A Comparison of Approaches For Maximizing Business Payoff of Prediction Models

Brij Masand and Gregory Piatetsky-Shapiro

GTE Laboratories,
40 Sylvan Rd. Waltham MA 02254, USA
email: brij@gte.com, gps@gte.com

Abstract

In many database marketing applications the goal is to predict the customer behavior based on their previous actions. A usual approach is to develop models which maximize accuracy on the training and test sets and then apply these models on the unseen data. We show that in order to maximize business payoffs, accuracy optimization is insufficient by itself, and explore different strategies to take the customer value into account. We propose a framework for comparing payoffs of different models and use it to compare a number of different approaches for selecting the most valuable subset of customers. For the two datasets that we consider, we find that explicit use of value information during the training process and stratified modelling based on value both perform better than post processing strategies.

1 Introduction

The rapidly growing business databases contain much potentially valuable knowledge that could be extracted by Data Mining and Knowledge Discovery Techniques (Piatetsky-Shapiro and Frawley 1991, Fayyad et al. 1996). One of the most widespread applications of data mining is targeted database marketing, which is the use of historical customer records to predict customer behavior.

In our application, historical customer records are used to group the customers into two classes -- those who respond to special offers and those who don't. Using past information collected over several months on usage of telephone services and responses to past offers, our task is to build a model for predicting the customer class in the next month and apply it to several hundred thousand customers. The prediction model is used to rank the customers according to their likelihood of response. Although the response rate for such applications is often low (e.g. 4%), and it is difficult or impossible to predict the class with high accuracy for all customers, a good model can concentrate the likely responders near the top of the list. For example, the top 5% of the list (sorted by the model prediction) may contain 30% of responders (compared to 4% base rate), giving the model a *lift* of 30/4=7.5. For a large customer base, even small improvements in prediction accuracy can yield large improvements in lift.

In this paper we argue that lift measure by itself is not sufficient and that we should take the customer value into account in order to determine the model payoff. Using the predicted behavior and a simple business model we estimate the payoffs from different models and examine different strategies to arrive at an optimal model that maximizes overall business value rather than just accuracy or lift.

In the rest of this paper we explain the business problem and model of business payoffs, present the main experimental hypotheses, results and conclusions.

2 Motivation

When using a predictive model to assign likelihood of response, typically overall accuracy or lift is maximized. To maximize business value however, we need to maximize not only prediction accuracy but identify a group of customers that are not only highly likely to respond but are going to be "high value" customers. How might one arrive at such a model? Is it enough to just select the estimated high value customers from the group of predicted reponders as a post processing step or might it be beneficial to have the predictive model itself take the value into account while building the model? We examine these questions by conducting experiments that contrast different strategies to take value into account.

3 Description of the business problem

Given billing information on customers for a given month, we want to predict the customer behavior for the next month

Table 1:
Example data fields from Customer Billing Record (Aug 95)

record number	unpaid Balance	total Amt Billed	prev Bill Amt	total Peak	cash Payments	monthly Service Charge	service length	optional Feature Charge	ldCarrier	customer service calls	response to offers
1	373.0	519.37	373.0	25.0	0.0	18.0	25.0	4.0	280	2	0
2	0.0	150.25	110.0	37.0	110.0	24.0	12.0	6.0	280	0	1
3	20.0	85.0	60.0	5.0	40.0	10.0	30.0	2.0	280	0	0

and arrive at a ranking of all subscribers for the purpose of extending offers to the top few percent from such a ranked list. The billing data (see Table 1) includes such information as total amount billed, unpaid balances, use of specific (telephone) services, customer service calls plus information on length of service and past marketing behavior e.g. whether they accepted certain offers or not. The sample database we use includes information for 100,000 customers with about two hundred fields per customer per month. A standard approach to this problem is to regard it as a classification problem and use one of the machine learning methods such as neural networks (Rumelhart 1986), decision tree induction (Quinlan 1993) or nearest neighbor classifiers (Dasarathy 1991) and build a predictive model that maximizes some criteria such as overall accuracy, lift in the top 5%.

For the experiments described in this paper, we use a commercial neural network software package from Neuralware. The following steps describe how we prepare the data for modelling.

3.1 Pre-processing and sampling data for analysis

Our first sample database comprises of 25,000 records from a particular market. Based on historical data, typical response rate for this application (responding to offers) are in the vicinity of 7%. For the purpose of building a model we need a more concentrated representation of responders. This is because with a sparse response rate, a learning method can achieve high accuracy by always predicting everyone to be a non-responder.

We create a random sample where we include about 2000 responders and then add enough non-responders to make a dataset of 50-50 concentration of responders. For each one of these individuals we include information from one previous month.

We divide this dataset (about 4000 records) into approximately 2/3 and 1/3 size training and test sets with a 50-50 concentration of responders. A separate random sample of 6000 records (with no overlap with the train, test sets) is held aside for evaluation purposes.

A second, larger sample is drawn from a different market with 100,000 records and a much lower response rate of about 2%. Following similar steps training and test sets of size about 4000 each are created and a non-overlapping heldaside set of 24,000 records is kept separately for evaluation.

3.2 Reducing the number of data fields for model development

As reported in (Kohavi & Sommerfield 1995) and (Almuallim & Dietterich 1991), reducing or eliminating irrelevant data fields can result in improved classification accuracy. We first excluded fields that have constant values as well l as dependent fields such as tax charges. In order to prune the data fields further, we correlated each field with the target field and arrived at a smaller list of fields per month (about 40). While in general it may not be desirable to exclude fields before the application of a learning procedure, in this case our goal is to compare different learning strategies to maximize business payoff from the models, therefore just including the "best n" fields still serves the purpose of contrasting different modelling strategies.

3.3 Methodology of testing

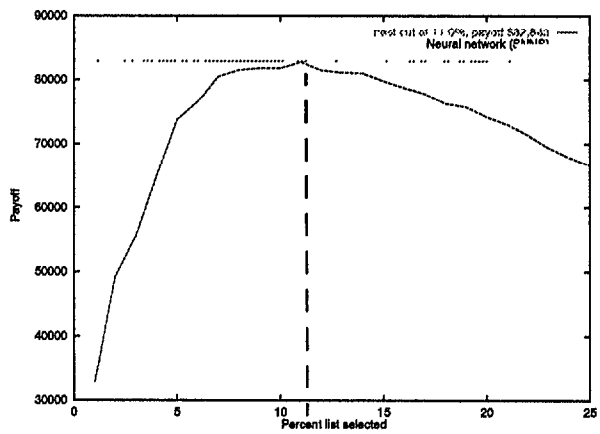
In order to estimate the accuracy and payoff of different models their ability to predict needs to be tested on a representation of "unseen" data. For our experiments the neural network training was done using the 50-50 train and the test sets, using the test set to prevent overtraining. Once the models were developed, they were applied to the heldaside set to compute the ranked list of subscribers and to estimate model payoffs on unseen data.

reaching non-responders overrides the benefit of potential response. There is an estimated "optimal" number of people to call, in this case about 15% giving an estimated optimal payoff of about $72 k for this group. If this was linearly scaled to 100,000 customers this payoff would represent about $1.1 million. (In practice we have observed a non-linear scaling, giving a higher lift and payoff for larger samples).

Table 3: Example lift table for ranked output log for 6,300 customers

seg-ment	hit rate per seg-ment	cum abs hits	cum% of all hits cap-tured	lift	payoff per seg (x 1000)	cum payoff (x 1000)
5%	45.71	144	30.00	6.00	61.13	61.13
10%	19.68	206	42.92	4.29	8.02	69.15
15%	15.24	254	52.92	3.53	3.14	72.29
20%	7.30	277	57.71	2.89	-0.48	71.81
30%	6.35	322	67.08	2.24	-0.15	71.40
50%	5.71	392	81.67	1.63	-2.08	67.97
70%	3.02	431	89.79	1.28	-4.54	58.99
80%	2.54	447	93.12	1.16	-4.63	54.36
90%	1.75	458	95.42	1.06	-5.05	49.31
100%	3.49	480	100.0	1.0	-3.37	45.94

The figure below (based on slightly different data from the above table) shows an optimal cutoff point from the ranked prediction log for an optimal payoff. Here the optimal point is about 11% for a payoff of $82k.



4 Main experimental hypotheses: Value based training vs. post processing

Given the above definition of business payoff in this domain, we now examine different strategies for the task of optimizing the payoff. Essentially we would like to rank the subscriber base in such a way that the people who are most likely to respond are at the top of the ranked list. In addition, we would like to identify a group of customers that are likely to respond to high value offers. For example, it might be better to extend offers to a smaller group of people with a moderate likelihood of response but with the high expected revenue, rather than a group of customers that with a higher likelihood of response but a lower expected revenue.

We ask the question: Is it enough to use a predictive model to arrive at ranked list of subscribers, and then, as a post processing step select the "high value" customers (as estimated from their past behavior) or is it necessary to somehow include the notion of value in the training process itself? Should we stratify customers into high value customers and low value customers and build different models for them? We examine and contrast the following four approaches:

4.1 Baseline payoff calculation

Using a basic predictive model we establish a baseline payoff against which all improvements can be measured. The payoff calculations are done as explained in sections 3.4 and 3.5. It is important to note that what we call "value" (various sources of revenue) are present as some of the inputs, although their use is to predict a likelihood of any response at all (a classification problem) rather than an absolute magnitude of the response (a regression problem).

4.2 Post Processing

We examine a simple strategy to re-rank the basic ranked list using a criteria that takes both likelihood of response and the actual estimated value (which is estimated from the input revenue variables). We use a product of the two factors as a composite rank factor. This simulates "expected value" even though the likelihood as estimated from the neural net prediction is not a strict probability. Thus subscribers with a high likelihood but low estimated value may not be near the top of the list because another subscriber with a moderate likelihood but high value might have a higher composite score.

merging the entire output logs from the two stratified models while the last row describes the results obtained by merging only optimum subsets from the output of the two stratified models.

Table 6: Dataset 2, Comparison of Optimum payoffs (x $1000)

	experiment	input 5	input 10	input 15	input 25	input 35	mean (std dev)
1	basic opt payoff	55.0	52.7	54.5	57.3	57.1	55.3 (1.9)
2	basic resorted opt payoff	57.6	56.2	56.4	59.0	58.2	57.5 (1.2)
3	value training opt payoff	58.6	59.3	58.8	58.5	58.8	58.8 (0.3)
4	strat1 merge opt payoff	54.7	53.3	54.0	53.1	52.9	53.6 (0.7)
5	strat2 opt-merge opt payoff	58.3	58.1	56.6	57.2	57.1	57.4 (0.7)

5.1 Improvements using post processing

We found that re-ranking based on the product of the likelihood of response and estimated revenue does not results in a significant change for Dataset 1 while for Dataset 2 there is an improvement (95% confidence level for a test comparing the difference between means of two populations)

5.2 Improvements using value based training

As can be seen by comparing the third and the first row of Tables 5 and 6, the results from the value based training are significantly better than the payoffs from the basic model and also consistently better than the post processing strategy

5.3 Improvements using training based on stratified models

We expected the results to improve significantly using the straight merge from the stratified models but the best comparable results to value based training were obtained by merging the optimum subsets from the stratified model outputs.

5.4 Comparison of optimal payoffs with best accuracy and lift

More details related to the experiments for Dataset 1 in Table 5 can be found in Table 7 where the first column indexes the experiments from Table 5 and the remaining

Table 7: Comparison of Optimum payoffs vs. accuracy and lift

	experiment	opt payoff	opt cutoff (%)	overall accuracy	lift at 5%
1	basic input 5	72.05	29.06	72.8	4.97
1	basic input 10	73.42	34.59	75.66	5.36
1	basic input 15	72.27	35.14	74.14	5.23
1	basic input 25	72.26	37.89	74.05	5.23
1	basic input 35	71.77	36.69	73.41	5.4
2	basic resorted 5	72.17	28.7	72.8	4.27
2	basic resorted 10	71.98	19.56	75.66	4.66
2	basic resorted 15	72.38	22.64	74.14	4.97
2	basic resorted 25	72.63	20.26	74.05	4.79
2	basic resorted 35	72.30	18.87	73.41	4.84
3	value training 5	77.79	35.39	65.62	4.47
3	value training 10	77.96	37.69	63.7	4.98
3	value training 15	77.58	28.12	59.25	4.66
3	value training 25	78.35	25.57	58.03	4.84
3	value training 35	77.73	33.7	60.04	4.84
4	basic merge strat1 5	73.65	39.32	73.22	5.14
4	basic mergestrat110	74.15	41.02	74.85	5.36
4	basic merge strat1 15	71.92	53.31	74.03	5.05
4	basic merge strat1 25	71.75	36.52	72.94	4.27
4	basic merge strat1 35	71.08	31.48	73.71	5.01
5	basic optmerge strat2 5	77.0	37.33	34.17	3.83
5	basic optmerge strat2 10	78.13	33.43	33.92	3.63
5	basic optmerge strat2 15	77.56	32.84	39.04	3.76
5	basic optmerge strat2 25	76.48	38.12	34.97	2.81
5	basic optmerge strat2 35	77.22	36.12	34.02	2.33

columns describe parameters such as accuracy and lift. It is generally expected that high accuracy and high lift will be correlated with a high payoff model, however as can be seen from Table 7, the best payoffs are not correlated with the best accuracy or highest lift. This is consistent with the explanation that when we rank subscribers with just the likelihood of response, there is no necessary correlation between high likelihood of response and high magnitude (high value) of response. Thus the strategy which achieves high lift in predicting subscribers may not have the highest payoff value.

Another dimension of comparison can be the optimal percent of subscribers selected for the optimal payoff. For a comparable payoff, a smaller set of selected subscribers would be preferable.

6 Conclusions

We address the problem of identifying an optimal subset of customers with highest estimated payoff for extending offers to them. We find that for our domain, using neural network models on two datasets, different in size and response rates, the value based training strategies and the stratified optimal merge approach outperform the simple post processing strategy based on re-ranking using expected value estimates.

6.1 Discussion and Analysis

One might ask whether the high value stratified group (top 25%) is not sufficient by itself to produce the highest payoffs. We found that there are enough moderate value responders in the remaining 75% such that the payoff from the top 25% alone cannot match the highest payoffs from the value based training.

While new re-ranking factors for post processing can perhaps be discovered by methods such as Genetic Programming or manual heuristics, we show a definite improvement using a value based response variable for training across a range of model complexity as measured by different number of inputs. It's not clear yet if this approach would apply to different domains with a similar sparse response rate and value criteria (e.g. a domain such as credit card attrition).

6.2 Extensions

Methods such C4.5 and KNN classification can also be modified for value based training. We are adding bootstrap error estimates for the optimum payoffs to better assess the statistical significance of the relative ranking of different strategies. We are also experimenting with variations of value based training such as re-ranking the output logs of the value based model and also doing value based training on stratified sets.

7 Acknowledgments

We would like to thank John Vittal for his support of this project.

8 References

Almuallim H. and Dietterich T. 1991. Learning with Many Irrelevant Features. In Proceedings of AAAI-91, 547-552. Menlo Park, CA: AAAI Press.

Dasarathy, B.V. 1991. *Nearest Neighbor Norms: NN Pattern Classification techniques.* Los Alamitos, CA: IEEE Press.

Fayyad, U., Piatetsky-Shapiro, G., P. Smyth, and Uthurusamy, R. 1996. *Advances in Knowledge Discovery and Data Mining*, Cambridge, MA: AAAI/MIT Press.

Kohavi, R. and Sommerfield, D. 1995. Feature Subset Selection Using the Wrapper Method: Overfitting and Dynamic Search Space Topology. In Proceedings of KDD-95: First International Conference on Knowledge Discovery and Data Mining, 192--197. Menlo Park, CA: AAAI Press.

Piatetsky-Shapiro, G. and Frawley, W. 1991. *Knowledge Discovery in Databases*, Cambridge, MA: AAAI/MIT Press.

Rumelhart, D. E., Hinton, G. E. & Williams, R. J. 1986. Learning internal representations by error propagation. In *Parallel Distributed Processing: Explorations in the microstructure of cognition. Volume I: Foundations.* Cambridge, MA: MIT Press/Bradford Books, pp 318 - 362.

Quinlan, J. R. 1993. *C4.5: Programs for Machine Learning.* Los Gatos, CA: Morgan Kauffman.