# Mining Customer Value: From Association Rules to Direct Marketing [*]

Ke Wang
Simon Fraser University
wangk@cs.sfu.ca

Senqiang Zhou
Simon Fraser University
szhoua@cs.sfu.ca

Jack Man Shun Yeung
Simon Fraser University
yeung@cs.sfu.ca

Qiang Yang
Hong Kong University of Science and Technology
qyang@cs.ust.hk

## 1 Introduction

*Direct marketing* refers to a process of identifying and mailing to potential customers: retail industries need to identify buyers of certain products, banks and insurance companies need to promote loan insurance products to customers, fundraising organizations need to identify potential donors, etc. Available is a *historical* database about the previous mailing campaign, including whether a customer responded and the dollar amount collected. The task is to build a model to predict the *current* customers who are likely to respond. Typically, most records, say 95%, are "not_respond" records. Thus, maximizing the accuracy of prediction does not work because simply predicting all customers as "not_respond" will give 95% accuracy.

In recent years, it is realized that cost-sensitive treatment is required in applications like direct marketing. [2] proposed the MetaCost framework for adopting accuracy-based classification to cost-sensitive learning by incorporating a cost matrix $C(i,j)$ for misclassifying class $j$ into class $i$. [6] examined the more general case where the benefit $B(i,j,x)$ depends not only on the classes involved but also on the individual customers $x$. A drawback of these approaches is that they need to estimate the the conditional class probability $P(j|x)$, which ignores the customer value of $x$ such as the donation amount. The customer value is only considered "after the fact" via the factor $B(i,j,x)$.

In this paper, we estimate the profit on a customer *directly*. This has two advantages. First, it takes into account the customer value from the very beginning. Second, it opens up new avenues for profit estimation. In particular, we propose a profit estimation method by combining association rules [1] and pessimistic estimation of errors [5]. Association rule approach offers an edge of finding correlated features that may never be found in a local search such as in decision tree induction and naive Bayes classifiers. The main contribution in this work is to make isolated association rules work as a team for maximizing the profit generated. On the well known, large, and challenging KDD-CUP-98 task [3], the proposed method generates 43% more profit than the winner of the competition and 37% more profit than the best known result. Although we consider the KDD-CUP-98 dataset for concreteness, the method proposed applies to the general cost-sensitive learning as described in [2, 6].

## 2 The Proposed Approach

Each record in the KDD-CUP-98 dataset [3] is described by 479 non-target variables and two target variables indicating the "respond"/"not_respond" classes and the actual donation in dollars. About 5% of records are "respond" records and the rest are "not_respond" records. The dataset has been pre-split into 50% for learning and 50% for validation. The competition task is to build a prediction model of the donation amount using the learning set. The participants are contested on $\Sigma(actual\ donation - \$0.68)$ over all the validation records with predicted donation greater than the mailing cost $0.68.

This real life dataset presents two challenges. First, "there is often an inverse correlation between the likelihood to respond and the dollar amount of the gift" (quoted from [4]). This inverse correlation invalids any probability based ranking because a valuable customer will be ranked low. Second, the high dimensionality of the dataset presents a big challenge for extracting correlated features. Among the 481 variables in the dataset, 208 variables have 10 or more distinct values after discretizing continuous variables, making a potential search space of size $10^{208}$.

We address these issues in three steps. In *rule generating*, we extract characteristics typical of "respond" records. In *model building*, we construct a prediction model using

[*] Research was supported in part by research grants from the Natural Science and Engineering Research Council of Canada

| Rule $r$ | Record $t$ | Profit model |
|---|---|---|
| FAR | "respond" record | $V - 0.68$ |
| "not_respond" rule | "respond" record | 0 |
| FAR | "not_respond" record | $-0.68$ |
| "not_respond" rule | "not_respond" record | 0 |

**Table 1.** $profit(r,t)$ **(**$V$ **is the donation amount in** $t$**)**

the found characteristics to maximize generated profit on the learning set. In *model pruning*, we prune overfitting characteristics to generalize the model to the whole population.

### 2.1 Step 1: Rule Generating

This step finds useful rules for predicting responders, called FARs, of the form

$$A_{i_1} = a_{i_1}, \ldots, A_{i_k} = a_{i_k} \rightarrow respond.$$

Despite many efficient algorithms for mining association rules (see [1], for example), we encountered a significant difficulty in this step because the dataset has 481 (non-binary) variables! We split the learning set into "respond" records denoted by $D_R$ and "not_respond" records denoted by $D_N$, and require that the support of each $A_i = a_i$ in $D_N$ be below some threshold, and the support of the antecedant $A_{i_1} = a_{i_1}, \ldots, A_{i_k} = a_{i_k}$ in $D_R$ be above some threshold. We call such rules *focused association rules (FARs)*. Finding FARs is similar to finding association rules, but we examine only $D_R$, which is 5% of the dataset, and the data items that are below the threshold for $D_N$ after the first iteration. We use only one "not_respond" rule, $\emptyset \rightarrow not\_respond$, which is used if a customer matches no FAR.

### 2.2 Step 2: Model Building

To predict on a customer record, we choose the rule that matches the record and has the largest observed profit. Table 1 shows the computation of the profit that $r$ generates on a learning record $t$, denoted $profit(r,t)$. The *observed profit* of $r$ is defined as:

$$O\_avg(r) = \Sigma_t profit(r,t)/N,$$

where $t$ is a learning record that matches $r$ and $N$ is the number of such records. Given a record $t$, the *prediction rule* of $t$ is the rule $r$ that matches $t$ and has the highest possible $O\_avg(r)$.

The mailing decision of a prediction rule $r$ is determined by the "estimated profit" of $r$. Suppose that $r$ predictions $N$ records in the learning set (following the principle of prediction rules), $E$ of which are predicted wrongly, i.e., do not match the consequent of $r$. The estimated profit of $r$ is computed by $x \times y$, where $x$ is the observed (in the learning set) profit per *correct* prediction by $r$, $y$ is the estimated number of correct predictions by $r$ for $N$ random customers from the whole population. We borrow the *pessimistic estimation* from [5], denoted $U_{CF}(N,E)$, to estimate the upper bound of the error rate of $r$ for a given confidence interval $CF$. The *estimated profit* of $r$ is defined as

- $Estimate(r) = 0$, if $r$ is the "not_respond" rule;
- $Estimate(r) = N \times (1 - U_{CF}(N,E)) \times x$, if $r$ is a FAR.

For a given current customer, if the prediction rule $r$ has a positive $Estimate(r)/N$, the customer will be contacted.

### 2.3 Step 3: Model Pruning

In the last step, we prune overly specific rules to generalize the profit maximization to the whole population. The idea is to prune rules on the basis of increasing the total estimated profit over all remaining rules. We organize rules into a specialization/generalization tree structure, in which a rule $r'$ is the *parent* of a rule $r$ if $r'$ is more general than (the antecedant of) $r$ and has the highest possible $O\_avg$. The essence of being a parent is that if a rule $r$ is pruned, the parent of $r$ will act as the prediction rule of the records previously predicted by $r$. We prune the rules in the tree in the bottom-up order. At each non-leaf node $r$, we compare the total estimated profit before and after pruning the subtree at $r$. If the pruning increases the total estimated profit, we prune the subtree; otherwise, we keep the subtree.

## 3 Validation

We validate the proposed method using the standard split of the KDD98-learning-set (95,412 records) and KDD98-validation-set (96,367 records) used by the KDD competition [3]. A model is built using only the KDD98-learning-set and is evaluated by the competition criterion, i.e., the *sum of actual profit* on the KDD98-validation-set, defined as $\Sigma_t(V - 0.68)$ for all validation records $t$ with a positive predicted profit $Estimate(r)/N$, where $V$ is the donation amount in $t$ and $r$ is the prediction rule for $t$. We choose the thresholds in the rule generating by testing on some records in the KDD98-learning-set that are hidden away from the model building.

Table 2 shows the comparison with several published results. The first row (in bold face) is our result. Next come the three categories of published results in the following

| Category | Algorithm | Sum of Actual Profit | # Mailed | Average Profit |
|---|---|---|---|---|
|  | **Our Algorithm** | **$21,045** | **23,437** | **$0.90** |
| KDD-CUP-98 Results [4] | GainSmarts (The winner) | $14,712.24 | 56,330 | $0.26 |
|  | SAS/Enterprise Miner (#2) | $14,662.43 | 55,838 | $0.26 |
|  | Quadstone/Decisionhouse (#3) | $13,954.47 | 57,836 | $0.24 |
|  | ARIAI/CARRL (#4) | $13,824.77 | 55,650 | $0.25 |
|  | Amdocs/KDD Suite (#5) | $13,794.24 | 51,906 | $0.27 |
| MetaCost [2, 6] | Smoothed C4.5 (sm) | $12,835 |  |  |
|  | C4.5 with curtailment (cur) | $11,283 |  |  |
|  | Binned naive Bayes (binb) | $14,113 |  |  |
|  | Average (sm, cur) | $13,284 |  |  |
|  | Average (sm, cur, binb) | $13,515 |  |  |
| Direct Cost-Sensitive [6] | Smoothed C4.5 (sm) | $14,321 |  |  |
|  | C4.5 with curtailment (cur) | $14,161 |  |  |
|  | Binned naive Bayes (binb) | $15,094 |  |  |
|  | Average (sm, cur) | $14,879 |  |  |
|  | Average (sm, cur, binb) | $15,329 |  |  |
| CART 4.0 | CART 4.0 | same as Mail to Everyone |  |  |
|  | Maximum possible profit | $72,776 | 4,873 | $14.93 |
|  | Mail to Everyone | $10,548 | 96,367 | $0.11 |

**Table 2. Comparison with published results**

order: the top five contestants of the KDD-CUP-98 as reported in [4], five algorithms of MetaCost and five algorithms of direct cost-sensitive decision making as reported in [6]. CART produced the same result as "Mail to Everyone". The last two rows show the maximum possible profit and the profit of "Mail to Everyone". Our method generated the sum of actual profit of $21,045, which is 43% more than the KDD-CUP-98 winner, 49% more than the best profit of MetaCost, and 37% more than the best profit of direct cost-sensitive decision making. From an analysis in [6], these differences are far larger than the required $1090 in order to be statistically significant. In fact, compared to the KDD-CUP winner, we generated 43% more profit by predicting less than an half number of contacts. This success is credited to the direct profit estimation and the global search of association rules.

## 4 Conclusion

In this paper, we push the customer value as the first class information. Our approach is to estimate directly the profit generated on a customer without estimating the conditional class probability. This methodology opens up new possibilities for profit estimation. In particular, we use association rules to summarize customer groups and to build a model for profit prediction. The advantage of the association rule approach is its scalability of finding correlated features that may never be found in a local search. The evaluation on the well known, large, and challenging KDD-CUP-98 task shows a breakthrough result.

## References

[1] R. Agrawal, T. Imilienski, and A. Swami. Mining association rules between sets of items in large datasets. In *SIGMOD*, pages 207–216, 1993.

[2] P. Domingos. Metacost: A general method for making classifiers cost sensitive. In *KDD 99*, pages 155–164. KDD, August 1999.

[3] KDD98. The kdd-cup-98 dataset. In *http://kdd.ics.uci.edu/databases/kddcup98/kddcup98.html*. KDD, August 1998.

[4] KDD98. The kdd-cup-98 result. In *http://www.kdnuggets.com/meetings/kdd98/kdd-cup-98.html*. KDD, August 1998.

[5] J. Quinlan. *C4.5: programs for machine learning*. Morgan Kaufmann, San Mateo, CA, USA, 1993.

[6] B. Zadrozny and C. Elkan. Learning and making decisions when costs and probabilities are both unknown. In *SIGKDD*, pages 204–213. SIGKDD, August 2001.