Some Remarks on our paper: Support Vector Machines for Large-Scale Regression Problems

Ronan Collobert
IDIAP,
CP 592, 1920 Martigny,
Switzerland,
collober@idiap.ch

Samy Bengio
IDIAP,
CP 592, 1920 Martigny,
Switzerland,
bengio@idiap.ch

August 16, 2000

Abstract

The following remarks have also been added in the appendix of the report, which is available at `ftp://ftp.idiap.ch/pub/reports/2000/rr00-17.ps.gz`.

1 Things that should have been in the paper

1.1 On the experiments

We forgot to specify that the kernel used in all the experiments was a Gaussian kernel. Thanks to Flake[1] for his remark on this.

In the conclusion of our paper, we said that we also implemented a classification version of the algorithm which was similar to the one proposed by Joachims [3] and that our implementation was twice as fast as *SVM-Light*. This assertion holds only for non-sparse data because *SVM-Light* has been specially designed for sparse data, while it was not the case in the first version of *SVMTorch*. The current version, which now includes sparse data format, is 1.33 times faster that *SVM-Light* for sparse data (and still 2 times faster for non-sparse data).

1.2 On the decomposition method

Since the publication of our technical report, we have been aware of many other decomposition algorithms for regression problems that we have not even cited. We try here to resume their work and the relation with our paper.

Shevade *et al* [8] proposed two modifications of the *SMO* algorithm for regression, based on a previous paper from the same team [5] for the classification problem. Laskov [6] proposed also a decomposition method for regression problems which is very similar to the second one from Shevade *et al.* In fact, it is easy to see that Laskov's method with a subproblem of size 2 uses the same selection algorithm as well as the same termination criterion.

Their method for selecting the working set is *very* similar to the one we proposed, but while we propose to select variables α_i independantly of their counterpart $\alpha_i^\star$, they propose to select simultaneously *pairs* of variables $\{\alpha_i,\ \alpha_i^\star\}$. Even if this seems to be a small difference, let us note that since $\alpha_i\,\alpha_i^\star = 0\ \forall i$,

[1] `http://www.neci.nj.nec.com/homepages/flake/`

one of the two variables α_i or $\alpha_i^\star$ is always equal to 0, and choosing the α_i and the $\alpha_i^\star$ independantly can thus help to quickly eliminate half of the variables, thanks to the *shrinking* phase[2], which of course have a direct impact on the speed of our program.

Similarly, Smola and Schölkopf [9] also proposed earlier to use a decomposition algorithm for regression based on *SMO* using an analytical solution for the subproblem, but again they propose to select 2 pairs of variables (2 α and their corresponding $\alpha^\star$) instead of 2 variables, which leads to a different mathematical formulation. As for Laskov and Shevada *et al*, they do not use *shrinking* which is, in our opinion, the main speed gain of our algorithm.

Finally, Flake and Lawrence [2] proposed again a modification of *SMO* for regression which uses the heuristics proposed by Platt [7] and those from Smola and Schölkopf [9] but works on a new variable $\lambda_i = \alpha_i - \alpha_i^\star$, which leads to a different analytical solution. In fact, all the analytical solutions proposed by these authors are different but need to handle multiple cases for the solution, except our method.

1.3 On the convergence of the algorithm

In our paper, we did not talked about the convergence of our algorithm. A paper from Chang *et al* [1] talks about the convergence of some SVM algorithms based on a decomposition method. However, using their arguments, we cannot conclude that our algorithm converges to the optimum, even when no shrinking is done. The hypothesis they use in their proof regarding their method to search for a feasible solution is slightly different from our method and thus we cannot use their proof in our case. Keerthi *et al* [4] also proposed a convergence proof for their method [5], but it applies only to their classification case. However, we will see in the next section that it also applies to our classification method *as well as* our regression method.

2 Remarks on the relation between many SVM algorithms

As we said in our paper, the algorithm we used in classification is the same, mathematically speaking, as the one proposed by Joachims [3]. Let us now consider[3] the paper from Keerthi *et al* [5] which proposes two algorithms based on Platt's algorithm, *SMO*. In particular, let us focus on the second method they propose and let us compare this method to the algorithm proposed by Joachims in the case of a working set of size 2.

We strongly suggest to the reader to refer to the papers from Joachims and Keerthi *et al* in order to understand the following notations which will not be re-explained here.

At each iteration, Keerthi *et al* propose to start by selecting two variables in their working set. Following their notation, let us denote

$$\begin{aligned} I_0 &= \{i : 0 < \alpha_i < C\} \\ I_1 &= \{i : y_i = 1,\ \alpha_i = 0\} \\ I_2 &= \{i : y_i = -1,\ \alpha_i = C\} \\ I_3 &= \{i : y_i = 1,\ \alpha_i = C\} \\ I_4 &= \{i : y_i = -1,\ \alpha_i = 0\} \end{aligned}$$

and let us denote also i_{low} and i_{up} the index of the two selected variables. They verify

$$F_{i_{low}} = b_{low} = \max\{F_i :\ i \in I_0 \cup I_3 \cup I_4\}$$

and

$$F_{i_{up}} = b_{up} = \min\{F_i :\ i \in I_0 \cup I_1 \cup I_2\}$$

[2]this is verified in practice

[3]Thanks to Patrick Haffner (`http://www.research.att.com/~haffner`) and Ryan Rifkin (`http://five-percent-nation.mit.edu/PersonalPages/rif/`) who have stimulated our interest on this.

where

$$F_i = \sum_j \alpha_j y_j k(\boldsymbol{x_i}, \boldsymbol{x_j}) - y_i.$$

One can easily remark that $F_i = \omega_i$, where ω_i is the sorting variable used by Joachims in his paper. Joachims defines the following constraints

$$\begin{array}{c} d_i \geq 0, \quad \forall i : \alpha_i = 0 \\ d_i \leq 0, \quad \forall i : \alpha_i = C \end{array} \tag{1}$$

and select the following working set variables:

- the one that corresponds to the highest ω_i such that $0 < \alpha_i < C$ or such that $d_i = -y_i$ verify (1), and
- the one that corresponds to the smallest ω_i such that $0 < \alpha_i < C$ or such that $d_i = y_i$ verify (1).

This is indeed equivalent to the choice made by Keerthi *et al.*

Both algorithms then solve the sub-problem and test the optimality of the general problem. The algorithm from Keerthi *et al* stops when

$$b_{low} - b_{up} \leq \tau$$

where τ is a tolerance factor defined by the user. The algorithm from Joachims stops when the following conditions are verified:

$$\begin{array}{c} \forall i \text{ such that } 0 < \alpha_i < C, \quad \lambda^{eq} - \tau \leq y_i - \sum_j \alpha_j y_j k(\boldsymbol{x_i}, \boldsymbol{x_j}) \leq \lambda^{eq} + \tau \\ \forall i \text{ such that } \alpha_i = 0, \quad y_i(\sum_j \alpha_j y_j k(\boldsymbol{x_i}, \boldsymbol{x_j}) + \lambda^{eq}) \geq 1 - \tau \\ \forall i \text{ such that } \alpha_i = C, \quad y_i(\sum_j \alpha_j y_j k(\boldsymbol{x_i}, \boldsymbol{x_j}) + \lambda^{eq}) \leq 1 + \tau \end{array} \tag{2}$$

where λ^{eq} is defined as follows

$$\lambda^{eq} = \frac{1}{|A|} \sum_{i \in A} \left[y_i - \sum_j \alpha_j y_j k(\boldsymbol{x_i}, \boldsymbol{x_j}) \right]$$

with

$$A = \{i : 0 < \alpha_i < C\}.$$

It is easy to see that equations (2) are equivalent to

$$\begin{array}{c} \forall i \in I_0, \quad \lambda^{eq} - \tau \leq -F_i \leq \lambda^{eq} + \tau \\ \forall i \in I_1 \cup I_2, \quad F_i \geq -\lambda^{eq} - \tau \\ \forall i \in I_3 \cup I_4, \quad F_i \leq -\lambda^{eq} + \tau. \end{array}$$

Moreover, since $-b_{low} \leq \lambda^{eq} \leq -b_{up}$, if the optimality conditions from Keerthi *et al* are verified, then

$$\forall i \in I_0 \cup I_3 \cup I_4, \quad F_i \leq b_{up} + \tau \leq -\lambda^{eq} + \tau$$

and

$$\forall i \in I_0 \cup I_1 \cup I_2, \quad -F_i \leq -b_{low} + \tau \leq \lambda^{eq} + \tau$$

which implies the optimality conditions from Joachims.

Since the optimality test of *SVM-Light* is weaker that the one from Keerthi *et al* [4], it is easy to see that one can apply their theorem to show that *SVM-Light* converges for subproblems of size 2 (as well as our classification algorithm).

In the same way, one can show that the optimality test we used in our regression algorithm is weaker than the general algorithm proposed by Keerthi *et al* [4] and it is easy to see that given the fact that our algorithm uses a selection method that choose independantly the α and the $\alpha^\star$, the proof from Keerthi *et al also applies to our regression algorithm when the subproblem size is set to 2.*

3 Conclusion

In conclusion, we note that all these decomposition algorithms are extremely related. For instance, subset selection algorithms from Keerthi *et al* and Joachims are strictly identical if the *shrinking* is not used and for a working subset of size 2 in *SVM-Light*.

Moreover, it is very easy to see that the regression method from Laskov (again with a subset of size 2) is equivalent to the one from Shevade *et al*, which is the same as the classification one from Keerthi *et al*. Note also that the method from Flake and Lawrence could be modified using the second modification from Keerthi *et al* and would thus be enhanced.

Finally, we think that shrinking makes the main difference with regard to speed, and the selection method we have chosen simplifies the resolution of the analytic quadratic problem and enables to obtain a convergence proof for the regression problem.

References

[1] C. Chang, C. Hsu, and C. Lin. The analysis of decomposition methods for support vector machines. In *IJCAI'99, Workshop on Support Vector Machines*, 1999.

[2] G.W. Flake and S. Lawrence. Efficient SVM regression training with SMO. Submitted to Machine Learning. Available at `http://external.nj.nec.com/homepages/flake/smorch.ps`.

[3] Thorsten Joachims. Making large-scale support vector machine learning practical. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods*. MIT Press, 1999.

[4] S.S. Keerthi and E.G. Gilbert. Convergence of a generalized SMO algorithm for SVM classifier design. Technical Report CD-00-01, Control Division, Dept. of Mechanical and Production Engineering, National University of Singapore, 2000. Available at `http://guppy.mpe.nus.edu.sg/~mpessk/svm/conv_ml.ps.gz`.

[5] S.S. Keerthi, S.K. Shevade, C. Bhattacharyya, and K.R.K. Murthy. Improvements to platt's SMO algorithm for SVM classifier design. Technical Report CD-99-14, Control Division, Dept. of Mechanical and Production Engineering, National University of Singapore, 1999. To appear in Neural Computation. Available at `http://guppy.mpe.nus.edu.sg/~mpessk/smo_mod.ps.gz`.

[6] P. Laskov. An improved decomposition algorithm for regression support vector machines. In S.A. Solla, T.K. Leen, and K.-R. Müller, editors, *Advances in Neural Information Processing Systems 12*. MIT Press, 2000. Available at `http://www.cis.udel.edu/~laskov/publications/NIPS-99.ps.gz`.

[7] John C. Platt. Fast training of support vector machines using sequential minimal optimization. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods*. MIT Press, 1999.

[8] S.K. Shevade, S.S. Keerthi, C. Bhattacharyya, and K.R.K. Murthy. Improvements to smo algorithm for svm regression. Technical Report CD-99-16, Control Division, Dept. of Mechanical and Production Engineering, National University of Singapore, 1999. To appear in IEEE Transaction on Neural Networks. Available at `http://guppy.mpe.nus.edu.sg/~mpessk/smoreg_mod.shtml`.

[9] A. J. Smola and B. Schölkopf. A tutorial on support vector regression. Technical Report NeuroCOLT Technical Report NC-TR-98-030, Royal Holloway College,University of London, UK, 1998.