Daily stock market forecasting using kernel principal component analysis, support vector regression, and teaching learning based optimization

Mohammed Siddique
Dept. of Mathematics,
Centurion University of Technology and Management, Odisha, India
siddique@cutm.ac.in

Sabyasachi Mohanty
Dept. of Computer Sc. & Engg.,
Centurion University of Technology and Management, Odisha, India
pikunimohanty@gmail.com

Debdulal Panda
Dept. of Mathematics, KIIT University, Bhubaneswar, India
debdulalpanda@yahoo.co.in

Abstract: This paper presents an extremely precise prediction method which improved the decision of the investors on daily direction of the stock market. Among the studies that focus on daily stock market forecasting, the hybrid machine learning techniques are more appreciated than the conventional data mining procedures. With an intent to produce such a model with more accurate predictions, this paper analyzes a series of technological indicators used in usual studies of the stock market and uses kernel principal component analysis (KPCA), which is a feature extraction algorithms, along with support vector regression (SVR) and teaching learning based optimization (TLBO) algorithm. Feature extraction is such a procedure that can remove the unnecessary and unrelated factors, and reduce the dimension of the input variables from the original dataset. The feasibility and efficiency of the proposed KPCA-SVR-TLBO hybrid model was applied to forecast the daily opening prices of stock index of TATA Steel. The performance of the proposed approach is evaluated with 4143days daily transactional data of Tata Steels stocks price available at Bombay Stock Exchange (BSE). We compared our results with SVR-PSO and OFS-SVR-TLBO hybrid models.The experimental results reflect that the new hybrid model incorporating KPCA is more practicable and performs better than the competing ones.

Keywords: Daily stock market; Feature extraction; Kernel principal component analysis; Support vector regression; Teaching learning based optimization.

1. Introduction:

Stock market is one of exciting and demanding monetary activities for individual investors, and financial analysts. Prediction of stock price on day to day basis has become a crucial issue for stock investors and brokers. The stock price is based on the state of market stability and the stock market is able to influence the day to day life of the common people. But, it is very difficult to find the best time to trade the stock as many constraints affect the stock market. The major aspect of the stock market is uncertainty due to the various factors such as political events, economic conditions and the demand of the traders. This uncertainty makes the scenario very complicated to predict the daily stock price. The best approach to forecast the stock price is by reducing the level of uncertainty by applying feature extraction method.Even though a lot of techniques have been developed for prediction of stock price, building an efficient model, which improves the prediction accuracy, can attract and put a positive impact to the investors. In forecasting of stock price feature extraction technique is a key research area. It evaluates a large number of factors that are measured important to the stock market, and gives the most significant ones for people to represent the market trend. In order to extract major features in the stock market, researches on effective feature extraction methods are really needed. Kernel principal component analysis (KPCA) (L.J.Cao, Chua, Lee, 2003) is one such technique that has outperformed principal component analysis (PCA) due to its non-linearity nature of feature extraction.

Researchers use various machine learning and artificial intelligent approaches to forecast future trends or price. In the current scenario, support vector regression (SVR) (Vapnik, 1995; Gilbert, 2007; Haykin, 2009), is one of the leading methodologies for prediction in order to resolve pattern recognition and regression issues and found to be acceptable due to its accuracy level among the competing methods. The kernel function in SVR also enhances the generalization of the model and this efficiency is highly influenced by the regularization and kernel parameters of SVR. In order to determine appropriate values for the hyper-parameters of SVR, a parameter optimization technique is required (Pai& Hong, 2006; Hong, 2011; S. Das & S. Padhy, 2015; Chen, Shen, & Huang, 2016). Teaching learning based optimization (TLBO) method (Rao, Savsani, &Vakharia, 2011; Rao, Savsani, & Balic, 2012; Rao &Patel, 2013) being a population-based optimization algorithm outperformed many nature-inspired optimization techniques such as particle swarm optimization (PSO), and artificial bee colony (ABC). The most suitable feature of TLBO is non-requirement of tuning of parameters (Rao et al., 2011; Rao & Patel, 2013; Rao, 2016).In this piece of research work, a hybrid machine learning model consisting of kernel principal component analysis, support vector regression and teaching learning based optimization is designed. The model, named as KPCA-SVR-TLBO, is applied for the daily stock market forecasting of Tata Steel Limited. Technical indicators used in this analysis are calculated from the historical trading data. The role of KPCA in the hybrid model is non-linear feature extraction that improves the efficiency of the model. While SVR is at the core of the prediction mechanism, TLBO helps in tuning of the hyper-parameters of SVR.The KPCA-SVR-TLBO modelis found to perform better than the previously designed hybrid models, i.e., SVR-PSO and OFS-SVR-TLBO.

The rest of this paper is organized as follows. In Section-2,literature review is mentioned and a brief explanation of KPCA, SVR and TLBO are given in Section 3. Then in Section-4, the methodology and the process involved in the hybrid model under study, i.e., KPCA-SVR-TLBO, is described. In Section-5, experimental analysis are presented and finally, the paper is concluded in Section-6.

2. Literature Review:

In this proposed paper we have introduce the feature extraction techniques. Feature extraction techniques has massively applied on the field of machine learning models and also in pattern recognition. Many authors are used KPCA with the machine learning modeds. In 2014, the authors G. Chandrashekar & F. Sahin reduced features by using the feature selection techniques. In 2011, the authors M. Khashei & M. Bijari has applied a hybrid ARIMA-ANNs models to forecast time series real data set. In 2004, the authors K.J. Kim & W. B. Lee proposed a genetic algorithm based feature transformation model with artificial neural network to forecast the stock market index. In 2013, the authors S. M. Vieira et al. analyzed that using modified binary PSO algorithm, the parameters of SVM can be optimized correctly.In 2009, the authors C.J. Lu, T.S. Lee & C.C. Chiu presented a two-stage ICA–SVR prediction model and applied to the financial time series data of Nikkei 225 and TAIEX index. In 2017, the authors A.Tharwat, A. E. Hassanien & B.E. Elnaghi proposed a Bat Algorithm based SVM model (BA-SVM), which search the parameters of SVM to minimize the error. In 2014, the authors Y. Zhang, D.Gong, Y. Hu & W. Zhang Proposed a binary PSO algorithm to solve feature selection problems. In 2009, the authors Cheng-Lung Huang & Cheng-Yi Tsai proposed a hybrid SOFM-SVR model which reduced the training time and improves the prediction accuracy of prediction in the financial stock index. In 2011, the authors Y. Kara, M.A. Boyacioglu & O. K.

Baykan analyzed the direction of movement of stock price of the Istanbul Stock Exchange using ANN and SVM. In 2018, the authors B.M.Henrique, V. A. Sobreiro & H. Kimura

developed a random walk-based SVR model to predict financial market. In 2015, the authors J. Patel, S. Shah, P. Thakkar & K. Kotecha proposed a two stage fusion approach with Support Vector regression model perform better results than a single stage ANN with random forest model. In 2016, the authors XiaoLi Zhang, XueFeng Chen & ZhengJia He optimized the SVR parameter using ant colony optimization algorithim. In 2015, the authors P.C. Chang & J.L.Wu improved the prediction performance of stock trading using kernel principal component analysis technique in SVR model. In 2017, the authors M.Siddique, D.Panda, S. Das & S.K. Mohapatra proposed an ANN, PSO hybrid model to predict stock price of Microsoft and yahoo. In 2015, the authors M. Ballings, D. V. Poel, N. Hespeels & R. Gryp compared the performance of Random Forest, Adaboost and Kernel Factory against Neural Networks, Support Vector Machines, K-Nearest Neighbors and Logistic Regression for prediction of stock price. In 2003, the authors L.J. Cao, K.S. Chua,W. K. Chong, H.P. Lee , Q. M. Gu and in 2014, F. Kuang, W. Xu, S. Zhang, were compared the performance of feature extraction techniques PCA, KPCA and ICA by applying to SVM model and concluded that KPCA perform better than PCA and ICA. In 2013, the authors Y. He, K. Fataliyev and L.Wang analyzed the performance of feature selection methodology of PCA and Sequential Forward Selection (SFS) with SVR and concluded that PCA perform better accuracy than SFS. In 2016, the authors I.Jaffel, O.Taouali, M.F. Harkat and H. Messaoud analyzed that moving window reduced kernel principal component analysis MW-RKPCA are perform better results than and moving window kernel principal component analysis MW-KPCA applied to monitoring the nonlinear dynamic system.

3. Methodology Used:

3.1 Kernel Principle Component Analysis:

Kernel principle component analysis (KPCA) is a statistical analysis method to remove the principle inconsistency of the data. The real meaning of this method is to expose the nature of data by resolving the main factors. KPCA is mainly used for dimensionality reduction of data. The calculation purpose of KPCA is to make a projection from the main components of high-dimensional data, onto a lower dimensional space. KPCA is a very familiar and well known data analysis technique and used to reduce the dimension of a large data set of variables to a small data set. New variables are able to be produced through transformation of original variables. Number of variables is less and most of the information is still retained. These new variables are called principal components. Extraction of a subset of variables from a larger data set depends upon the highest correlations of the principal component with the original variables. KPCA has several application areas in science and engineering. Consider a non linear transformation ϕ(x) from the original d-dimensional feature space to a D-dimensional feature space, where usually D >> d, then each data point x_i is projected to a point $\phi(x_i)$. Vapnic-chervononkis stated that kernel mapping provides greater classification power by transferring the dimension of input space into a higher dimensional space. KPCA extends conventional PCA to a high dimensional feature space using kernel trick and extracts finite number of non linear principal components. KPCA is useful when input data lie on a low dimensional nonlinear hyperplane.

In this method the input patterns $x_i \in R^d$ for I = 1, 2, 3 …….N (where N is the number of input samples) are first mapped onto a space H^D with more than dimension d using a nonlinear mapping $R^d \to H^D$. Their images $\phi(x_i)$ are projected along the orthonormal eigen vectors of the covariance matrix of $\phi(x_i)$. Those projections only involved the

inner product of $\phi(x_i)$in H^D. We assume that the projected new features have zero mean.
$\frac{1}{N}\sum_{i=1}^{n}\phi(x_i) = 0$ (1)
The projected features of covariance matrix can be calculated by
$C = \frac{1}{N}\sum_{i=1}^{n}\phi(x_i)\phi(x_i)^T$ (2)
The eigen values and eigen vectors of covariance matrix C are given by
$CV_n = \lambda_n V_n$, where n = 1, 2, 3………D (3)
From equations 2 and 3 we have $\frac{1}{N}\sum_{i=1}^{n}[\phi(x_i)\phi(x_i)^T]\, V_n = \lambda_n V_n$ (4)
As all the solutions V_n with λ = 0 lies in the span of $\phi(x_1), \phi(x_2)$ … … … … … … $\phi(x_n)$ so $V_n = \sum_{i=1}^{n}\alpha_{ni}\,\phi(x_i)$, substitute this value of V_n in equation 4 we get
$\frac{1}{N}\sum_{i=1}^{n}[\phi(x_i)\phi(x_i)^T]\ \sum_{j=1}^{n}\alpha_{nj}\,\phi(x_j) = \lambda_n \sum_{j=1}^{n}\alpha_{nj}\,\phi(x_j)$ (5)
Define the kernel function K (x_i, x_j) = $\phi(x_i)^T\phi\ (x_j)$ (6)
$\frac{1}{N}\sum_{i=1}^{n}\phi(x_i)\sum_{j=1}^{n}\mathrm{K}\,(x_i, x_j)\alpha_{nj} = \lambda_n \sum_{j=1}^{n}\alpha_{nj}\,\phi(x_j)$ (7)
Multiplying $\phi(x_k)^T$ from left of equation 7 we get
$\frac{1}{N}\sum_{i=1}^{n}\phi(x_k)^T.\phi(x_i)\sum_{j=1}^{n}\mathrm{K}\left(x_i, x_j\right)\alpha_{nj} = \lambda_n \sum_{j=1}^{n}\alpha_{nj}\,\phi(x_k)^T\ \phi(x_j)$ (8)
We can use the matrix notation $\frac{1}{N}K^2\alpha_n = \lambda_n\, K\, \alpha_n$ so $K^2\alpha_n = NK\,\lambda_n\,\alpha_n$, by removing the factor K from both sides we get $K\,\alpha_n = N\,\lambda_n\,\alpha_n$, where $K_{i,j} = \mathrm{K}\left(x_i, x_j\right)$, $\alpha_k = [\alpha_{n1}, \alpha_{n2}\ \ldots\ldots\ \alpha_{nN}]^T$and α_n is the N-dimensional column vector of α_{nj}.
$\alpha_n$can be solved by $K\,\alpha_n = \lambda_n N\,\alpha_n$ and the resulting kernel principal components can be calculated using $y_n(x) = \phi(x)^T V_n = \sum_{i=1}^{n}\alpha_{ni}\mathrm{K}\,(\mathrm{x}, x_i)$
If the projected data set $\phi(x_i)$ does not have zero mean, we can introduce Gram matrix $\breve{K}$to substitute the kernel matrix K. The Gram matrix is given by $\breve{K} = K - 1_N\,K - K\ 1_N + 1_N K_{\frac{1}{N}}$, where 1_N is the $N \times N$
Matrix with each element is $\frac{1}{N}$. We can directly construct the kernel matrix from the training data set $\{x_i\}$.We use Gaussian kernel K(x, y) = exp $\left(\frac{||x-y||^2}{2\sigma^2}\right)$ with parameter σ.

3.2 Support Vector Machine for Regression :

The principle of support vector machine (SVM) for regression is a supervised machine learning method developed by Vapnik and Cortes (1995). SVM makes a decision boundary by which the greater part of the data points of the relevant kind falls on the same side of the boundary. Let us consider the data points of an n-dimensional feature vector space X = (x_1, x_2, ... … ... ,x_n), from which we construct a hyper plane
$\alpha_0 + \sum_{j=1}^{n}\alpha_j x_j = 0$, where the boundary of the optimal hyperplane can be obtained by the maximizing the distance from any point to the plane. The maximum margin hyperplane (MMH) separates the similar types of data points. The necessary feature is that only neighboring points to the boundary of the hyperplane are participated in selection keeping all other points as it is. These points are well-known as the support vectors, and support vectors are separated in respective class by a hyperplane, which is called the Support Vector Classifier (SVC). The inner products of support vector classifier are weighted by their labels, and it helps to maximize the distance from support vectors to the hyperplane.
The fundamental perception of SVM is to maximize the hyperplane margin in the feature space. The Support Vector Machine for Regression (SVR) model is a supervised machine learning technique developed by Cortes and Vapnik et al. [1995], is described below.

Given a sample data-set S = (x_1; y_1); (x_2; y_2); ::: ::: (x_l; y_l) representing l input-output pairs, where each $x_i \in X \subset \mathbb{R}^n$, where X represents the n dimensional input sample space and matching target values $y_i \in Y \subset \mathbb{R}$ for i = 1, 2, … … l, where l is the size of the training data. The purpose of this regression problem is to construct a function f: $\mathbb{R}^n \rightarrow \mathbb{R}$, to approximate the value of y for unseen data x, which was not participated in training sample. By taking a nonlinear function ϕ, the input data is mapped from $\mathbb{R}^n$ to a high dimensional space $\mathbb{R}^m$, m> n, and consequently the estimation function f is defined as $f(x) = (w^T \phi(x)) + b$ (9)

where $w \in R^m$ is the regression coefficient vector, $b \in \mathbb{R}$, is the bias or threshold value. The main intention of the support vector regression is to build a function f which has the most ϵ-deviation from the target y_i. We need to find w and b for which the value of f(x) can be obtained by minimizing the risk. $R_{reg}(w) = \frac{1}{2}\|w\|^2 + C\sum_{j=1}^{l} L_\epsilon(y_j, f(x_j))$ (10)

where C is the user defined penalty factor, which determines the trade-off between the training error and the penalizing term $\|w\|^2$ and $L_\epsilon(y_j, f(x_j))$ is the ϵ-intensive loss function originally proposed by Vapnik et al, which is defined as

$$L_\epsilon\left(y_j, f(x_j)\right) = \begin{cases} |y_j - f(x_j)| - \epsilon & , |y_j - f(x_j)| \geq \epsilon \\ 0, & |y_j - f(x_j)| < \epsilon \end{cases} \quad (11)$$

The minimum risk functional equation (10) can be reformulated by introducing non-negative slack variables γ_i and ξ_i as

$$R_{reg}(w, \gamma_j, \xi_j) = \text{Minimize } \frac{1}{2}\|w\|^2 + C\sum_{j=1}^{l}(\gamma_j + \xi_j) \quad (12)$$

$$\text{subject to constraints} \begin{cases} y_j - W^T x_j - b \leq \epsilon + \gamma_j \\ W^T x_j + b - y_j \leq \epsilon + \xi_j \\ \gamma_j, \xi_j \geq 0 \end{cases} \quad (13)$$

where $\frac{1}{2}\|w\|^2$ is the regularization term preventing over learning $(\gamma_j + \xi_j)$ is the realistic risk and C > 0 is the regularization constant, that controls the trade-off between the empirical risk and regularization term.

By introducing Lagrange multipliers α_j, β_j, μ_j and η_j the quadratic optimization problem (12) and (13) can be formulated as

$$L = \frac{1}{2}\|w\|^2 + C\sum_{j=1}^{l}(\gamma_j + \xi_j) - \sum_{j=1}^{l}\alpha_j(\epsilon + \gamma_j - y_j + W^T x_j + b) - \sum_{j=1}^{l}\beta_j(\epsilon + \xi_j + y_j - W^T x_i - b) - \sum_{j=1}^{l}(\mu_j\gamma_j + \eta_j\xi_j) \quad (14)$$

The dual of the corresponding optimization problem (12) and (13) is represented as Maximize $-\frac{1}{2}\|w\|^2 + C\sum_{j,k=1}^{l}(\alpha_j - \beta_j)(\alpha_k - \beta_k)(x_j)^T x_k - \epsilon\sum_{j=1}^{l}(\alpha_j + \beta_j) + \sum_{j=1}^{l}(\alpha_j - \beta_j)$ Subject to constraints $\begin{cases} \sum_{j=1}^{l}(\alpha_j - \beta_j) = 0 \\ \alpha_j, \beta_j \in [0, C] \end{cases}$

By changing the equation $w = \sum_{j=1}^{l}(\alpha_j - \beta_j)x_j$, the function f(x) can be written as

$$f(x) = \sum_{j=1}^{l}[(\alpha_j - \beta_j)x_j]^T \phi(x) + b \quad (15)$$

accordingly by applying Lagrange theory and Karush-Kuhn-Tucker condition, the general support vector regression function can be expressed as

$f(x) = \sum_{j=1}^{l}(\alpha_j - \beta_j)K(x_j, x_j) + b$, where $C(x_j, x_k)$ known as Kernel function. The value of kernel function is equal to the inner product of x_j and x_k in the feature space $\phi(x_j)$ and $\phi(x_k)$ such that $K(x_j, x_k) = \phi(x_j).\phi(x_k)$.

3.3 Teaching Learning Based Optimization (TLBO)

The Teaching Learning based Optimization (TLBO) Algorithm is one of the population based optimization technique which is quite different from other population based algorithm like either evolutionary type or swarm intelligence type. Genetic Algorithm, Evolution Strategy, Evolution Programming, Artificial Immune Algorithm, etc are evolutionary type and Particle Swarm Optimization (PSO), Ant Colony Optimization

(ACO), Artificial Bee Colony (ABC) are examples of some of swarm intelligence type. All the above mentioned algorithms are probabilistic algorithms and needs common controlling parameters such as population size, number of generations, etc. Except common controlling parameters, each of evolutionary type as well as swarm intelligence type uses their own algorithm specific controlling parameters. Mentioning a few, mutation probability, crossover probability, selection operators are algorithm specific controlling parameters of Genetic Algorithm (GA). Similarly inertia weights, social and cognitive are algorithm specific controlling parameters for Particle Swarm Optimization (PSO); onlooker bees, employed bees, scout bees and limit are algorithm specific controlling parameters for Artificial Bee Colony (ABC) and like this other algorithm have also specific controlling parameters. The performances of above algorithms are greatly affected by proper choosing of algorithm specific controlling parameters. The improper choice of algorithm specific controlling parameters, leads to either increases computational effort or yields the local optimal solution instead of global optimal solution. Considering the above critics and mimics to one of natural system of learning, Rao et al. (2011) introduced the Teaching Learning Based Optimization (TLBO) algorithm. It does not required any algorithm- specified parameters but only common controlling parameters such as population size and the number of generations.

The TLBO algorithm consists of two phases. One is Teacher and another is Learner Phase

Teacher Phase:

It is the first part of TLBO where teacher will teach and the learners will learn. During this time, a teacher will try to raise his competence level through average result of the class in the subject taught by him/her.

Let at the ith iteration, 'm' number of subjects (i.e design variables) taught to a learner, 'n' number of learners (i.e population size, k=1,2,3,......, n) are there and M_{ij} be the mean result of the learners in a particular subject 'j'(j=1,2,3,......., m).The best overall result $X_{total-kbest,j}$ considering all the subjects together obtained in the entire population of learners can be considered as the result of best learner $kbest$ .The best learner identified is considered as teacher for next iteration. The difference between the existing mean result of each subject and the corresponding result of to be teacher i.e. best learner can be stated as, $Difference\_Mean_{j,k,i} = r_i(X_{j,kbest,i} - T_F M_{j,i})$ (16)

Where, $X_{j,kbest,i}$ is the result of the best learner in subject j. T_F is the teaching factor which decides the value of mean to be changed ,and r_i is the random number in the range [0,1]. The Value of T_F can be either 1 or 2. The value of T_F can be decided randomly with equally likely as $T_F = round[1 + rand(0,1)\{2 - 1\}]$, (17)

Where T_F is not a parameter of the TLBO algorithms as it is not an input to the algorithm rather its value is randomly decided by the above algorithm (17).It is concluded from large number of experiments that the algorithm gives better result for the value of T_F between 1 and 2 and far better for the value either 1 or 2. So it is suggested to take either 1 or 2 value for teaching factor T_F according the rounding values given by (17). The existing solution will be up-to-dated in the teacher phase according to following expression. $X'_{j,k,i} = X_{j,k,i} + Difference\_Mean_{j,k,i}$, Where, $X'_{j,k,i}$ is the updated value of $X_{j,k,i}$. The updated value of $X'_{j,k,i}$ is acceptable if it gives better values for the function. All accepted function values, at the end of teacher phase become input to learner phase.

Learner Phase:

In this phase the learners increase their knowledge by interaction among themselves randomly. A learner gain knowledge if the counterpart learner has more knowledge. Let us consider a population of seize 'n' and choose two learners P and Q randomly such that $X'_{total-p,i} \neq X'_{total-p,i}$, where $X'_{total-p,i}$, $X'_{total-Q,i}$ are updated function values of $X_{total-p,i}$, $X_{total-Q,i}$ at the end of teacher phase. $X''_{j,p,i} = X'_{j,p,i} + r_i(X'_{j,p,i} - X'_{j,Q,i})$, If $X'_{total-p,i} < X'_{total-Q,i} X''_{j,p,i} = X'_{j,p,i} + r_i(X'_{j,Q,i} - X'_{j,P,i})$, if $X'_{total-Q,i} < X'_{total-Q,i}$, Where $X''_{j,p,i}$ is accepted if it gives a better function value. The above two are for minimization problems. In case of maximization problem they can be modified as $X''_{j,p,i} = X'_{j,p,i} + r_i(X'_{j,p,i} - X'_{j,Q,i})$, if $X'_{total-p,i} > X'_{total-Q,i}$ and $X''_{j,p,i} = X'_{j,p,i} + r_i(X'_{j,Q,i} - X'_{j,P,i})$, if $X'_{total-Q,i} > X'_{total-Q,i}$.

4. Proposed Model KPCA-SVR-TLBO

The KPCA-SVR-TLBO hybrid modelis comprising of three main components. They are kernel principal component analysis (KPCA), support vector regression (SVR), and teaching learning based optimization (TLBO). Here, the role of KPCA is to extract features from the provided dataset, SVR to address the prediction mechanism and TLBOto optimize the free parameters of support vector regression. In this application of machine learning, the features under consideration may consists of irrelevant or dependent features and hence feature extraction can provide a newly created set of features by removing unrelated or unnecessary attributes from the existing data. In the practical sense, this decreases the number of features in order to improve the accuracy of the model. In KPCA-SVR-TLBO hybrid model, we have applied the kernel principal component analysis (KPCA) to reduce the dimension of the input data.As feature extraction is a technique to extract a feature subset from all the input features to prepare the constructed model better, it is desired that KPCA to be applied at the initial stages to the whole dataset to simplify and rearrange the original data structure.

Proper selection of kernel type, regularization parameter, and the ϵ-insensitive loss of SVR greatly influences the efficiency of the prediction model. Therefore, radial basis function (RBF) kernel is considered due to the nonlinearity nature of time series dataset under study. Mathematically, RBF kernel is defined as $K(u, v) = e^{-\gamma||u-v||^2}$, where $\gamma = \frac{1}{2\sigma^2}$.The hyper-parameters of SVR that requires to be optimized by TLBO are cost and gamma. The flowchart shown in Figure-1 describes the important steps involved in the prediction mechanism of the hybrid model. The first component that receives the dataset is KPCA. Once the task of feature extraction is complete, the generated features are forwarded to the SVR, which is optimized by TLBO.

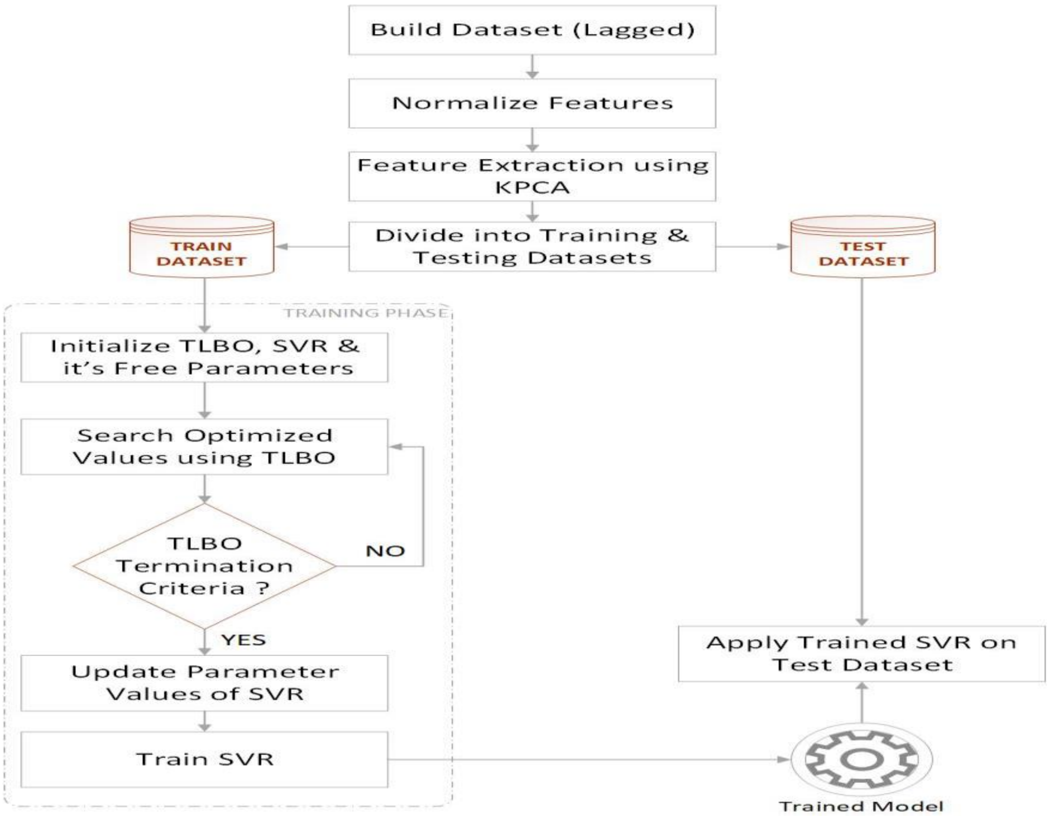

Figure-1: Flowchart of KPCA-SVR-TLBO model

For this study, the data comprises of seven features (mentioned in Table-1) that are daily stock market recordings of time-series data, along with lagged (past period) values of the last five days.

Table-1: Features

Sl.	Features	Description
1	Opening Price	Opening price ofstock exchange on a trading day.
2	Highest Price	Highest price on a trading day.
3	Lowest Price	Lowest price on a trading day.
4	Closing price	Closing price ofstock exchange on a trading day.
5	No. of Shares	Total quantity of shares traded on a trading day.
6	No. of Trades	Total number of trades happened on a trading day.
7	Turnover	Total value of stock traded on a trading day.

Normalization of data has been adopted to get rid of the numerical difficulties during computation and overcome the dominance of features with greater numerical ranges over smaller numerical ranges. Here, normalization has been achieved in the pre-processing stage by linearly scaling to [0, 1] using the formula mentioned below.

$$NV_i = \frac{A_i - A_{\min\_i}}{A_{\max\_i} - A_{\min\_i}} \text{ , } for i = 1, 2, 3, \ldots l$$

where, A_i is the actual value, $A_{\max\_i}$ and $A_{\min\_i}$ are the maximum and minimum values respectively, of the i-th feature, l is the total number of data points, and NV_i is the corresponding normalized value.

The data generated after the application of KPCA is divided into 2 sets i.e., training and testing, with a predefined proportion. Then, the parameters of TLBO and SVR are initialized, following which, the model trainingprocedurebeginsby the application of the training dataset. The optimized values of the free parameters of SVR are obtainedusing TLBO and thetermination criteria determines when to terminate the process of searching. Once the values of the hyper-parameters are obtained, the same is applied to build the SVR. Now to evaluate the performance of the model trained using the training dataset, it is applied to the testing dataset.

5. Experimental Results and Discussions

5.1 Evaluation Criteria

To evaluate the performance of the model addressing the daily stock market forecasting, we have considered 3 standard statistical metrics, i.e., mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE), whose definitions are mentioned in Table 2. As MAE, RMSE, and MAPE indicate variants of the differences between the estimatedand actual values, it should be noted that smaller the error value, better the performance.

Sl.	Metric	Definition
1	Mean Absolute Error (MAE)	$\frac{1}{l}\sum_{i=1}^{l} \lvert y_i - d_i \rvert$
2	Root Mean Squared Error (RMSE)	$\sqrt{\frac{1}{l}\sum_{i=1}^{l} (y_i - d_i)^2}$
3	Mean Absolute Percentage Error (MAPE)	$\frac{1}{l}\left(\sum_{i=1}^{l} \left\lvert \frac{y_i - d_i}{d_i} \right\rvert\right) 100$
where, *l* is the total number of instances or records under evaluation, d_i is the desired output value, i.e., actual or true value of interest, and y_i is the estimated value obtained using a prediction model.		

5.2 Comparison of Results

In this research work, the performance of our hybrid model i.e., KPCA-SVR-TLBO is compared with our earlier hybrid models, i.e., SVR-PSO and OFS-SVR-TLBO using the same set of financial data. Presently in the proposed model we have incorporated kernel principle component analysis (KPCA) as the feature extraction technique along with support vector regression (SVR) whose parameters are optimized by teaching learning based optimization (TLBO) technique.Here, the total dataset are divided into two parts, i.e., training and testing datasets, after the application of the feature extraction mechanism using KPCA.The KPCA is configured with kernel type as RBF and 95% to retain enough PC attributes to account for the proportion of variance in the original data. The original dataset was consisting of 35 features. The application of KPCA generated a dataset of 5 features. The training and testing datasets are applied to the above hybrid model for training and testing phases for prediction of the next day opening price. Out of 4143 numbers of data of Tata Steel (from 24-July-2001 to 19-March-2018) three-fourth of the data are used for building the training dataset and rest one-fourth for the testing dataset. For better generalization, cross validation is also applied. Errors evaluated with MAE, RMSE, and MAPE in training phase are 0.194310815, 0.30439309, and 0.143517405 % (approx) respectively and the errors in testing phase are 0.233326719, 0.351664082, 0.271837721 % (approx.) respectively. The Table-2 shows the error measures found for all the models, i.e., previously obtained results of PSO-SVR and OFS-SVR-TLBO and presently proposed KPCA-SVR-TLBO model. This empirical study shows that KPCA-SVR-TLBO performed better than PSO-SVR and OFS-SVR-TLBO in all the three evaluation criteria.

Table-2: Comparison of Performance of PSO-SVR, OFS-SVR-TLBO and KPCA-SVR-TLBO Models on Training and Testing Datasets

		Models		
		PSO-SVR	**OFS-SVR-TLBO**	**KPCA-SVR-TLBO**
Training	**MAE**	2.760213993	0.856451993	0.194310815
	RMSE	5.741340821	1.895755314	0.30439309
	MAPE	0.68994578 %	0.457514028 %	0.143517405 %
Testing	**MAE**	2.929112587	1.202996162	0.233326719
	RMSE	6.494903279	2.043010993	0.351664082
	MAPE	0.708516926 %	0.493052126 %	0.271837721 %

The Figures-2 to 5 shows the comparison of the actual stock value and prediction of stock values using KPCA-SVR-TLBO. It also includes the absolute error.

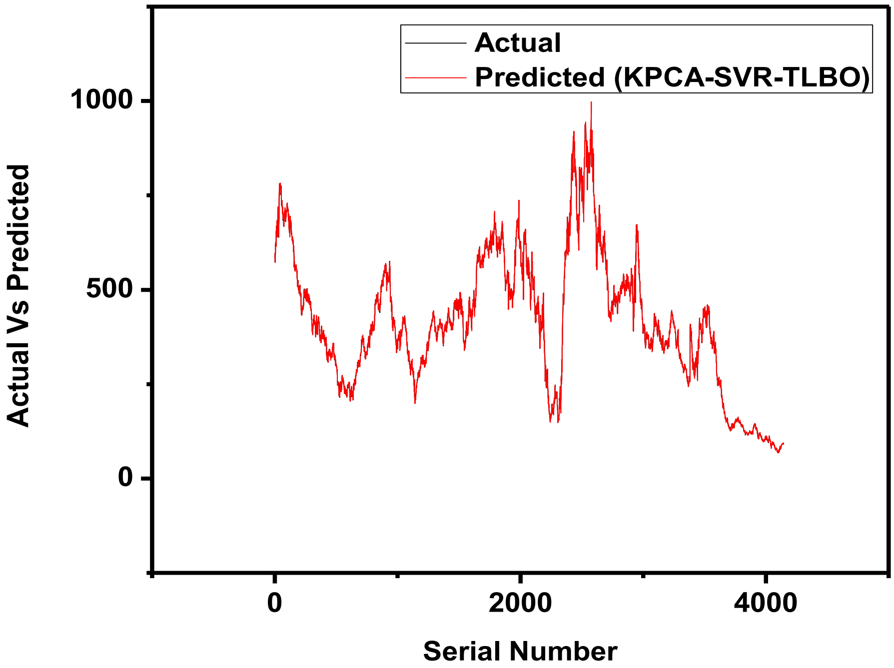

Figure-2: Actual Verses Prediction of KPCA-SVR-TLBO on complete dataset

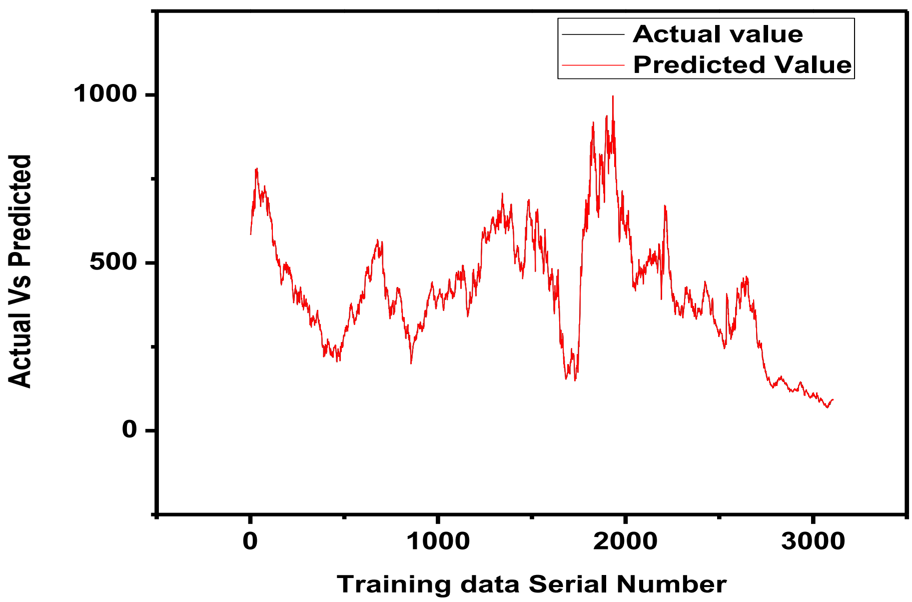

Figure-3: Actual Verses Prediction of KPCA-SVR-TLBO on training dataset.

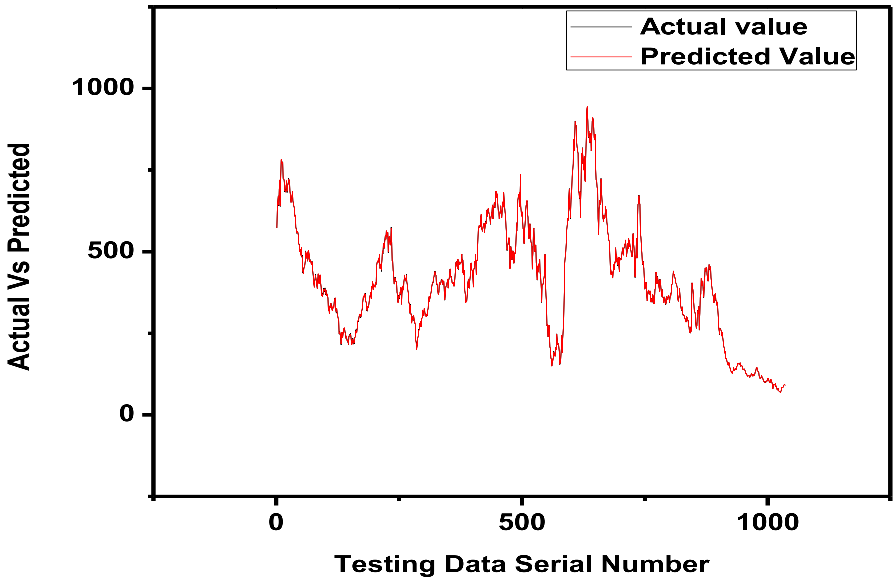

Figure-4: Actual Verses Prediction of KPCA-SVR-TLBO on testing dataset.

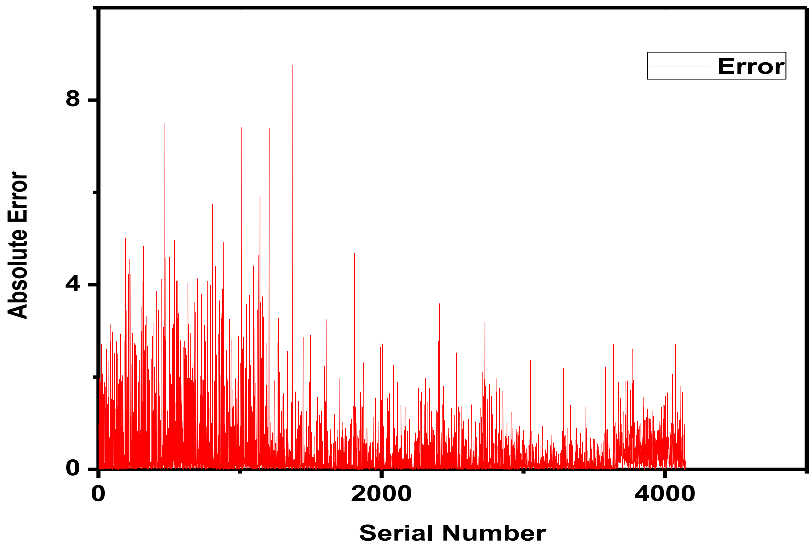

Figure-5: Absolute error of KPCA-SVR-TLBO.

6. Conclusion:

The stock market is an important and inter-connected economic international business. To address the daily stock market forecasting we proposed a machine learning based hybrid model using kernel principal component analysis (KPCA), support vector regression (SVR) and teaching learning based optimization (TLBO).The dataset was consisting of 35 attributes and the application of KPCA with RBF kernel generated a dataset with 5 features. The testing results obtained from the empirical study shows 0.27 % (approx.) mean absolute percentage error (MAPE). Empirical results showed that KPCA-SVR-TLBOhybrid model also outperformed PSO-SVR, and OFS-SVR-TLBO in MAE, RMSE, and MAPE.It shows that the proposed model (KPCA-SVR-TLBO) enhances the performance of the prediction model due to the presence of the feature extraction method, i.e., kernel principal component analysis (KPCA) and this proposed model can be used for taking better decision and more accurate predictions for financial investors on daily stock market forecasting.

7. Reference

[1] Girish Chandrashekar & Ferat Sahin, A survey on feature selection methods, Computers and Electrical Engineering Vol. 40 **(2014)** pp.16–28.

[2] Mehdi Khashei & Mehdi Bijari, A novel hybridization of artificial neural networks and ARIMA models for time series forecasting, Applied Soft Computing Vol. 11 **(2011)** pp. 2664–2675.

[3] Kyoung-jae Kim & Won Boo Lee, Stock market prediction using artificial neural networks

with optimal feature transformation, Neural Comput & Applic vol. 13 **(2004)** pp. 255–260.

[4] Susana M. Vieira, Luis F. Mendonc, Gonc¸ alo J. Farinha & Joao M.C. Sousa , Modified binary PSO for feature selection using SVM applied to mortality prediction of septic patients, Applied Soft Computing Vol. 13 **(2013)** pp. 3494–3504.

[5] Chi-Jie Lu, Tian-Shyug Lee & Chih-Chou Chiu, Financial time series forecasting using independent component analysis and support vector regression,Decision Support Systems vol. 47 **(2009)** pp.115–125.

[6] Alaa Tharwat, Aboul Ella Hassanien & Basem E. Elnaghi, A BA-based algorithm for parameter optimization of Support Vector Machine, Pattern Recognition Letters vol. 93 **(2017)** pp. 13–22.

[7] Yong Zhang, Dunwei Gong,Ying Hu & Wanqiu Zhang, Feature selection algorithm based on bare bones particle swarm optimization, Neurocomputing vol. 148 **(2015)** pp. 150–157.

[8] Cheng-Lung Huang & Cheng-Yi Tsai, A hybrid SOFM-SVR with a filter-based feature selection for stock market forecasting, Expert Systems with Applications vol. 36 **(2009)** pp. 1529–1539

[9] Yakup Kara, Melek Acar Boyacioglu & Omer Kaan Baykan, 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 vol. 38 **(2011)** pp. 5311–5319.

[10] Bruno Miranda Henrique, Vinicius Amorim Sobreiro & Herbert Kimura, Stock price prediction using support vector regression on daily and up to the minute prices, The Journal of Finance and Data Science vol. 4 **(2018)** pp. 183-201

[11] Jigar Patel, Sahil Shah, Priyank Thakkar & K Kotecha, Predicting stock market index using fusion of machine learning Techniques, Expert Systems with Applications vol. 42 **(2015)** pp. 2162–2172.

[12] XiaoLi Zhang, XueFeng Chen & ZhengJia He, An ACO-based algorithm for parameter optimization of support vector machines, Expert Systems with Applications vol. 37 **(2010)** pp. 6618–6628.

[13] Pei-Chann Chang & Jheng-Long Wu, A critical feature extraction by kernel PCA in stock trading model, Soft Comput vol. 19 **(2015)** pp. 1393–1408.

[14] Mohammed Siddique, Debdulal Panda, Sumanjit Das, Sudhir Kumar Mohapatra, A hybrid forecasting model for stock value prediction using soft computing technique, International Journal of Pure and Applied Mathematics Vol. 117, No. 19 **(2017)**, pp. 357-363.

[15] Michel Ballings, Dirk Van den Poel, Nathalie Hespeels & Ruben Gryp, Evaluating multiple classifiers for stock price direction prediction, Expert Systems with Applications vol. 42 **(2015)** pp. 7046–7056

[16] Shom Prasad Das, Sudarsan Padhy, A novel hybrid model using teaching–learning-based optimization and a support vector machine for commodity futures index forecasting, Int. J. Mach. Learn. & Cyber. **(2015)** DOI 10.1007/s13042-015-0359-0

[17] Chien-Feng Huang, A hybrid stock selection model using genetic algorithms and support vector Regression, Applied Soft Computing vol. 12 **(2012)** pp. 807–818

[18] Sabyasachi Mohanty & Sudarsan Padhy, A novel OFS–TLBO–SVR hybrid model for optimal budget allocation of government schemes to maximize GVA at factor cost, Journal of Management Analytics, **(2017)** DOI: 10.1080/23270012.2017.1373262

[19] M.Siddique, Sabyasachi Mohanty & Debdulal Panda, A hybrid forecasting model for prediction of stock value of tata steel using support vector regression and particle swarm optimization, International Journal of Pure and Applied Mathematics, Vol. 119 No. 14 **(2018)**, pp. 1719-1727.

[20] Shom Prasad Das, N. Sangita Achary & Sudarsan Padhy, Novel hybrid SVM-TLBO forecasting model incorporating dimensionality reduction techniques, Appl Intell vol. 45 **(2016)** pp. 1148–1165.

[21] L.J.Cao, K.S. Chua,W.K.Chong, H.P. Lee , Q.M.Gu, A comparison of PCA, KPCA and ICA for dimensionality reduction in support vector machine, Neurocomputing vol. 55 **(2003)** pp. 321 – 336.

[22] Fangjun Kuang, Weihong Xu, Siyang Zhang, A novel hybrid KPCA and SVM with GA model for intrusion detection, Applied Soft Computing vol. 18 **(2014)** pp. 178–184.

[23] Y. He, K. Fataliyev, and L. Wang (2013), Feature Selection for Stock Market Analysis, Springer-Verlag Berlin Heidelberg **(2013)**, pp. 737–744,.

[24] Ines Jaffel, OkbaTaouali, M. Faouzi Harkat, Hassani Messaoud, Moving window KPCA with reduced complexity for nonlinear dynamic process monitoring, ISA transactions vol. 64 **(2016)** pp.184–192.

[25] Cortes, C., & Vapnik, V. Support-vector networks. Machine Learning, vol. 20 **(1995)**, pp.273–297

[26] Haykin, S. Neural networks and learning machines **(2009)** (3rd ed). New York: Pearson Education

[27] Trafalis, T. B., & Gilbert, R.C.Robust support vector machines for classification and computational issues. Optimization Methods and Software, vol. 22 **(2007)**, pp. 187–198.

[28] Pai, P. F., & Hong, W. C. Software reliability forecasting by support vector machines with simulated annealing algorithms. Journal of Systems and Software, vol. 79 **(2006)**, pp. 747–755.

[29] Hong, W. C. Electric load forecasting by seasonal recurrent SVR (support Vector regression) with chaotic artificial bee colony algorithm. Energy, vol. 36 **(2011)**, pp. 5568–5578

[30] Rao, R. V., Savsani, V. J., & Balic, J.Teaching–learning-based optimization algorithm for unconstrained and constrained real-parameter optimization problems. Engineering Optimization, vol. 44, No. 12, **(2012)** pp. 1447–1462.

[31] Pudil, P., Novovicova, J., & Kittler, J. Floating search methods in feature selection. Pattern Recognition Letters, vol. 15 No. 11 **(1994)**, pp. 1119–1125.