International Journal of Computer Sciences and Engineering
Vol.**11**, Special Issue.**1**, pp.**120-126**, November **2023**
ISSN: 2347-2693 (Online)
Available online at: www.ijcseonline.org

Research Paper

An Empirical Comparison of Linear and Non-linear Classification Using Support Vector Machines

Sanjib Saha[1]

[1]Dept. of Computer Science and Engineering, National Institute of Technology, Durgapur and Dr. B. C. Roy Engineering College, Durgapur, India

Author's Mail Id: sanjib.saha@bcrec.ac.in

Abstract: Support Vector Machines (SVMs) are used in large-scale linear and non-linear non-probabilistic binary or multi-class classification. Classification using SVM techniques gives better accuracy than other machine learning classification methods. Various Support Vector Classification (SVC) algorithms are available in the literature, and many researchers are facing the problem of choosing the best methods for real-world applications. This paper integrates LibSVM and LibLINEAR tools with the Weka tool. The Radial Basis Function (RBF), Polynomial, Sigmoid and Linear kernel-based C-SVC and nu-SVC models, as well as predictive linear SVM models, are applied to six UCI machine learning datasets. The presentations of various SVC methods are empirically matched using Classification Accuracy (CA), Root Mean Square Error (RMSE), and Area Under Curve (AUC) metrics. The proposed method for this article is RBF kernel and linear kernel in C-SVC and nu-SVC models. The performance of the proposed models is trained and tested with UCI machine learning datasets for non-linear and linear classification. The results are compared with state-of-the-art SVC models. RBF kernel in C-SVC and nu-SVC models has achieved an accuracy of 97.3% and 98%, respectively, for non-linear classification on the Iris dataset. The linear kernel in C-SVC and nu-SVC models has achieved 96.6% and 98% accuracy for linear classification on the Iris dataset. L2-Regularized L2-Hinge Loss dual and primal SVC model has a classification accuracy of 96% for large-scale linear classification on the Iris dataset. Therefore, some conclusions based on overall performances on six datasets are as follows. (i) RBF kernel-based C-SVC model performs better than other non-linear SVC methods. (ii) Linear kernel-based C-SVC and nu-SVC methods perform better in the case of linear classification. (iii) In large-scale linear classification, L1-Regularized L2-Loss SVC, Multi-class SVC by Crammer Singer and L2-Regularized L2-Loss SVC methods perform better than other linear SVC methods. (iv) Most of these methods give good results in the case of datasets having all the most numeric attributes or dimensions and a large number of instances or vectors.

Keywords: Support Vector Machines, Kernel Functions, Regularized Losses, Classification

1. Introduction

Support Vector Machines [1-4] analyze data and recognize patterns established on direct learning models with related learning algorithms. SVMs are used for categorization as well as retrogression inspection. SVM inputs a set of data and forecasts an output among the two possible classes for each given input. SVM is a binary linear classifier. Training data samples are the input to the SVM to train the machine and generate a model that allocates new samples into one class or the other. An SVM model plots samples in a space so that a large space separates the samples of each class. The new samples are plotted and predicted to a class based on which side of the space they lie. The significance of calling SVM is that it is being taken like a deterministic machine as for a given data x_i, and value of α, the machine generates a similar output y_i or $f(x_i, \alpha)$. A distinct values of α produces a trained machine. Hence, the learning machine is a predetermined architecture with α respective to the weights and biases.

In addition to executing linear classification, SVMs can effectively do non-linear classifications that implicitly map the inputs into large-dimensional spaces. SVM constructs a suitable space achieved by the hyper-plane in an infinite or high-dimensional space with the most significant distance to the nearest training data point of any class. In general, a larger margin lowers the formation error of the classifier. The datasets to discriminate are often not linearly separable in limited dimensional space. The original limited dimensional space was plotted [5] into a much greater dimensional space for creating clear separations.

SVM algorithm is giving significant gains [6] in pattern recognition and machine learning on classification and regression problems. It is popularly implemented in various real-world applications like hand-written character recognition, text classification, image classification, face detection, bioinformatics and many more. However, it is not famous for large data sets, multi-class classification and unbalanced data sets. SVM depends on several parameters. C is one of the penalty parameters which controls between margin maximization and error minimization. Some other parameters control non-linear mapping into feature space. Therefore, automatically tuning multiple parameters for SVMs was proposed [7] and improved performance. Also, SVMs learning with kernels [8] are very effective. Especially, the Redial Basis Function kernel is the most promising.

It was suggested [9] to choose variables using the SVM-based concept by considering the weight vector or error bounds about a variable. Input vectors are nonlinearly mapped to very large feature space, and a linear decision plane is built in C-support vector classification (C-SVC) [10] based on regularization parameter C>0 for two-class classification. The nu-support vector classification (nu-SVC) [11] and nu-support vector regression (nu-SVR) [11] introduced a new parameter nu $\in$ (0, 1] which efficiently controls support vectors and training errors. A high-dimensional distribution's support is estimated via one class SVM [12] for unlabelled data. The epsilon support vector regression (epsilon-SVR) [13] is introduced based on parameters C>0 and epsilon>0 to solve regression problems effectively. The multi-class classification approach is implemented as "one-against-one" [14] for k number of classes and trains data from two classes from each of $k(k - 1)/2$ number of classifiers.

A library for SVMs is called LIBSVM [15]. It supports support vector classification on two or multi-class, support vector regression, and one-class SVM on unlabeled data. It is successfully used in bioinformatics, neuro-imaging, natural language processing, computer vision, and in many domains. In LIBSVM, all SVM formulations are provided as quadratic minimization issues. The shrinking and caching techniques in LIBSVM minimized SVM quadratic problems. The unbalanced data in different classes for classification problems are efficiently controlled in LIBSVM. SVM predicts a class label for classification problems and a target value for regression problems without providing a probability estimate. LIBSVM is the extension of SVM to offer probability information for classification [16], regression [17], and one-class SVM [18]. LIBSVM provides a grid of parameters to select and set values.

A library for large-scale linear classification is called LIBLINEAR [19]. It supports two binary linear classifiers: linear SVM and logistic regression (LR) [20]. Text classification is one of the large-scale classification problems. Linear classification learning techniques are promising for many instances and features with large sparse data. L1-SVM and L2-SVM [21] are developed based on loss function max(1 - yi α xi , 0) and squared loss function max(1 - yi α xi , 0)2 respectively. The LR [22] is implemented based on logarithmic loss from the probabilistic model. The multi-class classification approach is implemented as "one-vs-the-rest" [23] in LIBLINEAR. It inherits LIBSVM.

SVM is enhanced [24] to handle multi-class data with approaches such as directed acyclic graph SVM, and Crammer Singer SVM [25]. The training time of linear SVM is faster than non-linear SVM with a non-linear kernel, but test accuracy is comparable.

SVMs parameters such as penalty parameters and kernel function parameters are optimized by artificial intelligence optimization techniques like particle swam optimization [26, 27], bee colony optimization [28], and genetic algorithm [29]. It improves the performance of the SVMs model in machine learning applications. The various SVM algorithms are applied for a robust and practical solution on AI-based projects.

In deep learning, Convolutional Neural Networks (CNNs), Fully-connected Net-works (FCNs), and Recurrent Neural Networks (RNNs) mainly use softmax activation function for classification task and reduce cross-entropy loss. The linear support vector machine is replaced with a softmax layer in neural nets, and linear SVM reduces margin-based loss, which gives significant improvement [30] over cross-entropy loss.

Since its formation, SVM has been developing, and researchers have put forth several problem formulations, solutions, and techniques for handling SVM. The following section 2 describes the literature survey on state-of-the-art. The methodologies and datasets are demonstrated in section 3 and section 4, respectively. Section 5 illustrates the experiments and results. The performance analysis is discussed in section 6—finally, summaries with the conclusion in section 7.

2. Related Work

Kernel methods are increasingly popular and help in machine learning jobs like classification, regression, etc. The principal focuses on the renowned models based on kernel substitution, i.e., support vector machines (SVM). A general approach is bid-ding to a huge collection of machine learning jobs that can be applied in performing all probable learning machine architectures (RBF networks, feed-forward neural net-works) through a proper selection of kernel. Kernel methods were found to work satisfactorily in the field. The RBF is the appropriate kernel selection in a survey of cur-rent kernel methods [31, 8].

The performance comparison shows that if the number of attributes is much larger than the number of instances and vice versa, then linear SVM [15] is better than the Radial Basis Function (RBF) kernel of nonlinear SVM.
Compared to L1 and L2 SVM [32], the training time of L1 SVM is generally lesser than that of L2-SVM. Also, it is compared that the training by the exact KKT (Karush-Kuhn-Tucker) condition was sometimes slower than the approximate KKT condition. However, after experiments, it has been proved that the estimated KKT condition gives a stable guess of breaching variables, and training time using the estimated KKT situations is usually smaller.

The difficulty of dual and primal optimization [33, 34] can be answered successfully, both for linear and nonlinear SVMs.

SVMs were first introduced in their hard margin formulation, for which dual appears more naturalistic. The soft margin SVMs is selected to make a more robust decision boundary as more training points can be covered, even if the training data are separable. It is believed that an estimated primal solution is generally superior to a dual one since an estimated dual solution can induce a primal one.

Linear SVM [21] is a prevalent implement to contract with massive sparse data. A novel dual method was proposed for large linear SVM with L1-loss and L2-loss functions. This technique is simple and much faster than state-of-the-art solvers.

In a comprehensive survey [35] of current advancements in large-space linear categorization, the performance accuracy of linear SVM classification is closely com-parable with nonlinear SVM classification. Linear SVM has a more rapid training and testing rate than nonlinear SVM. Both linear, as well as nonlinear classifiers are functional in many situations. Linear classification is not limited to a standard outline; it can also be applied in many other places as effectively approaching nonlinear classifiers.

In some conditions, L1-based regularization is better to implement. These methods can supply exciting advancements in model simplification, computer graphics, and image processing. However, in practice, it is suggested [36] to use L2-based regularization methods, as it is uncomplicated and does not raise non-linearity.

Multilayer Perceptron (MLP), SVM, and Library for Large Linear (LIBLINEAR) are used as the agreement for the distinction of handwritten Bangla [37] integers. However, the result shows that all these classifiers are acceptable for this job, but LIBLINEAR is the quickest in the field of distinction outcome for the WBSUCS character dataset.

The effective machine learning process is the classification using a Support Vector Machine. SVM is applied by the Weka tool in which the Radial Basis Function proves to be a methodical Kernel for the categorization of portscan strike [38].

The rule-based classification and association rule mining techniques [39] are applied to UCI [40] medical database to identify the relation between diseases or symptoms and to predict medical diagnosis using Weka [41] tool. Different rule-based classification algorithms are state-of-the-art, and many researchers need help selecting the finest method for a data set. Best techniques are compared with their relative advantages and disadvantages to interpret their applicability in domain-specific tasks.

Resembling the execution of best machine learning classification methods [42] for categorical, continuous, and mixed attribute datasets and which types of attributes and how many instances will give better classification accuracy.

There are optimization methodologies for optimizing the training of SVMs [43] and finding solutions of SVMs.

Many conventional [44] and deep learning [45] based research works on applications and variants of SVMs have been found in the literature. It motivates researchers to work on variant SVMs and compare their merits and demerits.

3. Methods

In this section, different SVM methods have been discussed, and their usages are available in LibSVM [15] and LibLINEAR [19] packages. First, the methods illustrated are present in the LibSVM tool. In LibSVM, there are five types of SVM: three for classification and two for regression. Types of Classification SVM are: C-SVC, nu-SVC, and one-class SVM, and the reverting types are epsilon-SVR and nu-SVR.

In SVM-based classification, the training sample x_i is plotted into a larger dimensional area using Φ. The SVM obtains a linear separating hyper-plane with the most significant boundary in this higher dimensional space. $C > 0$ is the cost factor of the error term. Similarly, the kernel function [31, 8] can be written as equation 1:

$$K(x_i, x_j) \equiv \Phi(x_i)T\,\Phi(x_j) \quad (1)$$

SVM uses many kernel functions. Therefore, how to choose the best kernel function is also an analysis factor. However, for a common purpose, there are a few primary kernel functions equation 2-5:

RBF kernel: $K(x_i, x_j) = exp(-\gamma \left\| x_i - x_j \right\|^2), \gamma > 0$ (2)

Linear kernel: $K(x_i, x_j) = x_i^T x_j$ (3)

Sigmoid kernel: $K(x_i, x_j) = tanh(\gamma\, x_i^T x_j + r)$ (4)

Polynomial kernel: $K(x_i, x_j) = (\gamma\, x_i^T x_j + r)^d, \gamma > 0$ (5)

Here, kernel parameters: γ, r and d denote gamma, coefficient and degree, respectively. Default values of $\gamma=1/k$ *where* $k=1, 2, 3;\ r=0;\ d=0;\ c=1.$
The second tool is LibLINEAR, which includes the following SVC methods [21, 25, 32, 33] for data classification:
i. L1-Regularized L2-Loss SVC
ii. L2-Regularized L1-Loss SVC (dual)
iii. L2-Regularized L2-Loss SVC (dual)
iv. L2-Regularized L2-Loss SVC (primal)
v. Multi-class SVC by Crammer & Singer
The proposed method for this article is RBF kernel and linear kernel in C-SVC and nu-SVC models. The performance of the proposed models is trained and tested with UCI machine learning datasets for non-linear and linear classification. Finally, the results are compared with state-of-the-art SVC models.

4. Materials

WEKA version3 [41] is used here. It is developed by The University of Waikato. It is open-source software for machine learning and it supports the various format of the dataset but by default, it supports the ARFF format. Here, six UCI [40] datasets: Diabetes, Glass, Iris, Letter, Vehicle, and Zoo have been chosen in arff format for the computation, and the weka tool is used for obtaining the Classification Accuracy (CA), AUC-ROC, and RMSE result from these selected datasets. Weka supports various types of packages and some packages are already present in it. Here LibSVM and LibLINEAR packages are used for data classification. Weka3 has a package manager for installing the extra packages as per requirement. So, the required packages are downloaded and installed using the package manager.

LibSVM is unified software used in SVM classification composing C-SVC [10] and nu-SVC [11, 46] methods and for support vector regression using epsilon-SVR and nu-SVR. LibSVM was designed by, Chih-Chung Chang and Chih-Jen Lin. This also supports multi-class classification. LibSVM has several features like different SVM formulas, efficient multi-class classification, cross-validation for model selection, probability estimates, various kernel types to get better classification results, etc. LibSVM version 3.25 is integrated into Weka version3.

A library for large linear classification is LibLINEAR. It was designed by, The Machine Learning Group at National Taiwan University. It is a linear classifier for large-scale data categorization. LibLINEAR version 2.44 is used and installed in Weka version 3.

The following six UCI machine learning datasets are used, as listed in Table 1. These datasets have samples/vectors, attributes/dimensions, classes, and attribute types/features.

Table 1: Characteristics of datasets

Dataset	Vectors	Dimensions	Classes	Features
Diabetes	768	8	2	8 Numeric
Glass	214	9	7	9 Numeric
Iris	150	4	3	4 Numeric
Letter	20000	16	26	16 Numeric
Vehicle	846	18	4	18 Numeric
Zoo	101	17	7	1 Numeric, 16 Nominal

5. Results and Discussion

In the experimentation, classification accuracy depends on essential parameters like cost, gamma, coefficient, and degree. In the case of the RBF kernel, the accuracy result decreases by increasing the value of gamma, and by decreasing the gamma value to get a better result. In most cases, by increasing the cost, we get better accuracy in the RBF kernel. For linear kernel, the result does not depend on its cost parameter but depends on the Nu parameter. In the case of the polynomial kernel type, the accuracy result depends on gamma, coefficient, and degree. The value of the coefficient should increase to get a better result. However, it has been seen that when in-creases the coefficient value up to one particular point, it performs better than the previously selected point. By increasing the degree, the polynomial kernel performs better in data classification. The sigmoid kernel is less preferable because, in most cases, it gives poor results. In the case of Nu-SVC, the value of the Nu parameter should be kept between 0.1 and 0.5 to get the best accuracy result for each dataset. For the Glass dataset, the value of nu=0.2. For the Iris dataset, the value of nu=0.3. For Letter and Zoo dataset, the value of nu=0.1. For Diabetes and vehicle dataset, the value of nu=0.5. 10-fold cross-validation is used for model selection. In the first experiment, it has been tried to get accurate results for linear and non-linear classification using different kernel methods based on C-SVC and Nu-SVC models in the LibSVM tool. These results are shown in Table 2. The second experiment is focused on getting accurate results on large-scale linear classification using L1 or L2 regularized L1 loss SVC and L2 regularized L2 loss SVC models based on regularized hinge loss or logistic loss methods in the LibLINEAR tool. The third experiment has been done to get accurate results for the multi-class SVC method developed by Crammer and Singer. These results are revealed in Table 3.

We have applied all SVM classification techniques of LibSVM and LibLINEAR on nine datasets. Finally, we have chosen the six datasets, which are Diabetes, Glass, Iris, Letter, Vehicle, and Zoo. However, all techniques give different accuracy on a particular dataset. The accuracy results depend on several factors. Several kernel types exist in LibSVM, and many methods are in LibLINEAR. The higher the accuracy value, the performance is good, and the lower the accuracy value, the performance is poor. An AUC value near 1.0 is the best result, and a value near 0.5 is a poor result. RMSE value between 0.1 and 0.4 is the best result, whereas a value above 0.5 is a poor result.

From the first experiment, it has been found that, for the RBF kernel, accuracy becomes 74.77, 97.33, and 98 percent for non-linear classification on glass, iris, and letter datasets, respectively which are the best results among the other kernel methods like linear, polynomial, and sigmoid. Also, it is observed that the polynomial kernel gives better results than the linear kernel for non-linear classification, whereas the sigmoid kernel gives poor results, as shown in Figure 1.

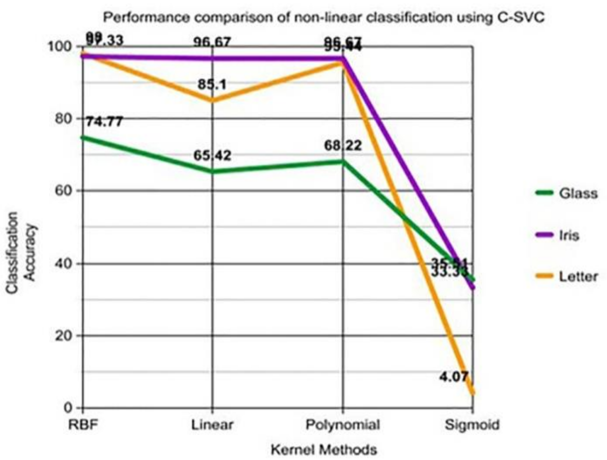

Figure 1. This given model illustrates graphically the statistics of non-linear classification using C-SVC methods, as demonstrated above in Table 2.

Table 2. Performance of SVM methods for classification by Accuracy and AUC value
The top four results are coloured Red, Green, Blue and Orange.

Tools	Model	Method	Accuracy% and AUC for Glass dataset	Accuracy% and AUC for Iris dataset	Accuracy% and AUC for Letter dataset
LibSVM [15]	C-SVC [10]	RBF [8, 31]	74.77 0.814	97.33 0.980	98.00 0.990
		Linear [8, 31]	65.42 0.746	96.67 0.975	85.10 0.923
		Polynomial [8, 31]	68.22 0.777	96.67 0.975	95.44 0.976
		Sigmoid [8, 31]	35.51 0.500	33.33 0.500	4.07 0.500
	Nu-SVC [11]	RBF [8, 31]	68.22 0.773	98 0.985	91.87 0.958
		Linear [8, 31]	30.37 0.565	98 0.985	79.25 0.892
		Polynomial [8, 31]	44.86 0.615	97 0.980	93.03 0.964
		Sigmoid [8, 31]	6.54 0.503	52 0.640	3.00 0.500
LibLINEAR [19]	Regularized and Hinge Loss	L2-R L2-Loss SVC (dual) [21]	44.39 0.631	96 0.970	56.11 0.772
		L2-R L2-Loss SVC (primal) [21]	60.75 0.712	96.67 0.975	69.73 0.843
		L1-R L2-Loss SVC [32]	65.42 0.746	69.47 0.841	69.47 0.841
	Regularized and Logistic Loss	L2-R L1-Loss SVC (dual) [21]	39.72 0.589	94 0.955	59.04 0.787

Table 3. Performance of SVM methods for classification by Accuracy and RMSE value

Tools	Model	Method	Accuracy% and RMSE for Diabetes dataset	Accuracy% and RMSE for Vehicle dataset	Accuracy% and RMSE for Zoo dataset
LibSVM [15]	C-SVC [10]	RBF [8, 31]	65.10 0.590	31.32 0.586	63.37 0.323
		Linear [8, 31]	77.47 0.474	80.26 0.314	96.04 0.106
		Polynomial [8, 31]	55.34 0.668	79.79 0.317	96.04 0.106
		Sigmoid [8, 31]	65.10 0.590	25.77 0.609	40.59 0.412
	Nu-SVC [11]	RBF [8, 31]	65.10 0.590	31.32 0.586	63.37 0.323
		Linear [8, 31]	76.04 0.489	80.26 0.314	96.04 0.106
		Polynomial [8, 31]	55.34 0.668	71.87 0.375	89.11 0.176
		Sigmoid [8, 31]	46.88 0.728	24.82	40.59 0.412
LibLINEAR [19]	Regularized and Hinge Loss	L2-R L2-Loss SVC (dual) [21]	65.76 0.585	70.21 0.385	92.08 0.150
		L2-R L2-Loss SVC (primal) [21]	72.14 0.527	78.01 0.331	97.03 0.092
		L1-R L2-Loss SVC [32]	77.86 0.470	72.10 0.373	94.06 0.130
	Regularized and Logistic Loss	L2-R L1-Loss SVC (dual) [21]	65.63 0.586	70.21 0.385	93.07 0.140
	Multi class	Multi-class SVC by Crammer and Singer [25]	76.17 0.488	70.69 0.382	94.06 0.130

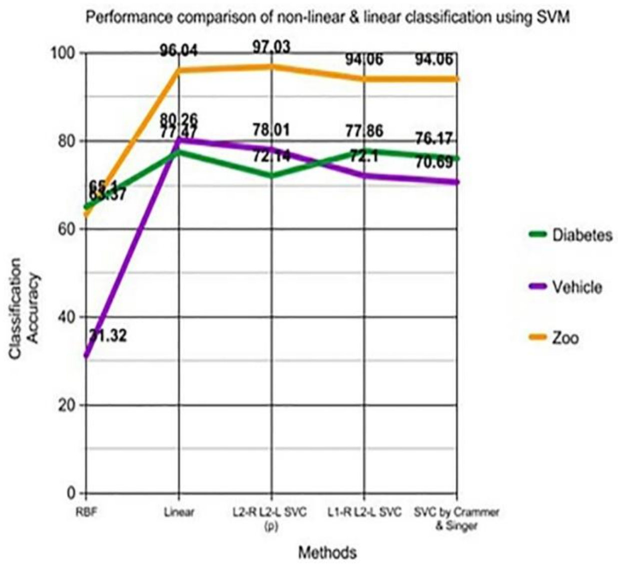

Figure 2. This given model illustrates graphically the statistics of linear classification using Kernel and Regularized methods, as demonstrated above in Table 3.

From the second experiment, it is found that, for the linear kernel, accuracy becomes 77.47, 80.26 and 96.04 percentages for linear classification on diabetes, vehicle and zoo dataset, respectively. In these cases, the linear kernel gives better results than the RBF kernel. Also, L1-R L2-L SVC and L2-R L2-L SVC (primal) give 77.86 and 97.03 percentage accuracy for large-scale linear classification of diabetes and zoo datasets, as shown in Figure 2.

From the third experiment, multi-class SVC by Crammer & Singer method gives an accuracy of 76.17, 70.69 and 94.06 percentages for large-scale linear classification on diabetes, vehicle, and zoo dataset, which are comparable results with other best methods as shown in Figure 2.
RBF kernel performs well because it nonlinearly plots samplings into a larger dimensional area; it has fewer factors than a polynomial kernel.

After performing these experiments, it has been found the top four SVM methods for data classification, which are:
i. L1-Regularized L2-Loss SVC
ii. Multi-class SVC by Crammer & Singer
iii. L2-Regularized L2-Loss SVC (primal)
iv. C-SVC
  a. Radial Basis Function Kernel
  b. Linear Kernel

6. Conclusion

Linear kernel SVMs and non-linear kernel SVMs are used for linear and non-linear classification, respectively. This paper integrates LibSVM and LibLINEAR tools with the Weka tool. The proposed method for this article is RBF kernel and linear kernel in C-SVC and nu-SVC models. The proposed model performed best and compared results with the state-of-the-art SVM methods. The RBF and linear kernel-based SVM model performs better for non-linear and linear classification, respectively. Further-more, the linear SVM performs better than the non-linear SVM on large-scale data.

Conflict of Interest
There is no conflict of interest for this article.

Funding Source
There is no funding source for this article.

Authors' Contributions
The single author of this article is the only contributor.

Acknowledgements
The author is thankful to his parents, wife, and daughter for their support.

References

[1]. Vapnik, Vladimir. The nature of statistical learning theory. Springer science & business media, 1999.
[2]. Hearst, Marti A., et al. "Support vector machines." IEEE Intelligent Systems and their applications 13.4 (1998): 18-28.
[3]. Zhang, Xian-Da, and Xian-Da Zhang. "Support vector machines." A Matrix Algebra Approach to Artificial Intelligence (2020): 617-679.
[4]. James, Gareth, et al. "Support vector machines." An introduction to statistical learning: with applications in R (2021): 367-402.
[5]. Das, Subhankar, and Sanjib Saha. "Data mining and soft computing using support vector machine: A survey." International Journal of Computer Applications 77.14 (2013).
[6]. Cervantes, Jair, et al. "A comprehensive survey on support vector machine classification: Applications, challenges and trends." Neurocomputing 408 (2020): 189-215.
[7]. Chapelle, Olivier, et al. "Choosing multiple parameters for support vector machines." Machine learning 46 (2002): 131-159.
[8]. Schölkopf, Bernhard, Alexander J. Smola, and Francis Bach. Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT press, 2002.
[9]. Rakotomamonjy, Alain. "Variable selection using SVM-based criteria." Journal of machine learning research 3.Mar (2003): 1357-1370.
[10]. Cortes, Corinna, and Vladimir Vapnik. "Support-vector networks." Machine learning 20 (1995): 273-297.
[11]. Schölkopf, Bernhard, et al. "New support vector algorithms." Neural computation 12.5 (2000): 1207-1245.
[12]. Schölkopf, Bernhard, et al. "Estimating the support of a high-dimensional distribution." Neural computation 13.7 (2001): 1443-1471.
[13]. Vapnik, Vladimir N. "An overview of statistical learning theory." IEEE transactions on neural networks 10.5 (1999): 988-999.
[14]. Knerr, Stefan, Léon Personnaz, and Gérard Dreyfus. "Single-layer learning revisited: a stepwise procedure for building and training a neural network." Neurocomputing: algorithms, architectures and applications. Springer Berlin Heidelberg, 1990.
[15]. Chang, Chih-Chung, and Chih-Jen Lin. "LIBSVM: a library for support vector machines." ACM transactions on intelligent systems and technology (TIST) 2.3 (2011): 1-27..
[16]. Wu, Ting-Fan, Chih-Jen Lin, and Ruby Weng. "Probability estimates for multi-class classification by pairwise coupling." Advances in Neural Information Processing Systems 16 (2003).
[17]. Lin, Chih-Jen, and Ruby C. Weng. "Simple probabilistic predictions for support vector regression." National Taiwan University, Taipei (2004).
[18]. Que, Z., and Lin, C J. "One-class SVM probabilistic outputs." Technical report, National Taiwan University, 2022. URL http://www.csie.ntu.edu.tw/~cjlin/papers/oneclass_prob/oneclass_prob.pdf.

[19]. Fan, Rong-En, et al. "LIBLINEAR: A library for large linear classification." the Journal of machine Learning research 9 (2008): 1871-1874.

[20]. Boser, Bernhard E., Isabelle M. Guyon, and Vladimir N. Vapnik. "A training algorithm for optimal margin classifiers." Proceedings of the fifth annual workshop on Computational learning theory. 1992.

[21]. Hsieh, Cho-Jui, et al. "A dual coordinate descent method for large-scale linear SVM." Proceedings of the 25th international conference on Machine learning. 2008.

[22]. Lin, Chih-Jen, Ruby C. Weng, and S. Sathiya Keerthi. "Trust region newton methods for large-scale logistic regression." Proceedings of the 24th international conference on Machine learning. 2007.

[23]. Keerthi, S. Sathiya, et al. "A sequential dual method for large scale multi-class linear SVMs." Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. 2008.

[24]. Chauhan, Vinod Kumar, Kalpana Dahiya, and Anuj Sharma. "Problem formulations and solvers in linear SVM: a review." Artificial Intelligence Review 52.2 (2019): 803-855.

[25]. Crammer, Koby, and Yoram Singer. "On the learnability and design of output codes for multiclass problems." Machine learning 47 (2002): 201-233.

[26]. Tan, Xian, Fasheng Yu, and Xifeng Zhao. "Support vector machine algorithm for artificial intelligence optimization." Cluster Computing 22 (2019): 15015-15021.

[27]. Chen, Yanhua, et al. "A hybrid application algorithm based on the support vector machine and artificial intelligence: An example of electric load forecasting." Applied Mathematical Modelling 39.9 (2015): 2617-2632.

[28]. Chiang, Hsiu-Sen, et al. "A novel artificial bee colony optimization algorithm with SVM for bio-inspired software-defined networking." International Journal of Parallel Programming 48 (2020): 310-328.

[29]. Frohlich, Holger, Olivier Chapelle, and Bernhard Scholkopf. "Feature selection for support vector machines by means of genetic algorithm." Proceedings. 15th IEEE International Conference on Tools with Artificial Intelligence. IEEE, 2003.

[30]. Tang, Yichuan. "Deep learning using linear support vector machines." arXiv preprint arXiv:1306.0239 (2013).

[31]. Campbell, Colin. "Kernel methods: a survey of current techniques." Neurocomputing 48.1-4 (2002): 63-84.

[32]. Koshiba, Yoshiaki, and Shigeo Abe. "Comparison of L1 and L2 support vector machines." Proceedings of the International Joint Conference on Neural Networks, 2003.. Vol. 3. IEEE, 2003.

[33]. Chapelle, Olivier. "Training a support vector machine in the primal." Neural computation 19.5 (2007): 1155-1178.

[34]. Nalepa, Jakub, and Michal Kawulok. "Selecting training sets for support vector machines: a review." Artificial Intelligence Review 52.2 (2019): 857-900.

[35]. Yuan, Guo-Xun, Chia-Hua Ho, and Chih-Jen Lin. "Recent advances of large-scale linear classification." Proceedings of the IEEE 100.9 (2012): 2584-2603.

[36]. Van Den Doel, Kees, Uri Ascher, and Eldad Haber. "The lost honour of l2-based regularization." Large Scale Inverse Problems, Radon Ser. Comput. Appl. Math 13 (2012): 181-203.

[37]. Halder, Chayan, Jaya Paul, and Kaushik Roy. "Comparison of the classifiers in Bangla handwritten numeral recognition." 2012 International Conference on Radar, Communication and Computing (ICRCC). IEEE, 2012.

[38]. Vidhya, M. "Efficient classification of portscan attacks using Support Vector Machine." 2013 International Conference on Green High Performance Computing (ICGHPC). IEEE, 2013..

[39]. Datta, R. P., and Sanjib Saha. "Applying rule-based classification techniques to medical databases: an empirical study." International Journal of Business Intelligence and Systems Engineering 1.1 (2016): 32-48.

[40]. Blake, C. and Merz, C. J. "UCI repository of machine learning datasets." University of California, Irvine, Dept. of Information and Computer Sciences.(http://www.cs.waikato.ac.nz/~ml/weka/)

[41]. WEKA3 tool for machine learning and knowledge analysis. Online available at http://www.cs.waikato.ac.nz/~ml/weka/

[42]. Saha, Sanjib, and Debashis Nandi. "Data Classification based on Decision Tree, Rule Generation, Bayes and Statistical Methods: An Empirical Comparison." Int. J. Comput. Appl 129.7 (2015): 36-41.

[43]. Shawe-Taylor, John, and Shiliang Sun. "A review of optimization methodologies in support vector machines." Neurocomputing 74.17 (2011): 3609-3618.

[44]. Saha, Sanjib. "Non-rigid Registration of De-noised Ultrasound Breast Tumors in Image Guided Breast-Conserving Surgery." Intelligent Systems and Human Machine Collaboration. Springer, Singapore, 2023. 191-206.

[45]. Saha, Sanjib, et al. "ADU-Net: An Attention Dense U-Net based deep supervised DNN for automated lesion segmentation of COVID-19 from chest CT images." Biomedical Signal Processing and Control 85 (2023): 104974.

[46]. Chen, Pai-Hsuen, Chih-Jen Lin, and Bernhard Schölkopf. "A tutorial on ν-support vector machines." Applied Stochastic Models in Business and Industry 21.2 (2005): 111-136.

AUTHORS PROFILE

Sanjib Saha earned his Bachelor of Engineering and Master of Technology from Burdwan University and Jadavpur University respectively. He is pursuing PhD in Computer Science and Engineering at National Institute of Technology, Durgapur, India. He is working as an Assistant Professor in Department of Computer Science and Engineering at Dr. B. C. Roy Engineering College, Durgapur. He has published research papers in SCI and Scopus journals including conferences. His main research work focuses on Machine Learning, Deep Learning, and Medical Image Classification, Segmentation & Registration.

