On General Definition of L1-norm Support Vector Machines for Feature Selection

Hai Thanh Nguyen, Katrin Franke, and Slobodan Petrovi'c

***Abstract*—In this paper, we introduce a new general definition of L1-norm SVM (GL1-SVM) for feature selection and represent it as a polynomial mixed 0-1 programming problem. We prove that solving the new proposed optimization problem reduces error penalty and enlarges the margin between two support vector hyper-planes. This possibly provides better generalization capability of SVM than solving the traditional L1-norm SVM proposed by Bradley and Mangasarian. We also propose a new search method that ensures obtaining of the global feature subset by means of the new GL1-SVM. The proposed search method is based on solving a mixed 0-1 linear programming (M01LP) problem by using the branch and bound algorithm. In this M01LP problem, the number of constraints and variables is linear in the number of full set features. Experimental results obtained over the UCI, LIBSVM, UNM and MIT Lincoln Lab data sets show that the new general L1-norm SVM gives better generalization capability, while selecting fewer features than the traditional L1-norm SVM in many cases.**

***Index Terms*—branch and bound, feature selection, L1-norm support vector machine, mixed 0-1 linear programming problem.**

I. Introduction

The feature selection problem for support vector machine (SVM) was studied in many previous works [1]-[7]. In this paper, we focus on feature selection for linear SVMs [10,11] for two-class classification problems. In particular, we consider the application of L1-norm SVM for feature selection, which was first proposed by Bradley and Mangasarian in 1998 [1]. Feature selection is an indirect consequence of the training process of SVMs. In fact, in the context of linear SVMs for two-class classification problems, the number of selected important features is the number of nonzero elements of the weight vector after the training phase. Bradley and Mangasarian [1] showed in many cases that utilizing the L1-norm SVM leads to a feature selection method, whereas utilizing the standard SVM [10,11] does not.

However, we realize that the Bradley and Mangasarian's method considers only one case of all *n* full-set features in the training phase. Since there probably exist irrelevant and redundant features [14,15], it is necessary to test all 2^n possible combinations of features for training SVM. In this paper, we propose a new general definition of L1-norm SVM (GL1-SVM) that takes into account all 2^n possible feature subsets. Therefore, the traditional L1-norm SVM proposed by Bradley and Mangasarian is just only a case of our new GL1-SVM. This is the reason why we call our method a general L1-norm SVM. The main idea of the GL1-SVM is that we encode the weight vector and the data matrix by utilizing binary variables $x_j (j = \overline{1,n})$. Following this encoding scheme, our GL1-SVM can be represented as a polynomial mixed 0-1 programming problem (PM01P). The objective function of this PM01P, which is a sum of the inverse value of margin by means of L1-norm and the error penalty, depends on binary variables. We prove that the minimal value of the objective function from the GL1-SVM is not larger than the one from the traditional L1-norm SVM. As a consequence, solving our new proposed optimization problem PM01P reduces error penalty and enlarges the margin between two support vector hyper-planes. This possibly provides better generalization capability of SVM than those obtained by solving the traditional L1-norm SVM.

In order to solve the obtained PM01P problem, we apply Chang's method to transfer it into a mixed 0-1 linear programming problem (M01LP), which can be solved by using branch and bound algorithm. The number of variables and constraints in this M01LP is linear in the number of full-set features and the number of instances. We have compared our new GL1-SVM with the traditional L1-norm SVM proposed by Bradley and Mangasarian [1] regarding the generalization capability and the number of selected important features. Experimental results obtained over the UCI [12], LIBSVM [17], UNM [19] and MIT Lincoln Lab (MIT LL) [18] data sets show that the new general L1-norm SVM gives better generalization capability, while selecting fewer features in many cases.

The paper is organized as follows. Section II formally defines a new general definition of L1-norm SVM (GL1-SVM). We show how to represent GL1-SVM as a polynomial mixed 0-1 programming problem (PM01P) by means of an encoding scheme. In this section, we also prove that the minimal value of the objective function from the GL1-SVM is not larger than the one from the traditional L1-norm SVM. Section III describes our new search approach that ensures globally optimal feature subsets by means of the GL1-SVM. We present experimental results in Section IV. The last section summarizes our findings.

Manuscript received April 22, 2011. This work was supported in part by the Norwegian Information Security Laboratory, Department of Computer Science and Media Technology, Gjøvik University College, Norway.

Authors are with the Norwegian Information Security Laboratory, P.O.Box 191, N-2802 Gjøvik, Norway.
(e-mail: hai.nguyen@ hig.no).
(e-mail: katrin.franke@hig.no).
(e-mail: slobodan.petrovic@hig.no).

II. A General L1-Norm Support Vector Machines

We are given a training data set *D* with *m* instances:

$$D = \{(a_i, c_i) \mid a_i \in R^n, c_i \in \{-1,1\}\}_{i=1}^m$$

where a_i is the i^{th} instance that has *n* features and a class label c_i; a_i can be represented as a data vector as follows: $a_i = (a_{i1}, a_{i2}, ..., a_{in})$, where a_{ij} is the value of the j^{th} feature in instance a_i.

For the two-class classification problem, SVM learns the separating hyper-plane *w*a=b* that maximizes the margin distance $\frac{2}{\|w\|_2^2}$, where *w* is the weight vector and *b* is the bias. The primal form of SVM is given below [10]:

$$\min_{w,b} \frac{1}{2}\|w\|_2^2 \qquad \text{(P1)}$$

subject to the following constraints:

$$\{c_i(wa_i - b) \geq 1, i = \overline{1,m}\}$$

In 1995, Cortes and Vapnik [11] proposed a modified version of SVM that allows for mislabeled instances. They called this version of SVM Soft Margin, which has the following form:

$$\min_{w,b,\xi} \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^m \xi_i \qquad \text{(P2)}$$

subject to the following constraints:

$$\begin{cases} c_i(wa_i - b) \geq 1 - \xi_i, \\ \xi_i \geq 0, i = \overline{1,m}. \end{cases}$$

where ξ_i is slack variable, which measures the degree of misclassification of instance a_i, $C > 0$ is the error penalty parameter.

In 1998, Bradley and Mangasarian [1] proposed to use L1-norm SVM for feature selection as a consequence of the resulting sparse solutions:

$$\min_{w,b,\xi} \|w\|_1 + C\sum_{i=1}^m \xi_i \qquad \text{(P3)}$$

subject to the following constraints:

$$\begin{cases} c_i(wa_i - b) \geq 1 - \xi_i, \\ \xi_i \geq 0, i = \overline{1,m}. \end{cases}$$

Define $w = p - q$ with $p, q \geq 0$. The problem (P3) is then equivalent to the following linear programming problem [1]:

$$\min_{p,q,b,\xi} e_n^T(p+q) + C\sum_{i=1}^m \xi_i,$$

subject to the following constraints:

$$\begin{cases} c_i(pa_i - qa_i - b) \geq 1 - \xi_i, \\ \xi_i \geq 0, i = \overline{1,m}, \\ p, q \geq 0, e_n^T = (1,1,...,1)^T \in R^n \end{cases} \qquad \text{(P4)}$$

It can be seen from (P4) that the vectors *p*, *q* and a_i are *n*-dimensional, where *n* is the number of full-set features. That means the Bradley and Mangasarian's method considers only a single case of the whole *m* x *n* data matrix, while skipping $2^n - 1$ cases of *m* x *k* data matrix ($1 \leq k \leq n$) in the training phase. In other words, $2^n - 1$ possible feature subsets are not taken into account by the traditional L1-norm SVM. Therefore, the general formulation of L1-norm SVM for feature selection is given below: We need to find the subset *S* of *k* features, which has the minimum value of $Merit_S(k)$ over all 2^n possible feature subsets:

$$\min_S \{Merit_S(k), 1 \leq k \leq n\}$$

where

$$Merit_S(k) = \min_{p^{(k)}, q^{(k)}, b, \xi} e_k^T(p^{(k)} + q^{(k)}) + C\sum_{i=1}^m \xi_i, \qquad \text{(P4)'}$$

subject to the following constraints:

$$\begin{cases} c_i(p^{(k)}a_i^{(k)} - q^{(k)}a_i^{(k)} - b) \geq 1 - \xi_i, \\ \xi_i \geq 0, i = \overline{1,m}, \\ e_k^T = (1,1,...,1)^T \in R^k \\ p^{(k)}, q^{(k)} \geq 0; p^{(k)}, q^{(k)}, a_i^{(k)} \in R^k \end{cases}$$

When the number of features *n* is small, we can apply the brute force method to scan all these subsets. But when this number becomes large, a more computationally efficient method that also ensures obtaining of the globally optimal subsets is required. In the following, we will show how to represent the problem (P4)' as a polynomial mixed 0-1 programming problem and show how to solve this optimization problem in order to get the globally optimal feature subset.

Firstly, we use the binary variables $x_j (j = \overline{1,n})$ for indicating the appearance of the j^{th} feature ($x_j = 1$) or the

absence of the j^{th} feature ($x_j = 0$) to encode the variables *p*, *q* and the data vector $a_i (i = \overline{1,m})$ as follows:

$$\begin{cases} p = (x_1 z_1, x_2 z_2, ..., x_n z_n), \\ q = (x_1 z_{n+1}, x_2 z_{n+2}, ..., x_n z_{2n}), \\ a_i = (a_{i1} x_1, a_{i2} x_2, ..., a_{in} x_n) \end{cases} \quad (*)$$

Proposition 1: With the encoding scheme (*), the problem (P4)' can be represented as a polynomial mixed 0-1 programming problem.

Proof. By using the encoding scheme (*), we generalize the problem (P4) into the following polynomial mixed 0-1 programming problem that takes into account all 2^n possible feature subsets and that is the same problem as (P4)':

$$\min_{x \in \{0,1\}^n} \left[\min_{z,b,\xi} \left\{ \sum_{j=1}^{n} x_j z_j + \sum_{j=1}^{n} x_j z_{n+j} + C \sum_{i=1}^{m} \xi_i \right\} \right]$$

subject to the following constraints:

$$\begin{cases} \sum_{j=1}^{n} c_i a_{ij} x_j^2 z_j - \sum_{j=1}^{n} c_i a_{ij} x_j^2 z_{n+j} - c_i b \geq 1 - \xi_j, \\ x = (x_1, x_2, ..., x_n), x_j \in \{0,1\}, j = \overline{1,n}, \\ \xi = (\xi_1, \xi_2, ..., \xi_m), \xi_i \geq 0, i = \overline{1,m}, \\ z = (z_1, ..., z_n, ..., z_{2n}), z_k \geq 0, k = \overline{1,2n}. \end{cases} \quad (P5)$$

Remark: We can even control the minimal number *T* ($T \leq n$) of selected features or the minimal number *T* ($T \leq n$) of nonzero elements of the weight vector *w* by adding the following constraint: $x_1 + x_2 + ... + x_n = T$ to the optimization problem (P5). We will show how to solve this optimization problem in the next section.

Proposition 2: Suppose that *S1*, *S2* are minimal values of the objective functions from (P4), (P5), respectively. The following inequality is true:

$$S2 \leq S1$$

Proof. It is obvious, since the problem (P4) is a case of the problem (P5) when $x = (x_1, x_2, ..., x_n) = (1,1,...,1)$.□

Remark: As a consequence of the Proposition 2, solving the problem (P5) reduces error penalty and enlarges the margin between two support vector hyper-planes, thus possibly providing better generalization capability of SVM than solving the traditional L1-norm SVM proposed by Bradley and Mangasarian [1].

In the next section, we show how to solve the problem (P5) by applying Chang's method [8,9]. The idea is to convert the (P5) into a mixed 0-1 linear programming problem, which can then be solved by utilizing the branch and bound algorithm for finding globally optimal feature subsets.

III. Optimizing the General L1-Norm Support Vector Machines

Since $x_j z_j = x_j^2 z_j$ when $x_j \in \{0,1\}$, the following problem is equivalent to (P5):

$$\min_{x,z,b,\xi} [\sum_{j=1}^{n} x_j^2 z_j + \sum_{j=1}^{n} x_j^2 z_{n+j} + C \sum_{i=1}^{m} \xi_i]$$

subject to the following constraints

$$\begin{cases} \sum_{j=1}^{n} c_i a_{ij} x_j^2 z_j - \sum_{j=1}^{n} c_i a_{ij} x_j^2 z_{n+j} - c_i b \geq 1 - \xi_i, \\ x = (x_1, x_2, ..., x_n), x_j \in \{0,1\}, j = \overline{1,n}, \\ \xi = (\xi_1, \xi_2, ..., \xi_m), \xi_i \geq 0, i = \overline{1,m}, \\ z = (z_1, z_2, ..., z_{2n}), z_k \geq 0, k = \overline{1,2n} \end{cases} \quad (P6)$$

Proposition 3: A polynomial mixed 0-1 term $x_j^2 z_j$ from (P6) can be represented by a continuous variable v_j, subject to the following linear inequalities [8,9]:

$$\begin{cases} v_j \geq M(2x_j - 2) + z_j, \\ v_j \leq M(2 - 2x_j) + z_j, \\ 0 \leq v_j \leq M x_j, \end{cases} \quad (P7)$$

where *M* is a large positive value.

Proof.

1. If $x_j = 0$, then (P7) becomes

$$\begin{cases} v_j \geq M(0 - 2) + z_j, \\ v_j \leq M(2 - 0) + z_j, \\ 0 \leq v_j \leq 0, \end{cases}$$

v_j is forced to be zero, as *M* is a large positive value.

2. If $x_j = 1$, then (P7) becomes

$$\begin{cases} v_j \geq M(2-2)+z_j \\ v_j \leq M(2-2)+z_j \\ 0 \leq v_j \leq M \end{cases}$$

v_j is forced to be z_j, as $z_j \geq 0$.

Therefore, the constraints on v_j reduce to

$$v_j = \begin{cases} 0, & if\ x_j = 0, \\ z_j, & if\ x_j = 1, \end{cases}$$

which is the same as $x_j^2 z_j = v_j$.□

Remark: As a consequence of the Proposition 3, the polynomial mixed 0-1 programming problem (P6) becomes a mixed 0-1 linear programming problem (M01LP). The total number of variables of the M01LP problem will be linear (*5n + m + 1*), as they are x_j, b, z_k, v_j, v_{n+j} and ξ_i ($j=\overline{1,n}$, $k=\overline{1,2n}$, $i=\overline{1,m}$). Therefore, the number of constraints on these variables will also be a linear function of *n* (number of full-set features) and *m* (number of instances). We can use branch and bound algorithm for solving this M01LP problem.

IV. Experiment

In order to validate our theoretical findings, we conducted an experiment on the UCI [12], the LIBSVM [17], the UNM [19] and the MIT LL [18] data sets. The goal was to compare our new method with the standard SVM [11] and the traditional L1-norm SVM [1] regarding the number of selected features and the generalization capability. Note that the number of selected features in context of linear SVM for binary classification problem is the number of nonzero elements of the weight vector.

We selected 2 UCI data sets [12], 4 LIBSVM data sets [17], 5 UNM data sets [19] and 2 MIT LL data sets [18]. The raw UNM and MIT LL data sets were not analyzed. Instead, we utilized the data sets with extracted features from [20]. For implementing the standard SVM and the traditional L1-norm SVM, we used the Mangasarian's code from the website [16]. For implementing the new general L1-norm SVM (GL1-SVM), the TOMLAB tool [13] was used for solving the mixed 0-1 linear programming problem. The values of the error penalty parameter *C* used for the experiment were: 2^{-7}, 2^{-6} ... 2^{6}, 2^{7}. We applied 10-fold cross validation for estimating the average classification accuracies as well as the average number of selected features. All the best results obtained over those penalty parameters were chosen and are given in the tables I and II.

We observed from Table I that our new proposed method GL1-SVM selects the smallest number of relevant features in comparison to the standard SVM and the traditional L1-norm SVM. At the same time, our new method still keeps or even yields better generalization capability than the traditional L1-norm SVM does. This can be seen from Table II.

TABLE I: NUMBER OF SELECTED FEATURES (ON AVERAGE)

Data Sets	Full set	SVM	L1-norm	GL1-SVM
a1a [17]	123	105.3	64	3.5
a2a [17]	123	107.6	74.9	3.8
w1a [17]	300	266.2	76.7	11.1
w2a [17]	300	270.5	99.9	10.2
Spectf [12]	44	44	31	6
haberman[12]	3	3	2.8	1.6
RawFriday [18]	53	33.9	8.4	1.9
RawMonday[18]	54	26	1	1
L-inetd [19]	164	33.5	13.5	2.4
Login [19]	164	46	9.6	2
PS [19]	164	22	5	2
S-lpr [19]	182	36.9	3.2	2
Xlock [19]	200	46.8	13.4	1
Average	**144.1**	**80.1**	**31**	**3.7**

TABLE II: CLASSIFICATION ACCURACIES (ON AVERAGE)

Data Sets	SVM	L1-norm	GL1-SVM
a1a [17]	83.99	65.40	75.37
a2a [17]	82.25	68.34	74.75
w1a [17]	96.76	88.49	97.09
w2a [17]	96.69	85.76	96.92
Spectf [12]	72.20	79.55	79.55
Haberman[12]	73.48	73.16	73.48
RawFriday [18]	98.40	54.02	98.80
RawMonday[18]	100	95.65	100
L-inetd [19]	88.33	85.00	85.83
Login [19]	80.00	65.00	81.67
PS [19]	100	100	100
S-lpr [19]	100	70	99.11
Xlock [19]	100	56.79	100
Average	**90.16**	**75.93**	**89.42**

V. Conclusions

We have proposed a new general L1-norm SVM (GL1-SVM) for feature selection that considers all possible feature subsets. The main idea was to utilize binary variables for encoding the weight vector and the data matrix. The new GL1-SVM can then be represent as a polynomial mixed 0-1 programming problem (PM01LP). We proved that the

traditional L1-norm SVM proposed by Bradley and Mangasarian is only a single case of our new GL1-SVM. Therefore, solving the new proposed optimization problem reduces error penalty and enlarges the margin between two support vector hyper-planes, thus possibly providing better generalization capability of SVM than solving the traditional L1-norm SVM problem from Bradley and Mangasarian. We also proposed a new search method that ensures obtaining of the global feature subset by means of the new GL1-SVM. By applying Chang's method, we transferred the PM01LP problem into a mixed 0-1 linear programming (M01LP) problem, which can be solved by using the branch and bound algorithm. In this M01LP problem, the number of constraints and variables is linear in the number of full set features.. Experimental results obtained over the UCI, the LIBSVM, the UNM and the MIT LL data sets showed that the new general L1-norm SVM gives better generalization capability, while selecting fewer features than the traditional L1-norm SVM in many cases.

REFERENCES

[1] Bradley, P. & Mangasarian, O.L., (1998). Feature selection via concave minimization and support vector machines. In *Proceedings of the Fifteenth International Conference (ICML)*, pp. 82-90.

[2] Mangasarian, O.L., (2007). Exact 1-Norm Support Vector Machines Via Unconstrained Convex Differentiable Minimization (Special Topic on Machine Learning and Optimization). *Journal of Machine Learning Research*, 7(2):1517- 1530.

[3] Weston, J., Mukherjee, S., Chapelle, O., Pontil, M., Poggio, T. & Vapnik, V., (2001). Feature selection for SVMs. *Advances in neural information processing systems*, pp. 668-674.

[4] Guyon, I.,Weston, J., Barnhill, S. & Vapnik, V., (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1):389-422.

[5] Guan, W., Gray, A. & Leyffer, S., (2009). Mixed-Integer Support Vector Machine. *NIPS Workshop on Optimization for Machine Learning*.

[6] Neumann, J., Schnorr, C. & Steidl, G., (2005). Combined SVM-based feature selection and classification, *Machine Learning*, 61(1):129-150.

[7] Rakotomamonjy, A., (2003). Variable selection using SVM based criteria. *Journal of Machine Learning Research*, 3:1357-1370.

[8] Chang, C-T., (2001). On the polynomial mixed 0-1 fractional programming problems, *European Journal of Operational Research*, vol. 131, issue 1, pages 224-227.

[9] Chang, C-T., (2000). An efficient linearization approach for mixed integer problems, *European Journal of Operational Research*, vol. 123, pages 652-659.

[10] Vapnik, V. (1995) *The Nature of Statistical Learning Theory*, Springer.

[11] Cortes, C., and V. Vapnik, V., (1995). Support-Vector Networks, *Machine Learning*.

[12] Murphy, P.M., and Aha, D.W., (1992). UCI repository of machine learning databases. *Technical report, Department of Information and Computer Science, University of California, Irvine*, 1992. www.ics.uci.edu/mlearn/MLRepository.html

[13] TOMLAB, *The optimization environment in MATLAB*. http://tomopt.com/tomlab/.

[14] Guyon, I., Gunn, S., Nikravesh, M., and Zadeh, L.A., (2006). *Feature Extraction: Foundations and Applications*. Series Studies in Fuzziness and Soft Computing, Physica-Verlag, Springer.

[15] Liu, H., Motoda, H., (2008). *Computational Methods of Feature Selection*. Chapman & Hall/CRC.

[16] DMI Classification Software . http://www.cs.wisc.edu/dmi/

[17] Chang, C.C. & Lin, C.J. (2001) LIBSVM: a library for support vector machines. Data sets and software are available at http://www.csie.ntu.edu.tw/ cjlin/libsvm/

[18] Lippmann, R. P., Graf, I., Garfinkel, S. L., Gorton, A. S., Kendall, K. R., McClung, D. J., Weber, D. J., Webster, S. E., Wyschogrod, D., and Zissman, M. A. (1998). The 1998 DARPA/AFRL off-line intrusion detection evaluation. Presented to The *First Intl. Work Workshop on Recent Advances in Intrusion Detection (RAID-98*), (No printed proceedings) Lovain-la-Neuve, Belgium, 14–16 September.

[19] UNM (University of New Mexico) audit data. http://www.cs.unm.edu/~immsec/systemcalls.htm

[20] Kang, D.-K., Fuller, D., and Honavar, V., (2005). "Learning Classifiers for Misuse and Anomaly Detection Using a Bag of System Calls Representation," Proceedings of *6th IEEE Systems Man and Cybernetics Information Assurance Workshop (IAW*), West Point, NY, June 15-17.

Hai Thanh Nguyen is doctoral researcher of information security at NISlab, Department of Computer Science and Media Technology, Gjøvik University College, Gjøvik, Norway. He received his diploma and Master degree in applied mathematics and computer science from Moscow State University named after M. V. Lomonosov in 2007. His research interests include machine learning, computational intelligence for security, intrusion detection, and cryptography.

Katrin Franke is professor of information security at NISlab, Department of Computer Science and Media Technology, Gjøvik University College, Gjøvik, Norway. She received a diploma in electrical engineering from the Technical University Dresden, Germany in 1994 and her Ph.D. in artificial intelligence from the Groningen University, The Netherlands in 2005. Her research interests include computational forensics, biometrics, document and handwriting analysis, computer vision and computational intelligence. She has published several scientific journal articles, peer-reviewed conference papers and edited books.

Slobodan Petrović is professor of information security at NISlab, Department of Computer Science and Media Technology, Gjøvik University College, Gjøvik, Norway. He received his Ph.D. degree in 1994 from the University of Belgrade, Serbia. His research interests include cryptology, intrusion detection, and digital forensic. He is the author of more than 40 papers published in renowned international journals and conferences.