Bias and Variance Multi-Objective Optimization for Support Vector Machines Model Selection

Alejandro Rosales-Pérez[1], Hugo Jair Escalante[1], Jesus A. Gonzalez[1], Carlos A. Reyes-Garcia[1], and Carlos A. Coello Coello[2]

[1] National Institute of Astrophysics, Optics and Electronics (INAOE)
Computer Science Department
Tonantzintla, Puebla, Mexico
{arosales,hugojair,jagonzalez,kargaxxi}@inaoep.mx
[2] CINVESTAV-IPN. Computer Science Department
Mexico City, Mexico
ccoello@cs.cinvestav.mx

Abstract. In this paper, we describe a novel model selection approach for a SVM. Each model can be composed by a feature selection method and a pre-processing method besides the classifier. Our approach is based on a multi-objective evolutionary algorithm and on the bias-variance definition. This strategy allows us to explore the hyperparameters space and to select the solutions with the best bias-variance trade-off. The proposed method is evaluated using a number of benchmark data sets for classification tasks. Experimental results show that it is possible to obtain models with an acceptable generalization performance using the proposed approach.

Keywords: Support vector machines; Model selection; Bias-variance trade-off; Multi-objective optimization

1 Introduction

Support vector machines (SVMs) are supervised learning algorithms able to build a classification model from a labeled data set. Due to their high performance and scalability, SVMs have gained popularity in regression and classification tasks. Nevertheless, as many other learning algorithms, SVMs have some parameters whose fine-tuning can affect their performance. These tunable parameters are called hyperparameters and determining their appropriate values is a key issue when using SVMs; this problem is known as *model selection*. Additionally, other components that could improve SVMs' performance are the set of training features and the type of preprocessing applied to the data. Therefore, a classification model can be seen as a combination of those components with their corresponding hyperparameters values.

The final goal of the model selection task is that the selected model has the highest possible generalization ability. Since generalization is the key, this implies that the model should predict new samples with the lowest possible error.

It is well known that error can be decomposed into two components: bias and variance. However, computing bias and variance is not straightforward, due to the fact that the target model and the probability density function that generated the data are usually unknown. Nevertheless, approximate values of them can be obtained from a data set with a finite number of samples.

Bias and variance are closely related to the model accuracy and complexity. In general, bias describes the extent to which the systematic error of the learning algorithm contributes to the error, while variance describes the extent to which variations in the training data or a random behavior of the learning algorithm contributes to the error [11]. Therefore, both components should be as minimum as possible in order to get a better generalization performance. Nonetheless, they are conflicting components and the best model is the one which has the best bias-variance trade-off. So, in this sense, model selection can be seen as a multi-objective optimization problem.

Previous studies have faced the SVM model selection task as a multi-objective optimization problem, and some of them try to minimize the number of features and an estimation of the generalization error [9], looking to construct diverse models for being considered into an ensemble; others try to reduce the error rate between the positive and negative classes which are defined as the objectives [2, 3, 12], looking to mitigate the effect of the majority class in the model selection task for unbalanced datasets. Other works have considered the accuracy and the number of support vectors as the objectives to be optimized [1,15], under the assumption that the number of support vectors is directly associated to the model complexity, but this assumption does not apply for full model selection[3]. To the best of our knowledge, estimated values of bias and variance have not been previously used in a multi-objective approach for model selection. In this paper, we propose a multi-objective evolutionary algorithm for model selection, optimizing bias and variance estimates, which are approximated from a finite data set. We used the NSGA-II [5] as our search algorithm because of its efficiency and because it can provide a diverse set of solutions along the Pareto front. We evaluate our proposal using a series of benchmark data sets for classification. Our experimental results show that our proposed approach selects highly effective classification models, when compared with respect to single-objective formulations, and a related method for model selection.

2 Multi-Objective Optimization

A multi-objective optimization problem (MOOP) is defined as follows:

$$\begin{aligned} &\text{minimize } \mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), \ldots, f_l(\mathbf{x})] \\ &\text{subject to } g_i(\mathbf{x}) \leq 0 \; i = 1, \ldots, p \\ &\qquad\qquad h_j(\mathbf{x}) = 0 \; j = 1, \ldots, q \end{aligned}$$

[3] The full model selection problem consist of choosing a combination of pre-processing, features selection methods, and learning algorithm together with their hyperparameters

where $\mathbf{x} = [x_1, \ldots, x_n] \in \mathbb{R}^n$ is a decision variables vector, l is the number of objectives, and $p + q$ is the number of constraints.

The notion of optimum in MOOP refers to obtaining a good trade-off between the objectives. In order to establish this trade-off, the most accepted notion of optimum in MOOP is the so-called *Pareto optimality*.

Most modern multi-objective optimization algorithms (MOEAs) use the concept of **Pareto dominance** to determine if a solution is better than another. We say that a solution $\mathbf{x^{(1)}}$ dominates a solution $\mathbf{x^{(2)}}$ ($\mathbf{x^{(1)}} \preceq \mathbf{x^{(2)}}$) if and only if $\mathbf{x^{(1)}}$ is better than $\mathbf{x^{(2)}}$ at least in one objective and it is not worse in the rest.

The notion of **Pareto optimality** says that a solution $\mathbf{x}^*$ is a Pareto optimal if it is not possible to improve one objective without worsening another. This definition does not produce a single solution, but a set of trade-off solutions among the different objectives. The set of trade-off solutions (in decision variable space) is known as **Pareto optimal set**. The objective function values corresponding to the elements of the Pareto optimal set constitute the so-called **Pareto front**. The use of MOEAs presents some advantages, because evolutionary algorithms are less susceptible than mathematical programming techniques to the shape and the continuity of the Pareto front. Additionally, MOEAs require less problem domain information to operate than mathematical programming techniques.

3 Bias-Variance Trade-Off

From a statistical point of view, the expected error over a sample can be decomposed into two components: the squared bias and the variance. In a general sense, square bias is a measure of the contribution to the error of the central tendency (i.e. the class with the most votes across the multiple predictions) when a model is trained with different data sets. The variance is a measure of the deviations to the central tendency when a model is trained with different data sets [16].

In order to obtain a better generalization error, both components should be minimized. However, reducing one of them causes an increment in the other one. This is known as the bias-variance dilemma [8]. In general, it is said that a model with low bias is too flexible and has a low training error rate, but its generalization capability is poor; this is known as the *overfitting* problem. In contrast, a model with low variance is too simple, has low complexity and does not have the ability to learn the training set and its generalization performance is also poor; this is known as the *underfitting* problem. Therefore, a good model is the one which provides a good trade-off between these two components. So, we face the model selection task as a multi-objective optimization problem. We used as objectives estimates of bias and variance, trying to select the model with the best trade-off between both components.

In classification tasks, different ways to estimate the bias and the variance have been proposed [8,10,11,16]. Notwithstanding the different definitions, all of them are able to give insights of the bias and variance contribution to the model error. In our study we adopted the Webb's definition [10], because it is close to the bias/variance decomposition formulated for regression tasks. This definiton

is based on the idea of training N models with different partitions of a data set. Then, the N models are tested using the samples that were not used during the training phase. The predictions are recorded and finally an estimation of bias and variance is computed based on a given definition.

4 SVM Multi-Objective Model Selection

The main goal of our study is to select a classification model that has the best bias-variance trade-off. To tackle this task, evolutionary algorithms (EAs) are particularly well-suited for solving multi-objective problems, because they can obtain several elements of the Pareto optimal set in a single run [4] and because they are less susceptible than mathematical programming techniques to the shape and continuity of the Pareto front. A comprehensive review of these methods can be found in [4]. In this work, we adopt NSGA-II [5], which is one of the most popular MOEAs. We refer to [5] for details about the NSGA-II.

4.1 Model Selection Approach

The proposed approach to model selection adopts a multi-objective optimization technique for exploring the hyperparameters space, searching which one of them gives the best bias-variance trade-off. Figure 1 shows our model selection process. We have to highlight that besides optimizing the hyperparameters of the SVM, we also optimize the feature selection and pre-processing hyperparameters.

First, given a labeled data set for model training, we divide it into two different sets called learning set and validation set. The learning set is used to fit the parameters of the model during the hyperparameters space exploration. The NSGA-II is used for the exploration task. Once the search process is completed, a set of trade-off solutions is obtained (i.e., the Pareto optimal set). Each solution in the Pareto optimal set corresponds to a model that satisfies a trade-off between the two objectives considered: bias and variance. The next step is to choose one solution from that set. In order to avoid the selection of an underfitted or overfitted (see Figure 2) solution, the validation set is used to test each model in the Pareto optimal set. We select the solution with the lowest balanced error rate in the validation set. Finally, the model is trained using both, the learning set and the validation set. The selected model is tested over a new data set to evaluate its performance.

We used the Challenge Learning Object Package (CLOP) [14]. This Matlab toolbox has available several methods for pre-processing (such as standardize, normalize, shift and scale, and PCA) and feature selection (such as s2n, relief, zfilter, aucfs), as well as learning algorithms (such as SVM).

4.2 Representation Scheme

Under the adopted approach, we need to represent each potential model as an individual for NSGA-II. Therefore, each model is encoded in a 12-dimensional

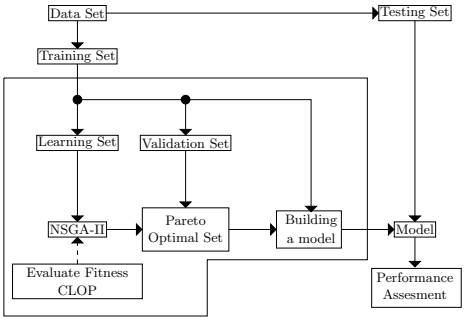


Fig. 1: General Architecture of the proposed model selection method

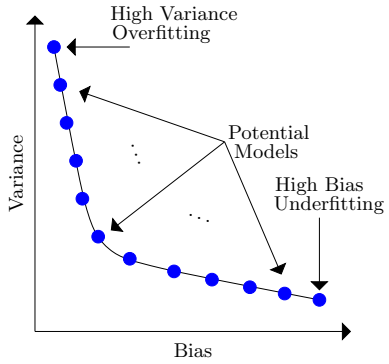


Fig. 2: Models with high bias or variance in Pareto front.

vector as follows: $\mathbf{x^{(i)}} = [x_1, \ldots, x_{12}]$, where x_1 represents the type of model, i.e., if the model is just a SVM classifier, or if it is composed by a pre-processing method and/or a feature selection method, as well as the order in which they are applied. x_2 represents a feature selection method and, x_3 is a pre-processing method. x_4, x_5 and x_6 are the hyperparameters for the feature selection method, while x_7 and x_8 are the hyperparameters for the pre-processing method. Finally, x_9, x_{10}, x_{11} and x_{12} are the hyperparameters for the SVM classifier (kernel, and its parameters). We emphasize that we used a real codification for the hyperparameters in order to achieve as precision as possible.

4.3 Fitness Function

The fitness function determines how good a solution is with respect to others. In our model, the selection task is treated as a multi-objective optimization problem, where the bias and variance are the objectives to be minimized. It is important to note that we just have a limited number of samples, and, therefore the estimations are approximations of expected values for those components. We used Webb's definition [16] of bias and variance, because it is one of the most widely used, but other definitions can be applied as well. In our case, bias and variance are estimated in the following way:

$$bias^2 = P_{(X,Y),D}\left(f_D\left(x\right) \neq f\left(x\right) \wedge f_D\left(x\right) = C_{f,D}\left(x\right)\right)$$
$$var = P_{(X,Y),D}\left(f_D\left(x\right) \neq f\left(x\right) \wedge f_D\left(x\right) \neq C_{f,D}\left(x\right)\right)$$

where $f_D\left(x\right)$ is the predicted output with the model trained with data set D, $f\left(x\right)$ is the desired output and $C_{f,D}$ is the central tendency.

Under this definition, N models should be trained and tested. We used k-fold cross validation repeated n times, because it has the advantage that every sample is used for training and testing, and each of them is evaluated n times. We fixed the values of k to 3 and n to 10, as employed by Webb [16]. Note that

Table 1: Data sets used in our experiments.

ID	Data set	Feat.	Training	Testing	ID	Data Set	Feat.	Training	Testing
1	Banana	2	400	4900	8	Ringnorm	20	400	7000
2	BC	9	200	77	9	Splice	60	1000	2175
3	Diabetes	8	468	300	10	Thyroid	5	140	75
4	FS	9	666	400	11	Titanic	3	150	2051
5	German	20	700	300	12	Twonorm	20	400	7000
6	Heart	13	170	100	13	Waveform	21	400	4600
7	Image	20	1300	1010					

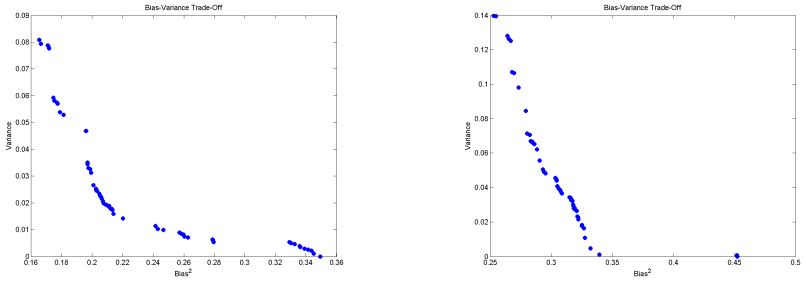
a model should be trained 30 times in order to assign it a merit, thus resulting in a high computational cost of this task.

5 Experiments and Results

For our experiments, we used a suite of thirteen binary classification benchmark data sets [13]. These data sets are described in Table 1 and they have been used in related works [7, 13]. For each data set, we randomly selected 10 partitions, which were used to evaluate the performance of our multi-objective SVM model selection (MOSVMMS) approach. For each partition, we performed the model selection procedure independently, and thus, the proposed method was applied a total of 130 times.

In Figure 3, we show the Pareto front obtained in a particular trial of some data sets. These Pareto fronts show the trade-off solutions between bias and variance. We can observe that both objectives are in conflict, as indicates the behavior depicted in Figures 3a and 3b. Therefore, each point in the Pareto front corresponds to different trade-offs between the objectives, when a model is trained with a particular set of hyperparameters. From these plots, we can also observe that several solutions are distributed along the Pareto front. One of these is chosen based on its error in a validation set. That solution is tested with the test set and the performance model is evaluated.

Table 2 shows the error rates and the standard deviation of 10 replications for each data set. For illustrative purposes, in Table 2 we show the obtained results when either bias and variance is minimized. This table also shows the obtained results by the standard SVM, without performing hyperparameters selection, and those obtained by PSMS [7]. PSMS is a full model selection method reported in the literature, which has obtained good performance over data sets from different domains, including those adopted here. PSMS uses particle swarm optimization (PSO) for selecting a combination of the feature selection method, a preprocessing method, a learning algorithm and their associated hyperparameters. In these experiments, we fixed the learning algorithm to SVM, and we used the same 10 partitions for both approaches in order to allow a fair comparison. Note that in both tables, the standard deviation represents the variability from both the partitions for the training set and from the model selection method.



(a) diabetes (b) flare-solar

Fig. 3: Pareto fronts for each data set. The point in the Pareto front represents a trade-off between the bias and the variance.

From Table 2 we can observe that our proposal obtains in average a better performance than the other ones. Wilcoxon Signed Rank test was recommended by Demšar [6] for comparing two methods over different data sets. This statistical test is applied with 95% of confidence, and showed that our proposal outperformed significantly to the single-objective formulations, as it was expected. MOSVMMS also outperformed SVM, showing a statistical difference in datasets 1, 7, 8, 9, 10, 12, and 13. With respect to PSMS, the statistical test indicated that there exists a significant difference on datasets 3, 4, 5, 8, 9, 11, and 13. We should highlight that for datasets where MOSVMMS obtained the worst results, the difference is not significant, according to the test. This provides an empirical evidence of the advantages of treating model selection as a multi-objective problem, minimizing the bias and variance components.

6 Conclusions and Future Work

We introduced a novel approach for SVM model selection, where the model can be composed by a combination of the feature selection and pre-processing methods. We adopted a multi-objective approach, choosing models with the best bias-variance trade-off. The estimation of the bias and variance is computed using Webb's definition. This definition has the advantage that can be applied to full model selection formulation. We tested our method in a benchmark of classification data sets from different domains. Our results indicated that our proposed approach had a good performance. A statistical test showed that our approach significantly improved the results of PSMS, which is the closest approach to our proposal. This improvement gives evidence of the suitability of treating model selection as a multi-objective optimization problem. Because of the intensive search done to explore the hyperparameters space, this scheme is computationally expensive. However, our current work is focused on tackling this drawback and produce another approach that is more computationally efficient.

Table 2: Comparison of the multi-objective SVM model selection (MOSVMMS), the standard SVM and PSMS. We report the error rates obtained over ten trials for each method. The best result for each data set is shown in **boldface**.

ID	Bias	Variance	SVM	PSMS	MOSVMMS
1	$\mathbf{10.51} \pm 0.44$	43.24 ± 2.01	46.05 ± 2.37	10.66 ± 0.62	10.56 ± 0.43
2	30.13 ± 10.65	$\mathbf{24.94 \pm 3.91}$	26.37 ± 4.15	27.40 ± 4.17	27.40 ± 4.48
3	25.00 ± 1.89	30.00 ± 4.17	$\mathbf{22.87 \pm 1.10}$	27.97 ± 3.61	24.20 ± 0.95
4	38.30 ± 4.04	40.88 ± 6.65	$\mathbf{33.18 \pm 1.29}$	35.88 ± 1.82	33.53 ± 1.58
5	27.03 ± 4.31	28.47 ± 2.39	$\mathbf{24.53 \pm 1.68}$	27.63 ± 2.28	24.87 ± 2.94
6	20.60 ± 5.91	28.20 ± 8.85	17.00 ± 2.11	16.20 ± 1.75	$\mathbf{15.80 \pm 1.75}$
7	4.69 ± 1.88	29.20 ± 5.44	15.57 ± 0.92	3.32 ± 0.56	$\mathbf{2.63 \pm 0.53}$
8	1.74 ± 0.25	29.01 ± 18.65	25.17 ± 0.74	3.86 ± 6.72	$\mathbf{1.62 \pm 0.13}$
9	43.56 ± 1.90	24.59 ± 8.49	16.61 ± 0.79	8.32 ± 2.22	$\mathbf{6.34 \pm 0.67}$
10	4.13 ± 1.72	6.40 ± 6.59	11.47 ± 3.88	$\mathbf{4.00 \pm 1.89}$	4.53 ± 2.28
11	29.44 ± 14.18	25.22 ± 4.48	22.52 ± 0.35	22.67 ± 0.56	$\mathbf{21.72 \pm 0.27}$
12	2.95 ± 0.49	21.82 ± 22.43	3.64 ± 0.62	$\mathbf{2.12 \pm 1.59}$	2.55 ± 0.19
13	10.58 ± 1.30	27.83 ± 6.86	13.44 ± 0.65	11.19 ± 1.34	$\mathbf{10.35 \pm 0.99}$
Ave.	19.13 ± 3.77	27.68 ± 7.76	21.42 ± 1.59	15.48 ± 2.24	$\mathbf{14.32 \pm 1.27}$

Future research directions include the study of the feasibility of the proposed approach for full model selection. We are also interested in studying the selection of members in an ensemble and in studying the effect of the population size, the number of iterations as well as the used search strategy in the model selection context. Finally, we are interested in extending our method to multi-class classification problems.

References

1. Aydin, I., Karakose, M., Akin, E.: A multi-objective artificial immune algorithm for parameter optimization in support vector machine. Appl. Soft Comput. 11(1), 120 – 129 (2011)
2. Chatelain, C., Adam, S., Lecourtier, Y., Heutte, L., Paquet, T.: Multi-objective optimization for svm model selection. In: Ninth ICDAR. vol. 1, pp. 427 –431 (2007)
3. Chatelain, C., Adam, S., Lecourtier, Y., Heutte, L., Paquet, T.: A multi-model selection framework for unknown and/or evolutive misclassification cost problems. Pattern Recogn. 43(3), 815 – 823 (2010)
4. Coello Coello, C.A., Lamont, G.B., Veldhuizen, D.A.V.: Evolutionary Algorithms for Solving Multi-Objective Problems. Genetic and Evolutionary Computation, Springer US (2007)
5. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii. In: Proceedings of the Parallel Problem Solving from Nature PPSN VI, LNCS, vol. 1917, pp. 849–858. Springer Berlin / Heidelberg (2000)
6. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. 7, 1–30 (2006)
7. Escalante, H.J., Montes, M., Sucar, L.E.: Particle swarm model selection. J. Mach. Learn. Res. 10, 405–440 (2009)

8. Friedman, J.H.: On bias, variance, 0/1loss, and the curse-of-dimensionality. Data Min. Knowl. Discov. 1, 55–77 (1997)
9. Ji, Y., Sun, S.: Feature selection for ensembles using non-dominated sorting in genetic algorithms. In: Natural Computation (ICNC), 2010 Sixth International Conference on. vol. 2, pp. 888 –891 (aug 2010)
10. Kohavi, R., Wolpert, D.: Bias plus variance decomposition for zero-one loss functions. In: Proceedings of the 13th ICML. pp. 275–283. Morgan-Kaufmann Publishers (1996)
11. Kong, E., Dietterich, T.: Error-correcting output coding corrects bias and variance. In: Proceedings of the 12th ICML. pp. 313–321. Morgan-Kaufmann Publishers (1995)
12. Li, W., Liu, L., Gong, W.: Multi-objective uniform design as a svm model selection tool for face recognition. Expert Syst. Appl. 38(6), 6689 – 6695 (2011)
13. Rätsch, G., Onoda, T., Müller, K.R.: Soft margins for adaboost. Mach. Learn. 42, 287–320 (2001)
14. Saffari, A., Guyon, I.: Quick start guide for clop. Tech. rep., Technical Report, May 2006. http://ymer.org/research/files/clop/QuickStartV1. 0. pdf (2006)
15. Suttorp, T., Igel, C.: Multi-objective optimization of support vector machines. In: Jin, Y. (ed.) Multi-Objective Machine Learning, Studies in Computational Intelligence, vol. 16, pp. 199–220. Springer Berlin / Heidelberg (2006)
16. Webb, G.I.: Multiboosting: A technique for combining boosting and wagging. Mach. Learn. 40, 159–196 (2000)