From: KDD-98 Proceedings. Copyright © 1998, AAAI (www.aaai.org). All rights reserved.

Direct Marketing Response Models using Genetic Algorithms

Siddhartha Bhattacharyya
Information and Decision Sciences Department
College of Business Administration,
University of Illinois at Chicago
601 S. Morgan Street (M/C 294), Chicago, IL 60607
sidb@uic.edu

Abstract

Direct marketing response models seek to identify individuals most likely to respond to marketing solicitations. Such models are commonly evaluated on classification accuracy and some measure of fit-to-data. Given large customer files and budgetary limitations, only a fraction of the total file is typically selected for mailing promotional material. This desired mailing-depth presents potentially useful information that is not considered by conventional methods. This paper presents a genetic algorithm based approach for developing response models aimed at maximizing performance at the desired mailing depth. Here, depth of file information is explicitly taken into account during model development. Two modeling objectives, response maximization at selected mailing depth and fit-to-data, are considered and tradeoffs amongst these empirically explored. Resampling approaches, effective for controlling overfit to training data, are also investigated.
Keywords: Genetic Algorithms, Response Models, Database Marketing, Data Mining, Resampling.

Introduction

Data analysts in direct marketing (DM) seek techniques that maximize response to direct-mailing solicitations. The identification of target audiences for specific marketing promotions involves detailed analyses of the customer database to seek out individuals most likely to respond. A key task here is the development of models to identify the most promising individuals to mail to. Models are defined over attributes characterizing potential responders to marketing promotions. Developed models are used to score individuals in a customer file such that higher scores indicate greater mailing preference. Individuals are then ranked by their model-obtained scores, and the final mailing list determined through budgetary considerations.

Direct marketers typically have to contend with limited budgets. Models are thus used to mail to a fraction of individuals in the customer file. For instance, out of a total customer file of a million individuals, resources might permit mailing only to, say, 20% of them. Obviously, the most promising 200,000 individuals, as indicated by the model, will be mailed to. This mailing *depth-of-file* provides potentially useful information that should be considered in model determination.

Response models are built from data on prior purchases that identify individuals as either responders or non-responders. Models are typically evaluated on their prediction accuracy on unseen "test" data, and some measure of fit-to-data. Model performance assessment, however, should also consider how the model will be implemented. Traditional accuracy estimates are noted to be inadequate for maximizing business payoffs (Massand and Piatetsky-Shapiro 1996). Given that DM models are used to identify a subset of total customers expected to maximize response to a mailing solicitation, model performance is assessed at different mailing depths. Mailing decisions are then undertaken considering mailing costs and expected returns at different depths-of-file.

A decile analysis is typically used to examine model performance. Individuals are ranked in descending order of their respective model scores – higher scores indicating better performance – and separated into 10 equal groups. Table 1 shows a typical decile analysis. The first row, or top decile, indicates performance for the best 10% of individuals as identified by the model. The Cumulative Lifts at specific depths of file provide a measure of improvement over a random mailing, and is calculated as:

$$\text{Cumulative Lift}_d = \frac{\text{cumulative average profit}_d}{\text{overall average profit}_d} * 100$$

for a depth-of-file d. Thus, in Table 1, a cumulative lift of 346 in the top decile indicates that the model in question is expected to provide a mailing response that is 3.46 times the response expected from a random mailing (no model) to 10% of the file. The cumulative lift at the bottom decile is always 100 and corresponds to a mailing to the entire file. An ideal model should exhibit "smoothly" decreasing performance from the top through bottom -- this is not true in the table shown, and can be considered indicative of model lack of fit to the data.

The modeling approach taken in this paper seeks to determine models with explicit consideration of the mailing depth of interest. In this decile-maximizing (DMAX) approach, the depth-of -file provides an input parameter to the model search process. One can thus obtain models tailored to the specific mailing depth of interest. Thus, if resource limitations allow mailing to, say, only 20% of the total customer file or database, a DMAX model can be obtained with the usually chosen predictor variables *and* knowledge that only 20% of the database will be mailed to.

Such a model will then seek to insure that the total expected response, amongst the 20% of individuals identified by the model, is maximized. Models optimized for different depths-of-file also provide decision-makers a useful view of expected performance at different deciles, thereby aiding in the selection of a mailing depth with desired tradeoffs.

A genetic algorithm based modeling approach is used. The fitness estimation of population members, a crucial aspect of genetic search, is based on two criteria. While the first criterion focuses on maximizing performance at the desired depth-of-file, the second criterion seeks model fit to data. Experiments with a real-life data set examine tradeoffs between these two potentially conflicting objectives. Overfit to the training data and consequent shrinkage in performance from the training to the test data is common in application of machine learning techniques. While the model fit-to-data criteria contributes to robustness, the use of resampling (Efron and Gong 1983) within the genetic learning process is investigated to further control such shrinkage.

The following section discusses the use of genetic algorithms for developing response models, including the formulation of fitness functions, and use of resampling. Experimental results are then provided in Section 3, followed by a discussion of future research issues.

Genetic Algorithms

Genetic algorithms provide a stochastic search procedure based on principles of natural genetics and survival of the fittest. They operate through a simulated evolution process on a population of string structures representing candidate solutions in the search space. Evolution occurs through (1) a selection mechanism that implements a survival of the fittest strategy, and (2) genetic recombination of the selected strings to produce offspring for the next generation. GAs are considered suitable for application to complex search spaces not easily amenable to traditional techniques, and are noted to provide an effective tradeoff between exploitation of currently known solutions and a robust exploration of the entire search space. The selection scheme operationalizes exploitation and recombination effects exploration. Goldberg (1989) provides a thorough account of the mechanics of genetic search.

Model Representation

Each population member, in the present problem context, can specify a model expressed in symbolic rule form (DeJong, Spears and Gordon 1993) or a weight vector on the predictor variables (Koehler 1991). This paper takes the latter approach, with models expressing a linear combination of attributes. Given their easier interpretation and higher predictive reliability, such linear models are often preferred for decision-making (Hand 1991). The use of linear models also allows a direct comparison with traditional statistical techniques commonly use in DM, and helps focus attention on the decile-maximizing approach. Models, here, thus specify a vector of weights $\mathbf{w}$ corresponding to the attributes; each population member represents such a vector. We restrict $\mathbf{w} \in [-1,1]$, since a weight vector $\mathbf{w}$ scores and ranks individuals similarly to $\lambda\,\mathbf{w}$ for any $\lambda \in \Re$ ($\Re$ denotes real numbers). Since the dependent variable seeks to model response likelihood, the model scores are filtered using a logistic function: $\hat{y} = 1/(1+\exp(-\mathbf{wx}))$, yielding values in [0,1]. A direct real number representation is used in the genetic search (Davis, 1991; Michalewicz 1994).

Genetic search operators

Standard fitness-proportionate selection (Goldberg 1989) is used, and an elitist selection strategy is implemented where a certain fraction of good solutions is retained intact into the next generation. Crossover and mutation are the two basic recombination operators. Given the real-string representation, standard operators reported in the literature are used -- arithmetic and exchange crossover, and uniform mutation, together with two operators fostering finer local search: heuristic crossover and non-uniform mutation (Michalewicz 1994).

Fitness criteria

The fitness function, embodying modeling objectives, guides the genetic search. Two evaluation criteria are used in estimating the fitness of each population member, and the fitness function is designed to allow a tradeoff amongst these possibly conflicting objectives. The first criterion pertains to decile performance maximization, while the second seeks to enhance a solution's fit to the data.

The decile performance maximization objective is modeled by the total number of responders provided by a model at the specified file-depth. Model fit to data is estimated using the Hosmer-Lemeshow goodness of fit measure (Hosmer and Lemeshow, 1989), which calculates the Pearson Chi-square statistic based on a grouping of observations as found in the decile analysis. This is defined as:

$$C = \sum_{d=1}^{10} \frac{(r_d - n_d \bar{\pi}_d)}{n_d \bar{\pi}_d (1 - \bar{\pi}_d)},$$

where d is the decile index, r_d and n_d give the number of responders and total number of observations respectively in the d-th decile; the average estimated response probability for the d-th decile is

$$\bar{\pi}_d = (\sum_{j=1}^{n_d} \hat{y}_j) / n_d.$$

The C values are used here to compare the desirability of models based on goodness-of-fit. While other measures can also be used, the Hosmer-Lemeshow statistic is well suited for our purpose of assessing fit based on the aggregated information in the deciles.

The fitness of a solution is then determined as the weighted average:

$$f = W_1 \sum_{i=1}^{N_d} (\hat{y}^s)_i + W_2 C$$

where N_d = (Nd) gives the number of observations in the top d deciles out of a total of N observations, and $\hat{y}^s$ denotes the requirement that the $\hat{y}$-values be sorted in descending order prior to summation. The parameters W_1 and W_2 determine the relative importance assigned to the two criteria, and can be used to effect varied tradeoff amongst the two criteria.

Resampling

Resampling techniques, utilized mainly as a bias reduction tool in the estimation of error rates in classifiers (Efron and Gong, 1983), offer potential advantages for GA-based learning of robust models. Here, the fitness of population members is estimated from fitness values obtained from multiple sub-samples of the training data. This biases the search towards solutions that perform uniformly well across the different sub-samples; solutions exhibiting a high performance, but with large variation across the sub-samples, do not survive the selective pressure effected by this bias. Sub-samples for training may be generated by repeated sampling with replacement -- bootstrap -- or by considering mutually exclusive partitions on the data. The latter approach was found to lead to varying cumulative-lift values on the different sub-samples; this arises from the low response rates typical in DM data, leading to uneven splits of responders amongst the different sub-samples. Bootstrapped sub-samples were thus used.

Sub-samples may be used in different ways in estimating the fitness of population members: (a) member-wise, where a different set of samples is generated for each population member to be evaluated; (b) generation-wise, where the same sub-sample set is used for evaluating population members in a generation, but with different samples used in different generations; or (c) run-wise, where a fixed set of sub-samples is used for evaluating all population members across the GA-run. The effectiveness of a resampling scheme for use with genetic search depends on two desirable characteristics. First, it is desirable that competing members of the same population be evaluated on an equitable basis. Secondly, good solutions should also display robustness in performance across a range of sub-samples. While the member-wise sub-sampling scheme does not satisfy the first criterion, the run-wise resampling approach does not adequately meet the second condition. Generation-wise resampling satisfies both the properties and is thus the method of choice here. An elitist selection scheme that retains the best solutions from the previous 10 generations helps ensure that solutions with somewhat lower performance on one sub-sample set but with greater robustness across different sub-samples, survive the GA selection process. Hillis (1992) presents another approach to using resampling in the context of overfit with GAs.

Experimental Results

A series of experiments help determine the effectiveness of the designed GA based scheme in obtained models tailored to different file-depths. Separate training and holdout (test) data sets were maintained, and cumulative lifts on the test data provide the performance measures of interest; models exhibiting robust performance across the training and test data are also sought. A sole focus on the decile performance maximization objective was found to yield over-optimistically high performance on the training data, with large shrinkage on the unseen data. The second fitness objective, model-fit to data, was thus examined for reducing such shrinkage in performance. Experiments analyze performance across specified decile-levels with varying importance assigned to the two evaluation criteria. A second set of experiments investigates potential performance enhancements through the use of resampling.

A real-life data set with a total of 20,160 observations was separated into training and test sets of 10,098 and 10,062 cases. The data, after the usual variable transformation and reduction (Tukey 1977), contained 11 attributes. All attributes were normalized to zero mean and unit standard deviation prior to application of the GA procedures. The GA parameters were set at values found to yield overall good performance in preliminary experiments, and in accordance with usage reported in the literature (Michalewicz 1994). The same set of values was used across the different experimental conditions. A population size of 50 was used, and learning was terminated after 100 generations.

Experimental results in Tables 2-5 below present the cumulative lift aggregated over 10 independent runs of the GA, each run being conducted with a different random number seed. A set of common random numbers was used across different experimental conditions to provide a more equitable performance comparison. Performance of a logit model, widely in DM industry use for such problems, at different deciles is also shown for a comparison baseline. Varying levels of tradeoff between the decile-maximization and model-fit objectives were obtained by manipulating the parameters W_1 and W_2. These were set at values to learn models with roughly the following distribution of fitness values derived from the decile maximization (dmax) and model-fit criteria: 100%dmax, 75%dmax-25% fit, and 50%dmax-50%fit. The Tables below show the cumulative lift at the indicated depths-of-file obtained with DMAX models tailored for that file-depth. Values in brackets are the standard deviation. Table cells marked by an asterisk indicate a lack of significance (at the 0.05 level) amongst the respective models.

Results indicate that focussing on the dmax criterion leads to solutions that, in attempting to maximize response at the specified depth-of-file, also tend to overfit the training data with subsequent large shrinkage on test data performance. Such shrinkage is seen to be countered by assigning increasing importance to the model-fit criterion in the fitness function. At the 50%dmax-50%fit level, the

performance of models show little improvement over the baseline logit model. At the top decile, the difference in test data performance between the 100%dmax and 75%dmax-25%fit models is not found significant; given the larger shrinkage with the 100%dmax models, 75%dmax-25%fit can be considered to provide the best tradeoff; 50%dmax-50%fit shows lower shrinkage, but significantly lower test data performance. At the second decile too, the 75%dmax-25%fit tradeoff is seen to be most effective. At the third and seventh deciles, there is no significant difference in test data performance between the 75%dmax-50%fit and 50%dmax-50%fit models; given that shrinkage in performance is least with the 50%dmax-50%fit setting, this seems to effect the best tradeoff. Though the exact results here hold only for the specific data considered, a general higher focus on the model-fit objective seems desirable for models tailored for higher file-depths.

For resampling, fitness was estimated as the average of that over 30 bootstrapped samples from the training data. Given the computationally intensive nature of this operation, the results in Table 6 present the performance of the median model of 5 independent GA runs. Since the model-fit objective serves primarily to decrease shrinkage, and considering the same essential motive for incorporating resampling, fitness is estimated using only the dmax criterion. As anticipated, resampling is seen to yield substantial reductions in shrinkage, even without any model-fit objective. For models at the first, second and third deciles, resampling based models also exhibit higher performance than the best models obtained without resampling. While performance of the resampling based model for the first decile is substantially higher that of other 10%file-depth models (Table 2), improvements with resampling at the second and third decile levels is smaller. At the seventh decile, no improvement is seen over the baseline logit model's performance.

Table2: Performance in top decile (DMAX 10%)

Cum Lift	100%dmax 0%fit	75%dmax 25% fit	50%dmax 50% fit	Logit
Train data	453.4 (8.47)	423.8 (7.42)	417.2 (5.35)	407
Test data	352.3* (8.6)	346.0* (8.78)	330.1 (6.93)	328

Table3: Performance in second decile (DMAX 20%)

Cum Lift	100%dmax 0%fit	75%dmax 25% fit	50%dmax 50% fit	Logit
Train data	335.1 (9.3)	317.6 (8.97)	304 (5.35)	297
Test data	258.9* (8.65)	269.5 (8.79)	253.5* (7.96)	247

Table4: Performance in third decile (DMAX 30%)

Cum Lift	100%dmax 0%fit	75%dmax 25% fit	50%dmax 50% fit	Logit
Train data	269.9 (8.84)	252.5 (9.08)	243.1 (7.46)	237
Test data	212.6 (8.67)	230.2* (7.19)	225.1* (5.08)	207

Table5: Performance in seventh decile (DMAX 70%)

Cum Lift	100%dmax 0%fit	75%dmax 25% fit	50%dmax 50% fit	Logit
Train data	138.8 (6.06)	132 (3.62)	127.9 (2.68)	129
Test data	118.5 (6.06)	126.1* (3.78)	125.9* (3.9)	125

Table 6: Performance with Resampling

Cum Lift	First Decile	Second Decile	Third Decile	Seventh Decile
Train	409	293	240	125
Test	361	274	234	124

Discussion and Future Research

The proposed GA based approach provides decision-makers a technique for learning models that explicitly seek to maximize response at different desired mailing depths arising from resource constraints or other considerations. Results indicate superior performance obtained through this genetic learning facility over traditional logit models common in practice. While this study considers a decile analysis, other grouping schemes can also be used and the depth-of-file can be other than strict multiples of 10.

Overfit, resulting from the sole use of the decile maximization criterion in the fitness evaluation, is seen to be controllable using the model-fit criterion and obtaining a balance between the two. Tradeoffs amongst these two criteria effective for models developed for different file-depths are reported in the previous section. The exact tradeoff levels used and results reported, however, should be interpreted only as indicative of general expected behavior, and not viewed as values to be used across diverse data; the choice of optimal tradeoff will be data dependent. The presented analyses shows how varying performance with different DMAX models may be investigated. A simple resampling scheme is seen to foster the learning of robust models. Both with and without resampling, performance improvements over the baseline logit model is noticed to decrease with increasing file-depths. In the data considered, for example, no improvements were observed at the seventh decile level. This decreasing performance is to be expected, since greater file-depths imply larger proportions of individuals to be mailed to, leaving fewer "degrees of freedom", as it were, for the GA to manipulate.

A range of issues await further investigation. Noting that maximal advantage of the proposed decile maximization approach is obtained at the upper deciles, future research should evaluate alternative mailing strategies that seek to

take advantage of higher performance at lower mailing depths. Though this research has considered simple linear models, the proposed GA-based approach is equally applicable with more sophisticated representations. Genetic programming (Koza, 1993), with its parse tree representation of solutions, can obtain non-linear scoring models with potentially higher performance; rule-based representations capable of discerning complex patterns in the data can also be advantageous. The general modeling approach is, further, amenable to application of other learning techniques. A comparison of various techniques for maximizing decile performance will provide useful insights as to their relative benefits. Further research should also investigate the proposed decile-maximizing approach under different data conditions and characteristics. The use of simulated data here will allow greater control and evaluation under strictly known conditions, thus helping relate defining data characteristics with the technique.

Acknowledgements. The author would like to thank Bruce Ratner for introduction to data mining issues in direct marketing, and for providing the data used in the study.

References

David Shepard Associates. 1995. *The New Direct Marketing: How to Implement a Profit-Driven Database Marketing Strategy*, 2nd Edition, Irwin.

Davis, L. ed. 1991. Handbook of Genetic Algorithms. NY: Van Nostrand Reinhold.

DeJong, K., Spears, W.M. and Gordon, D.F. 1993. Using Genetic Algorithms for Concept Learning. *Machine Learning* 13: 161-188.

Efron B. and Gong, G. 1983. A Leisurely Look at the Bootstrap, the Jacknife, and Cross-Validation. *American Statistician* 37: 36-48.

Goldberg, D.E. 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*. Mass: Addison-Wesley.

Hand, D.J. 1981. *Discrimination and Classification*. NY: John Wiley and Sons.

Hillis, W.D. 1992. Co-evolving parasites improve simulated evolution as an optimization procedure. In C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen Eds., *Artificial Life II*, 313-324. Mass: Addison-Wesley.

Hosmer, D.W. and Lemeshow, S. 1989. *Applied Logistic Regression*. NY: John Wiley and Sons.

Koehler, G.J. 1991. Linear Discriminant Functions Determined through Genetic Search. *ORSA Journal on Computing* 3(4): 345-357.

Koza, J.R. 1993. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press.

Massand, B. and Piatetsky-Shapiro, G. 1996. A Comparison of Different Approaches for Maximizing the Business Payoffs of Prediction Models. In E. Simoudis, J. W. Han, and U. Fayyad Eds. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. 195-201.

Michalewicz, Z. 1994. *Genetic Algorithms + Data Structures = Evolution Programs*, 2nd Edition, Springer.

Matheus, C.J., Chan, P.K. and G. Piatetsky-Shapiro. 1993. Systems for Knowledge Discovery in Databases. *IEEE Transactions on Knowledge and Data Engineering* 5(6): 903-913.

Tukey, J.W. 1977. *Exploratory Data Analysis*, Mass: AddisonWesley.

Table 1: Sample Decile Analysis

Decile	Number of Customers	Number of Responses	Response Rate	Cumulative Responses	Cumulative Response Rate	Cumulative Response Lift
top	1006	207	20.58	207	20.58	346
2	1006	90	8.95	297	14.76	248
3	1006	60	5.96	367	11.83	199
4	1006	59	5.86	416	10.34	173
5	1006	32	3.18	448	8.90	150
6	1006	41	4.08	489	8.10	136
7	1006	32	3.18	521	7.40	125
8	1006	30	2.98	551	6.84	115
9	1006	21	2.09	572	6.32	106
bottom	1006	26	2.58	598	5.94	100
Total	10,060	598	5.94			

Copyright © 1998, American Association for Artificial Intelligence (www.aaai.org). All rights reserved.