Unsupervised Classification via Convex Absolute Value Inequalities

Olvi L. Mangasarian *

Abstract

We consider the problem of classifying completely unlabeled data by using convex inequalities that contain absolute values of the data. This allows each data point to belong to either one of two classes by entering the inequality with a plus or minus value. By using such absolute value inequalities (AVIs) in support vector machine classifiers, unlabeled data can be successfully partitioned into two classes that capture most of the correct labels dropped from the data. Inclusion of partially labeled data leads to a semisupervised classifier. Computational results include unsupervised and semisupervised classification of the Wisconsin Breast Cancer Wisconsin (Diagnostic) Data Set.

Keywords: unsupervised classification, absolute value inequalities, support vector machines

1 Introduction

We begin with the following convex absolute value inequality (AVI):

$$|x'w - \gamma| \leq 1, \tag{1.1}$$

where $|\cdot|$ denotes the absolute value. Here the column vector x represents any data point in an n-dimensional space R^n, $w \in R^n$ is the normal vector to the classifying plane $x'w - \gamma = 0$, γ determines the distance from the origin of the plane, and the prime denotes the transpose of the column vector x. The AVI (1.1) is equivalent to dividing R^n into two overlapping halfspaces by the following two linear inequalities:

$$\begin{array}{rcl} x'w & \leq & \gamma + 1, \\ x'w & \geq & \gamma - 1. \end{array} \tag{1.2}$$

The key to our approach is to represent the last two inequalities by the single absolute value inequality AVI (1.1). Thus if $x'w - \gamma \geq 0$ then AVI (1.1) reduces to the first linear inequality of (1.2), whereas if $x'w - \gamma \leq 0$ then AVI (2.3) reduces to the second linear inequality of (1.2). Hence if we impose the AVI (1.1) on an unlabeled dataset, the dataset will be divided into two categories to best fit the AVI (1.1) or equivalently the two linear inequalities (1.2). Our objective will then be to minimize the overlap between the bounding planes $x'w = \gamma \pm 1$.

There have been approaches to semisupervised classification utilizing support vector machines such as [3, 7], but none of them have utilized absolute value inequalities. As far as unsupervised classification is concerned there have been no approaches such as ours here that does not use any labeled data if desired. There are of course many clustering techniques such as [1, 2, 4, 5, 16] that do not utilize any labels. However these approaches are quite different from what we are proposing here. Furthermore none of these approaches utilize either absolute value inequalities [13] or absolute value equations [20, 21, 14, 12].

*Computer Sciences Department, University of Wisconsin, Madison, WI 53706 and Department of Mathematics, University of California at San Diego, La Jolla, CA 92093. *olvi@cs.wisc.edu*.

Our approach here, as will be made clear in Section 2, will be based on concave minimization on a polyhedral set [9, 11, 12] which is solved by a few linear programs, typically four or less.

We briefly describe now the contents of the paper. In Section 2 we outline the theory behind our approach and in Section 3 we state our iterative algorithm for minimizing a concave function on a polyhedral set that consists of solving a succession of linear programs with a linearized objective function. In Section 4 we give computational results that show the effectiveness of our approach by recovering most of the labels that have been dropped from either all (unsupervised) or most (semisupervised) data points. Section 5 concludes the paper.

We describe now our notation. The scalar product of two column vectors x and y in a n-dimensional real space will be denoted by $x'y$. For a vector u, u^i represents the ith iterate of u during an iterative process. The identity matrix in a real space of arbitrary dimension will be denoted by I, while a column vector of ones of arbitrary dimension will be denoted by e and a column of zeros by 0. The notation $\partial f(x)$ stands for a subgradient if $f(x)$ is a convex function, and a supergradient if $f(x)$ is a concave function. We shall use the MATLAB [18] symbol $1e \pm n$ to denote $10^{\pm n}$. The abbreviation "s.t." stands for "subject to".

2 Unsupervised and Semisupervised Classification

We begin with an unlabeled dataset consisting of m points in the n-dimensional space R^n represented by the $m \times n$ matrix A and the labeled dataset consisting of k points in R^n represented by the $k \times n$ matrix H and labeled by the $k \times k$ diagonal matrix D with entries of ± 1 which denote which class of $+1$ or -1 each row of H belongs to. Thus we wish to find two planes $x'w - \gamma = \pm 1$ in R^n that specify the ± 1 feasible regions generated by the two inequalities of (1.2) and which satisfy with minimal error vector s the following inequalities:

$$\begin{array}{rcl} |Aw - e\gamma| & \leq & e, \\ D(Hw - e\gamma) + s & \geq & e, \\ s & \geq & 0. \end{array} \tag{2.3}$$

Here the nonnegative slack variable s is to be driven toward zero by the following optimization problem:

$$\begin{array}{rl} \min\limits_{(w,\gamma,s)\in R^{n+1+k}} & -e^{'}|w| - |\gamma| + \mu e^{'}s \\ \text{s.t.} \quad |Aw - e\gamma| & \leq e, \\ D(Hw - e\gamma) + s & \geq e, \\ s & \geq 0, \end{array} \tag{2.4}$$

which in addition, maximizes the 1-norm of (w, γ) in order to minimize the distance between the two overlapping feasible regions of the inequalities of (1.2), while μ is a positive parameter that balances the two groups of objectives of (2.4). We note that the constraints of (2.4) are convex while the objective function is concave, which we shall minimize by the finite successive linearization techniques of [9, 11, 12]. In order to handle the nonlinear absolute value inequality in (2.4) above we proceed as follows. We replace the term $|Aw - e\gamma|$ in the absolute value inequality by an upper bound r on it: $-r \leq (Aw - e\gamma) \leq r$ whose 1-norm is minimized with objective function weight μ. This results in the following linearly constrained concave minimization problem:

$$\begin{array}{rll} \min\limits_{(w,\gamma,r,s)\in R^{n+1+m+k}} & -e^{'}|w| - |\gamma| + \nu e^{'}r + \mu e^{'}s & \\ \text{s.t.} -r \leq & Aw - e\gamma & \leq r, \\ & r & \leq e \\ & -D(Hw - e\gamma) - s & \leq -e, \\ & s & \geq 0, \end{array} \tag{2.5}$$

which will be solved by a finite sequence of linear programs as described in the next section.

3 Successive Linear Programming Solution of Unsupervised & Semisupervised Classification

We note first that the objective function of our optimization problem (2.5) is concave and the constraints are linear. We further note, under the generally satisfied condition that the $n+1$ columns of the $m \times (n+1)$ matrix $[A \quad e]$ are linearly independent, it follows that the iterates $(w^i, \gamma^i, r^i, s^i)$ of our successive linearization algorithm 3.1 below are bounded. Hence, by [10, Theorem 3] we have that the iterates $(w^i, \gamma^i, r^i, s^i)$ strictly decrease the objective function of (2.5) and terminate in a finite number of steps (typically less than five) at a point satisfying the minimum principle necessary optimality condition for our problem (2.5).

We now state our successive linearization algorithm.

ALGORITHM 3.1. **SLA: Successive Linearization Algorithm** *Choose parameter values for (μ, ν) in (2.5), typically $1e-4$.*

(I) Initialize the algorithm by choosing an initial nonnegative random vector in R^{n+1} for (w^0, γ^0). Set iteration number to $i = 0$.

(II) Solve the following linear program, which is a linearization of (2.5) around $(w^i, \gamma^i, r^i, s^i)$, for $(w^{i+1}, \gamma^{i+1}, r^{i+1}, s^{i+1})$:

$$\begin{array}{rcl} \min\limits_{(w,\gamma,r,s)\in R^{n+1+m+k}} & -sign(w^i)'w - sign(\gamma^i)\gamma + \nu e'r + \mu e's & \\ s.t. -r \leq & Aw - e\gamma & \leq \quad r, \\ & r & \leq \quad e \\ & -D(Hw - e\gamma) - s & \leq \quad -e, \\ & s & \geq \quad 0, \end{array} \tag{3.6}$$

(III) If $w^{i+1} = w^i$ stop.

(IV) Set $i = i+1$ and go to Step (II).

By invoking [10, Theorem 3] we have the following finite termination result for our SLA 3.1.

PROPOSITION 3.1. **Finite Termination of SLA 3.1** *Let $z = (w, \gamma, r, s)$, let $f(z)$ denote the concave objective function of (2.5) and let Z denote the feasible region of (2.5). Then the SLA 3.1 generates a finite sequence of feasible iterates $\{z^1, z^2, \ldots, z^i\}$ of strictly decreasing function values $f(z^1) > f(z^2) > , \ldots, f(z^i)$, such that z^i satisfies the minimum principle necessary optimality condition:*

$$\partial f(z^i)(z - z^i) \geq 0, \quad \forall z \in Z, \tag{3.7}$$

where $\partial f(z^i)$ denotes the supergradient of f at z^i.

We note that all the above results can be extended to nonlinear kernel classification [17, 8, 6] by replacing the linear absolute value inequality (1.1) by one representing a nonlinear surface $|K(x', A')u - \gamma| \leq 1$ that is still linear in the unknowns (u, γ), but nonlinear in the data variable $x \in R^n$, where K is any nonlinear kernel.

4 Computational Results

We begin with a simple 2-dimensional example consisting of six points depicted in Figure 1. When Algorithm 3.1 is applied to the six points $[0,0;0,1;1,0;4,4;3,4;4,3]$ starting with a random (w,γ), it terminates after solving three linear programs with the separating line $x_1 = 2$, which is quite appropriate for the given six unlabeled data points.

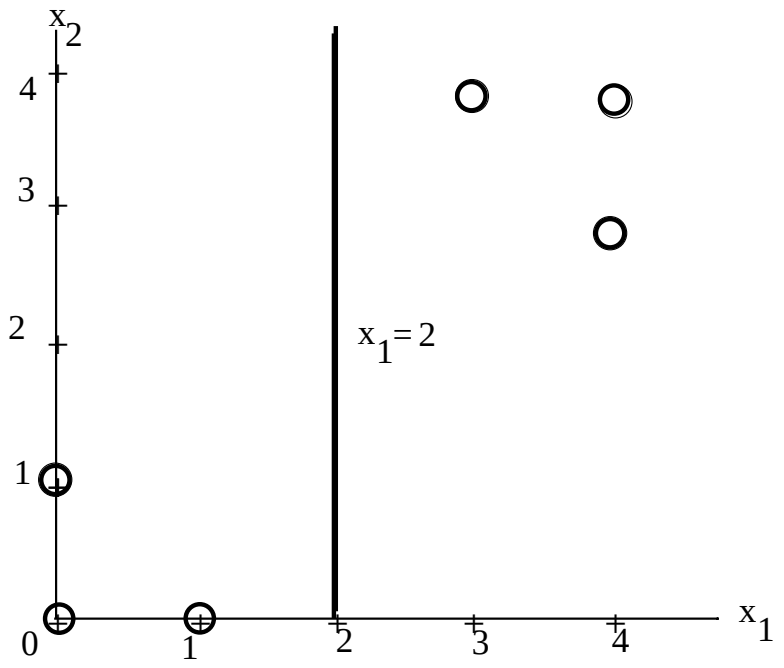

Figure 1: **Simple 2-dimensional example consisting of the six unlabeled points** $[0,0;0,1;1,0;4,4;3,4;4,3]$ **and the separating line** $x_1 = 2$ **generated by Algorithm 3.1 by solving three linear programs.**

Our second example is one of the most popular datasets available from the University of California Irvine Machine Learning Repository [19]: the Wisconsin Diagnostic Breast Cancer Dataset WDBC [22]. For our purposes we have extracted from WDBC an $m \times n$ matrix A with $m = 569$ and $n = 30$. Here m is the total number of patients in WDBC and n is the total number of features obtained from the fine needle aspirate of each patient [15]. We have rearranged the rows of A so that the first 357 rows are the data of benign aspirates, while the last 212 rows A are those of malignant aspirates. We shall now present results of our Algorithm 3.1 applied to this dataset.

We first ran Algorithm 3.1 on the whole matrix A as a completely unsupervised problem, that is without the last two constraints of (3.6) and obtained a correctness of 65.55%. We then ran Algorithm 3.1 twice with two different ten labeled cases, five benign and five malignant cases and obtained correctness values of 75.40% and 81.72% respectively. These results are summarized in Table 1.

5 Conclusion and Outlook

We have proposed the use of absolute value inequalities for classifying unlabeled data. We have also combined the approach with standard methods for classifying labeled data. It will be interesting to utilize other absolute value inequality formulations to handle unlabeled data, as well as utilizing nonlinear kernels in addition to the linear kernels employed here. Hopefully this may lead to effective tools for handling large unlabeled data.

No. of Unlabeled Data Used	Labeled Data Rows Used	No. of Linear Programs Solved	Correctness
569	None	2	65.55%
559	Rows 201-205 Benign Rows 361-365 Malignant	3	75.40%
559	Rows 1-5 Benign Rows 565-569 Malignant	4	81.72%

Table 1: **Unsupervised and semisupervised results for Algorithm 3.1 on the Wisconsin Diagnostic Breast Cancer Dataset WDBC [22] consisting of** 569 **cases the first** 357 **of which being benign and the last** 212 **cases being malignant.**

Acknowledgments The research described here is based on Data Mining Institute Report 14-01, March 2014.

References

[1] K. Al-Sultan. A Tabu search approach to the clustering problem. *Pattern Recognition*, 28(9):1443–1451, 1995.

[2] M. R. Anderberg. *Cluster Analysis for Applications*. Academic Press, New York, 1973.

[3] K. P. Bennett and A. Demiriz. Semi-supervised support vector machines. In M. S. Kearns, S. A. Solla, and D. A. Cohn, editors, *Advances in Neural Information Processing Systems -10-*, pages 368–374, Cambridge, MA, 1998. MIT Press.

[4] P. S. Bradley, O. L. Mangasarian, and W. N. Street. Clustering via concave minimization. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems -9-*, pages 368–374, Cambridge, MA, 1997. MIT Press. ftp://ftp.cs.wisc.edu/math-prog/tech-reports/96-03.ps.

[5] G. Celeux and G. Govaert. Gaussian parsimonious clustering models. *Pattern Recognition*, 28:781–793, 1995.

[6] G. Fung and O. L. Mangasarian. Proximal support vector machine classifiers. In F. Provost and R. Srikant, editors, *Proceedings KDD-2001: Knowledge Discovery and Data Mining, August 26-29, 2001, San Francisco, CA*, pages 77–86, New York, 2001. Association for Computing Machinery. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/01-02.ps.

[7] G. Fung and O. L. Mangasarian. Semi-supervised support vector machines for unlabeled data classification. *Optimization Methods and Software*, 15:29–44, 2001. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/99-05.ps.

[8] G. Fung, O. L. Mangasarian, and A. Smola. Minimal kernel classifiers. *Journal of Machine Learning Research*, pages 303–321, 2002. University of Wisconsin Data Mining Institute Technical Report 00-08, November 200, ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/00-08.ps.

[9] O. L. Mangasarian. Machine learning via polyhedral concave minimization. In H. Fischer, B. Riedmueller, and S. Schaeffler, editors, *Applied Mathematics and Parallel Computing - Festschrift for Klaus Ritter*, pages 175–188. Physica-Verlag A Springer-Verlag Company, Heidelberg, 1996. ftp://ftp.cs.wisc.edu/math-prog/tech-reports/95-20.ps.

[10] O. L. Mangasarian. Solution of general linear complementarity problems via nondifferentiable concave minimization. *Acta Mathematica Vietnamica*, 22(1):199–205, 1997. ftp://ftp.cs.wisc.edu/math-prog/tech-reports/96-10.ps.

[11] O. L. Mangasarian. Minimum-support solutions of polyhedral concave programs. *Optimization*, 45:149–162, 1999. ftp://ftp.cs.wisc.edu/math-prog/tech-reports/97-05.ps.

[12] O. L. Mangasarian. Absolute value equation solution via concave minimization. *Optimization Letters*, 1(1):3–8, 2007. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/06-02.pdf.

[13] O. L. Mangasarian. Absolute value programming. *Computational Optimization and Applications*, 36(1):43–53, 2007. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/05-04.ps.

[14] O. L. Mangasarian and R. R. Meyer. Absolute value equations. *Linear Algebra and Its Applications*, 419:359–367, 2006. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/05-06.pdf.

[15] O. L. Mangasarian, W. N. Street, and W. H. Wolberg. Breast cancer diagnosis and prognosis via linear programming. *Operations Research*, 43(4):570–577, July-August 1995.

[16] O. L. Mangasarian and E. W. Wild. Feature selection in k-median clustering. Technical Report 04-01, Data Mining Institute, Computer Sciences Department, University of Wisconsin, Madison, Wisconsin, January 2004. SIAM International Conference on Data Mining, Workshop on Clustering High Dimensional Data and Its Applications, April 24, 2004, La Buena Vista, FL, Proceedings Pages 23-28. http://www.siam.org/meetings/sdm04. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/04-01.pdf.

[17] O. L. Mangasarian and E. W. Wild. Nonlinear knowledge-based classification. *IEEE Transactions on Neural Networks*, 19:1826–1832, 2008. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/06-04.pdf.

[18] MATLAB. *User's Guide*. The MathWorks, Inc., Natick, MA 01760, 1994-2006. http://www.mathworks.com.

[19] P. M. Murphy and D. W. Aha. UCI machine learning repository, 1992. http://archive.ics.uci.edu/ml/.

[20] J. Rohn. Systems of linear interval equations. *Linear Algebra and Its Applications*, 126:39–78, 1989. http://www.cs.cas.cz/ rohn/publist/47.doc.

[21] J. Rohn. On unique solvability of the absolute value equation. *Optimization Letters*, 3:603–606, 2009.

[22] W. H. Wolberg, W. N. Street, and O. L. Mangasarian. WDBC: Wisconsin Diagnostic Breast Cancer Database. Computer Sciences Department, University of Wisconsin, Madison, ftp://ftp.cs.wisc.edu/math-prog/cpo-dataset/machine-learn/cancer/WDBC/, 1995.