aggregate features approach for texture analysis. In: 2012 Nirma University international conference on engineering (NUICONE). IEEE, pp 1–5

21. Tsai CF, Hsiao YC (2010) Combining multiple feature selection methods for stock prediction: union, intersection, and multi-intersection approaches. *Decis Support Syst* 50(1):258
22. Ray R, Khandelwal P, Baranidharan B (2018) A survey on stock market prediction using artificial intelligence techniques. In: 2018 International conference on smart systems and inventive technology (ICSSIT). IEEE, pp 594–598
23. Ryll L, Seidens S (2019) Evaluating the performance of machine learning algorithms in financial market forecasting: a comprehensive survey. [arXiv:1906.07786](https://arxiv.org/abs/1906.07786)
24. Abiodun OI, Jantan A, Omolara AE, Dada KV, Mohamed NA, Arshad H (2018) State-of-the-art in artificial neural network applications: a survey. *Heliyon* 4(11):e00938
25. Nguyen D (1990) Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. In: Proceedings of the international joint conference on neural networks, vol. 3. pp 21–26
26. Drago GP, Ridella S (1992) Statistically controlled activation weight initialization (SCAWI). *IEEE Trans Neural Netw* 3(4):627
27. Enke D, Thawornwong S (2005) The use of data mining and neural networks for forecasting stock market returns. *Expert Syst Appl* 29(4):927
28. Castillo E, Guijarro-Berdiñas B, Fontenla-Romero O, Alonso-Betanzos A (2006) A very fast learning method for neural networks based on sensitivity analysis. *J Mach Learn Res* 7(Jul):1159
29. Mingyue Q, Cheng L, Yu S (2016) Application of the artificial neural network in predicting the direction of stock market index. In: 2016 10th international conference on complex, intelligent, and software intensive systems (CISIS). IEEE, pp 219–223
30. Qiu M, Song Y (2016) Predicting the direction of stock market index movement using an optimized artificial neural network model. *PLoS ONE* 11(5):e0155133
31. Wang H (2018) A study on the stock market prediction based on genetic neural network. In: Proceedings of the 2018 international conference on information hiding and image processing, pp 105–108
32. Kim KJ, Han I (2000) Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert Syst Appl* 19(2):125
33. Abraham A, Nath B, Mahanti PK (2001) Hybrid intelligent systems for stock market analysis. In: International conference on computational science. Springer, pp 337–345
34. Hadavandi E, Shavandi H, Ghanbari A (2010) Integration of genetic fuzzy systems and artificial neural networks for stock price forecasting. *Knowl Based Syst* 23(8):800
35. Csáji BC et al (2001) Approximation with artificial neural networks. Faculty of Sciences, Eötvös Loránd University, Hungary 24(48):7
36. Kara Y, Boyacıoğlu MA, Baykan ÖK (2011) Predicting direction of stock price index movement using artificial neural networks and support vector machines: the sample of the Istanbul Stock Exchange. *Expert Syst Appl* 38(5):5311
37. Chorowski J, Zurada JM (2015) Learning understandable neural networks with nonnegative weight constraints. *IEEE Trans Neural Netw Learn Syst* 26(1):62
38. Thakkar A, Lohiya R (2020) Attack classification using feature selection techniques: a comparative study. *J Ambient Intell Humaniz Comput* 12:1–18
39. Benesty J, Chen J, Huang Y, Cohen I (2009) Pearson correlation coefficient. In: Noise reduction in speech processing. Springer, pp 1–4
40. 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 Syst Appl* 42(1):259
41. NSE (1992) NSE-National Stock Exchange of India Ltd., NSE-National Stock Exchange of India Ltd. <http://www.nseindia.com/>. Last Accessed 31, Oct 2018
42. Chollet F et al (2015) Keras. <https://keras.io>
43. Kingdom S (2019) Wilcoxon signed-rank test. http://www.statskingdom.com/175wilcoxon_signed_ranks.html. Last Accessed 21 Mar 2019
44. Abdi H (2010) Coefficient of variation. *Encycl Res Des* 1:169
45. Thakkar A, Chaudhari K (2020) Fusion in stock market prediction: a decade survey on the necessity, recent developments, and potential future directions. *Inf Fusion* 65:95
46. Chaudhari K, Thakkar A (2021) iCREST: international cross-reference to exchange-based stock trend prediction using long short-term memory. In: International conference on applied soft computing and communication networks. Springer, pp 323–338. https://doi.org/10.1007/978-981-33-6173-7_22
47. Thakkar A, Chaudhari K (2021) A comprehensive survey on deep neural networks for stock market: the need, challenges, and future directions. *Expert Syst Appl* 177:114800. <https://doi.org/10.1016/j.eswa.2021.114800>
48. Thakkar A, Chaudhari K (2020) Predicting stock trend using an integrated term frequency-inverse document frequency-based feature weight matrix with neural networks. *Appl Soft Comput* 96:106684

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.