Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 18 (2013) 1730 – 1736

Procedia Computer Science

International Conference on Computational Science, ICCS 2013

The Support Vector Regression with Adaptive Norms

Chunhua Zhang[a], Dewei Li[a], Junyan Tan[b]

[a]*Information School, Renmin University of China, Beijing 100872, China*
[b]*College of Science, China Agricultural University, Beijing 100083, China*

Abstract

This study proposes a new method for regression – l_p-norm support vector regression (l_p SVR). Some classical SVRs minimize the hinge loss function subject to the l_2-norm or l_1-norm penalty. These methods are non-adaptive since their penalty forms are fixed and pre-determined for any types of data. Our new model is an adaptive learning procedure with lp-norm ($0 < p < 1$), where the best p is automatically chosen by data. By adjusting the parameter p, l_p SVR can not only select relevant features but also improve the regression accuracy. An iterative algorithm is suggested to solve the l_p SVR efficiently. Simulations and real data applications support the effectiveness of the proposed procedure.

Keywords: Regression; Support vector machine; Norm; Feature selection;

1. Introduction

Support vector machines (SVMs), being computationally powerful tools for pattern classification and regression, have been successfully applied to a variety of real-world problems([1]- [6]). Regards to the support vector regression (SVR), some classical SVRs minimize the hinge loss function subject to the l_2-norm or l_1-norm penalty ([7]). We call them l_2 SVR or l_1 SVR correspondingly. These methods are non-adaptive since their penalty forms are fixed and pre-determined for any types of data.

Recently, l_p-norm ($p \in (0, 1)$) attracts great attention in the optimization framework, the idea that using l_p-norm can find sparse solutions is considered in [8]-[11]. Correspondingly, [12]-[17] propose l_p-norm ($0 < p < 1$) support vector machine for classification (l_p SVC), which replace the l_2-norm penalty by the l_p-norm ($p \in (0, 1)$) penalty in the objective function in the primal problem in the standard linear l_2 SVC. Compared with SVC with a fixed norm, l_p SVC is desired for feature selection since it can automatically select relevant features by adjusting the parameter p. However, l_p SVC is used only to solve classification problems. This motivates us to consider a new model with l_p-norm for regression problems.

This paper proposes a new method for regression – l_p-norm support vector regression (l_p SVR), which replaces l_2-norm by l_p-norm ($0 < p < 1$) in the classical l_2 SVR. Our new model is an adaptive learning procedure with l_p-norm ($0 < p < 1$), where the best p is automatically chosen by data. By adjusting the parameter p, l_p SVR can not only select relevant features but also improve the regression accuracy. In order to solve the non-convex problem in our model, an efficient algorithm is constructed using the successive linear approximation algorithm (SLA)([18]).

E-mail address: zhangchunhua@ruc.edu.cn Tel.:+86-010-8250-0694.

Selection and peer review under responsibility of the organizers of the 2013 International Conference on Computational Science
doi:10.1016/j.procs.2013.05.341

Now we describe our notation. All vectors are column vectors unless transposed to a row vector by a super script T. For a vector x in R^n, $[x]_i (i = 1, 2, \cdots, n)$ denotes the i-th component of x. $|x|$ denotes a vector in R^n of absolute value of the components of x. $\|x\|_p$ denotes that $(|[x]_1|^p + \cdots + |[x]_n|^p)^{\frac{1}{p}}$. Strictly speaking, $\|x\|_p$ is not a general norm when $0 < p < 1$, but we still follow this term l_p-norm, because the forms are same except that the values of p are different. $\|x\|_0$ is the number of nonzero components of x. For two vectors $x \in R^n$ and $y \in R^n$, $(x \cdot y)$ denotes the inner product of x and y.

This paper is organized as follows. In section 2, the l_p SVR is introduced. In section 3, the SLA is proposed to solve l_p SVR. In section 4, the lower bounds for the absolute value of nonzero entries in any local optimal solution is established. In section 5, numerical experiments are given to demonstrate the effectiveness of our method. We conclude this paper in section 6.

2. l_p Support Vector Regression

Suppose that the training set T is given by

$$T = \{(x_1, y_1), \cdots, (x_l, y_l)\} \in (R^n \times R)^l, \tag{1}$$

where $x_j \in R^n$, $y_j \in R$ $(j = 1, \cdots, l)$, the linear regression problem is to find a decision function $f(x) = (w \cdot x) + b$ to derive the value of y for any x by the function $y = f(x)$.

In the classical l_2 SVR, the decision function is decided by the solution to the following optimization problem:

$$\min_{w,b,\xi,\eta} \quad \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{l}(\eta_i + \xi_i)\,, \tag{2}$$

$$\text{s.t.} \quad ((w \cdot x_i) + b) - y_i \le \varepsilon + \eta_i\,,\ i = 1, \cdots, l\,, \tag{3}$$

$$y_i - ((w \cdot x_i) + b) \le \varepsilon + \xi_i, i = 1, \cdots, l\,, \tag{4}$$

$$\xi_i, \eta_i \ge 0\,,\ i = 1, \cdots, l. \tag{5}$$

Replacing the first term $\frac{1}{2}\|w\|_2^2$ in the objective function of the above problem by $\|w\|_p^p$ $(0 < p < 1)$, l_p SVR proposes the following problem:

$$\min_{w,b,\xi,\eta} \quad \|w\|_p^p + C\sum_{i=1}^{l}(\eta_i + \xi_i)\,, \tag{6}$$

$$\text{s.t.} \quad ((w \cdot x_i) + b) - y_i \le \varepsilon + \eta_i,\ ,\ i = 1, \cdots, l\,, \tag{7}$$

$$y_i - ((w \cdot x_i) + b) \le \varepsilon + \xi_i, i = 1, \cdots, l\,, \tag{8}$$

$$\xi_i, \eta_i \ge 0\,,\ i = 1, \cdots, l, \tag{9}$$

where C $(C > 0)$, p $(0 < p < 1)$ and ε $(\varepsilon > 0)$ are parameters. The algorithm of l_p SVR is described as follows:

Algorithm 1

1. Give the training set $T = \{(x_1, y_1), \cdots, (x_l, y_l)\} \in (R^n \times R)^l$, where $x_i \in R^n$,$y_i \in R, i = 1, \cdots, l$;
2. Select proper parameters C, p, ε, where $C > 0, 0 < p < 1$ and $\varepsilon > 0$;
3. Solve problem (6)-(9) and get the solution (w^*, b^*);
4. Select the feature set: $\{i|[w^*]_i \ne 0, (i = 1, \cdots, n)\}$;
5. Construct the decision function $y = (\tilde{w}^* \cdot x) + b^*$, where the components of $\tilde{w}^*$ are nonzero components of w^* and the components of $\tilde{x}$ are also corresponding to nonzero components of w^*.

3. The SLA for problem (6)-(9)

Consider the problem (6)-(9), the objective function is not differentiable, because of the absolute value in the first item. In order to make this problem smooth, we introduce the variable $v = ([v]_1, \cdots, [v]_n)^T$, and get the

following equivalent problem:

$$\min_{w,v,b,\xi,\eta} \quad \sum_{i=1}^{n}[v]_i^p + C\sum_{i=1}^{l}(\eta_i+\xi_i)\,, \tag{10}$$

$$\text{s.t.} \quad ((w\cdot x_i)+b)-y_i \le \varepsilon+\eta_i,\ ,\ i=1,\cdots,l\,, \tag{11}$$

$$y_i-((w\cdot x_i)+b) \le \varepsilon+\xi_i, i=1,\cdots,l\,, \tag{12}$$

$$\xi_i,\eta_i \ge 0\,,\ i=1,\cdots,l, \tag{13}$$

$$-v \le w \le v. \tag{14}$$

When $p\in(0,1)$, we note that the problem (10)-(14) is differentiable, but not convex. In fact, it is the minimization of a concave objective function over a polyhedral set. Even though it is difficult to find a global solution to this problem, a fast successive linear approximation (SLA) algorithm ([18]) terminates finitely at a stationary point which satisfies the necessary optimality condition for problem (10)-(14). For convenience we state the SLA algorithm below.

Algorithm 2 (SLA for problem (10)-(14))

1. Select the proper parameters $C>0, 0<p<1, \varepsilon>0$ and a precision $\delta(0<\delta<<1)$, start with a random $v^0=([v]_1^0,[v]_2^0,\cdots,[v]_n^0)^T$ and let $k=1$;
2. Solve the following problem

$$\min_{w,v,b,\xi,\eta} \quad \sum_{i=1}^{n}[v^{k-1}]_i^{p-1}[v]_i + C\sum_{i=1}^{l}(\eta_i+\xi_i)\,, \tag{15}$$

$$\text{s.t.} \quad ((w\cdot x_i)+b)-y_i \le \varepsilon+\eta_i,\ ,\ i=1,\cdots,l\,, \tag{16}$$

$$y_i-((w\cdot x_i)+b) \le \varepsilon+\xi_i, i=1,\cdots,l\,, \tag{17}$$

$$\xi_i,\eta_i \ge 0\,,\ i=1,\cdots,l, \tag{18}$$

$$-v \le w \le v. \tag{19}$$

   where $(v^{k-1})^{p-1}=([v^{k-1}]_1^{p-1},\cdots,[v^{k-1}]_n^{p-1})^T$, and get its solution $(w^k,b^k,\eta^k,\xi^k,v^k)$;
3. If $\left|\sum_{i=1}^{n}[v^{k-1}]_i^{p-1}([v^k]_i-[v^{k-1}]_i^{p-1})+C\sum_{i=1}^{l}(\eta_i^k-\eta_i^{k-1}+\xi_i^k-\xi_i^{k-1})\right| < \delta(0<\delta\ll 1)$, then stop and get the solution $w^*=w^k, b^*=b^k$; Otherwise, let $k=k+1$ and go back to step 2.

4. The lower bounds for nonzero components in solutions

In Algorithm 1, it is easy to see that our l_p SVR can accomplish feature selection and regression simultaneously. Feature selection needs to find the nonzero components of the solution to the problem (6)-(9). However, usually the above Algorithm 2 can only provide an approximate local solution where nonzero components in the solution can not be identified theoretically. Using a similar strategy in [8], we get the following theorem 1, which can be used to identify nonzero components in any local optimal solutions to the problem (6)-(9), even though the Algorithm 2 can only find the approximate local optimal solution.

Theorem 1 For any local optimal solution (w^*,b^*,ξ^*) to the problem (6)-(9), we have

$$|[w^*]_j| \ge (p/(C\sum_{i=1}^{l}|x_i|_j))^{\frac{1}{1-p}},\ j=1,2,\cdots,n.$$

Proof. Suppose $\|w^*\|_0=k,(1<k\le n)$, without loss of generality, let $w^*=([w]_1^*,\cdots,[w]_k^*,0,\cdots,0)^T$ where $[w]_i^*\ne 0, i=1,\cdots,k$. Let $\tilde{w}^*=([w]_1^*,\cdots,[w]_k^*)^T$, $\tilde{x}_i=([x_i]_1,\cdots,[x_i]_k)^T\in R^k, i=1,\cdots,l$, we consider a new optimization problem:

$$\min_{\tilde{w},\tilde{b},\tilde{\xi},\tilde{\eta}} \quad \|\tilde{w}\|_p^p + C\sum_{i=1}^{l}(\tilde{\eta}_i+\tilde{\xi}_i)\,, \tag{20}$$

$$\text{s.t.} \quad ((\tilde{w} \cdot \tilde{x}_i) + \tilde{b}) - y_i \le \varepsilon + \tilde{\eta}_i, \ , \ i = 1, \cdots, l \ , \tag{21}$$

$$y_i - \ ((\tilde{w} \cdot \tilde{x}_i) + b) \le \varepsilon + \tilde{\xi}_i, i = 1, \cdots, l \ , \tag{22}$$

$$\tilde{\xi}_i, \tilde{\eta}_i \ge 0 \ , \ i = 1, \cdots, l, \tag{23}$$

where $\tilde{w} \in R^k, \tilde{b} \in R, \tilde{\eta} \in R^l, \tilde{\xi} \in R^l$. Obviously, $(\tilde{w}^*, b^*, \eta^*, \xi^*)$ is a local minimizer of problem (20)-(23). According to the KKT condition, there exist Lagrange multipliers $\alpha_i^*, \beta_i^* (i = 1, \cdots, l)$ satisfy:

$$p(|\tilde{w}^*|^{p-1} \cdot sign(\tilde{w}^*)) - \sum_{i=1}^{l} (\beta_i^* - \alpha_i^*)\tilde{x}_i = 0, \tag{24}$$

$$0 \le \alpha_i^* \le C, \ 0 \le \beta_i^* \le C. \tag{25}$$

According to (24), we have

$$p(|\tilde{w}^*|^{p-1} \cdot sign(\tilde{w}^*)) = \sum_{i=1}^{l} (\beta_i - \alpha_i)\tilde{x}_i.$$

Furthermore, by (25), we have

$$p\,|\tilde{w}^*|^{p-1} = \left| \sum_{i=1}^{l} (\beta_i - \alpha_i)\tilde{x}_i \right| \le \sum_{i=1}^{l} |\beta_i - \alpha_i|\,|\tilde{x}_i| \le C \sum_{i=1}^{l} |x_i|, 0 < p < 1$$

So, $\left|w_j^*\right| \ge (p/(C \sum_{i=1}^{l} |x_i|_j))^{\frac{1}{1-p}}$, for $j = 1, \cdots, n$.

According to Theorem 1, we can identify the nonzero components of the local optimal solution to (6)-(9). Based on the Algorithm 2 and Theorem 1, the new algorithm is established as follows:

Algorithm 3 (l_p-SVR):

1. Give the training set $T = \{(x_1, y_1), \cdots, (x_l, y_l)\} \in (R^n \times R)^l$, where $x_i \in R^n, y_i \in R, i = 1, \cdots, l$;
2. Select proper parameters C, p, ϵ, where $C > 0, 0 < p < 1$;
3. Solve problem (6)-(9) by Algorithm 2 and get the solution (w^*, b^*);
4. Compute $L_j = (p/(C \sum_{i=1}^{l} |x_i|_j))^{\frac{1}{1-p}}, j = 1, \cdots, n$; select the feature index set: $F' = \{i | [w^*]_i | \ge L_i, i = 1, \cdots, n\}$;
5. Construct the decision function $f(x) = \text{sgn}((\tilde{w}^* \cdot \tilde{x}) + b^*)$, where $\tilde{w}^*$ are composed by the components in the F' of w^* and the components of $\tilde{x}$ are also corresponding to components in the feature set F' of w^*.

In the following section, our experiments are conducted according to the algorithm 3.

5. Numerical experiments

In this section, some experiments on simulation datasets and real datasets are conducted respectively, by comparing l_p SVR with l_2 SVR, l_1 SVR. Note that, the performance of each method depend on the parameters (C, p and ε in l_p SVR; C and ε in l_2 SVR and l_1 SVR). Therefore, these parameters should be adjusted properly. In our experiments, the best value of these parameters are chosen by five-fold cross validation. C is obtained through searching in the range $2^{-7} - 2^4$, p is chosen from $0.1 - 0.9$ and ε is chosen from 0.01 to 0.1.

In order to evaluate the performance of algorithms, some evaluation criteria ([19], [20]) commonly used should be introduced in the following:

$$MSE = \frac{1}{m} \sum_{i=1}^{m} (f(x_i) - y_i)^2, MAPE = \frac{\sum_{i=1}^{m} \frac{|y_i - f(x_i)|}{|y_i|}}{m} \times 100\%, NMSE = \frac{\sum_{i=1}^{m} (y_i - f(x_i))^2}{\sum_{i=1}^{m} (y_i - \bar{y})^2},$$

$$R^2 = \frac{\sum_{i=1}^{m} (f(x_i) - \bar{y})^2}{\sum_{i=1}^{m} (y_i - \bar{y})^2}, r = \frac{m \sum_{i=1}^{m} f(x_i) y_i - \sum_{i=1}^{m} f(x_i) \sum_{i=1}^{m} y_i}{\sqrt{(m \sum_{i=1}^{m} f(x_i)^2 - (\sum_{i=1}^{m} f(x_i))^2)(m \sum_{i=1}^{m} y_i^2 - (\sum_{i=1}^{m} y_i)^2)}},$$

where m is the number of testing samples, $f(x_i)$ denotes the predict value of x_i and $\bar{y}$ is the average value of $y_1, \cdots, y_l$.

5.1. Simulation datasets

The simulation datasets are generated as follows. The inputs $x_i \in R^n$ are stochastic vectors independently generated in [0, 1], and the numbers of samples and features are described in Table 1. The outputs are determined

Table 1. Description of simulation datasets

Datasets	No. of samples	No. of features
1	400	50
2	500	60
3	500	50
4	100	120

by some simple functions. For example, in dataset 1, the output y_i is given by

$$y_i = 2[x_i]_1 + 3[x_i]_2 + 4[x_i]_3 + 0.1 \times rand(1);$$

In dataset 2,

$$y_i = 8[x_i]_1 - 7[x_i]_5 + 6[x_i]_9 - 5[x_i]_{13} + 4[x_i]_{20} - 3[x_i]_{31} + 2[x_i]_{45} - [x_i]_{49} + rand(1);$$

In dataset 3,

$$y_i = 100[x_i]_3 + 20[x_i]_{17} + 3[x_i]_{21} + 0.4[x_i]_{36} + 0.05[x_i]_{44};$$

In dataset 4,

$$y_i = 2[x_i]_1 + 3[x_i]_2 + 4[x_i]_3 + rand(1).$$

The results on four datasets are illustrated in Table 2. We show the effectiveness of l_p SVR from two aspects: feature selection and regression accuracy. On the one hand, from the data in 3th column, it is easy to see that l_p SVR selects the minimal features. On the other hand, the data in 4th-8th column show that l_p SVR derives the smallest MSE, MAPE, NMSE, and the largest R^2, r among these methods in most datasets. This indicates that the statistical information in these datasets is well presented by our l_p SVR with fairly small feature sets and regression errors.

Table 2. Results on simulation datasets

Datasets	Regressor	No. of selected features	MSE	MAPE	NMSE	R^2	r
1	l_p SVR	3	0.0008	0.0064	0.0003	1.0080	0.9999
	l_2 SVR	50	0.0017	0.0084	0.0006	1.0059	0.9997
	l_1 SVR	3	0.0008	0.0062	0.0003	0.9963	0.9999
2	l_p SVR	8	0.0811	0.1561	0.0044	0.9848	0.9978
	l_2 SVR	60	0.0871	0.1623	0.0048	0.9999	0.9976
	l_1 SVR	16	63.0302	5.6449	3.4472	3.6134	0.7724
3	l_p SVR	5	0.0763	0.0346	0.0221	0.9596	0.9889
	l_2 SVR	50	0.1038	0.0393	0.0301	0.9622	0.9849
	l_1 SVR	16	0.3486	0.0802	0.1011	1.0428	0.9879
4	l_p SVR	3	0.1222	0.0780	0.0377	0.9023	0.9843
	l_2 SVR	120	1.2196	0.2420	0.3758	0.6735	0.7999
	l_1 SVR	3	0.1420	0.0734	0.0438	0.7510	0.9875

5.2. Real datasets

For further evaluation of our method, we choose four real datasets: ”bodyfat”, ”cpusmall”, ”housing” and ”insurance”, which are commonly used in testing machine learning algorithms. More detailed description can be found in Table 3.

Table 4 lists the results of three methods on four real datasets. It can be seen that our l_p SVR can accomplish the desired feature selection and achieve the good regression accuracy. The reason maybe that it can balance these two aspects better than the other two methods by adjusting the parameter p.

Table 3. Description of real datasets

Datasets	No. of samples	No. of features
bodyfat	252	14
cpusmall	500	12
housing	452	13
insurance	500	85

Table 4. Results on real datasets

Datasets	Regressor	No. of selected features	MSE	MAPE	NMSE	R^2	r
bodyfat	l_p SVR	1	0.0000	0.0017	0.0004	1.0017	0.9998
	l_2 SVR	14	0.0000	0.0016	0.0004	1.0041	0.9998
	l_1 SVR	1	0.0305	0.0981	1.1352	0.0550	0.1710
cpusmall	l_p SVR	8	0.0010	0.0124	0.1508	0.7845	0.9354
	l_2 SVR	12	0.0009	0.0122	0.1445	0.7973	0.9378
	l_1 SVR	2	0.0992	0.1625	15.4089	14.7173	0.2143
housing	l_p SVR	3	0.0097	0.0446	0.2607	0.7223	0.8607
	l_2 SVR	13	0.0097	0.0416	0.2587	0.7710	0.8641
	l_1 SVR	3	0.1372	0.2673	3.6738	3.7220	0.7769
insurance	l_p SVR	9	2.3464	0.1123	0.0141	0.9411	0.9933
	l_2 SVR	83	2.8524	0.1121	0.0172	0.9916	0.9917
	l_1 SVR	2	3.3524	0.1556	0.0198	0.9070	0.9927

6. Conclusions

For regression problems, a new model l_p SVR is proposed in this paper. The main contribution is that the desired feature selection and good regression performance are implemented simultaneously by introducing the adaptive norms – l_p-norm, where the parameter p can be chosen flexibly in $(0, 1)$ by data. Computational comparisons between our l_p SVR and other popular methods including l_2 SVR and l_1 SVR indicate the effectiveness of our method. We believe that its good performance mainly comes from the fact that the parameter p is adjusted properly.

Acknowledgements

This work has been partially supported by grants from National Natural Science Foundation of China (No.11271361, No.11201480) and Chinese Universities Scientific Fund (No.2011JS039).

References

[1] V.Vapnik, Statistical Learning Theory, Wiley, New York (1998).
[2] Y. Tian,Y. Shi, X. Liu, Recent advances on support vector machines research, Technological and Economic Development of Economy, 2012, 18(1): 5-33.
[3] Z. Qi, Y. Tian, Y. Shi, Robust twin support vector machine for pattern classification, Pattern Recognition, 2013, 46(1): 305-316.
[4] Z. Qi, Y. Tian, Y. Shi, Laplacian twin support vector machine for semi-supervised classification, Neural Networks, 2012, 35:46-53.
[5] Z. Qi, Y. Tian, Y. Shi, Twin support vector machine with Universum data, Neural Networks, 2012, 36C:112-119.
[6] N.Deng, Y.Tian, C.Zhang, Support Vector Machines – optimization based theory, algorithms and extensions, CRC Press (2012).
[7] P.Bradley, O.Managsarian, Feature selection via concave minimization and support vector machines, The Fifth International Conference on Machine Learning (1998), 82-90.
[8] X.Chen, F.Xu, Y.Ye, Lower bound theory of nonzero entries in solutions of l_2-l_p minimization (2009)http://www.standardford.edu/ yyye/.
[9] A. Bruckstein, D. Donoho, M. Elad, From sparse sulutions of systems of equations to sparse modeling of signals and images. SIAM Reviewer 51 (2009) 34-81.
[10] J. Fan, R. Li, Varible selection via nonconcave penalized likelihood and its oracle properties, J Amer Statis Assoc 96 (2001) 1348-1360.
[11] Z. Xu, H. Zhang, Y. Wang, X. Chang, $L_{\frac{1}{2}}$ regularizer, Science in China Series F-InfSci 52 (2009) 1-9.
[12] W. Chen, Y.Tian, L_p-norm proximal support vector machine and its applications, Procedia Computer Science, ICCS 1(1) (2010) 2417-2423.
[13] Y.Tian, J.Yu, W.Chen, l_p-norm support vector machine with CCCP, In Proc. the 7th FSKD (2010) 1560-1564.
[14] J.Tan, C.Zhang, N.Deng, Cancer related gene identification via p-norm support vector machine, The 4th International Conference on Computational Systems Biology (2010) 101-108.

[15] C.Zhang, J.Tan, etc. Feature Selection in multi-instance learning, The International Symposium on Operations Research and its Applications (2010) 462-469.
[16] J.Tan, Z.Zhang, L.Zhen, C.Zhang, N.Deng, Adaptive feature selection via a new version of support vector machine. Neural Comput & Applic, (2012) doi:10.1007/s00521-012-1018-y.
[17] C.Zhang, Y.Shao, J.Tan, N.Deng, Mixed-norm linear support vector machine, Neural Comput & Applic, (2012) doi:10.1007/s00521-012-1166-0.
[18] P.Bradley, O.Mangasarian, W.Street, Feature selection via mathematical programming, INFORMS Journal on Computing (1998), doi:10.1287/ijoc.10.2.209.
[19] S.Weisberg, Applied linear regression seconded, Wiley, New York (1985).
[20] R.Staudte, S.Sheather, Robust estimationand testing: Wiley series in probability and mathematical statistics, Wiley, New York (1990).