Interval Nonlinear Solver with Symbolic Preprocessing for Training AI Tools in Presence of Perturbations

Bartłomiej Jacek Kubica[1], Jarosław Kurek[2]

[1] *Department of Applied Informatics, Warsaw University of Life Sciences, Poland, bartlomiej_kubica@sggw.pl*
[2] *Department of Applied Informatics, Warsaw University of Life Sciences, Poland, jaroslaw_kurek@sggw.pl*

Training various artificial intelligence (AI) tools is a hard problem, often requiring to solve a difficult nonlinear optimization problem. Numerical finding its solution can be significantly accelerated by proper symbolic techniques.

This is the case as for artificial neural networks (ANN); e.g., [8], as for support vector machines (SVM); e.g., [10], and for many other techniques.

To be succinct, we can either solve the optimization problem:

$$\min_w \sum_i ||f(x_i, w) - y_i|| \;, \tag{1}$$

or the nonlinear system:

$$f(x_i, w) - y_i = 0 \;\text{ for all } i = 1, \ldots, N \;. \tag{2}$$

In the above formulae, (x_i, y_i) are training examples and w is the vector of parameters, we are trying to determine in the learning process. For an ANN w represents weights of links between neurons; for SVM – parameters of the Gaussian kernel and of the soft margin.

To train the AI tools we can, in particular, solve the system of nonlinear equations, representing the necessary conditions for optimality of (1) or solve the system (2) directly. Other equations systems also arise in training such tools (e.g., [3]).

Solutions of such systems can be found by a few algorithms. We propose using interval methods (see, e.g., [5]), as this approach has proven to be useful in solving nonlinear systems – both well-determined and underdetermined ones.

One of the advantages of interval calculus is that it can deal with uncertainties in data, in a natural manner: instead of taking specific numbers (x_i, y_i) as inputs, we can use intervals $(\boldsymbol{x}_i, \boldsymbol{y}_i)$, containing the perturbed values.

The solver we use for training all these tools is HIBA_USNE [4], described, i.a., in [6], [7]. Interval arithmetic is augmented by the use of algorithmic differentiation [1] and symbolic preprocessing techniques, based on CoCoALib [2] to improve the performance of the interval solver. For an ANN, the systems of equations are non-polynomial, but it can still benefit from some symbolic techniques. As we encounter terms of the form $\exp\Big(\sum_{i=1}^{N} w_i x_i\Big)$, we can add new terms $t_i = \exp(w_i x_i)$. In case the terms repeat in other equations, they can be removed using the Gröbner basis theory. The paper is going to discuss possible improvements, obtained by this approach.

As an illustrative example, we consider the problem of determining the state of a drill (good, suspicious, damaged). We apply ANNs and SVMs to solve it.

In the paper we are going to present computational results for both AI tools. We show how interval methods combined with computer algebra and algorithmic differentiation help to model perturbations and tune the classifiers in their presence.

References

[1] *ADHC*, Algorithmic Differentiation and Hull Consistency enforcing, C++ library, https://www.researchgate.net/publication/316687827_HIBA_USNE_Heuristical_Interval_Branch-and-prune_Algorithm_for_Underdetermined_and_well-determined_Systems_of_Nonlinear_Equations_-_Beta_25 (2017).

[2] J. Abbott and A. M. Bigatti, *CoCoALib: a C++ library for doing Computations in Commutative Algebra*, http://cocoa.dima.unige.it/cocoalib (2017).

[3] M. Beheshti and A. Berrached and A. de Korvin and C. Hu and O. Sirisaengtaksin, *On interval weighted three-layer neural networks*, in *Proceedings of Simulation Symposium*, IEEE, pp. 188-194 (1998).

[4] *HIBA_USNE*, Heuristical Interval Branch-and-prune Algorithm for Underdetermined and well-determined Systems of Nonlinear Equations, C++ library, https://www.researchgate.net/publication/316687827_HIBA_USNE_Heuristical_Interval_Branch-and-prune_Algorithm_for_Underdetermined_and_well-determined_Systems_of_Nonlinear_Equations_-_Beta_25 (2017).

[5] R. B. Kearfott, *Rigorous Global Search: Continuous Problems*, Kluwer, Dordrecht, 1996.

[6] B.J. Kubica, *Presentation of a highly tuned multithreaded interval solver for underdetermined and well-determined nonlinear systems*, Numerical Algorithms, **70**, 4, pp. 929–963 (2015).

[7] B.J. Kubica, *Parallelization of a bound-consistency enforcing procedure and its application in solving nonlinear systems*, Journal of Parallel and Distributed Computing, published online https://doi.org/10.1016/j.jpdc.2017.03.009 (2017).

[8] J. Kurek and S. Osowski, *Support vector machine for fault diagnosis of the broken rotor bars of squirrel-cage induction motor*, Neural Computing and Applications, **19**, 4, pp. 557-564 (2010).

[9] *LIBSVM* C++ library, `https://www.csie.ntu.edu.tw/~cjlin/libsvm/` (2017).

[10] B. Świderski and J. Kurek and S. Osowski, *Multistage classification by using logistic regression and neural networks for assessment of financial condition of company*, Decision Support Systems, **52**, 2, pp. 539-547 (2012).