ICIC Express Letters
Part B: Applications
Volume **13**, Number **5**, May **2022**

ICIC International ©2022 ISSN 2185-2766
pp. **479–486**

IMPROVING EFFICIENCY OF SUPPORT VECTOR MACHINE CLASSIFIER WITH FEATURE SELECTION

Auapong Yaicharoen[1,2], Kotaro Hashikura[3], Md Abdus Samad Kamal[3] and Kou Yamada[3]

[1]Electronic and Telecommunication Engineering
King Mongkut's University of Technology Thonburi
126 Pracha Uthit Rd., Bang Mot, Thung Khru, Bangkok 10140, Thailand
auapong.yai@mail.kmutt.ac.th

[2]Graduate School of Science and Technology
[3]Division of Mechanical Science and Technology
Gunma University
1-5-1 Tenjincho, Kiryu, Gunma 376-8515, Japan
{ t202b001; k-hashikura; maskamal; yamada }@gunma-u.ac.jp

Received September 2021; accepted November 2021

Abstract. *In this paper, a relationship among feature selection, time to train a classifier and quality of that classifier is investigated. By choosing different sets of features in a data set to build classifiers, every information from processing time to the quality of each classifier is inspected. The focus is on time used to create a classifier and accuracy of that classifier. Four data sets are used in this investigation. The results from our investigation show that, most of the time, not all features in the data set are necessary to build a good classifier. Those features with higher importance are the ones needed. Also, when an optimum value of threshold is set, a train data set with features that have total value of their importance equal to or better than that threshold can be used to create an equally good quality classifier as the original but required less execution time.*
Keywords: Principal component analysis, Support vector classification, Feature selection

1. **Introduction.** In the information age, applications and research topics in data analytic become increasingly important. One of the well-known fields in data analytic and machine learning is "Big Data", coined and popularized by John Mashey and his scientist cohort at Silicon Graphics in mid-1990s [1]. The three main characteristics of "Big Data" are volume, velocity, and variety [1,2]. This means the system for Big Data has to handle a large number of information (volume) that moves fast (velocity) and in various forms (variety). These properties allow researchers to choose any information of interests. However, some drawbacks such as an amount of storage required [2,3] to keep all the information, some irrelevant pieces of information that have been stored among all those useful ones in the system [3] may require a lot of computational resources to process and yield no useful information. Among the tasks perform on these data, classification is one of those important tasks. While training the classifier with relevant information can help create a good classifier, using misleading information may yield the opposite result. In this paper, the amount of information in terms of data features, will be the focal point of our investigation. For a given data set, if the useful and useless information can be separated and carefully chosen to train the classifier, the resulting classifier should perform as well as the one trained by all information. To optimize the use of these resources, many data optimization techniques have been proposed, such as principal component analysis (PCA), Isomap, and Diffusion maps [4]. Also, there is a feature engineering (FE) method

DOI: 10.24507/icicelb.13.05.479

where steps of feature transformation or creation have been done to improve classification performance [5] or combine FE with feature learning to increase accuracy of the classification [6]. In this paper, the dimensionality reduction technique (and also one of the FE tools) proposed in 1901 by Pearson [7] is chosen because of its simplicity and ability to create a more relevant feature vector space. This characteristic ensures that the redundant information will not be processed; therefore, the processing time should be reduced. Also, in this paper, a machine learning model proposed in 1995 by Vapnik and Uapnik [8] called support vector classification (SVC), is used to demonstrate the effect of dimensionality reduction on a classifier.

The purpose of this paper is to show that a combination of feature reduction and selection can be used to reduce the computation resources when training a classifier and still yields a comparable quality result. This paper is organized as follows. In Section 2, we summarize preliminaries and explain the problem considered in this paper. In Section 3, we propose a new modification to improve the SVM classifier. In Section 4, the results of our tests are shown, and the meanings are explained. Section 5 gives concluding remarks.

2. **Problem Statement and Preliminaries.** Dealing with a large information usually requires a lot of resources. In our previous study [9], we proposed selecting the most important component in data features using PCA technique and used it to train a classifier. We found that the reduction of information helps reduce the training time. However, accuracy of some classifiers is worsened. Also, in that study, the benefit is not clearly obtained and there are some outliers in the results where some of the training time increased significantly (approximately 5 times of the regular SVC) even though the number of information has been reduced by 50%. In this paper, that drawback of reducing too much information from a data set on a classifier training time and quality is explained in Section 4. Also, we proposed how to effectively choose important information and performed thorough investigation on several samples of real-world data sets to confirm the validity of the proposed method. The decision criteria on factors from dimensionality reduction technique, feature selection and optimal importance threshold setting, are presented. Finally, the benefit of the proposed solution in terms of classifier training time and the quality of each classifier are measured.

3. **Proposed Method.** In this paper, we proposed that combining two popular techniques in machine learning, which are PCA and SVC, with the right criteria of feature selection, the classifier training time can be reduced while its quality is preserved. A set of experiment steps has been performed on four different data sets to demonstrate the results of our proposed method. The information of each data set is shown in Table 1.

For each data set, the following experiment steps have been performed.

STEP 1: Separate data for training and testing

The first step is to randomly divide all data samples into a training and a testing data set. The training data set contains 80% from the data samples and the testing data set contains 20% from the data samples. This is one of the cross validation techniques called a holdout method. The K-fold cross validation is not used here because it takes more time and from our earlier experiments on the random data set, the accuracy scores do not vary much among each test.

STEP 2: Create three groups of data sets for training

In this step, three groups of input are generated based on number of features and will be later used to train an SVM. The time used to create each data set is recorded and shown in the row labelled *PCA execution time (second)* of every table.

Group 1: Original data set with all features

In this group, all features in the data set will be used when running the SVM. The result of this group will be shown in the first column of the result table. In this group,

Table 1. Detailed information of four data sets used in this study

Data set	Source	Description	Samples	Features
1. data_1	Randomly generated using *make_blobs* function in the *sklearn* library obtained from scikit-learn.org	A data set consists of two clusters. Each cluster contains 50,000 Gaussian distributed samples and can be linearly separable.	100,000	2
2. iris	Obtain by calling the built-in function sklearn.datasets.load_iris which is a part of sklearn library from scikit-learn.org	A data set of three types of iris flowers (Setosa, Versicolour, and Virginica) with petal and sepal information (length and width) as features.	150	4
3. heart disease	The Cleveland database on heart disease obtained from kaggle.com [10]	A heart disease data set. The original data set consists of 76 features where this data set uses only subset of 14 features to determine if there is a presence of heart disease or not.	303	14
4. credit card	A data set containing credit card transaction from European cardholders in September 2013. Available on kaggle.com [11]	A data set used to determine if there will be a credit card fraudulent or not based on 30 features.	284,807	30

the time to perform the PCA task is equal to 0 and the total time consists of only the SVM training time. The execution time and accuracy score of the first column (Group 1) in each table are used as baseline data to compare with results from other groups.

Group 2: Reduced data set from one feature

The data set in this group is obtained by performing a PCA technique on the original data set to find an eigen value and an eigen vector related to each feature. In the first sub-column of *Group 2*, a feature with the highest eigen value, which means the highest importance, is chosen and used to reduce the dimensionality of the data set. The result data set from this reduction is then used by the SVM to create a classifier. The execution time of running PCA on the original data set and the accuracy score of the obtained classifier are shown in the first column of Group 2. The second feature (data sets 1-4), third feature (data sets 2-4) and fourth feature (data set 2) are also selected and used to reduce the original data set. The results are shown in the related sub-column of *Group 2* to show the differences in execution time and accuracy score in case the less important features are used.

Group 3: Reduced data set from a group of features that their total importance value meets the required threshold

After running a PCA to obtain a set of eigen values and eigen vectors, the algorithm will select a group of features based on the following criteria.

$$\Sigma(\text{eigen values from selected features})/\Sigma(\text{all eigen values}) \geq \text{Desired threshold} \quad (1)$$

The desired thresholds used in this paper are 90% and 95% in some cases. Then the original data set is converted to a reduced data set based on that selected group of features.

STEP 3: Using support vector classification technique on data obtained from step 2

Each of the data set obtained from step 2 will be used as an input for an SVM to obtain a classifier. The time used to train the classifier is recorded and shown in each table in the row labelled *SVC execution time (second).*

STEP 4: Test the accuracy of the classifier

Group 1: To test the accuracy of the obtained classifier, the test data set will be used with the classifier and the confusion matrix and accuracy score are calculated.

Groups 2 and 3: To test the accuracy of the classifier, the test data set has to be reduced using the same selected features as the training set. Then use the reduced test set with the classifier to obtain the confusion matrix and calculate the accuracy score.

The results accuracy scores are shown in the row labelled *Accuracy score*.

4. **Results.** In this section, the results of our tests are shown, and the meanings of these numbers are explained.

The result table has the following format.

The first column contains an explanation of data in each row.

The second column (*Group 1*) contains the results obtained from running the SVC on the original data.

The third column (*Group 2*) contains the results obtained from reducing the data set using only one feature before running the SVC on the reduced data set.

The last column is the result when performing the SVC on a reduced data set which uses a combination of features starting from the one with the highest importance and the ones with second highest importance and so on until the summation of importance reach the desired threshold. In our paper, the desired threshold is set to 90% (and 95% in some tables).

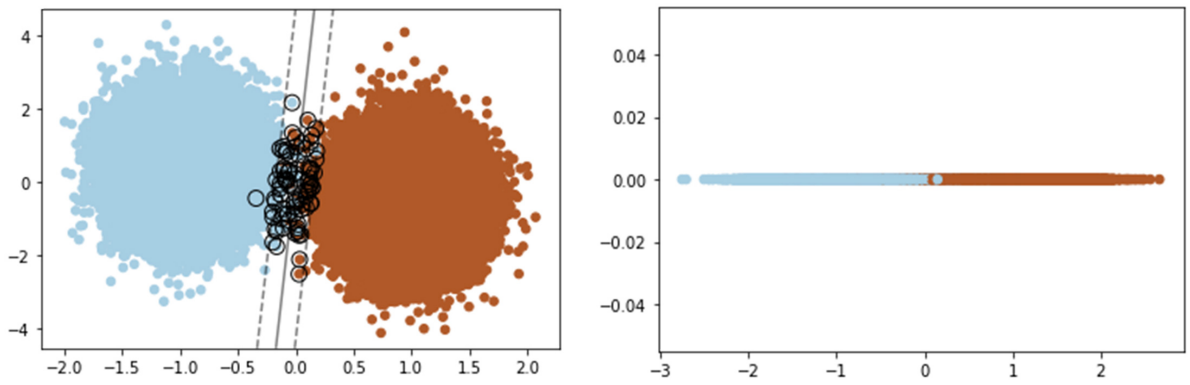
First, a set of randomly generated data with 100,000 samples, each sample has 2 features is investigated. This data set contains 2 clusters. The details and results of classification are shown in Table 2.

Table 2. Results from our first data set (randomly generated data)

data_1 Sample size = 100,000 Feature size = 2	Group 1 Original SVM	Group 2 PCA with 1 feature and SVM		Group 3 PCA with features importance $\geq 90\%$ and SVM
Feature chosen	All	1	2	1, 2
% importance	100.00	73.62	26.38	100.00
PCA execution time (s)	0	0.0053	0.0039	0.0032
SVC execution time (s)	0.2868	61.8487	156.0622	0.3116
No. of support vectors	84	11058	54706	84
Accuracy score	0.9999	0.9447	0.6926	0.9999

In the first test, the difference of execution time between performing the SVC on the data set and performing PCA (selecting either feature 1 or feature 2) followed by the SVC on the reduced data set, is significantly large (0.2868 seconds vs. 61.8487 or 156.0622 seconds). This is not what we expected, since our assumption is that by using the less features in the data set it should lead to the faster training time. Upon further inspection, the reason behind the time difference was found. The reduced data set has many data samples from cluster 1 and cluster 2 mixed together (same values after reduced to the selected feature), which causes no clear boundary between two clusters.

Therefore, SVC took longer time to find the support vectors. This can be seen in the *Number of support vectors* row. The number of support vectors obtained from using the SVC alone is 84, which can be seen as those samples in black circles in Figure 1(a), while from the PCA and SVC are 11058 and 54706. In Figure 1(b), there is an overlap area between blue samples and brown samples, and the number of samples in that area is 11058, which is difficult to notice in this plot. Notice that, in both cases of the reduced method, the accuracy scores are also noticeably lower than using only the SVC since the importance of features 1 and 2 is both lower than 90%.



(a) (b)

FIGURE 1. (color online) 2D plots from 100,000 samples where x-axis represents value in feature 1 and y-axis represents value in feature 2: in Figure 1(a), the SVC technique is performed on original data, and in Figure 1(b), the SVC technique is performed after feature reduction

In the second experiment, the iris data set, which is a well-known data set for machine learning study, is being observed. Since this data set contains only four features, the results from all features are shown.

TABLE 3. Results from the iris data set

iris Sample size = 150 Feature size = 4	Group 1	Group 2				Group 3
	Original SVM	PCA with 1 feature and SVM				PCA with features importance ≥ 90% and SVM
Feature chosen	All	1	2	3	4	1, 2
% importance	100.00	72.34	22.98	4.12	0.56	95.32
PCA execution time (s)	0	0.0003	0.0004	0.0004	0.0007	0.0008
SVC execution time (s)	0.0012	0.0009	0.0013	0.0011	0.0017	0.0008
No. of support vectors	25	44	107	116	116	33
Accuracy score	0.9000	0.9667	0.3000	0.1667	0.1667	0.9333

The result from the table shows that when using the support vector machine technique with the original data set without any dimensionality reduction, the number of support vectors obtained is 25 and the accuracy score is 0.9000. However, when a PCA technique is used to find principal components among the data features, the higher value of the component, the higher the accuracy score in the classification. For example, using only feature with 72.34 percent importance and then performing the SVC technique on the reduced data can lead to the classification result with accuracy score of 0.9667. Notice that the result in this case yields a better accuracy than when using all features. So, it might be possible that some features are less useful for classifying the data cluster. For the execution time viewpoint, the feature reduction process followed by the SVC does not cost much time; in fact, with the feature with the highest importance case, the total execution time is the same.

The third test case uses a heart disease data set from kaggle website. This data set has 303 samples, and each sample contains 13 features.

The results from this data set confirm that the importance of features plays an important part on the quality of a classifier. The feature with a higher percent importance leads to a higher accuracy score of the classifier. However, in this test data set, the additional last column (threshold 95% or more) is added to show that adding more features may not

Table 4. Results from the heart disease data set

heart disease Sample size = 303 Feature size = 13	Group 1	Group 2			Group 3	
	Original SVM	PCA with 1 feature and SVM (Showing 3 highest weighted features)			PCA with features importance ≥ 90% and SVM	PCA with features importance ≥ 95% and SVM
Feature chosen	All	1	2	3	1, 2, 3, 7, 8, 12, 13, 11, 10, 9	1, 2, 3, 7, 8, 12, 13, 11, 10, 9, 6
% importance	100.00	20.83	11.74	10.05	94.16	97.32
PCA execution time (s)	0	0.0020	0.0009	0.0007	0.0034	0.0035
SVC execution time (s)	0.0049	0.0015	0.0023	0.0019	0.0026	0.0032
No. of support vectors	100	124	224	227	100	100
Accuracy score	0.9016	0.8525	0.5738	0.5902	0.9016	0.9016

improve the accuracy of the classifier once the accuracy score reaches the highest value. (However, in a real situation, we cannot know the highest accuracy score, though.) Since the accuracy score at 90% is already equal to the accuracy score of the original SVC result, adding more features to the data set may not improve the accuracy score. The total execution time in case of using only reduced data with one feature is all lower than the time used when performing SVC directly on the original data set with 13 features. However, when performing the feature selection to obtain a group of features that reach 90% or 95% importance threshold, the total execution time of both cases exceeds the original method since at least 10 features have to be inspected and combined and this process takes more time than when performing the data reduction of only one feature.

The last data set is a huge set of data. This data set is chosen because of its size and number of features. The sample size is 284807 and each sample has 30 features. It is used to detect if there will be a credit card fraudulent or not.

Table 5. Results from the credit card data set

credit card data Sample size = 284,807 Feature size = 30	Group 1	Group 2			Group 3	
	Original SVM	PCA with 1 feature and SVM (Only 3 highest weighted features are shown)			PCA with features importance ≥ 90% and SVM	PCA with features importance ≥ 95% and SVM
Feature chosen	All	1	2	3	26 features	27 features
% importance	100.00	6.24	5.68	0.14	92.77	95.87
PCA execution time (s)	0	0.1518	0.0972	0.0987	0.1363	0.1816
SVC execution time (s)	7015.91	3.4591	3.2655	3.3367	3848.57	5299.64
No. of support vectors	466	783	783	783	473	415
Accuracy score	0.9994	0.9982	0.9982	0.9982	0.9994	0.9994

For this data set, the difference of sample size between two classes is significantly large, since class 1 (fraudulent transaction) has only 423 samples while class 0 has 284,384 samples. Therefore, when only one feature is selected to reduce original data using PCA technique before training a classifier, the result obtained is insignificantly varied from the original data. For example, using only feature 1 to train the SVM, the confusion matrix from the classification is $[[56861\ 0], [101\ 0]]$. While using all features to train the SVM, the result classifier yields a confusion matrix of $[[56847\ 14], [19\ 82]]$. Even

though, the classification of data for class 1 is all wrong, the accuracy score is barely lower than the original classifier. Therefore, no matter how small the importance score of that feature is, when used to convert data set using a PCA technique, the result caused by misclassification is still very small when computing an accuracy score. This may be the reason why the accuracy score does not change much even when the importance score of selected features is very low. However, since the sample size is very large, the effect of time saved from feature reduction is noticeable. For example, running SVC on the original data set takes almost 2 hours while reducing data set to one feature takes less than 4 seconds. Even using 26 features, to reach 90% importance threshold, reduce computation time almost 50%.

5. **Conclusions.** With the right criteria, feature selection can significantly improve training time of a classifier while its quality is intact. The effects of selecting different features on a classifier training time and quality are shown. A series of experiments are performed on four different data sets. The results from the first data set show that choosing the maximum weighted feature with insufficient importance can cause the reduced data to become noisy (many overlap data points between two clusters). The noisy data increases classifier training time and creates a bad classifier. The second data set shows that not all features are equally important. This can be seen when the classifier created from a reduced data set has a better accuracy score than the one created from the original data set. The third data set shows that when each feature has low important score, more features are required to train the classifier. In the last data set, the significant imbalance in number of the data between two clusters causes the accuracy result to be almost the same no matter which feature or how many features are selected. On the execution time point of view, when there are a lot of features presented in the data set, the SVM alone used more time to perform the training. However, if the SVM is performed after feature selection, the training time is usually reduced. Except in the first data set, when the reduced data set become noisy, the SVM takes much more time to train the classifier. The quality of the classifier also becomes worse than the original one. The PCA process when combined with the SVM can take more time than performing only the SVC technique if the number of features is close to the total number of features. However, when the number of selected features is much lower than the number of all features, the total execution time is usually smaller. Our experiments show that with the right feature selection criteria, number of features and importance value threshold, the training time and the quality of the classifier can be optimized.

REFERENCES

[1] F. X. Diebold, *On the Origin(s) and Development of "Big Data": The Phenomenon the Term, and the Discipline*, PIER Working Paper No. 12-037, 2012.

[2] S. Mukherjee and R. Shaw, Big data – Concepts, applications, challenges and future scope, *International Journal of Advanced Research in Computer and Communication Engineering*, vol.5, no.2, pp.66-74, 2016.

[3] R. Patgiri, Issues and challenges in big data: A survey, in *Distributed Computing and Internet Technology. ICDCIT 2018. Lecture Notes in Computer Science*, A. Negi, R. Bhatnagar and L. Parida (eds.), Cham, Springer International Publishing, 2018.

[4] L. van der Maaten, E. Postma and J. Van Den Herik, Dimensionality reduction: A comparative review, *J. Mach. Learn. Res.*, vol.10, pp.66-71, 2009.

[5] T. Rawat and V. Khemchandani, Feature engineering (FE) tools and techniques for better classification performance, *International Journal of Innovations in Engineering and Technology (IJIET)*, vol.8, no.2, pp.169-179, 2017.

[6] I. A. Siradjuddin, A. Sakinah and M. K. Sophan, Combination of feature engineering and feature learning approaches for classification on visual complexity images, *International Journal of Innovative Computing, Information and Control*, vol.17, no.3, pp.991-1005, 2021.

[7] K. Pearson, On lines and planes of closest fit to systems of points in space, *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol.2, no.11, pp.559-572, DOI: 10.1080/14786440109462720, 1901.
[8] C. Cortes and V. Vapnik, Support-vector networks, *Machine Learning*, vol.20, pp.273-297, 1995.
[9] A. Yaicharoen and K. Yamada, Improving support vector classification efficiency with principal component analysis, *The 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pp.862-865, 2021.
[10] *Heart Disease UCI Data Set*, https://www.kaggle.com/ronitf/heart-disease-uci, Accessed on July 9, 2021.
[11] *Credit Card Fraud Detection Data Set*, https://www.kaggle.com/mlg-ulb/creditcardfraud, Accessed on July 9, 2021.