1.1 ON GENETIC ALGORITHMS AND DISCRETE PERFORMANCE MEASURES

Caren Marzban*
Center for Analysis and Prediction of Storms
University of Oklahoma, Norman, OK
Department of Statistics, University of Washington, Seattle, WA
Applied Physics Laboratory, University of Washington, Seattle, WA

Sue Ellen Haupt
Applied Research Laboratory
The Pennsylvania State University, State College, PA

Abstract

A relation exists between the manner in which a statistical model is developed and the measure employed for gauging its performance. Often the model is developed by optimizing some continuous measure of performance, while its final performance is assessed in terms of some discrete measure. The question then arises as to whether a model based on the direct optimization of the discrete measure may be superior to or significantly different from the model based on the optimization of continuous measure. Some Artificial Intelligence parameter estimation techniques allow the optimization of discrete measures. Genetic Algorithms constitute one such technique, and therefore, allow for an examination of this question. Here, one type of genetic algorithm is employed to optimize three discrete performance measures of a parametric model for the prediction of hail. A more conventional technique is then employed to optimize the same discrete measures. The former outperforms the latter. In other words, the direct optimization of three discrete measures via genetic algorithms yields better fits to the data than alternatives requiring the intermediate step of optimizing a continuous measure.

Corresponding author address: Caren Marzban, National Research Center for Statistics and the Environment Dept. of Statistics, University of Washington Box 354323 Seattle, WA 98195-4323 e-mail: marzban@caps.ou.edu

1. INTRODUCTION

There is a relationship between how a statistical model is developed and the measure used for gauging its performance. This is due to the fact that the estimation of the parameters of a statistical model almost always involves the optimization of some quantity, such as likelihood or mean squared error. Often the model is developed by optimizing one measure of performance, while its final performance is assessed in terms of another measure. For example, in meteorology a model is often developed by first minimizing a ``continuous" measure such as mean squared error, and then its final performance is gauged in terms of some ``discrete" measure such as the critical success index. The question then arises as to whether a model based on the direct optimization of the discrete index may be superior to or significantly different from the model based on the optimization of mean squared error. This question is difficult to address, because most optimization methods require the measure of performance to be continuous and differentiable in the parameters (e.g., mean squared error), while many performance measures common in meteorology (e.g., the critical success index) do not satisfy these conditions. However, some Artificial Intelligence parameter estimation techniques do not require either of these constraints. Genetic Algorithms (GAs) constitute one such technique, and therefore, allow for an examination of this question. Here, one type of genetic algorithm is employed to optimize three discrete performance measures of a parametric model for the prediction of hail. A more conventional

technique is then employed to optimize the same discrete measures. The former outperforms the latter. In other words, the direct optimization of three discrete measures via genetic algorithms yields better fits to the data than alternatives requiring the intermediate step of optimizing a continuous measure.

A parametric statistical model consists of two specifications: 1) The precise parametric form of the model, and 2) the distribution of the errors. For example, a simple linear regression model relating *N* observations on two variables, x_i and y_i, $i = 1, 2, \cdots, N$, can be written as

$$y_i = \alpha x_i + \beta + \varepsilon_i \quad (1)$$

where α and β are parameters that must be estimated from the data on *x* and *y*. However, the estimated values depend on the distribution of the errors, ε_i. Ordinarily, one seeks the most likely values of the parameters given the data. These estimates are called the maximum likelihood estimates. It can be shown that the maximum likelihood estimates are equivalent to those arrived at by the minimization of the mean square error (MSE)

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\varepsilon_i^2 = \frac{1}{N}\sum_{i=1}^{N}(y_i - \alpha x_i - \beta)^2 \quad (2)$$

with respect to α and β, if the distribution of ε is Gaussian (Draper and Smith 1981). It is this fact - that the resulting parameter estimates are maximum likelihood estimates - which underlies the ubiquitous use of MSE in a wide range of problems.

The connection between maximum likelihood estimates and the mean square error is one of many such relations between the choice of model parameters and the choice of the error function. As a result, it has become common practice in model building to simply optimize some error function without consideration of the likelihood of the estimated parameters. This practice has been historically successful in that models developed by the optimization of some error function (usually MSE) often perform well on independent data.

Another common practice is to use an error function that is continuous and differentiable in the parameters. The reason for this practice is that most traditional optimization techniques (or training algorithms) rely on the computation of the gradient of the error function. Examples include Gradient decent, Newton's method, Conjugate Gradient, Levenberg-Marquardt (Press et al. 1999).

Generally, many regression models minimize MSE, and many classification models are based on the minimization of cross-entropy (Bishop 1986). This is because the former assume that the predictand is unbounded (with Gaussian errors), while the latter assume that the predictand is a probability (of belonging to a class). As mentioned previously, both of these measures are continuous and differentiable in model parameters. However, most common measures of classification performance are based on the contingency table, rather than MSE or cross-entropy. The critical success index is one example which is common in meteorology; numerous others are discussed by Marzban (1998). Because, they are based on a discrete table - the contingency table – they are neither continuous nor differentiable in the model parameters. This is why they are referred to as discrete measures. Given that the final model is to be assessed in terms of such discrete measures, it is natural to optimize them directly.

Recent advances in Artificial Intelligence have led to numerous optimization techniques which do not require continuity or differentiability of the error function. This allows for the possibility of directly optimizing discrete measures of performance based on the contingency table. One family of such techniques is referred to as Genetic Algorithms (Holland 1975; Haupt and Haupt 1998, 2004).

In this article, a number of such measures are optimized directly using GAs, and the results are compared to the more traditional approach of first minimizing cross entropy, followed by the optimization of a discrete performance measure. One may interpret this task as equivalent to the comparison of two parameter estimation (or training) algorithms - one capable of optimizing discrete measures, the other not. It should be noted that, strictly

speaking, the latter is not an optimization algorithm because it is a 2-stage procedure with different measures optimized at each stage.

The next section will delve further into the methodology of the study. It is followed by results, and a discussion thereof. It will be shown that direct optimization with a genetic algorithm yields a model that outperforms models based on the 2-stage procedure, where performance is gauged in terms of three discrete measures - the fraction correct, the critical success index, and the Heidke Skill Statistic.

2. METHODOLOGY

This study is inherently empirical in that the findings are specific to the data set examined. This is true of most studies dealing with performance measures since one cannot separate properties of different performance measures from properties of data. The data set consists of over 21,000 cases with every case including three predictors and one predictand. The predictors are related to both Doppler radar-derived parameters and parameters representing the near-storm environment. The predictand is a binary number labeling the occurrence or nonoccurrence of hail. This data set has been used in the development of a neural networks to aid the National Severe Storms' Hail Detection Algorithm in detecting hail and estimating hail size (Marzban and Witt 2000, 2001). The neural network for predicting the occurrence of hail was trained by minimizing cross-entropy, and that for predicting the size of hail was based on the minimization of MSE. As mentioned above, both of these measures are continuous and differentiable. Given that the latter network is a classifier, its performance was assessed in terms of discrete measures - specifically, the Critical Success Index (CSI) and the Heidke Skill Statistic (HSS). As such, its development was a 2-stage process involving the maximization of cross-entropy followed by the maximization of the discrete measures. The discrete measures were computed from the contingency table, which in turn was formed by placing a threshold on the probability (of hail) produced by the network. As such, the maximization of the discrete measure (at the second stage) is tantamount to identifying the threshold which yields that highest performance. In addition to CSI and HSS, one other measure will be employed here - The fraction correct (FRC).

The parametric form of the model examined here is motivated by the above mentioned neural network. Specifically, it is

$$y = g\left(\sum_{i=1}^{H} \omega_i f\left(\sum_{j=1}^{N_{in}} \omega_{ij} x_j - \theta_j \right) - \omega \right) + \varepsilon \quad (3)$$

where the ω's and θ's are all parameters to be estimated from data. It is the estimates of these parameters (analogous to those in equation 1) which is the subject of this study. For clarity, the index *i=1,2,...,N,* referring to the data case is not shown. N_{in} refers to the number of predictors, and *H* is a parameter that gauges the nonlinearity of the function. Note that *H* is analogous to the order of a polynomial. Although there are techniques for estimating it from data, the optimal value for *H* is not important in the current study. It will be fixed at *H=2*, and *4*. Recall that the goal of the study is to examine different training algorithms, and not to develop the ``best'' model for hail detection.

On a related note, recall that model building in a nonlinear setting often calls for at least two data sets, called training and validation sets. The combination of the two data sets is employed to estimate all the ω and θ parameters as well as *H*. Such approaches are called resampling techniques, because they involve drawing many different training and validation sets from the given data set (Bishop 1996). The aim of the approach is to preclude overfitting the data. However, again, in the current study, overfitting is not of concern, for we are not attempting to develop the best model in the sense of one that performs best on independent data. The task is to compare two different training procedures for the purpose of identifying the one that yields the lowest fit error.

In short, the two procedures under comparison are defined as follows:

A) Minimize cross-entropy to build a model producing a continuous predictand (i.e. probability). Place a threshold on the predictand in order to construct a contingency table, and compute performance measures. Vary the threshold across the full range of the predictand in order to identify the threshold at which the maximum performance measure occurs. This maximum value is taken to represent optimal performance in this approach.

B) Maximize the discrete measure, directly, by employing a genetic algorithm.

2.1 Conjugate Gradient and the Genetic Algorithms

As mentioned above, the two models under comparison are essentially based on two different training algorithms. The minimization cross-entropy is performed by Conjugate Gradient (Press et al 1999). Details of this method are unimportant; suffice it to say that it is a gradient-based method. In fact, any of the above-mentioned gradient-based method could have been employed here. Conjugate Gradient (CG) is simply one of the better ones in terms of speed and the ability to avoid local minima.

Genetic algorithms (GAs) are an artificial intelligence technique based on the biological concepts of genetic combination and natural selection. Parameters to be optimized (known as genes) are concatenated into data strings called chromosomes. The algorithm begins with a population of randomly generated chromosomes. These chromosomes undergo the operations of mating and mutation. Mating combines the information from two parent chromosomes to produce new individuals, exploiting the best of the current generation, while mutation, or randomly changing some of the parameters allows exploration into other regions of the solution space. Natural selection via a problem specific cost function assures that only the most fit chromosomes remain in the population to mate and produce the next generation. Upon iteration, the GA converges to a global solution.

Some of the advantages of GAs include that they can be used to find discrete or continuous parameters, no differentiation is necessary, they can simultaneously search different regions of the solution space, they work well with data from physical or numerical experiments, they do not get stuck in local minima, there is no need for a good first guess, and they work well on parallel computers. They are not known to be particularly fast when used on parabolic problems for which more traditional techniques are formulated; however, they are remarkably robust at finding solutions in highly complex solution spaces.

The GA used for this study is a continuous parameter GA. The parameters, or genes, are coded as real numbers. More details on the GA and the operators can be found in Haupt and Haupt (2004).

One common feature of both conjugate gradient and genetic algorithm is that they are iterative. One typically begins with a random set of values for the parameters, and applies the training algorithm until the performance measure converges to some optimal value. To assure that the training algorithms are not trapped in shallow local minima, both CG and GAs are applied to five different initial parameter values.

In summary, a neural network is trained by two different training algorithms - conjugate gradient and genetic algorithm. The former minimizes cross-entropy, which means that the network produces a continuous quantity between 0 and 1. This output is then dichotomized by the introduction of a threshold, and the discrete measure is optimized as a function of this threshold. The latter training algorithm maximizes the discrete measure directly. It also directly optimizes the threshold as part of the calculation. Finally, the two algorithms are compared in terms of the discrete measure.

2.1 The Performance Measures

In the Bayesian approach to forecasting, performance measures are necessary to translate the dependence of posterior probabilities on prior probablilities. We write the measure in terms of the rates of correct

prediction of events and of false alarms. The measures are then defined in terms of a contingency table matrix

$$\mathrm{C} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \tag{4}$$

where *a* and *d* are the correct forecasts of nonevents and events, respectively, and *b* and *c* are the number of false alarms and misses, respectively. If we call the total number of nonevents (no hail occurred) N_0 and events (hail) N_1, then $N_0 = a + b$ and $N_1 = c + d$.

We define three separate measures for comparison below:

1. Fraction correct (FRC):

$$FRC = \frac{a+d}{N_0 + N_1} \tag{5}$$

2. Critical Success index (CSI):

$$CSI = \frac{d}{b + N_1} \tag{6}$$

3. Heidke's Skill Statistic (HSS):

$$HSS = \frac{2(ad+bc)}{N_0(b+d) + N_1(a+c)} \tag{7}$$

These measures are widely used as discussed by Marzban (1998).

3. RESULTS

Figures 1a through 1f (end of paper) display the results. The horizontal lines correspond to the performance measures reached by the GA for 5 different initializations. They are not a function of threshold since the threshold is optimized directly by this approach. The remaining 5 curves correspond to the performance measures based on the minimization of cross-entropy by CG. It can be seen that the curves for all 3 measures display similar behavior, although they reach their peaks at somewhat different values of the threshold. The behavior of these curves is in complete agreement with their theoretical behavior in Gaussian models (Marzban 1988).

The important point of these figures is that the 2-stage optimization of the measures does not yield performance values as high as those obtained in their direct optimization. This is true of all five curves. As such, the GA has a (slight) advantage over the alternatives that require continuous and differentiable performance measures.

The difference between the two approaches, however, is rather small in that the curves approach and even cross some of the horizontal lines. The question arises as to whether the difference is statistically significant. To that end, a t-test is performed and 2σ confidence intervals are computed. Note that 2σ corresponds the a 97% confidence interval. The latter are displayed in Table 1. The t-values (not shown in table) for the *H=2* case are in the 2.8 range, and those of the *H=4* case are in the 2.4 range. It can be seen that the differences between the two approaches are statistically significant. The slightly lower t-value of the *H=4* case is anticipated from the larger variations between the 5 curves in the right figures in Fig. 1.

Table 1. The average performance values and confidence intervals, for the different measures, and for H=2 and H=4.

H = 2

Measure	GA	CG
FRC	0.92163±0.00019	0.92130±0.00006
CSI	0.50370±0.00105	0.49970±0.00037
HSS	0.62076±0.00266	0.61768±0.00038

H = 4

Measure	GA	CG
FRC	0.92157±0.00022	0.92109±0.00018
CSI	0.50360±0.00147	0.50208±0.00146
HSS	0.62192±0.00158	0.61958±0.00106

3. SUMMARY AND DISCUSSION

This study has demonstrated that when a neural network is trained directly using the performance measure that will be used to judge its success, it is somewhat more skillful

than if trained using the traditional mean square error approach. To train the network using the discrete performance measures, however, requires use of an artificial intelligence technique that can work with discrete numbers. In this case we used a genetic algorithm and were able to demonstrate a reasonable level of success. We should note that the genetic algorithm does take considerably more CPU time to complete the optimization of the neural network weights than the competing methodology.

The results reported here are demonstrated with only a single data set; therefore, they should be considered preliminary until they can be confirmed with other data sets of differing types and sizes. We do expect that with more experiments we will be able to generalize these results to other cases.

ACKNOWLEGEMENTS – Part of this work was supported by in house funds from the director of the PSU Applied Research Laboratory. We thank Randy Haupt who is coauthor of the genetic algorithm used for this study.

REFERENCES

Bishop, C. M., 1996: *Neural networks for pattern recognition.* Clarendon Press, Oxford, pp. 482.

Draper, N. R., and H. Smith, 1981: Applied Regression Analysis, John Wiley \& Sons. pp. 706.

Haupt, R.L. and S.E. Haupt, 2004, *Practical Genetic Algorithms, 2nd Edition with CD*, John Wiley & Sons, New York, NY, 255 pp.

Haupt, R.L. and S.E. Haupt, 1998, *Practical Genetic Algorithms,* John Wiley & Sons, New York, NY, 177 pp.

Hollard, J.H., 1975: *Adaptation in Natural and Artificial Systems*, Ann Arbor, The University of Michigan Press.

Marzban, C., and A. Witt, 2001: A Bayesian neural network for hail size prediction. *Wea. Forecasting*, **16**, 5, 600-610.

Marzban, C., and A. Witt, 2000: *Bayesian neural networks for severe hail prediction. A report.* Available at http://www.nhn.ou.edu/\~{}marzban/hda_class.pdf .

Marzban, C. 1998: Bayesian probability and scalar performance measures in Gaussian models. *Jour. Appl. Meteo*, **37**, 72-82.

Press, W. H., S. A. Teukolsky, W. T. Vetterling, B. P. Flannery 1999: *Numerical Recipes in C*. Cambridge University Press, 994 pp.

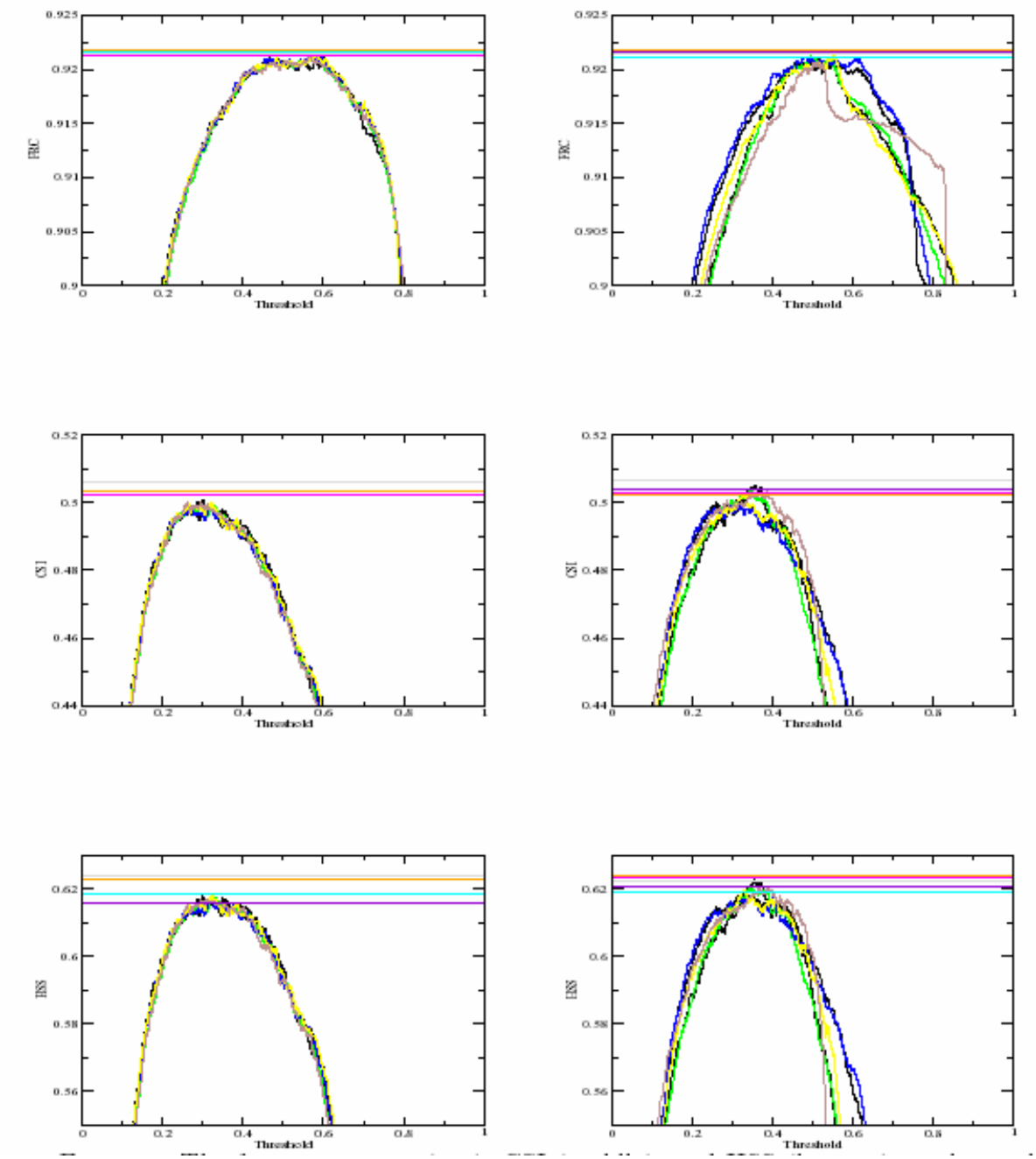

Figure 1. The fraction correct (top), CSI (middle), and HSS (bottom) as obtained from five different initializations of conjugate gradient with *H=2* (left) and *H=4* (right). The horizontal lines are the corresponding scores from GA.