

Using Artificial Intelligence for Model Selection

Darin Goldstein
Department of Computer Science
California State University, Long Beach
daring@cecs.csulb.edu

Will Murray
Department of Mathematics
California State University, Long Beach
wmurray@csulb.edu

Binh H. Yang
Department of Epidemiology
University of California, Los Angeles
School of Public Health
binhyang@ucla.edu

Abstract

We apply the optimization algorithm Adaptive Simulated Annealing (ASA) to the problem of analyzing data on a large population and selecting the best model to predict the probability that an individual with various traits will have a particular disease. We compare ASA with traditional forward and backward regression on computer simulated data. We found that the traditional methods of modeling are best for smaller datasets whereas a numerically stable ASA seems to perform better on larger and more complicated datasets.

Keywords: artificial intelligence, modeling, simulated annealing

1 Introduction

In this paper we apply a new method for model selection on large data sets using the artificial intelligence algorithm Adaptive Simulated Annealing (ASA). We focus here on an epidemiological setting, although our techniques could be applied more widely. The objective is to analyze a large amount of data on many different characteristics of a given population and to select the model that best predicts the probability that an individual with various traits will have a particular disease outcome. The inclusion of categorical predictors that split up into multiple “dummy” variables and the possibility of cross terms between the different characteristics make the number of variables in a complete model prohibitively large. Thus it is necessary to select a small number of variables that will give the most informative model. Traditionally, this is done using forward regression or backward regression. Instead, we place an upper bound on the number of variables we want in our final model in advance and use ASA to decide which characteristics (or combinations of characteristics) each variable should represent to produce the model with the lowest C_p statistic, a reflection of bias and variance. We ran computer simulations on populations of 100,000, 500,000, and 1,000,000, equivalent to the data that might be available for a major metropolitan area. Our results suggest that this method produces a model with a C_p

statistic that is consistently close to optimal whereas both forward and backward regression occasionally do not.

2 The Problem

We assume that we have access to a large data set with complete records of a population with a variety of fields, including both continuous variables (such as age and alcoholic consumption) and categorical variables (such as gender, ethnicity, and blood type). We also assume that we know whether each individual in the population has a particular disease, for example, liver cancer. (This is a binary response, since the outcome is either “yes” or “no”, but our method could be easily modified to predict a continuous response, for example, years of life lost due to liver cancer.) In our computer simulations we used models of varying numbers of people with characteristics randomly generated according to standard probabilities. We assigned probabilities of getting the disease according to various risk factors among the characteristics and let the computer randomly choose which individuals to afflict. The characteristics we used were age, gender, Hepatitis B viral infection, Hepatitis C viral infection, aflatoxin exposure, genetic marker, alcohol, and tobacco. Simulations with 1,000,000 people, equivalent to a major metropolitan area, resulted in about 400-450 people getting the disease in any given trial. We then selected an equal number of healthy people randomly from the population and added these to the diseased people to form a data set for a simulated population-based case-control study.

A classical problem of public health is then to develop a model from the data set that predicts an individual’s probability of getting the disease based on his/her combination of characteristics. To do this, we create a set of variables $\vec{w} = \{w_i\}$ as follows: For each continuous characteristic (such as age or alcohol consumption) we create one variable, and for each categorical characteristic (such as gender or blood type) with n categories, we create $n - 1$ indicator variables that can take values of 0 or 1. The i th indicator formed from a particular categorical variable will indicate inclusion in category i . (We only need $n - 1$ of these indicator variables because if all such variables are 0, then we are guaranteed inclusion into the final category.) We then posit a model of the form

$$g(\vec{w}) = \gamma_0 + \sum_i \gamma_i w_i + \sum_{i < j} \gamma_{i,j} w_i w_j$$

in which the γ_i and $\gamma_{i,j}$ are constant coefficients and we omit the cross term $w_i w_j$ when w_i and w_j arise from the same categorical variable. This model is linear (in the sense that if all variables but one are held constant, it is linear in the nonconstant variable), but we could as easily include higher degree terms and cross terms involving more than two variables if necessary. In practice it is uncommon for biological models to include terms involving three or more variables; we exclude these terms because they are difficult to interpret biologically.

To simplify the notation, we can change variables from \vec{w} to \vec{x} , where each x_i represents either 1 (the constant term), one of the w_i ’s, or one of the $w_i w_j$ ’s. Then, replacing the various γ ’s with a single set of coefficients $\{\beta_i\}$, the model above becomes

$$g(\vec{w}) = f(\vec{x}) = \sum_i \beta_i x_i = \vec{\beta} \cdot \vec{x} \tag{1}$$

In the traditional approach to this problem, which, as we will see below, uses forward or backward regression, the ordering of the x_i ’s, in other words, which x_i is assigned to which w_i or $w_i w_j$, is not important. However, our method using Adaptive Simulated Annealing depends heavily on finding a structured way of assigning the x_i ’s so that x_i will have some relation to x_j when i is close to j . We will discuss this further in Section 4 below.

One final modification is necessary to our model. To account for the fact that the response from a known individual is either “healthy” or “sick”, we can, at the end, transform the model (1) by the logistic function

$$P(\vec{x}) = \frac{1}{1 + e^{-f(\vec{x})}}.$$

For a detailed explanation of this transformation, see Section 12.12 in [33].

The problem now is that of economy of terms; in our simulation, for example, the eight characteristics we studied produced 61 variables, x_0 to x_{60} . We therefore try to find a model using fewer terms and identify the x_i 's (and hence the individual characteristics or combinations thereof) that produce the most informative results. A commonly used measure of the efficiency of a model is the C_p statistic

$$C_p = p + \frac{(s^2 - \hat{\sigma}^2)(n - p)}{\hat{\sigma}^2},$$

where p is the number of variables in the current model, s^2 is the mean square error, and n is the total possible number of variables. Finally, $\hat{\sigma}^2$ is an estimate of σ^2 , the error variance in the population; since this latter quantity is unavailable, we take $\hat{\sigma}^2$ to be the mean square error for the most complete model, i.e. the one incorporating all the variables. Then a lower C_p value indicates a more desirable model. For a general overview of these terms, see Chapters 11 and 12 in [33]; for a more detailed development of the the C_p statistic, see [28].

Two traditional methods of solving this problem are forward regression and backward regression. In the former, one starts with only the constant variable and adds variables to the model one at a time, each time selecting the variable that lowers the C_p statistic the most. The process terminates when no variables can be added that lower the statistic. Backward regression is similar, starting with all the variables and progressively deleting them. Both procedures are computationally intensive, since they involve at each step computing a least squares solution to $\mathbf{X}\vec{\beta} = \vec{y}$ (where \mathbf{X} represents the data matrix) for each variable that is under consideration to be added or deleted from the model. This in turn requires finding the inverse (or pseudoinverse) of the matrix $\mathbf{X}^T\mathbf{X}$, which has dimension equal to the number of variables currently under consideration (up to 61x61 in our simulations). As greedy algorithms, they are also not guaranteed to produce an optimal solution.

As we noted above, since forward and backward regression test at each step every remaining variable to see which one should be added or deleted, these methods are not dependent on the ordering of the variables x_i .

Our new method described in Section 4 below is to determine an appropriate number of variables in advance and then to use ASA to determine which variables to include in the model to yield the lowest C_p statistic. We will see that the ordering of the x_i 's is important to this method.

3 Adaptive Simulated Annealing

Simulated annealing was introduced in 1983 by Kirkpatrick, Gelatt, and Vecchi ([24]) as a method of function optimization that is particularly well suited to functions that are difficult to evaluate in any continuous manner. It is based on the physical process of annealing, in which the molecules of a material are brought into a crystalline arrangement by gradually cooling the material. Since the crystal is the most ordered configuration of the molecules, it is the one that minimizes the total energy of the system, so the molecules should ultimately come to rest in that configuration. However, there may be other arrangements of the molecules that are stable despite having a higher total energy than the minimum (i.e. local minima of the energy function), and to avoid the system coming to rest in one of these arrangements the scientist must be careful to give the system enough initial energy and to avoid cooling it too quickly. At each stage of the cooling, small temperature fluctuations within the material will create and destroy defects until equilibrium for that temperature is achieved.

Simulated annealing is a procedure for minimization of a function of several variables in which the values of the variables represent configurations of the molecules of the system and the objective function represents its total energy. The computer initially assigns random values to the variables and then gives the system a certain “temperature”, i.e. a tendency of the variables to move randomly. Each move will affect the total energy of the system as measured by the objective function, and the temperature is used to determine the probability that a move that raises the total energy will be accepted. After enough moves have been made to simulate the equilibrium activity of the material at that temperature, the temperature of the system is lowered and the process begins again. Eventually, the temperature is low enough that the variables no longer change significantly and a minimum is achieved. It is important to note, however, that although physical annealing is known in theory to produce the global minimum of the total energy of a system, the efficiency of simulated annealing depends on the function being minimized and it is difficult to guarantee that a given annealing schedule will produce a global minimum rather than a local one.

In our investigation we use an adaptive version of simulated annealing introduced by Lester Ingber ([16], [17], [19], [22]) as an improvement on his earlier algorithm for Very Fast Simulated Reannealing ([12], [9]). ASA takes advantage of structure on the input of the objective function (in our case, the ordering of the x_i 's described below in Section 4) to decrease the running time of the algorithm and increase the probability that it will find a global minimum of the function. Optimization using VFSR and ASA has been effectively applied in wide variety of situations, including three dimensional image compression ([6], [5]), modeling of financial markets ([20], [11], [8], [13], [10], [30], [29]), dairy farming ([25], [26]), neural networks ([4], [3], [7], [15]), geophysical inversion ([31]), magnetic resonance imaging ([2]), electroencephalography ([14], [21]), and combat simulation ([1], [18]). An article on the many applications of ASA has appeared in The Wall Street Journal ([34]).

4 The Algorithm

The point at which our new method departs from traditional solutions to this problem is in the variable change from the w_i 's to the x_i 's in the model (1) above. As we have seen at the end of Section 2, the efficiency of forward and backward regression is not dependent on the ordering of the x_i 's. In our method, however, we use the following “zipper” algorithm to ensure that if i is close to j , then x_i and x_j represent similar values of the w 's: Suppose, for example, that there are only four w_i 's. Then the x_i 's would be assigned as follows:

$$\begin{aligned}
 x_0 &:= 1, \\
 x_1 &:= w_1, & x_2 &:= w_1 w_2, & x_3 &:= w_1 w_3, & x_4 &:= w_1 w_4, \\
 x_5 &:= w_4, & x_6 &:= w_3 w_4, & x_7 &:= w_2 w_4, \\
 x_8 &:= w_2, & x_9 &:= w_2 w_3, \\
 x_{10} &:= w_3.
 \end{aligned}$$

(Here we have assumed that there were no categorical characteristics with more than two categories, so that no w_i and w_j arose from the same characteristic. If they had, then the corresponding cross term $w_i w_j$ would be omitted from the assignment above.) By using this algorithm, we attempt to structure the search space in such a way as to make the search for the optimal solution as easy as possible for ASA. To get an idea of why this structuring of the search space is important, consider the difference between trying to find the minimum of a straight line and trying to find the minimum of a series of randomly placed points in the plane. Obviously, the former is much easier (both for a human and a computer). However, the former situation can become the latter very quickly if we allow the points along the x -axis to shuffle themselves randomly (thereby causing the x -coordinates of the line to shuffle themselves randomly and removing the ordered structuring of the search space). In the model above, by guaranteeing that x_i and x_{i+1} both contain exactly one

of the w_j 's in common, we hope that the C_p value does not change too radically when transitioning between the two. Of course, we can make no guarantees.

We decide in advance the number p of the x_i 's we want our model to include. This could be any number from 0 to n , but we found in our simulations with $n = 60$ that $p \approx 5 - 10$ was usually optimal. This range is based on the outcomes of forward and backward regression, on trial and error with our ASA program, on running time (trials with ten variables took about ten minutes on a Pentium 4 850 MHz processor), and on current practice among researchers in epidemiology.

We then set up dummy variables z_i , $1 \leq i \leq p$, which may take integer values between 1 and n depending on which of the x_i 's would be included in the model. Our goal is then to find the values of the z_i 's that minimize the objective function defined by the C_p statistic for a given model. This is where we use simulated annealing.

The publicly available C-code for ASA ([16]) allows the user to define how to evaluate the objective function to be minimized. In our case, evaluation on a particular set of values of the z_i 's required building a model with the corresponding x_i 's included, finding the coefficients β_i by calculating a least squares solution to $\mathbf{X}\vec{\beta} = \vec{y}$, and calculating the C_p statistic for that model. We modified the public code accordingly and ran multiple simulations with various sizes of data sets.

5 The Results

We ran several simulations on computer-generated data. In all cases, we used our ASA method and also ran forward and backward regression on the data as a control. We compared the three methods according to the final C_p statistic produced and the running time required to achieve it as measured by the number of function evaluations involved. Here one function evaluation is defined to be the calculation of the C_p statistic for a given set of variables as described at the end of Section 4, since forward and backward regression use the same procedure as they search for variables to be added to or deleted from the model. It is true that not all function evaluations will take equal amounts of processor time, since an evaluation with more variables will involve finding the pseudoinverse of a larger matrix; however, we believe, in keeping with standard practice in computer science, that such a measure is still meaningful enough to merit study.

We describe first the results of our simulations on computer-generated data. Using the characteristics described above in Section 2, there were 60 possible variables; we set our ASA program to search for the best seven variables. We ran five simulations each on populations of size 100,000, 500,000, and 1,000,000 with the following results.

Simulated populations of 100,000 (approximately 20 diseased, 20 healthy studied).

	Forward Regression	Backward Regression	ASA
C_p statistic	-46.387	-29.574	-46.734
	-45.831	-43.318	-48.968
	-54.046	-50.251	-53.009
	-56.199	-56.199	-56.199
	-54.936	-49.022	-54.724
Function evaluations	177	1725	1341
	119	1802	1677
	177	1824	1587
	119	1830	2827
	177	1820	199

Simulated populations of 500,000 (approximately 225 diseased, 225 healthy studied).

	Forward Regression	Backward Regression	ASA
C_p statistic	-47.511	-41.831	-47.510
	-43.141	-37.691	-43.152
	-50.803	-41.506	-51.160
	-36.406	-35.767	-41.127
	-45.648	-41.693	-46.063
Function evaluations	119	1815	2442
	399	1785	3336
	290	1764	1305
	119	1739	3780
	452	1752	2994

Simulated populations of 1,000,000 (approximately 450 diseased, 450 healthy studied).

	Forward Regression	Backward Regression	ASA
C_p statistic	-47.162	-39.287	-48.347
	-44.149	-41.134	-44.148
	-40.755	-21.738	-40.743
	-16.518	-22.845	-46.679
	-43.012	-40.003	-38.235
Function evaluations	234	1764	461
	345	1764	329
	234	1659	678
	177	1620	690
	504	1752	1913

Asymptotically, the number of function evaluations necessary to compute forward regression is easily seen to be $O(nk)$ where n represent the total number of variables under consideration and k represents the final number of variables chosen in the model. It is a notoriously difficult problem of theoretical computer science to analyze the computational complexity of ASA [23, 27, 32]. We can only present our results and leave it to the interested reader to draw his own conclusions.

6 Advantages and Limitations

We found that running ASA in most cases produces a slightly lower C_p statistic especially on the trials with large data sets. The cases in which ASA did not produce a lower C_p statistic were ones in which the optimal solution required significantly more or fewer than the seven variables we programmed ASA to search for; in these cases the C_p statistic produced by ASA was still close to that of the other methods, and when we reconfigured ASA to search for an appropriate number of variables it produced a lower C_p statistic. It appears that for small data sets with few parameters, the traditional methods of modeling are slightly preferable to ASA. The benefits of ASA seem to be more pronounced for larger data sets.

Another advantage of ASA is that its results seem to be more predictable than those of the other methods. Although in many tests the results of the three methods were comparable, in several instances forward regression and backward regression produced C_p statistics that were considerably higher than the lowest obtained C_p statistic; the statistics produced by ASA were never far from the best statistics produced by the three methods.

One disadvantage of our method is that it requires the user to determine the number of variables in the model beforehand. Simulations run under the same conditions can have optimal solutions (as determined by forward and backward regression) requiring quite different numbers of variables; in our simulations with 60 possible nonconstant variables we found some for which the C_p statistic was optimized with just one variable and some requiring as many as 21.

In practice this should not be a serious concern for three reasons. One is that a researcher in epidemiology will usually have a rough idea of the desired number of variables in advance as a compromise between the needs of the problem and the availability of computing power. (The assumption above that the model is linear already requires that the researcher know in advance something about the desired model.) A second reason is that as part of the random component of the program it automatically checks the possibility of deleting variables from the model, so if the minimum involves fewer variables than the user selects, the program will still find it. Finally, this property may even be considered an advantage, since our method allows the user to control the sophistication of the model (i.e. the number of variables it will use) in advance, whereas with forward and backward regression the models produced may have widely varying numbers of variables.

Another disadvantage of our method is that it is not guaranteed to find the global minimum of the function we are trying to optimize. Neither, however, do the traditional methods of regression, and indeed, such a goal would be impractical because of the number of parameters in the problem.

7 Further Research

Though the results of the computer simulations are suggestive, we would like to see more done in the area of practical epidemiology, perhaps a large-scale analysis of publicly available datasets. Such an undertaking is non-trivial, however, even given this previous research. Most publicly available datasets are enormous in comparison with the relatively small ones used in this study (several hundreds versus over fifteen thousand). Also, the number of possible choices for variables is immense. The investment of computer time would be substantial even for a modestly ambitious study. An even greater hurdle seems to be the numerical

stability of the algorithm itself. For datasets that are extremely large or that have a very wide range of data values, finding the least squares solution (which involves inverting a matrix) is inherently a numerically unstable undertaking. We have attempted programming the bulk of the algorithm in both C and Mathematica. In C, the numerical stability of the unusually large number of computations seems to be an issue; whereas, in Mathematica, even though the kernel controls the numerical precision well, the time necessary for even a relatively small computation is enormous.

References

- [1] M. Bowman and L. Ingber. Canonical momenta of nonlinear combat. In *Proceedings of the 1997 Simulation Multi-Conference, 6-10 April 1997, Atlanta, GA*, San Diego, CA, 1997. Also available at http://www.ingber.com/combat97_cmi.pdf.
- [2] M. Buszko, D. C. Wang, M. F. Kempka, E. Szczesniak, and E. R. Andrew. Application of adaptive simulated annealing to optimization of gradient coils with concentric return paths. University of Florida NMR Poster Session 1995 (<http://micro.ifas.ufl.edu/>), Gainesville, FL, 1995. Also available at <http://micro.ifas.ufl.edu/marian/txt/optim.html>.
- [3] B. Cohen. Training synaptic delays in a recurrent neural network. Master's thesis, Tel-Aviv University, Tel-Aviv, Israel, 1994. Also available at <ftp://archive.cis.ohio-state.edu/pub/neuroprose/Thesis/cohen.thesis.ps.gz>.
- [4] R.A. Cozzio-Bueler. *The design of neural networks using a priori knowledge*. PhD thesis, Swiss Federal Institute of Technology, Zurich, Switzerland, 1995. Also available at <ftp://archive.cis.ohio-state.edu/pub/neuroprose/Thesis/cozzio.thesis.ps.gz>.
- [5] M. C. Forman. *Compression of Integral Three Dimensional Television Pictures*. PhD thesis, De Montfort University, Leicester, United Kingdom, 2000.
- [6] M. C. Forman, A. Aggoun, and M. McCormick. Simulated annealing for optimisation and characterisation of quantisation parameters in integral 3D image compression. In J. M. Blackledge and M. J. Turner, editors, *Image Processing II: Mathematical Methods, Algorithms and Applications*, pages 399–413. The Institute of Mathematics and its Applications, Horwood Publishing, Berlin, 2000. Also available at <http://www.imtech.cse.dmu.ac.uk/3d-med/pubs/ima98.pdf>.
- [7] G. Indiveri, G. Nateri, L. Raffo, and D. Caviglia. A neural network architecture for defect detection through magnetic inspection. Technical report, University of Genova, Genova, Italy, 1993. Also available at <ftp://archive.cis.ohio-state.edu/pub/neuroprose/indiveri.nn.defect.detect.ps.gz>.
- [8] L. Ingber and R. P. Mondescu. Optimization of trading physics models of markets. *IEEE Trans. Neural Networks*, 12(4):776–790, 2001. Also available at http://www.ingber.com/markets01_optim_trading.pdf.
- [9] L. Ingber and B. Rosen. Genetic algorithms and very fast simulated reannealing: A comparison. *Mathematical Computer Modelling*, 16(11):87–100, 1992. Also available at http://www.ingber.com/asa92_saga.pdf.
- [10] L. Ingber, M. F. Wehner, G. M. Jabbour, and T. M. Barnhill. Application of statistical mechanics methodology to term-structure bond-pricing models. *Mathematical Computer Modelling*, 15(11):77–98, 1991. Also available at http://www.ingber.com/markets91_interest.pdf.

- [11] L. Ingber and J. K. Wilson. Statistical mechanics of financial markets: Exponential modifications to Black-Scholes. *Mathematical Computer Modelling*, 31(8/9):167–192, 2000. Also available at http://www.ingber.com/markets00_exp.pdf.
- [12] Lester Ingber. Very fast simulated re-annealing. *Mathematical Computer Modelling*, 12(8):967–973, 1989. Also available at http://www.ingber.com/asa89_vfsr.pdf.
- [13] Lester Ingber. Statistical mechanical aids to calculating term structure models. *Physical Review A*, 42(12):7057–7064, 1990. Also available at http://www.ingber.com/markets90_interest.pdf.
- [14] Lester Ingber. Statistical mechanics of neocortical interactions: A scaling paradigm applied to electroencephalography. *Physical Review A*, 44(6):4017–4060, 1991. Also available at http://www.ingber.com/smni91_eeg.pdf.
- [15] Lester Ingber. Generic mesoscopic neural networks based on statistical mechanics of neocortical interactions. *Physical Review A*, 45(4):R2183–R2186, 1992. Also available at http://www.ingber.com/smni92_mnn.pdf.
- [16] Lester Ingber. Adaptive simulated annealing (ASA) global optimization C-code. <http://www.ingber.com/#ASA-CODE>, 1993.
- [17] Lester Ingber. Simulated annealing: Practice versus theory. *Mathematical Computer Modelling*, 18(11):29–57, 1993. Also available at http://www.ingber.com/asa93_sapvt.pdf.
- [18] Lester Ingber. Statistical mechanics of combat and extensions. In C. Jones, editor, *Toward a Science of Command, Control, and Communications*, pages 117–149. American Institute of Aeronautics and Astronautics, Washington, D.C., 1993. ISBN 1-56347-068-3. Also available at http://www.ingber.com/combat93_c3sci.pdf.
- [19] Lester Ingber. Adaptive simulated annealing (ASA): Lessons learned. *Control and Cybernetics*, 25(1):33–54, 1996. Also available at http://www.ingber.com/asa96_lessons.pdf.
- [20] Lester Ingber. Statistical mechanics of nonlinear nonequilibrium financial markets: Applications to optimized trading. *Mathematical Computer Modelling*, 33(7):101–121, 1996. Also available at http://www.ingber.com/markets96_trading.pdf.
- [21] Lester Ingber. Statistical mechanics of neocortical interactions: Canonical momenta indicators of electroencephalography. *Physical Review E*, 55(4):4578–4593, 1997. Also available at http://www.ingber.com/smni97_cmi.pdf.
- [22] Lester Ingber. Adaptive simulated annealing (ASA) and path-integral (PATHINT) algorithms: Generic tools for complex systems. Invited talk at the University of Calgary, Canada, April 2001. Lecture plates available at http://www.ingber.com/asa01_lecture.pdf.
- [23] M. Jerrum and G. Sorkin. Simulated annealing for graph bisection. In *Proc. 34th Ann. Symp. on Foundations of Comp. Sci.*, 1993.
- [24] S. Kirkpatrick, Jr. C. Gelatt, and M. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):498–516, May 1983.
- [25] D. G. Mayer, J. A. Belward, and K. Burrage. Use of advanced techniques to optimize a multi-dimensional dairy model. *Agricultural Systems*, 50:239–253, 1996.

- [26] D. G. Mayer, J. A. Belward, K. Burrage, and M. A. Stuart. Optimization of a dairy farm model - comparison of simulated annealing, simulated quenching and genetic algorithms. In *Proceedings, 1995 International Congress on Modelling and Simulation, 27-30 November 1995, University of Newcastle*, volume 50, pages 33–38, 1995.
- [27] D. Mitra, F. Romeo, and A. Sangiovanni-Vincentelli. Convergence and finite-time behavior of simulated annealing. *Advances in Applied Probability*, 18:747–771, 1986.
- [28] Raymond H. Myers. *Classical and Modern Regression with Applications*. Duxbury Press, Boston, second edition, 1990.
- [29] S. Sakata. *High breakdown point estimation in econometrics*. PhD thesis, University of California at San Diego, La Jolla, CA, 1995.
- [30] S. Sakata and H. White. High breakdown point conditional dispersion estimation with application to S&P 500 daily returns volatility. *Econometrica*, 66:529–567, 1998.
- [31] M. K. Sen and P. L. Stoffa. *Global Optimization Methods in Geophysical Inversion*. Elsevier, The Netherlands, 1995. ISBN 0-444-81767-0.
- [32] G. Sorkin. Efficient simulated annealing on fractal energy landscapes. *Algorithmica*, 6:367–418, 1991.
- [33] R. E. Walpole, R. H. Myers, S. L. Myers, and K. Ye. *Probability and Statistics for Engineers and Scientists*. Prentice Hall, Inc., Upper Saddle River, New Jersey, seventh edition, 2002.
- [34] M. Wofsey. Technology: Shortcut tests validity of complicated formulas. *The Wall Street Journal*, 24 September 1993. Page B1.