



Unveiling social network clans: improving genealogical clan classification with SVM neural classifiers and enhanced kernels

S. N. Deepa¹ · Karam Ratan Singh² · Arun Joram²

Received: 22 April 2024 / Accepted: 17 August 2024 / Published online: 12 September 2024
© Bharati Vidyapeeth's Institute of Computer Applications and Management 2024

Abstract In this study, we developed a variant of the support vector machine (SVM) neural classifier and utilized it to categorize clans in a genealogical dataset. For each of the five kernels, all four variants, twin SVM (TSVM), proximal SVM (PSVM), twin proximal SVM (TPSVM), and multi-class SVM (MCSVM) classifier are simulated and tested. The analysis of variance - radial basis function (ANOVA RBF) kernel outperformed all other SVM variants, in terms of classification accuracy with the lowest error value. Additionally, it is found that for the considered dataset, TPSVM neural classifier with ANOVA RBF Kernel generated 98.91% classification accuracy, and the TPSVM classifier has achieved the minimized mean square error (MSE) value of 0.00015. The Twin Proximal SVM classifier has produced enhanced classification accuracy with better precision and F1-score in comparison to all other developed and simulated SVM classifier models.

Keywords Machine learning · Clan classification · Support vector machine (SVM) · Social network model

1 Introduction

Analysis and understanding of genealogical networks may benefit significantly from machine learning. Individuals can be classified into different clans or groups utilising machine learning techniques such as Support Vector Machines (SVMs), decision trees, or neural networks based on naming patterns, geographic locations, or additional characteristics found in the genealogical data. Additionally, by learning from recognized patterns in the data, machine learning systems can infer connections between individuals, like parent–child, sibling, or cousin relationships.

Machine learning techniques have become growing in popularity as possible answers to this issue in recent years. SVM is a powerful supervised learning algorithm, which can be used for classification. It is an excellent choice for clan classification in genealogical data because it can handle both linear and non-linear relationships in high-dimensional feature spaces [1].

The process of categorizing data or items based on their features or characteristics is referred to as classification. Classification in machine learning is a type of supervised learning where an algorithm is trained on a labelled data set to predict the class or category of new data [2]. The basic goal of classification is to develop a model that can accurately categorize or label new data based on their features [3]. Determining the optimum hyperplane to divide various classes in the feature space is its primary goal. SVM attempts to identify the hyperplane that maximizes the margin between the two classes in the context of binary classification. Support vectors are the data points that have the closest connection to the hyperplane. The objective behind SVM is to employ kernel functions to transform the data into a higher-dimensional space where the classes are linearly separable.

S. N. Deepa, Karam Ratan Singh, and Arun Joram have contributed equally to this work.

✉ Karam Ratan Singh
karamratan7@gmail.com

S. N. Deepa
sndeepa@nitc.ac.in

Arun Joram
arunjoram23@gmail.com

¹ Department of Electrical Engineering, NIT, Calicut, Kerala 673601, India

² Department of Basic and Applied Science, NIT, Jote, Arunachal Pradesh 791113, India

A modification of the conventional SVM technique used for classification problems is the Twin Support Vector Machine (Twin SVM) [4]. It has been proposed as an approach for enhancing the performance of SVMs, especially when the data is imbalanced or contains noisy samples. Twin SVM was initially introduced in the 2007, by Jayadeva et al [5]. In a typical SVM, the objective is to select a hyperplane that maximizes the margin between classes while best classifying them. Twin SVM improves on this idea by employing two non-parallel hyperplanes for each class instead of just one [6, 7]. In order to maximize the distance between the two hyperplanes while simultaneously ensuring that they accurately classify the training data, these hyperplanes are improved. This may end up in better generalization as well as better management of datasets having imbalances [8]. As a variant of the support vector machine (SVM), the proximal support vector machine (PSVM) generates two non-parallel hyperplanes for classification [9]. In the proximal support vector machine (PSVM), the data points are assigned according to their closeness to the hyperplanes that are as distant as possible, instead of splitting the space into separate areas for each class [10]. When the labelled classes are provided with the hyperspectral datasets with two proximal hyperplanes based on the distance from their original hyperplanes, the twin proximal support vector machine (TPSVM) classifier model is applied [11]. Binary classification and splitting of data points into two classes can be performed by multi-class classification using SVM. After dividing the multiclassification problem into multiple binary classification problems, the same technique is then used to multiclass classification [12]. Each data instance goes through to the kernel function, which converts the original non-linear observations into a higher-dimensional space from which they can be segregated. For the present study, kernel functions such as Gaussian SVM kernel, Bessel function kernel, ANOVA radial basis function (RBF) kernel, Linear spline kernel, and Pearson VII universal kernel (PUK) are used for the above classifiers [13].

With respect to various clan models and classifiers, a framework was modelled to explore simultaneous correlation between social organization, settlement configuration and the architecture of ancestral house in both regions of Kerala and Minangkabau resulting in a social clan study and analysis. The work presented a review on geopolitical-based scholarly classification in the traditional and vernacular architecture studies in general, such as Southeast Asian architecture and Indian architecture, and on discourse of cultural diffusion such as “Indianized Southeast Asia” [14, 15]. An efficient algorithm using local monotone vertex property functions for determining the cores decomposition of a given social network has been carried out [16]. In a work of [17], studied the genetic structure of an ethnic group occupying a single village in north-eastern Ghana. The correspondence

between chromosome lineages and clan membership due to the combined effects of the strict patrilocal and patrilineal structure has also been discussed here. A study on comparative analysis of clan names and clan praises in Khumalo and Msogwaba Settlements. A sample of twenty respondents or informants from each settlement was considered for interview in relation to their specific clan names and clan praises. Random sampling was used to obtain the information which represents the population of both settlements [18].

A work demonstrated by demystifying the classification of various animal totems, categorising them into clans or groups, and evaluating their distinct features or characteristics and their impact on the human being. In this case also, the ethnographic approach was used to collect data [19]. A first-hand ethnographic data was employed to analyse the kinship classifications of the Enawene-Nawe, an Aruak-speaking people located in Meridional Amazonia. A formula was derived for matrimonial alliance as a direct exchange between two families [20]. A graphic comparison was made based on spatial types and their provincial distribution characteristics, and analyzing them with the conflicts in Fujian history, and then a dynamical relationship was arrived between clan groups. Study was conducted over 459 historic villages in this work [21]. Machine Learning (ML) algorithms were used to predict individuals’ susceptibility to be tricked by social engineering attacks. For each participant, different types of social engineering scores were calculated according to their responses. Along with simulations, questionnaires related to demographics, technology usage, and personality traits were filled out by the participants and ML algorithm resulted in better prediction accuracy [22, 23]. The applicability of machine learning algorithms has been significantly increased for various domains which includes, analysing brain activity patterns [24], opinions to be induced for ontology treatments [25], analysis of nutrient quality in foods [26], employability predictability for students using support vector machine models [27], fault tolerance in ubiquitous networks [28] and so on [29, 30]. Always steps are been taken to further improve the performance of machine learning based neural network models, like in weight parameter adjustment and bias variations as discussed in the work of [31]. Based on all the discussions made above and identifying the importance of machine learning models, the social network clan classification is done by variants of support vector machine models in this work. Table 1 provides a literature review on few existing works and their limitations.

In this research paper, the problem under consideration is to perform clan classification of the genealogical network, to achieve better classification results, novel SVM neural classifier and its variants are devised. The significance of this research lies in its potential to accelerate ancestral lineage analysis, assist genealogists to identify clan affiliations, and uncover undiscovered familial connections. We hope

Table 1 Literature review of existing works

Author and year	Methodology adopted	Limitations
Batagelj and Zaveršnik. (2011)	Decomposition technique based on k-cores employing vertex property function	Difficulty to handle large graphs and networks
Sanchez-Faddeev et al. (2013)	Gene sub-decomposition for skewed male population substructure and pattern variation matching	Unable to identify and classify strict patrilineal and patrilineal structure
Silva (2022)	Formulation of two interdictions: unions between people from the same clan and between kin who share close genealogical ties	Unable to locate indirect exchange between two families
Qin et al. (2022)	Using spatial types and their provincial distribution characteristics, and analyzing them with the conflicts in Fujian history	Difficulty in locating the morphological changes.
Makgopa (2019)	Indigenous knowledge systems (IKS) based rule base formulation	Computational complexity is high
Thwala (2018)	Random sampling technique was employed for analysis of clan names and clan praises in Khumalo and Msogwaba Settlements	Difficulty in grouping as the cultural life is to be considered
Huseynov and Ozdenizci Kose (2024)	Predictive classification and regression machine learning models	Occurrence of global minima problem
Widiastuti and Vedomuthu (2009)	Geopolitical-based scholarly classification in the traditional and vernacular architecture and clan analysis	Difficulty to obtain the contextual information of clans

to contribute to the advancement of our understanding of human history and legacy by utilizing the possibilities of SVM in genealogy studies.

2 Data

Data for this study was gathered through in-person interviews with a few learned members of *Heche* clan of the *Galo* tribe of Arunachal Pradesh, India. Data is also compiled from the genealogical book maintained by the clan in text format for generations [32].

3 Methods

Machine learning algorithms, SVM are used for clan classification. Basically, SVM classifier operates on formulating the decision hyperplane and classifying the hyperspace to attain required social network clans for the considered data model. In SVM classification, always the separating margin between the hyperplanes are kept maximal for better classification accuracy. This study focused on developing modified variants of SVM classifiers with different kernels to carry out the clan classification. In the existing SVM classifier models, modifications are made in respect of applying the kernel functions and the new variants are devised. The following section presents the variants of SVM kernels used in this study with the different kernels. The kernel functions of the SVM variants will operate and handle the non-linearity in the considered social

network clan problem and even though the data is non-linearly separable tends to achieve superior classification results.

3.1 Twin SVM classifier

In case of basic SVM classifier, only one hyperplane will be formulated to classify the data samples, but in Twin SVM (TSVM) model, two hyperplanes are attained based on the distance metric and their equations corresponds to:

$$w_1^T x_i + b_1 = 0 \quad (1)$$

$$w_2^T x_i + b_2 = 0 \quad (2)$$

The w and b corresponds to the weights and bias, x_i is the data sample, based on all these parameters the decision plane of SVM model is formulated. Each of the hyperplane exists close to its data class samples and to the farthest from the opposite data class sample. For the TSVM classifier, the hyper-planes are obtained through the below given equations:

$$\min \frac{1}{2} (Pw_1 + vb_1)^T (Pw_1 + vb_1) + r_1 v^T \gamma \quad (3)$$

$$\text{such that } -(Qw_1 + vb_1) + \gamma \geq v, \quad \gamma \geq 0 \quad (4)$$

$$\min \frac{1}{2} (Qw_2 + vb_2)^T (Qw_2 + vb_2) + r_2 v^T \gamma \quad (5)$$

$$\text{such that } -(Pw_2 + vb_2) + \gamma \geq v, \quad \gamma \geq 0 \quad (6)$$

In the above equations, γ indicates the non-negative vector component and hence is greater than or equal to 0, v specifies the error of the slack variable n and r_1 and r_2 are the penalty parameters.

3.2 Proximal SVM classifier model

The linear support vector machine predictor model operates using,

$$\text{Minimize } P(w, \gamma) = \frac{1}{2}w^T w + Cv^T \gamma \quad (7)$$

$$\text{Subject to } D(Aw - e\mu) + \gamma \geq e \quad \text{with } \gamma \geq 0 \quad (8)$$

with, $\gamma = (\gamma_1, \gamma_2, \dots, \gamma_p)^T$ is the error variable and $e = (1, 1, \dots, 1)^T$ is a vector of ones pertaining to slack variable of size n . Equation (7) represents the classification problem with the constraint satisfied by Eq. (8). The classifier solution lies at (w, γ) , for classifying the hyperplanes using the support vectors. For any data vector of the samples, the separating hyperplane acts as a classifier with length n specified as,

$$x^T w - \gamma \begin{cases} > 0, \text{ then } x \in P+ \\ < 0, \text{ then } x \in P- \\ = 0, \text{ then } x \in P+ \text{ or } x \in P- \end{cases} \quad (9)$$

Based on this linear SVM model, the proximal support vector machine (PSVM) classifier is designed to operate on the proximal support vectors. For an orientation vector, w , and existence of the separating planes γ , for maximization of the margin of separation, L_1 -norm of error vector, ξ , gets replaced by L_2 -norm and the inequality constraint will be replaced by means of an equality constraint.

The tailored equality constraints for the classification problem using the Proximal SVM is specified with,

$$\text{Minimize } P''(w, \xi, \gamma) = \frac{1}{2}(k^T k + \gamma^2) + \frac{1}{2}C\|\xi\|^2 \quad (10)$$

$$\text{Subject to } F(Z(Q, Q^T)Bu - v\gamma) + \xi = v \quad (11)$$

Linear Proximal SVM classification problem solution is evaluated based on the real time data set matrices $\mathbf{Q}$, $\mathbf{B}$ and some positive C , performing few algebraic simplifications, and is given by (Fung and Mangasarian 2005):

$$L = B[Q - v] \quad (12)$$

$$E = BL = [Q - v] \quad (13)$$

$$[\dot{w}^T \dot{\gamma}]^T = (I_{y+1}/C + E^T E)^{-1} E^T Bv \quad (14)$$

In Eqs. (12) and (13), the v vector (of length n) is appended to the $\mathbf{Q}$ matrix (of size $x \times y$) for a matrix of size $x \times (y + 1)$ and $\mathbf{I}_{y+1}$ is the unit matrix of size $(y + 1)$. The computation of w and γ involves matrix inversion of size $(y + 1)$ which is significantly smaller than x for a larger data-set. Figure 1, shows the planes, $w^T x = \gamma \pm 1$, in case of Proximal SVM classifier it will change their operations from general bounding classifying planes to proximal planes. Only at these proximal planes the points of the classifying classes get grouped. These 'proximal' planes margins get maximized by the first term of the minimization function given in Eq. (10).

In case of proximal support vector neural classifier, the numbers of proximal support vectors are 15–30% of the entire training dataset. In Proximal SVM classifier model, epsilon ' ϵ ' support vectors are defined as the vectors of predictive data point's x_i for which error ξ_i is minimal than its absolute value. The proximal epsilon support vector's tolerance parameter ϵ is selected such that about 1% of total data points form ϵ -proximal SV and the number of ϵ -proximal support vectors are substantially smaller than the support classifier vectors in basic SVM classifier.

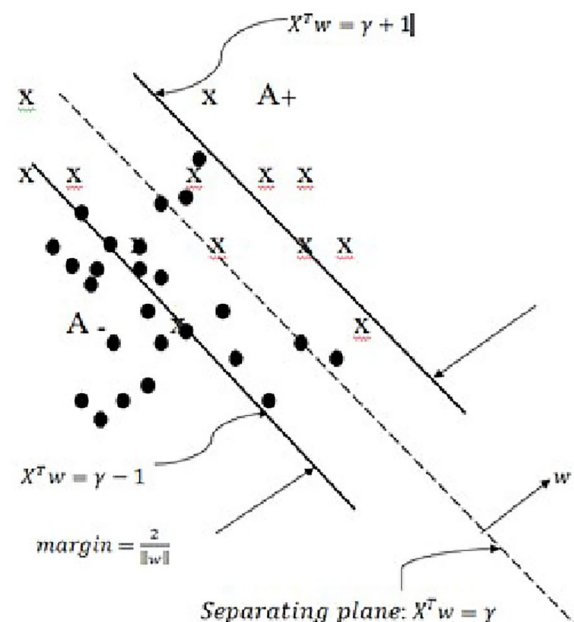


Fig. 1 Proximal support vector classifier model

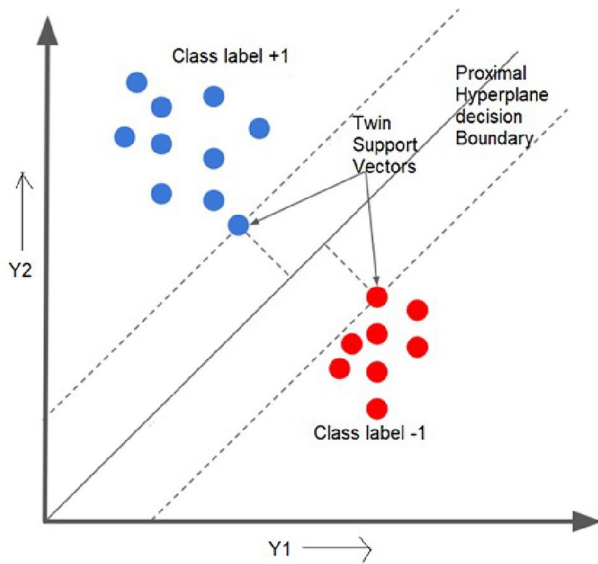


Fig. 2 Twin proximal SVM neural classifier

3.3 Twin proximal SVM classifier model

In the proximal support vector machine presented in Sect. 3.2, proximal hyper-planes are formed by including a weight orientation vector V_w along the position of the separation planes (χ) for maximization of the separating margins in the hyper-space. This weight orientation vector locates the proximal hyperplanes with the separation margins and obtains better classification metrics. The variant Twin proximal support vector machine (TPSVM) neural classifier model creates two proximal hyperplanes with respect to the distance from their original hyperplanes. The characteristics of the proximal SVM and that of twin SVM models with two hyperplanes are combined and the twin proximal support vector model with multi kernel functions is derived and used for clan classification.

Twin PSVM deduces the two non-parallel proximal hyperplanes considering the two set of labelled classes and the balance data samples are merged between a region of the above optimal proximal hyperplanes. Consider the data samples $A \in \mathbb{R}^{k_1 \times n}$ pertains to the class ‘+1’ and $B \in \mathbb{R}^{k_2 \times n}$ represents

the class ‘-1’ and the matrix $C \in \mathbb{R}^{k_3 \times n}$ represents the other data points labelled to be ‘0’. The Twin proximal SVM determines two non-linear proximal kernel surfaces in the input plane defined to be:

$$H_k(y^T, N_*^T, \chi)v_+ + b_+ = 0 \quad (15)$$

$$H_k(y^T, N_*^T, \chi)v_- + b_- = 0 \quad (16)$$

where, $N = [A; B; C]$; v_+ and $v_- \in \mathbb{R}^n$, ‘ H_k ’ is the kernel function and ‘ χ ’ represents the position of the proximal hyperplanes. Here, the classification problem is the quadratic problem and solution is obtained with the following pair of equations corresponding to data points of the classes ‘ $A \rightarrow +1$ ’ and ‘ $B \rightarrow -1$ ’. The two hyperplanes of Twin proximal SVM is obtained from below given equations.

$$\min_{(v_+, b_+, \lambda_1, \sigma_1)} \frac{1}{2} \|H_k(A, N_*^T, \chi)v_+ + e_1 b_+\|^2 + r_1 e_2^T \lambda_1 + r_2 e_3^T \sigma_1$$

such that:

$$-(H_k(B, N_*^T, \chi)v_+ + e_2 b_+) + \lambda_1 \geq e_2$$

$$-(H_k(C, N_*^T, \chi)v_+ + e_3 b_+) + \sigma_1 \geq e_3(1 - \epsilon)$$

with $\lambda_1 \geq 0$ and $\sigma_1 \geq 0$

$$\min_{(v_-, b_-, \lambda_2, \sigma_2)} \frac{1}{2} \|H_k(B, N_*^T, \chi)v_- + e_2 b_-\|^2 + r_3 e_1^T \lambda_2 + r_4 e_3^T \sigma_2$$

such that:

$$(H_k(A, N_*^T, \chi)v_- + e b_-) + \lambda_2 \geq e_1 \quad (17)$$

$$-(H_k(C, N_*^T, \chi)v_- + e_3 b_-) + \sigma_2 \geq e_3(1 - \epsilon) \quad (18)$$

with $\lambda_2 \geq 0$ and $\sigma_2 \geq 0$.

In the above equations, ‘ $r-j$ ’ ($j = 1$ to 4) denotes the real positive penalty factors, $e - 1$, e_2 and e_3 are the unit vectors of specific dimensions and the slack variables includes the λ_1 , σ_1 and λ_2 , σ_2 . In the twin proximal SVM model,

Table 2 Multi-kernels used for SVM variant classifiers

Multi-kernels	Kernel function definition
Gaussian SVM kernel	$K_{\text{kernel}}(z_m, z_n) = \exp\left(-\frac{\ z_m - z_n\ ^2}{2\sigma^2}\right)$
Bessel function kernel	$K_{\text{kernel}}(z_m, z_n) = -\text{Bessel}_{(lq+1)}^l(\sigma\ z_m - z_n\ ^2)$
ANOVA radial basis function (RBF) kernel	$K_{\text{kernel}}(z_m, z_n) = \left(\sum_{j=1}^q \exp\left(-\sigma(z_m^j - z_n^j)^2\right)\right)^d$
Linear spline kernel	$K_{\text{kernel}}(z_m, z_n) = 1 + z_m z_n \min(z_m + z_n) - \frac{z_m + z_n}{2} (\min(z, z^p)^2) + \frac{z_m + z_n}{3} (\min(z, z^p)^3)$
Pearson VII universal kernel (PUK)	$K_{\text{kernel}}(z_m, z_n) = \frac{1}{\left[1 + \left(2\sqrt{\ z_m - z_n\ ^2} \sqrt{2\left(\frac{1}{\omega}\right) - \frac{1}{\sigma}}\right)\right]^p}$

Table 3 Algorithmic steps for the SVM classifier and its variants

Step 1	Start
Step 2	Initialize the necessary weight parameters, learning rate and kernel parameters
Step 3	Set the maximum number of epochs(max_ epochs)
Step 4	Load the respective social network clan dataset
Step 5	While (number of epochs!=max_epochs) Do
Step 6	Separate the input features and target variables
Step 7	Build the SVM classifier. Use the respective hyperplane equations of the classifier and formulate the linear separable line
Step 8	Define each of SVM kernels, Gaussian (z_m, z_n), Bessel (z_m, z_n), RBF (z_m, z_n), Spline (z_m, z_n), PUK (z_m, z_n)
Step 9	Train the SVM classifiers using the kernel functions
Step 10	Obtain the performance metrics for least MSE values
Step 11	Check if the stopping condition is reached
Step 12	If Yes, Goto next step 10 else Goto step 4
Step 13	Output the performance metrics values for minimum MSE
Step 14	Stop
–	The test for stopping condition may be the mean square error reaching the most possible least minimum value or the set number of maximum iterations.
–	The algorithm is given for the classic SVM classifier and in this algorithm by changing the hyperplane equation, the other variants - Proximal SVM, Twin SVM, Proximal Twin SVM and Multi-class SVM can be simulated for the clan classification.

Table 4 Parameters for SVM classifier variants

Parameters	Metric values
Learning rate	0.2
Learning approach	Supervised learning
Learning rule	Gradient Descent learning
Activation function	Tangential sigmoidal activation
No. of epochs	Until error convergence
No. of trials	36
Error convergence	10^{-6}
Neuron layers	Based on data-sets used
σ (kernel)	0.25

$$\max_{\xi=[\alpha, \beta]} u_5^T \xi - \frac{1}{2} \xi^T V_1 (Q^T Q)^{-1} O_2^T \xi \quad \text{such that} \quad 0 \leq \xi \leq D_2$$

$$\text{where } P = [H_k(A, N_*^T, \chi)u_1], \quad Q = [H_k(Q, N_*^T, \chi)u_2] \quad \text{and} \\ R = [H_k(R, N_*^T, \chi)u_3],$$

$$O_1 = [Q; R]; \quad O_2 = [P; R] \quad (19)$$

$$D_1 = [r_1 u_2; r_2 u_3]; \quad D_2 = [r_3 u_1; r_4 u_3] \quad (20)$$

$$\omega = [\zeta, \delta] \quad \text{and} \quad \xi = [\alpha, \beta] \quad (21)$$

$$u_4 = [u_2; u_3(1 - \varepsilon)] \quad \text{and} \quad (22)$$

optimality conditions are with respect to the Lagrangian multiplier given by,

$$\max_{\omega=[\zeta, \delta]} u_4^T \omega - \frac{1}{2} \omega^T V_1 (P^T P)^{-1} O_1^T \omega \quad \text{such that} \quad 0 \leq \omega \leq D_1$$

In the novel twin proximal SVM classifier, the weight orientation vector is calculated using,

$$[O_w]_1^T = \left(\frac{P_{j+1}}{R} + N^T N \right)^{-1} N^T \chi u_1 \quad (23)$$

Table 5 Performance metrics evaluated

Performance metric	Evaluation formula
Precision	$\frac{TP}{TP+FP}$ (Correctly classified positive data samples, depends on both positive and negative samples)
F1-Score	$\frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$ (Harmonic mean of the precision and recall)
Classification Accuracy (CA)	$\frac{TP+TN}{TP+TN+FP+FN}$ (Number of correct classifications to the total number of classifications) ^a
Mean Square Error (MSE)	$\frac{1}{N} \sum_{k=1}^n (O_d - O_c)^2$ (Mean square between the difference of the desired output and the actual calculated output) ^b

^aTP (True Positive), TN (True Negative), FP (False Positive), and FN (False Negative)

^bN (Total number of samples), O_d (Desired Target Output), O_c (Actual calculated output)

Table 6 Clan classification results

Proposed SVM	Kernels used	Precision	F1-Score	Classification accuracy	MSE
Variant					
Neural Classifier					
Modified Convolutional Neural approach	–	0.7015	0.6934	71.54	5.9965
Fairsna technique	–	0.7127	0.7066	75.38	4.8624
Traditional SVM Classifier	Gaussian kernel	0.8311	0.8453	87.22	5.9418
	Bessel kernel	0.7219	0.7451	77.90	5.0247
	ANOVA RBF kernel	0.9278	0.9149	88.27	4.5168
	Linear spline Kernel	0.8063	0.8206	81.51	4.3105
	PUK Kernel	0.7849	0.7531	76.39	5.1453
Twin SVM Classifier	Gaussian kernel	0.8941	0.9056	88.21	3.2417
	Bessel kernel	0.7548	0.8549	79.65	3.0945
	ANOVA RBF kernel	0.9315	0.9547	93.62	0.00128
	Linear spline Kernel	0.8176	0.8379	84.56	0.9947
	PUK Kernel	0.8053	0.7648	78.49	1.6648
Proximal SVM Classifier	Gaussian kernel	0.8987	0.9236	90.22	1.2478
	Bessel kernel	0.7719	0.8725	83.46	0.8849
	ANOVA RBF kernel	0.9549	0.9847	95.41	0.00349
	Linear spline Kernel	0.8430	0.8414	87.02	0.0647
	PUK Kernel	0.8277	0.7954	81.44	1.0148
Twin Proximal SVM Classifier	Gaussian kernel	0.9025	0.9415	92.35	0.7714
	Bessel kernel	0.8144	0.8936	85.41	0.06473
	ANOVA RBF kernel	0.9965	0.9904	98.91	0.00015
	Linear spline Kernel	0.8647	0.8719	90.63	0.00249
	PUK Kernel	0.8295	0.8655	85.49	0.07516
Multi-class SVM Classifier	Gaussian kernel	0.8846	0.8741	86.42	6.5847
	Bessel kernel	0.7149	0.7266	77.59	4.1159
	ANOVA RBF kernel	0.9028	0.9114	90.47	2.3149
	Linear spline Kernel	0.8025	0.8149	82.11	4.2178
	PUK Kernel	0.7691	0.7398	75.46	5.0064

$$[O_w]_2^T = \left(\frac{Q_{j+1}}{R} + N^T N \right)^{-1} N^T \chi u_2 \quad (24)$$

Figure 2 illustrates the classification mechanism of the proposed Twin Proximal Support Vector Machine classifier. In Fig. 2, the separating hyperplane $O_w^T Y = \chi \pm 1$, changes the basic SVM decision plane to that of the proximal separating hyperplane using the twin support vectors. In the location of these proximal support vector planes, each and every specific classifying data points gets grouped together.

3.4 Multi-class SVM classifier

Traditional SVM classifier is devised to handle binary classification, which is capable of handling only two classes. For the clan classification problem considered in this research study, it may be required to handle more than two classes for social network analysis. As a result, this work employed multi-class support vector machine (MCSVM) classifier that handles more than two classes, each class in MCSVM neural model is formulated as an individual SVM classifier model. In the proposed study of clan classification from a social network, an MCSVM regression model is developed with $N = \frac{n(n-1)}{2}$ binary SVM classifiers. The multi-class SVM neural model is solved with directed acyclic graph SVM. For the given training

Fig. 3 Convergence plot using basic SVM classifier model

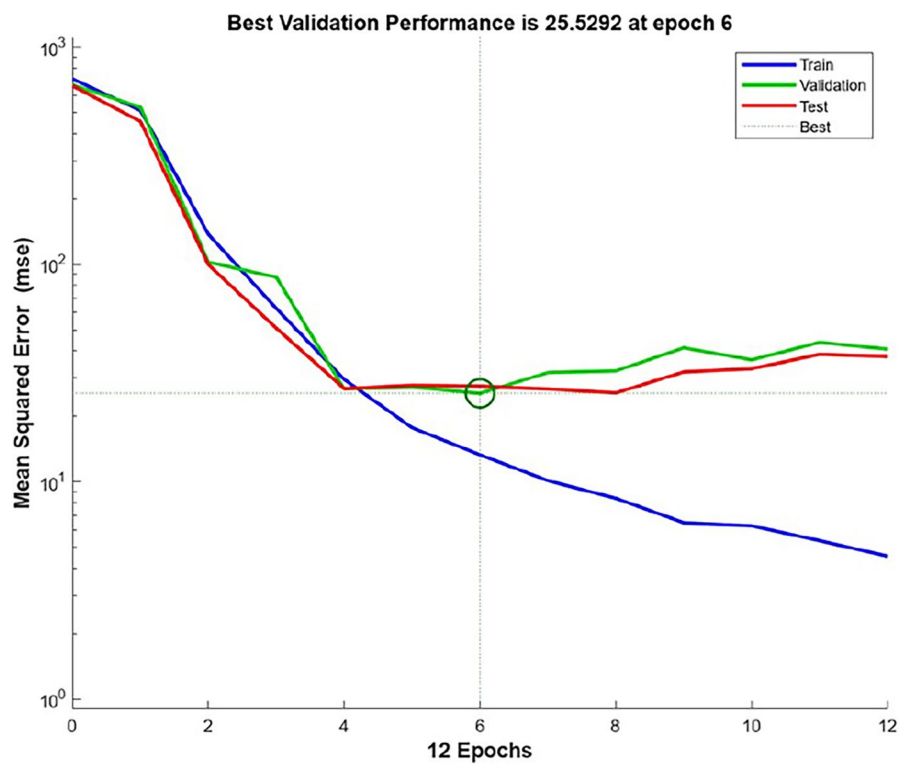


Fig. 4 Convergence plot using twin SVM classifier model

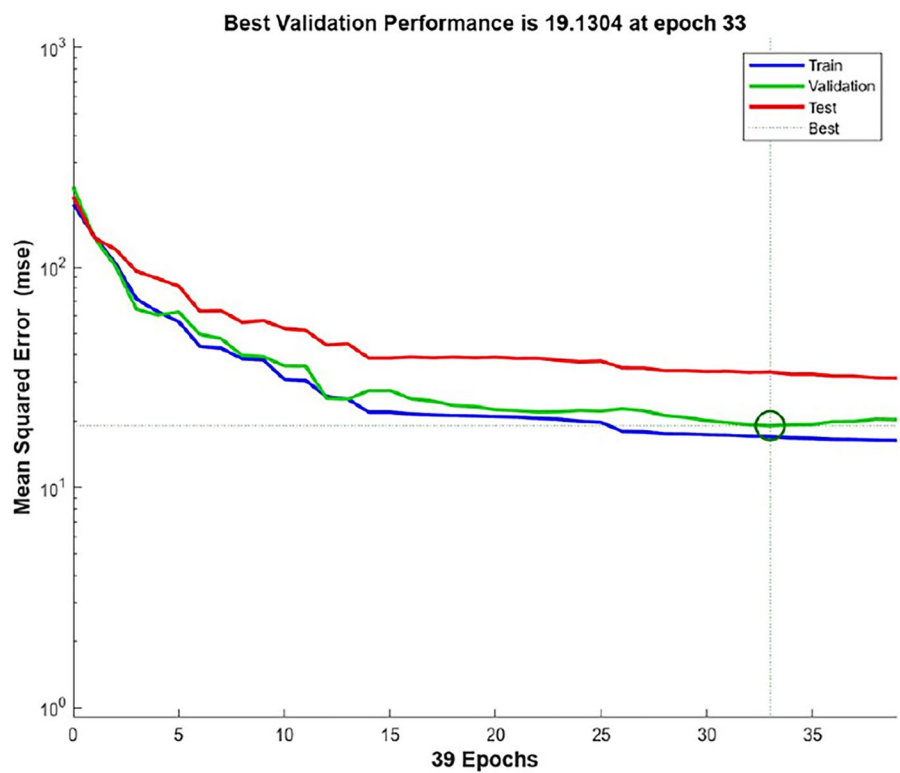


Fig. 5 Convergence plot using PSVM classifier model

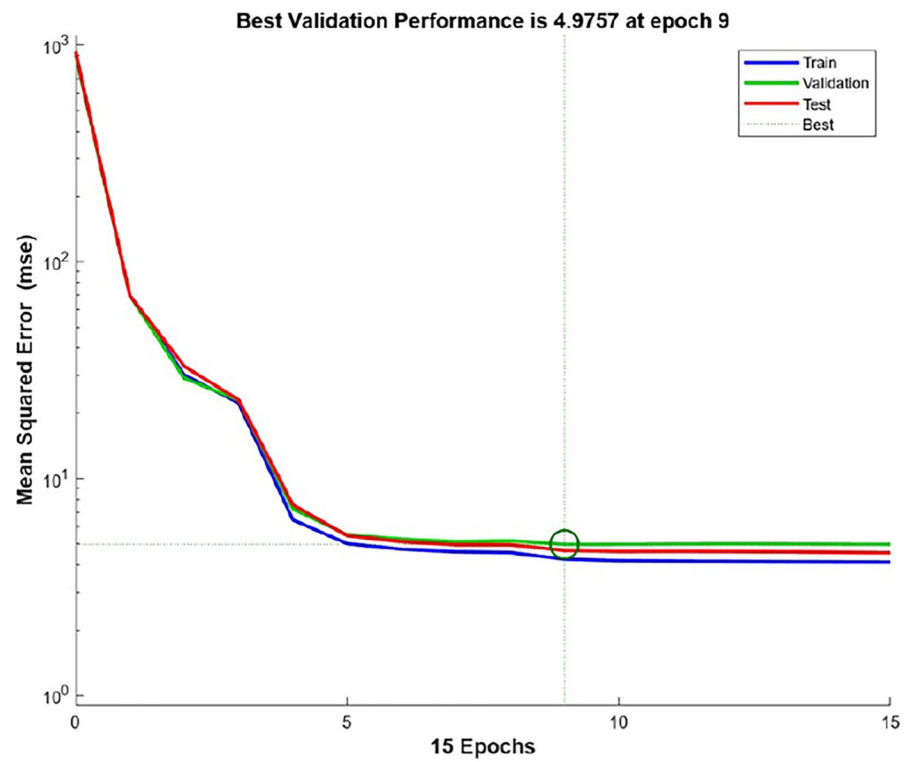


Fig. 6 Convergence plot using twin PSVM classifier model

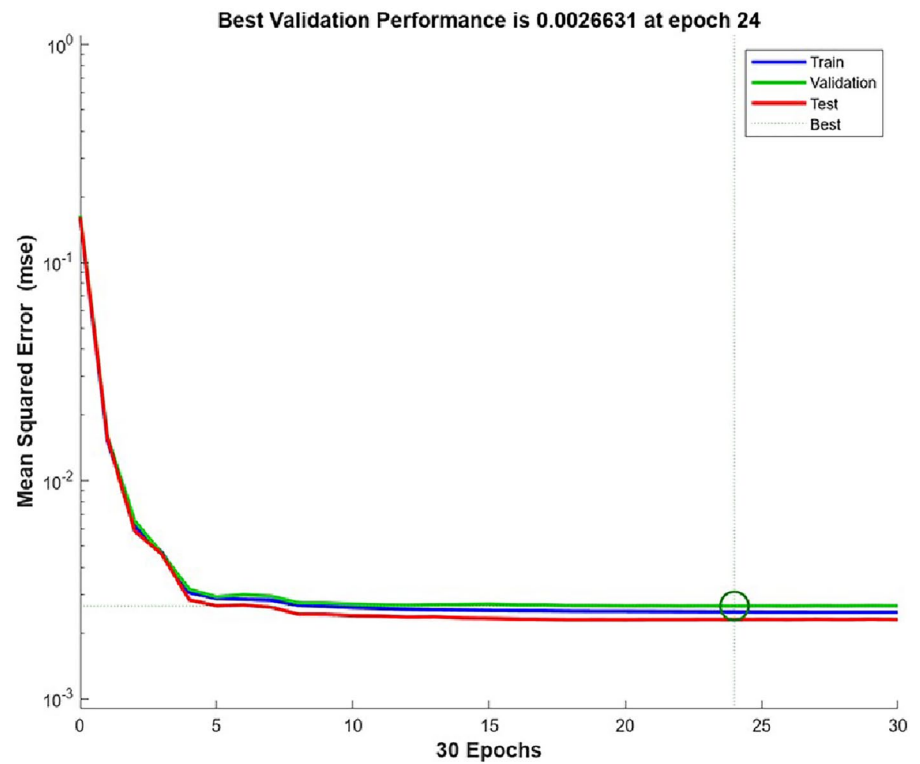
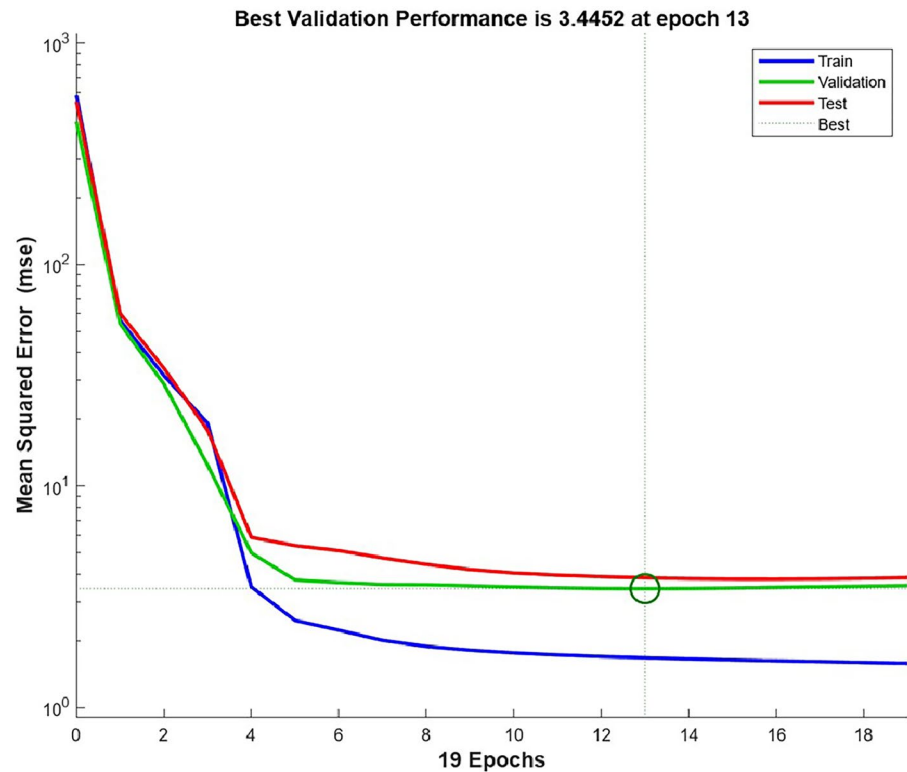


Fig. 7 Convergence plot for clan dataset using MCSVM neural classifier



clan data samples, the multiclass regression problem is built to be,

$$\min_{v^{ij}, b^{ij}, \zeta_t^{ij}} \frac{1}{2} (v^{ij})^T (v_{ij}) + \phi \sum_t \zeta_t^{ij} \hat{\zeta}_t^{ij} (v^{ij})^T$$

subject to:

$$\begin{aligned} (v^{ij})^T \alpha(x_t) + b^{ij} &\geq 1 - \zeta_t^{ij} & \text{if } y_t = i \\ (v^{ij})^T \alpha(x_t) - b^{ij} &\geq 1 - \hat{\zeta}_t^{ij} & \text{if } y_t = i \\ (v^{ij})^T \alpha(x_t) + b^{ij} &\geq -1 + \zeta_t^{ij} & \text{if } y_t = j \\ (v^{ij})^T \alpha(x_t) - b^{ij} &\geq -1 + \hat{\zeta}_t^{ij} & \text{if } y_t = j \\ \zeta_t^{ij}, \hat{\zeta}_t^{ij} &\geq 0 \end{aligned}$$

where $\zeta_t^{ij}, \hat{\zeta}_t^{ij}$ are the slack variables, ϕ is the trade-off parameter of the margin size and the error.

3.5 Kernel functions for the classifiers

The clan dataset of the considered social network is non-linearly separable data and thus it is very tedious to obtain the decision hyperplane of all the classifiers SVM, TSVM, PSVM, TPSVM and MCSVM neural classifiers. All this machine learning based support vector machine classifiers are designed to classify the non-linearly separable clan data. To achieve this, the best possible hyperplanes, proximal hyperplanes, twin hyperplanes, kernels are used.

In this study, 5 different kernels are considered to operate with the proposed variants of the SVM classifiers. Each of these kernels are defined with their specific parameters and are tabulated in Table 2. All the variants of SVM classifiers proposed in this research study operates using these kernels to perform the clan classification.

The algorithm in a general form for the SVM and its variants employed in this research paper for social network clan classification is given in Table 3.

4 Results and discussion

Variants of the SVM neural classifier developed in this research study are applied to the datasets of the clan classification of the considered social network. All the four variants - Twin SVM classifier, Proximal SVM classifier, Twin Proximal SVM classifier and Multi-class SVM classifier are simulated and tested for each of the five kernels in respect of the clan datasets. The parameters set for the modelled SVM variants are listed in Table 4. Performance metrics evaluated for testing and validating the novel SVM variant classifiers over the clan datasets are presented in Table 5. Simulation is carried out in MATLABR2022b version in an Intel hard core i7 2100 MHz 16GB processor and the classification results are obtained to prove the superiority of the variants of SVM neural classifiers.

Table 7 Performance metrics based on percentage of dataset features (proposed SVM variant classifiers)

Performance Metrics	Developed SVM variant classifiers	Percentage of clan dataset features									
		10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Precision	Traditional SVM classifier	0.1145	0.1658	0.2019	0.2248	0.3417	0.5148	0.6475	0.7122	0.7741	0.9278
	Twin SVM Classifier	0.3614	0.3948	0.4478	0.4519	0.4961	0.6615	0.6922	0.7714	0.8976	0.9315
	Proximal SVM Classifier	0.4487	0.4813	0.5075	0.5496	0.5844	0.6971	0.7481	0.8246	0.8991	0.9549
	Twin Proximal SVM Classifier	0.5487	0.5911	0.6248	0.6492	0.7384	0.7726	0.8845	0.8961	0.9247	0.9965
	Multi-class SVM Classifier	0.2748	0.3012	0.3148	0.3594	0.4178	0.4689	0.4872	0.7015	0.8814	0.9028
F1-Score	Traditional SVM Classifier	0.2314	0.3148	0.3557	0.3647	0.4297	0.4518	0.6147	0.6324	0.8046	0.9149
	Twin SVM Classifier	0.5124	0.6237	0.6684	0.7548	0.7741	0.7934	0.8524	0.8841	0.8923	0.9547
	7Proximal SVM Classifier	0.5247	0.6349	0.7012	0.7561	0.8014	0.8126	0.8744	0.9126	0.9349	0.9847
	Twin Proximal SVM Classifier	0.6623	0.6947	0.7519	0.8147	0.8327	0.8529	0.9012	0.9165	0.9463	0.9904
	Multi-class SVM Classifier	0.3014	0.3647	0.4021	0.4876	0.5514	0.5629	0.6347	0.6947	0.8215	0.9114
Classification Accuracy	Traditional SVM Classifier	60.23	62.66	63.48	70.15	72.34	71.29	76.55	80.12	83.64	88.27
	Twin SVM Classifier	68.77	70.23	76.44	79.25	79.63	80.12	82.66	84.19	86.29	93.62
	Proximal SVM Classifier	71.54	76.48	76.93	77.49	80.56	82.31	85.49	87.66	90.18	95.41
	Twin Proximal SVM Classifier	75.48	79.88	84.57	85.06	88.67	88.91	90.64	93.22	97.01	98.91
	Multi-class SVM Classifier	61.26	63.94	64.03	71.56	73.99	75.48	78.19	82.46	85.05	90.47
Mean Square Error	Traditional SVM Classifier	7.1546	7.0594	6.9418	6.6647	6.0289	5.6147	5.2348	5.0016	4.8179	4.5168
	Twin SVM Classifier	5.4189	5.0466	4.5179	3.0248	2.4196	2.0034	1.4276	0.9248	0.0647	0.00128
	Proximal SVM Classifier	3.1564	3.0248	2.4761	2.3178	2.0463	1.4718	1.2246	0.3417	0.0151	0.00349
	Twin Proximal SVM Classifier	2.3147	2.0169	1.4781	1.2347	1.0415	0.6649	0.1428	0.0249	0.0069	0.00015
	Multi-class SVM Classifier	7.0238	6.66348	6.4179	6.2245	6.0048	5.2248	5.0647	4.9036	3.5479	2.3149

Fig. 8 Plot of precision for varied percentage of data features

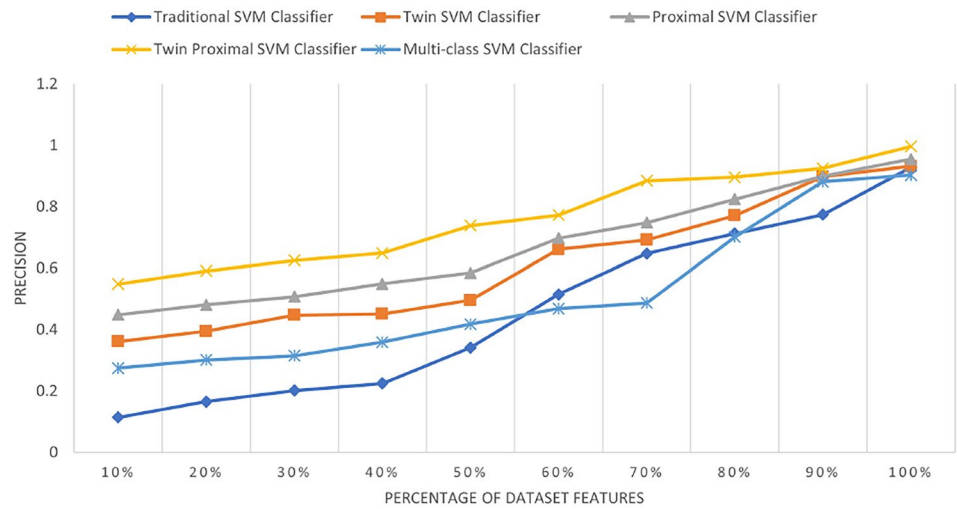


Fig. 9 Plot of F1-Score for varied percentage of data features

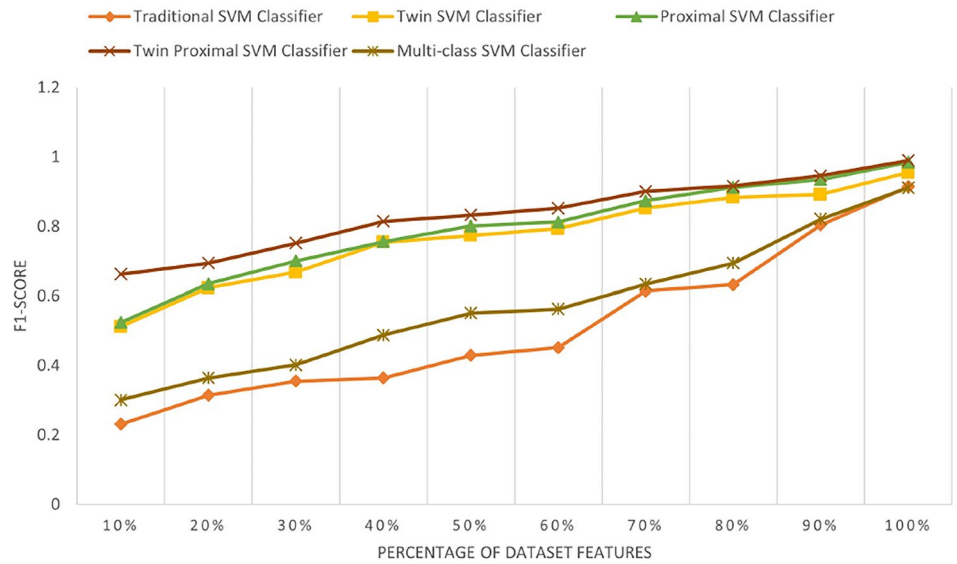


Fig. 10 Plot of classification accuracy for varied percentage of data features

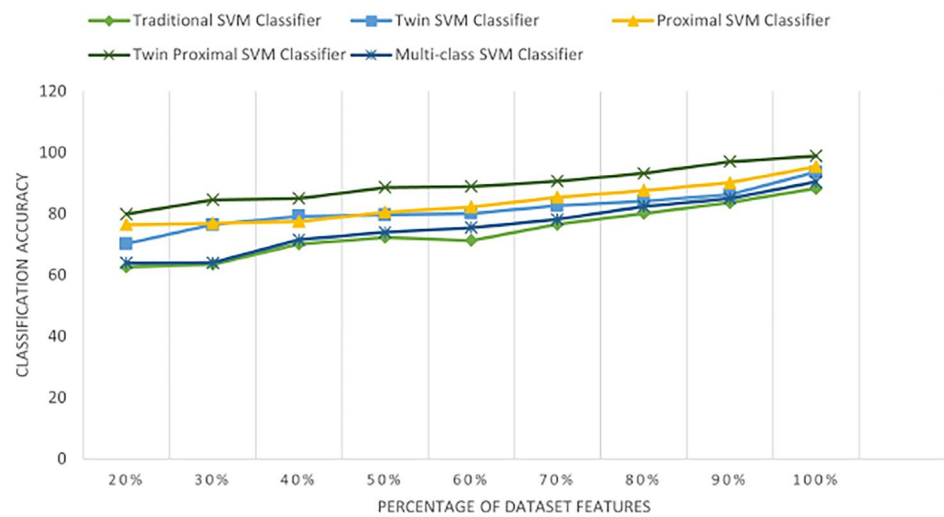


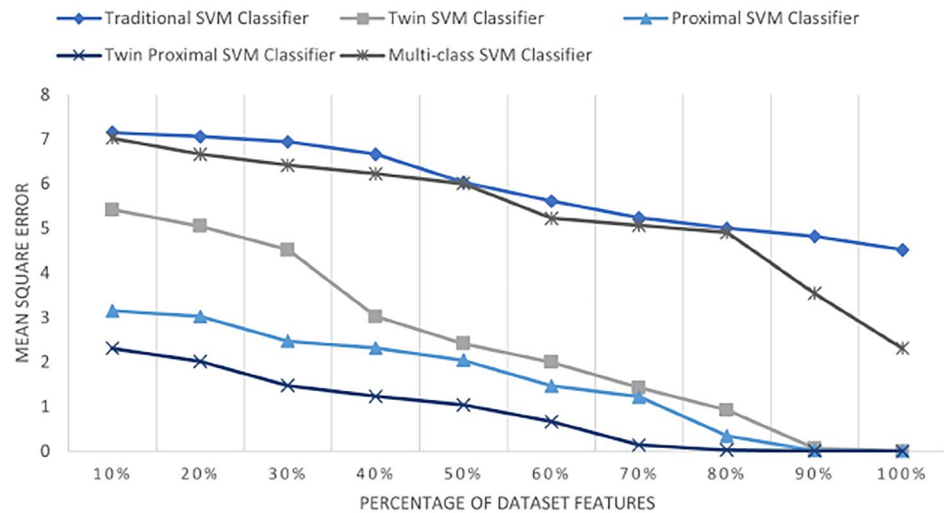
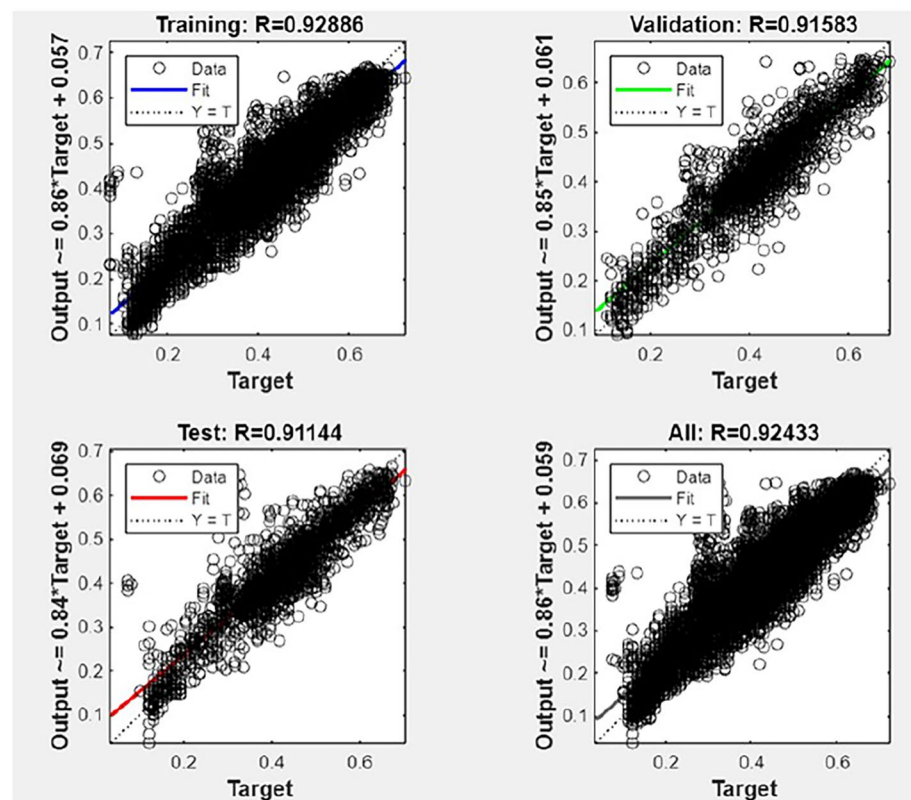
Fig. 11 Plot of MSE for varied percentage of data features**Fig. 12** Regression plot for the clan dataset using TPSVM classifier model

Fig. 13 Error histogram plot with TPSVM classifier model

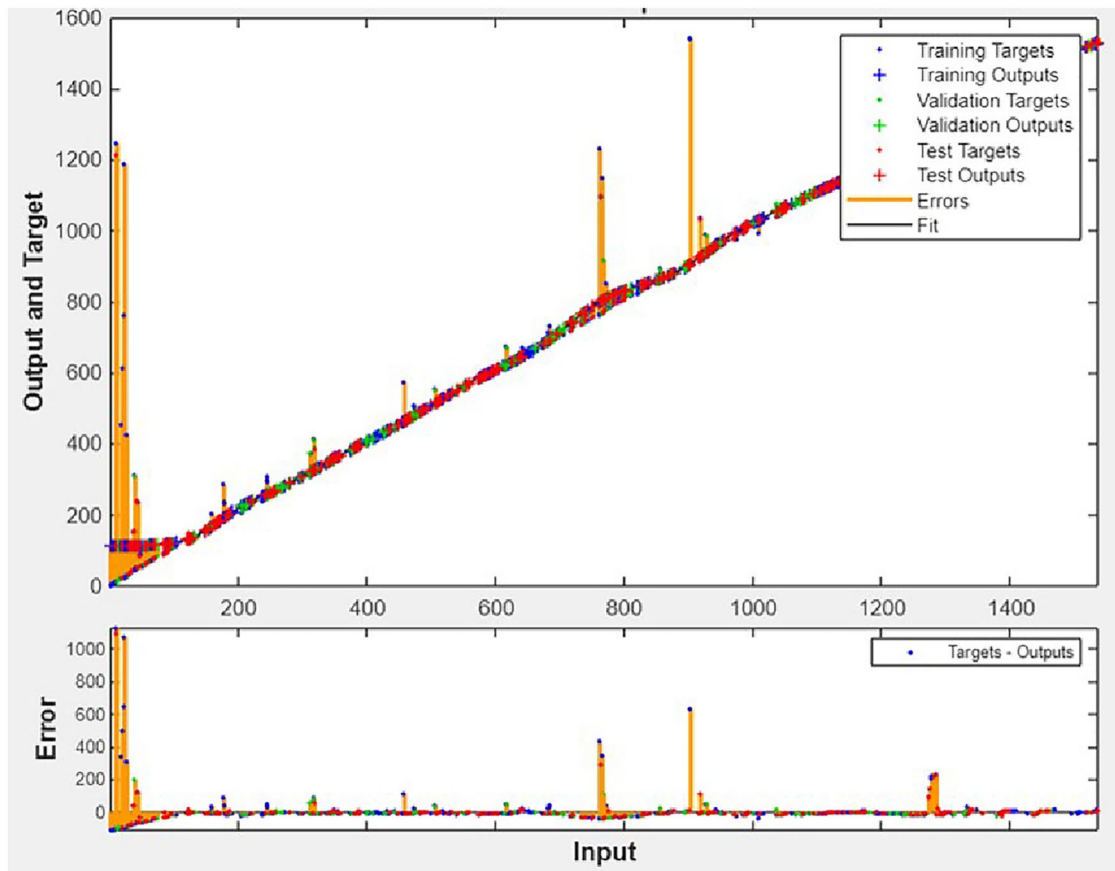
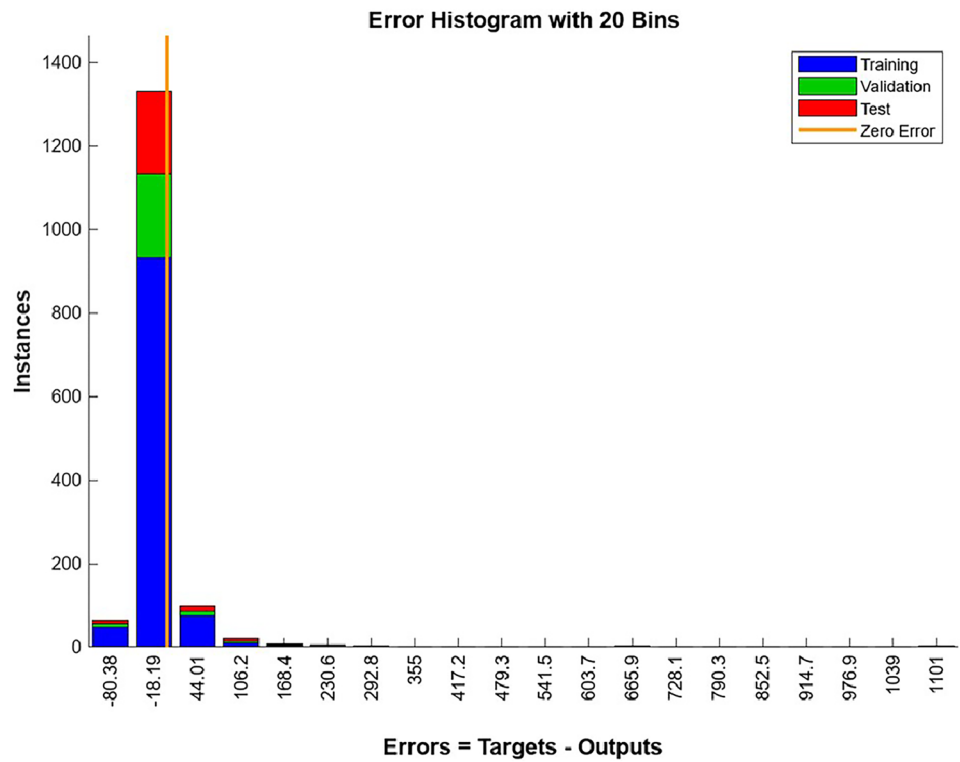


Fig. 14 Plot of training, testing and validation data using TPSVM classifier model

Figures 3, 4, 5, 6 and 7 shows the sample convergence plots simulated using the proposed classifier models for the clan dataset with different SVM kernels. Among all the kernel functions employed for this non-linearly separable data, it was noted that the ANOVA radial basis function kernel outperformed all the other kernels. In respect of all the SVM variants - TSVM, PSVM, TPSVM and MCSVM neural classifiers, ANOVA RBF kernel resulted in better classification accuracy with minimized error value proving its superiority over other kernels. This is due to the fact that the regularization aspect in RBF kernel is increased due to the closest neighborhood points considered in the radius for the attaining the maximal separating margins. 5-fold cross validation was used for the considered clan datasets of the particular social network. Table 6 confirms that better classification accuracy is obtained using the TPSVM neural classifier with ANOVA RBF Kernel with 98.91% for the considered dataset. As well the minimized MSE value of 0.00015 has been attained using the Twin PSVM classifier compared with the other proposed variants. Among all the designed and simulated SVM classifier models, the Twin Proximal SVM classifier has resulted in superior classification accuracy with better precision and F1-score than other methods compared. Compared to the other methods from the literature as shown in Table 6, [9] that employed modified convolutional neural network and [31], that proposed Fairsna, the ANOVA RBF Kernel has resulted in better clan classification accuracy.

Table 7 presents the evaluated results using the variants of SVM classifiers with respect to the percentage of clan dataset features. For about nearly 70–100% of the dataset features, superior results are evaluated and among all the variants modelled, twin proximal support vector machine classifier attains better classification accuracy with minimized mean square error value. In Table 7, all the SVM, TSVM, PSVM, TPSVM and MCSVM neural classifiers are simulated with ANOVA radial basis function kernel as this kernel outperformed all other kernels used as given in Table 6. For varied percentage of features, the classifier models intend to achieve most suitable clan classification mechanism. Figures 8, 9, 10 and 11 illustrates the plot of all SVM classifier models for the considered dataset for varied percentage of dataset features. It is inferred from Table 7 and Figs. 8, 9, 10 and 11, that the modelled twin proximal support vector machine has resulted in better performance metrics than the other compared variants of SVM neural classifiers. Figure 12 provides the regression plot of the outperformed twin proximal support vector neural classifier for the clan dataset. Figures 13 and 14 shows the error histogram plot of TPSVM classifier model and plot of training, testing and validation data using TPSVM classifier model respectively.

In many areas, SVM classification can be a beneficial tool for genealogical studies. Based on genealogical data, SVM algorithms may help classification of individuals into different ancestral lineages or clans. SVM may identify patterns that are characteristic of particular ancestral groups by analyzing particular genetic markers or characteristics, enabling researchers to trace the genealogy of individuals or families. By utilizing the possibilities of SVM in genealogy studies, the present work is an attempt to classify clans by employing SVM algorithms, which in turn will improve our understanding of human history and heritage.

The future scope of this research study is to apply the SVM model and its variants to other different social clan networks. Also, to develop better machine learning models with improved weight update criterion or tuning the weights appropriately to achieve better classification accuracy of the clan networks. Development of stochastic population based optimization algorithms and applying it to the parameter optimization of machine learning models for attaining an effective social clan network classifier.

Funding The authors affirm that they did not accept any funds, grants, or other assistance for the creation of this manuscript.

Data availability Data that supports the findings of this study are available from the corresponding author upon request.

Declarations

Conflict of interest The authors of this work have no financial or other conflicts to disclose.

References

1. Arora A, Chakraborty P, Bhatia M (2023) Identifying digital biomarkers in actigraph based sequential motor activity data for assessment of depression: a model evaluating svm in lstm extracted feature space. *Int J Inf Technol* 15(2):797–802
2. Joram A, Singh KR (2024) Comparative study of kinship network community detection approaches. *Discrete mathematics, algorithms and applications*. World Scientific, Singapore
3. Durgesh KS, Lekha B (2010) Data classification using support vector machine. *J Theoret Appl Inf Technol* 12(1):1–7
4. Ding S, Zhang N, Zhang X, Wu F (2017) Twin support vector machine: theory, algorithm and applications. *Neural Comput Appl* 28:3119–3130
5. Jayadeva, Khemchandani R, Chandra S (2007) Twin support vector machines for pattern classification. *IEEE Trans Pattern Anal Mach Intell* 29(5):905–910. <https://doi.org/10.1109/TPAMI.2007.1068>
6. Tomar D, Agarwal S (2015) Twin support vector machine: a review from 2007 to 2014. *Egypt Inform J* 16(1):55–69
7. Roy A, Islam M, Karim M, Ahmed KA, Khan AR, Uddin M, Xames MD (2023) Comparative analysis of KNN and SVM in multicriteria inventory classification using topsis. *Int J Inf Technol* 15(7):3613–3622

8. Shao Y-H, Lv X-J, Huang L-W, Bai L (2023) Twin svm for conditional probability estimation in binary and multiclass classification. *Pattern Recogn* 136:109253
9. Krishna SL, Jeya IJS, Deepa SN (2022) Fuzzy-twin proximal svm kernel-based deep learning neural network model for hyperspectral image classification. *Neural Comput Appl* 34(21):19343–19376
10. Bairagi D (2019) Proximal support vector machine classifier based on lms algorithm. In: 2019 International Conference on Innovative Trends and Advances in Engineering and Technology (ICITAET), pp. 116–123 . <https://doi.org/10.1109/ICITAET47105.2019.9170231>
11. Cano Lengua MÁ, Papa Quiroz EA, Lopez Luis JC, Ichpas Tapia R (2023) A symmetric cone proximal multiplier algorithm. *Int J Eng Trends Technol* 71(1):257–270
12. Chamasemani FF, Singh YP (2011) Multi-class support vector machine (svm) classifiers – an application in hypothyroid detection and classification. In: 2011 Sixth International Conference on Bio-Inspired Computing: Theories and Applications, pp. 351–356 . <https://doi.org/10.1109/BIC-TA.2011.51>
13. Karal Ö (2020) Performance comparison of different kernel functions in svm for different k value in k-fold cross-validation. In: 2020 Innovations in Intelligent Systems and Applications Conference (ASYU), pp. 1–5 . IEEE
14. Widiastuti I, Vedomuthu R (2009) A comparative study of vernacular settlement and dwelling culture: a case study in Kerala, South India, and Minangkabau in Sumatra, Indonesia. *Int J Interdiscipl Soc Sci Ann Rev* 4(2):15–30
15. Joram A, Singh KR, Mishra LN, Rathour L, Vanav Kumar A (2024) Tracing roots and linkages: Harnessing graph theory and social network analysis in genealogical research, based on the kin naming system. *Discrete mathematics, algorithms and applications*. World Scientific, Singapore
16. Batagelj V, Zaveršnik M (2011) Fast algorithms for determining (generalized) core groups in social networks. *Adv Data Anal Classif* 5(2):129–145
17. Sanchez-Faddeev H, Pijpe J, Van Der Hulle T, Meij HJ, Gaag KJ, Slagboom PE, Westendorp RGJ, De Knijff P (2013) The influence of clan structure on the genetic variation in a single Ghanaian village. *Eur J Human Genet* 21(10):1134–1139
18. Thwala JJ (2018) A comparative study of clan names and clan praises in Khumalo and Msogwaba settlements. *J Sociol Soc Anthropol* 9(1):1–9
19. Makgopa M (2019) Totemism: A symbolic representation of a clan with specific reference to the basotho ba leboa-an ethnographical approach. *S Afr J Afr Lang* 39(2):159–164
20. Silva M (2022) An amazonian iroquois system: Enawene-nawe kinship and alliance. *Rev Antropol* 65:195922
21. Qin Q, Xiao D, Ma J, Tao J (2022) Using graphic comparison to explore the dynamic relationship between social conflicts and Fujian defense-dwellings in China. *J Asian Archit Build Eng* 21(6):2285–2305
22. Huseynov F, Ozdenizci Kose B (2024) Using machine learning algorithms to predict individuals' tendency to be victim of social engineering attacks. *Inf Dev* 40(2):298–318
23. Mostafa MM (2024) A social network analysis of the citation diffusion of academic articles. *Int J Inf Technol*. <https://doi.org/10.1007/s41870-024-02034-2>
24. Cho R, Zaman M, Cho KT, Hwang J (2024) Investigating brain activity patterns during learning tasks through eeg and machine learning analysis. *Int J Inf Technol* 16:2737–2744
25. Oussaid M, Bouarab-Dahmani F (2024) Ontology enrichment from opinions using machine learning algorithms. *Int J Inf Technol*. <https://doi.org/10.1007/s41870-024-01873-3>
26. Balamurugan S, Gurumoorthi E, Devi PP, Maruthamuthu R (2024) Impact of nutrients in food quality and safety by machine learning classifier using internet of things. *Int J Inf Technol* 16:2803–2812
27. Jayachandran S, Joshi B (2024) Customized support vector machine for predicting the employability of students pursuing engineering. *Int J Inf Technol* 16:3193–3204
28. Bhajantri LB, Ayyannavar VV (2024) Cognitive agent based fault tolerance in ubiquitous networks: a machine learning approach. *Int J Inf Technol* 16:2363–2377
29. Verma V, Shalender K (2024) A triad ensemble learning framework for forecasting disengagement behaviour among employees using machine learning (ml) algorithms. *Int J Inf Technol* 16(5):3001–3007
30. Reyaz N, Ahamad G, Khan NJ, Naseem M, Ali J (2024) Svmcti: support vector machine based cricket talent identification model. *Int J Inf Technol* 16(3):1931–1944
31. Mittal A, Chandra P (2024) Improving learning in artificial neural networks using better weight initializations. *Int J Inf Technol*. <https://doi.org/10.1007/s41870-024-01869-z>
32. Bage L, Kamdak J, Bage B (2011) Genealogy of Heche Clan vol. 2, pp. 16–54. Heche Welfare Society, Dumptorijo, Arunachal Pradesh

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.