CHAPTER 7

CASE STUDY: MODELING RESPONSE TO DIRECT MAIL MARKETING

CROSS-INDUSTRY STANDARD PROCESS FOR DATA MINING

The case study in this chapter is carried out using the Cross-Industry Standard Process for Data Mining (CRISP–DM). According to CRISP–DM, a given data mining project has a life cycle consisting of six phases, as illustrated in Figure 7.1. Note that the phase sequence is *adaptive*. That is, the next phase in the sequence often depends on the outcomes associated with the preceding phase. The most significant dependencies between phases are indicated by the arrows. For example, suppose that we are in the modeling phase. Depending on the behavior and characteristics of the model, we may have to return to the data preparation phase for further refinement before moving forward to the model evaluation phase. The six phases are as follows:

1. *Business understanding phase*. The first phase in the CRISP–DM standard process may also be termed the *research understanding phase*.
   a. Enunciate the project objectives and requirements clearly in terms of the business or research unit as a whole.
   b. Translate these goals and restrictions into the formulation of a data mining problem definition.
   c. Prepare a preliminary strategy for achieving these objectives.

Data Mining Methods and Models By Daniel T. Larose

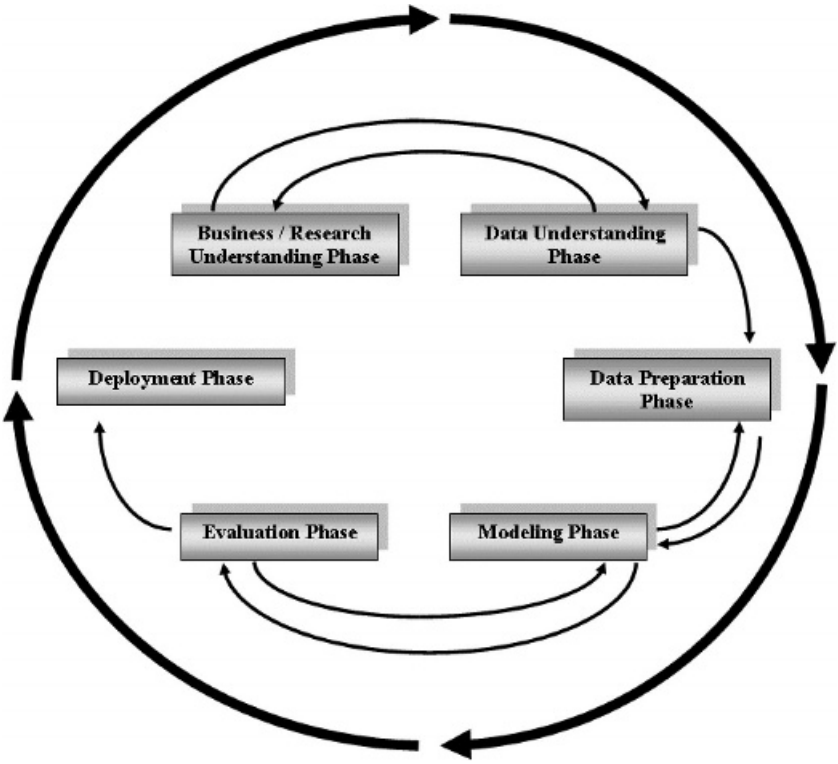


Figure 7.1 CRISP–DM is an iterative, adaptive process.

2. *Data understanding phase*
   a. Collect the data.
   b. Use exploratory data analysis to familiarize yourself with the data, and discover initial insights.
   c. Evaluate the quality of the data.
   d. If desired, select interesting subsets that may contain actionable patterns.

3. *Data preparation phase*
   a. This labor-intensive phase covers all aspects of preparing the final data set, which will be used for subsequent phases, from the initial, raw, dirty data.
   b. Select the cases and variables you want to analyze and that are appropriate for your analysis.
   c. Perform transformations on certain variables, if needed.
   d. Clean the raw data so that it is ready for the modeling tools.

4. *Modeling phase*
   a. Select and apply appropriate modeling techniques.
   b. Calibrate model settings to optimize results.
   c. Often, several different techniques may be applied for the same data mining problem.

d. Loop back to the data preparation phase as required to bring the form of the data into line with the specific requirements of a particular data mining technique.

5. *Evaluation phase*

a. The modeling phase has delivered one or more models. These models must be evaluated for quality and effectiveness before we deploy them for use in the field.

b. Determine whether the model in fact achieves the objectives set for it in phase 1.

c. Establish whether some important facet of the business or research problem has not been accounted for sufficiently.

d. Finally, come to a decision regarding the use of the data mining results.

6. *Deployment phase*

a. Model creation does not signify the completion of the project. Need to make use of created models according to business objectives.

b. Example of a simple deployment: Generate a report.

c. Example of a more complex deployment: Implement a parallel data mining process in another department.

d. For businesses, the customer often carries out the deployment based on your model.

For more on CRISP–DM, see Chapman et al. [1], Larose [2], or `www.crisp-dm.org`.

BUSINESS UNDERSTANDING PHASE

Direct Mail Marketing Response Problem

In this detailed case study, our task is to predict which customers are most likely to respond to a direct mail marketing promotion. The *clothing-store* data set [3], located at the book series Web site, represents actual data provided by a clothing store chain in New England. Data were collected on 51 fields for 28,799 customers. More information about the data set is provided in the data understanding phase below.

Our data mining task is a classification problem. We are to classify which customers will respond to a direct mail marketing promotion based on information collected about the customers. How does this problem fit into the business as a whole? Clearly, for the clothing store, the overriding objective is to increase profits. Therefore, the goal of our classification model should also be to increase profits. Model evaluative measures that assess the effect of the classification model on the business's bottom line will therefore be applied. In the following sections we examine this case study using Clementine 8.5 data mining software, available from SPSS, Inc.

Building the Cost/Benefit Table

Classification models are often evaluated on accuracy rates, error rates, false negative rates, and false positive rates. These measures can be applied to any classification

problem. However, for a particular classification problem, these measures may not select the optimal model. The reason is that each classification problem carries with it a unique set of costs and benefits, which stem from the particular set of circumstances unique to that business or research problem.

The cost of a false positive (wrongly predicting positive response) may be low in certain environments but higher in other environments. For example, in direct marketing, a false positive may cost no more than a postcard, while in HIV testing, a false positive on the ELISA test will be more expensive, leading to second-level HIV testing. (On the other hand, of course, false negatives in HIV testing are very serious indeed, which is why the ELISA test allows a higher rate of false positives, to maintain the false negative rate as low as possible.)

In business problems, such as our direct mail marketing problem, company managers may require that model comparisons be made in terms of cost/benefit analysis. Recall from *Discovering Knowledge in Data: An Introduction to Data Mining* [2] that it is useful to construct a cost/benefit table when performing classification. This is done to provide model comparison in terms of anticipated profit or loss by associating a cost or benefit with each of the four possible combinations of correct and incorrect classifications.

Let us consider each of the four possible decision outcomes (true negative, true positive, false negative, and false positive) and assign reasonable costs to each decision outcome.

1. *True negative* (TN). The model predicted that this customer would not respond to the direct mail marketing promotion, so no postcard was mailed to him or her. In reality, this customer would not have responded to the promotion. Therefore, the correct decision was made. No costs were incurred, since no postcard was sent; no sales were made, and no prospective sales were lost.
2. *True positive* (TP). The model predicted that this customer would respond to the direct mail marketing promotion, so a promotion was mailed to him or her. In reality, this customer would indeed have responded to the promotion. Therefore, again the correct decision was made. The direct mailing cost, with materials, postage, and handling, is $2 per promotion unit mailed. However, this particular TP customer, upon receiving the postcard, would have come into the store to make purchases. The question then becomes: How much money would we reasonably expect the customer to spend, and how much of that amount spent could be considered profit? Table 7.1 shows the statistics associated with the average amount spent per visit for all 28,799 customers. The mean is $113.59, which we shall use as our estimate of the amount this customer will spend on the visit after receiving the promotion. (The median is another reasonable estimate, which we did not use in this example. By the way, why is the mean larger than the median? *Hint*: Check out the maximum: Imagine spending an average of $1919.88 per visit to a clothing store.) Assume that 25% of this $113.59, or $28.40, represents profit. Then the benefit associated with this customer is the profit expected from the visit, $28.40, minus the cost associated with the mailing, $2.00, that is, $26.40.

TABLE 7.1 Statistics Associated with the Average Amount Spent per Visit for All Customers

Count	28,799
Mean	113.588
Minimum	0.490
Maximum	1,919.880
Standard deviation	86.981
Median	92.000

3. *False negative* (FN). In most marketing problems, which decision error is worse, a false negative or a false positive? A false positive means that you contacted a nonresponsive customer, which is not very costly. But a false negative means that you failed to contact a customer who would have responded positively to the promotion. This error is much more expensive, and marketing classification modelers should endeavor to minimize the probability of making this type of error. What is the cost associated with making a false negative decision in this case? There is no cost of contact for this customer, since we did not contact him or her. But had this customer been in fact contacted, he or she would have responded, and spent money at the clothing store. The estimated amount is the same as above, \$113.59, of which \$28.40 would have been profit. Therefore, the lost profit associated with this customer is \$28.40.
4. *False positive* (FP). False positives are much less serious for marketing models. Here, the cost associated with contacting a nonresponsive customer is the \$2 for postage and handling. We can therefore see that in the context of this particular problem, a false negative is $28.40/2.00 = 14.2$ times as expensive as a false positive.

We may thus proceed to construct the cost/benefit table for this clothing store marketing promotion example, as shown in Table 7.2. Note that benefit is shown as

TABLE 7.2 Cost/Benefit Decision Summary for the Clothing Store Marketing Promotion Problem

Outcome	Classification	Actual Response	Cost	Rationale
True negative	Nonresponse	Nonresponse	\$0	No contact, no lost profit
True positive	Response	Response	−\$26.4	Estimated profit minus cost of mailing
False negative	Nonresponse	Response	\$28.40	Lost profit
False positive	Response	Nonresponse	\$2.00	Materials, postage, and handling cost

negative cost. The cost/benefit table in Table 7.2 will be the final arbitrator of which model we select as optimal for this problem, error rates notwithstanding.

DATA UNDERSTANDING AND DATA PREPARATION PHASES

Clothing Store Data Set

For this case study we meld together the data understanding and data preparation phases, since what we learn in each phase immediately affects our actions in the other phase. The *clothing-store* data set contains information about 28,799 customers in the following 51 fields:

- Customer ID: unique, encrypted customer identification
- Zip code
- Number of purchase visits
- Total net sales
- Average amount spent per visit
- Amount spent at each of four different franchises (four variables)
- Amount spent in the past month, the past three months, and the past six months
- Amount spent the same period last year
- Gross margin percentage
- Number of marketing promotions on file
- Number of days the customer has been on file
- Number of days between purchases
- Markdown percentage on customer purchases
- Number of different product classes purchased
- Number of coupons used by the customer
- Total number of individual items purchased by the customer
- Number of stores the customer shopped at
- Number of promotions mailed in the past year
- Number of promotions responded to in the past year
- Promotion response rate for the past year
- Product uniformity (low score = diverse spending patterns)
- Lifetime average time between visits
- Microvision lifestyle cluster type
- Percent of returns
- Flag: credit card user
- Flag: valid phone number on file
- Flag: Web shopper

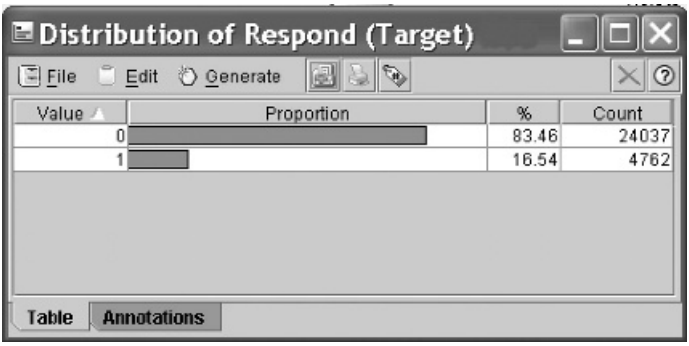


Figure 7.2 Most customers are nonresponders.

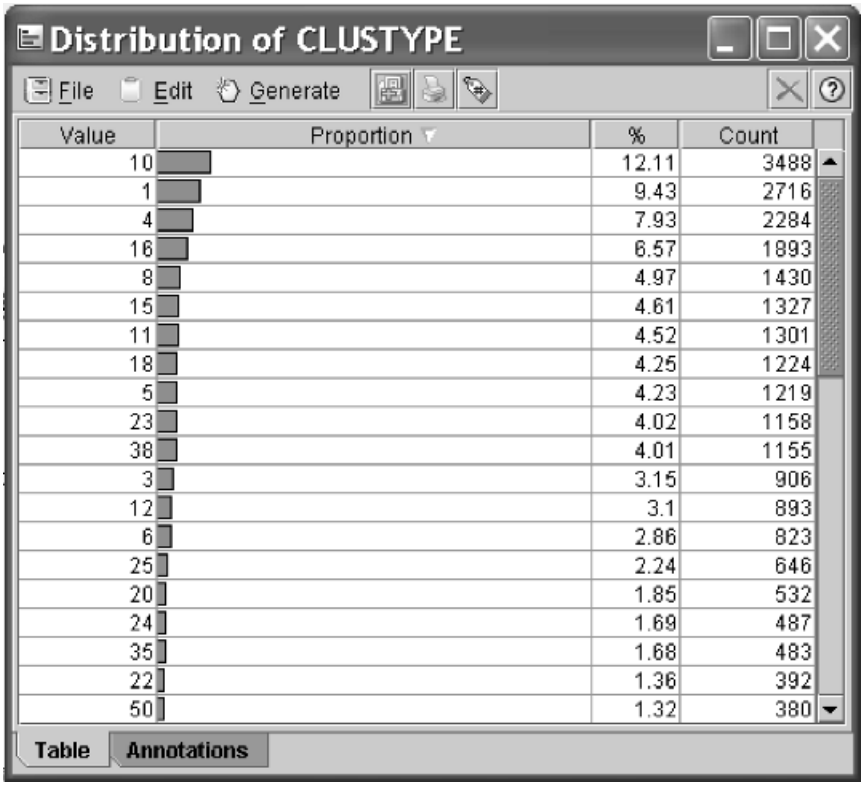
- 15 variables providing the percentages spent by the customer on specific classes of clothing, including sweaters, knit tops, knit dresses, blouses, jackets, career pants, casual pants, shirts, dresses, suits, outerwear, jewelry, fashion, legwear, and the collectibles line; also a variable showing the brand of choice (encrypted)
- Target variable: response to promotion

These data are based on a direct mail marketing campaign conducted last year. We use this information to develop classification models for this year's marketing campaign. In the data understanding phase, we become more familiar with the data set using exploratory data analysis (EDA) and graphical and descriptive statistical methods for learning about data. First, what is the proportion of responders to the direct mail marketing promotion? Figure 7.2 shows that only 4762 of the 28,799 customers, or 16.54%, responded to last year's marketing campaign (1 indicates response, 0 indicates nonresponse.) Since the proportion of responders is so small, we may decide to apply balancing to the data prior to modeling.

One of the variables, the Microvision lifestyle cluster type, contains the market segmentation category for each customer as defined by Claritas Demographics [4]. There are 50 segmentation categories, labeled 1 to 50; the distribution of the most prevalent 20 cluster types over the customer database is given in Figure 7.3.

The six most common lifestyle cluster types in our data set are:

1. *Cluster 10: Home Sweet Home*—families, medium-high income and education, managers/professionals, technical/sales
2. *Cluster 1: Upper Crust*—metropolitan families, very high income and education, homeowners, manager/professionals
3. *Cluster 4: Midlife Success*—families, very high education, high income, managers/professionals, technical/sales
4. *Cluster 16: Country Home Families*—large families, rural areas, medium education, medium income, precision/crafts
5. *Cluster 8: Movers and Shakers*—singles, couples, students, and recent graduates, high education and income, managers/professionals, technical/sales
6. *Cluster 15: Great Beginnings*—young, singles and couples, medium-high education, medium income, some renters, managers/professionals, technical/sales



Value	Proportion	%	Count
10		12.11	3488
1		9.43	2716
4		7.93	2284
16		6.57	1893
8		4.97	1430
15		4.61	1327
11		4.52	1301
18		4.25	1224
5		4.23	1219
23		4.02	1158
38		4.01	1155
3		3.15	906
12		3.1	893
6		2.86	823
25		2.24	646
20		1.85	532
24		1.69	487
35		1.68	483
22		1.36	392
50		1.32	380

Figure 7.3 The 20 most prevalent Microvision lifestyle cluster types.

Overall, the clothing store seems to attract a prosperous clientele with fairly high income and education. Cluster 1, *Upper Crust*, represents the wealthiest of the 50 cluster types and is the second most prevalent category among our customers.

Moving to other variables, we turn to the customer ID. Since this field is unique to every customer and is encrypted, it can contain no information that is helpful for our task of predicting which customers are most likely to respond to the direct mail marketing promotion. It is therefore omitted from further analysis.

The zip code can potentially contain information useful in this task. Although ostensibly numeric, zip codes actually represent a categorization of the client database by geographic locality. However, for the present problem, we set this field aside and concentrate on the remaining variables.

Transformations to Achieve Normality or Symmetry

Most of the numeric fields are right-skewed. For example, Figure 7.4 shows the distribution of *product uniformity*, a variable that takes large values for customers who purchase only a few different classes of clothes (e.g., blouses, legwear, pants) and

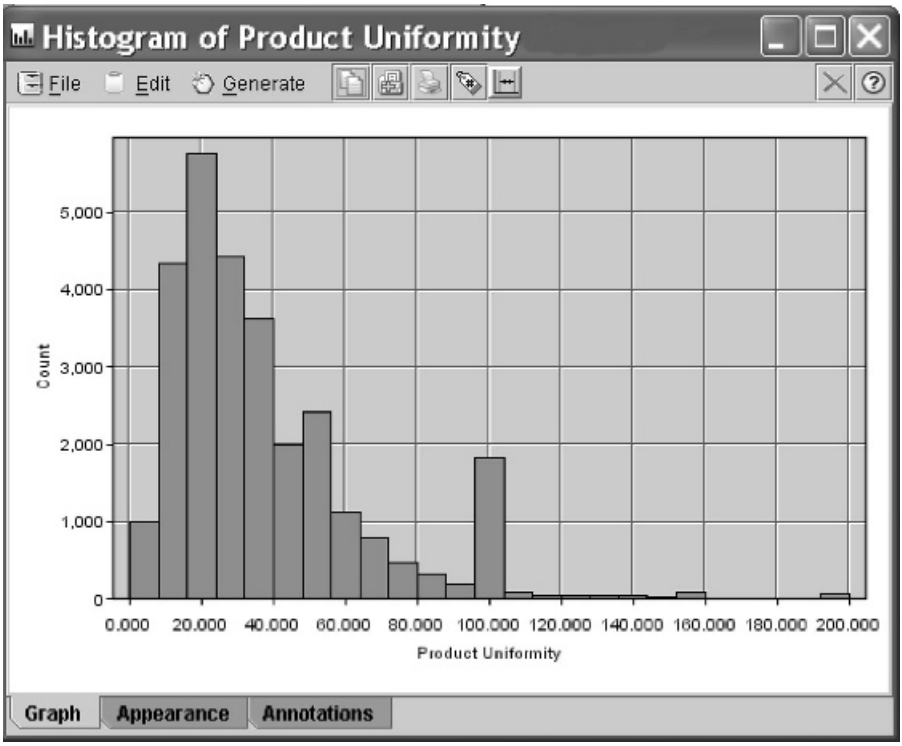


Figure 7.4 Most of the numeric fields are right-skewed, such as product uniformity.

small values for customers who purchase many different classes of clothes. Later we shall see that high product uniformity is associated with low probability of responding to the promotion. Figure 7.4 is right-skewed, with most customers having a relatively low product uniformity measure, while fewer customers have larger values. The customers with large values for product uniformity tend to buy only one or two classes of clothes.

Many data mining methods and models, such as principal components analysis and logistic regression, function best when the variables are normally distributed or, failing that, at least symmetric. Thus, we therefore apply transformations to all of the numerical variables that require it, to induce approximate normality or symmetry. The analyst may choose from the transformations indicated in Chapter 2, such as the *natural log transformation*, the *square root transformation*, a *Box–Cox transformation*, or a *power transformation* from the ladder of re-expressions. For our variables which contained only positive values, we applied the natural log transformation. However, for the variables that contained zero values as well as positive values, we applied the square root transformation, since $\ln(x)$ is undefined for $x = 0$.

Figure 7.5 shows the distribution of product uniformity after the natural log transformation. Although perfect normality is not obtained, the result is nevertheless much less skewed than the raw data distribution, allowing for smoother application of

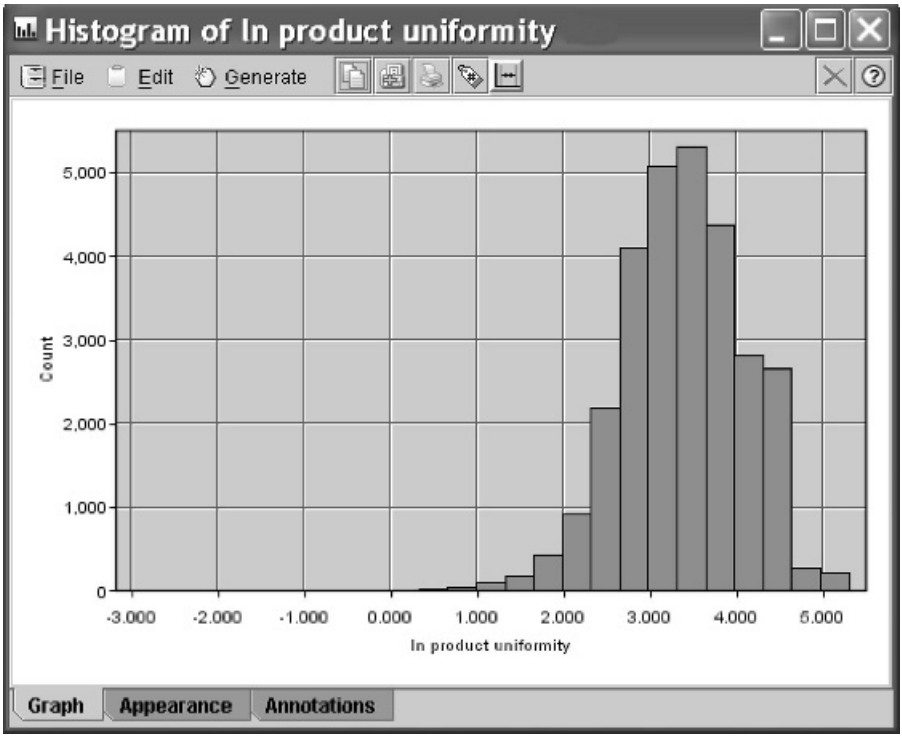


Figure 7.5 Distribution of *ln product uniformity* is less skewed.

several data mining methods and models. Recall that the data set includes 15 variables providing the percentages spent by the customer on specific classes of clothing, including sweaters, knit tops, knit dresses, blouses, and so on. Now, a small percentage (usually, $<1\%$) of records contain percentage values that are negative. It is not clear how percentages can take negative values, or what the meaning of these negative values is, in the context of this problem. Communication with the database analyst or other domain specialist is in order. However, absent that option, we adjust these anomalous values upward to zero dollars. Another option would have been to take the absolute value of these negative amounts, on the assumption that the figures represent returns of earlier purchases.

Figure 7.6 shows the distribution, after adjustment, of the *percentage spent on blouses*. We see a spike at zero, along with the usual right-skewness, which calls for a transformation. The square root transformation is applied, with results shown in Figure 7.7. Note that the spike at zero remains, while the remainder of the data appear nicely symmetric. The dichotomous character of Figure 7.7 motivates us to derive a flag variable for all blouse purchasers. Figure 7.8 shows the distribution of this flag variable, with about 58% of customers having purchased a blouse at one time or another. Flag variables were also constructed for the other 14 clothing percentage variables.

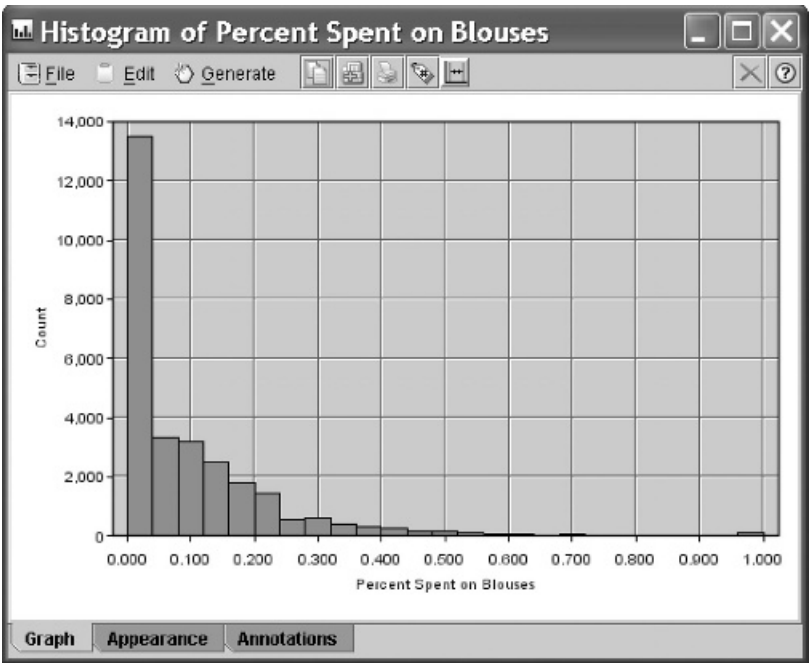


Figure 7.6 Distribution of *percentage spent on blouses*.

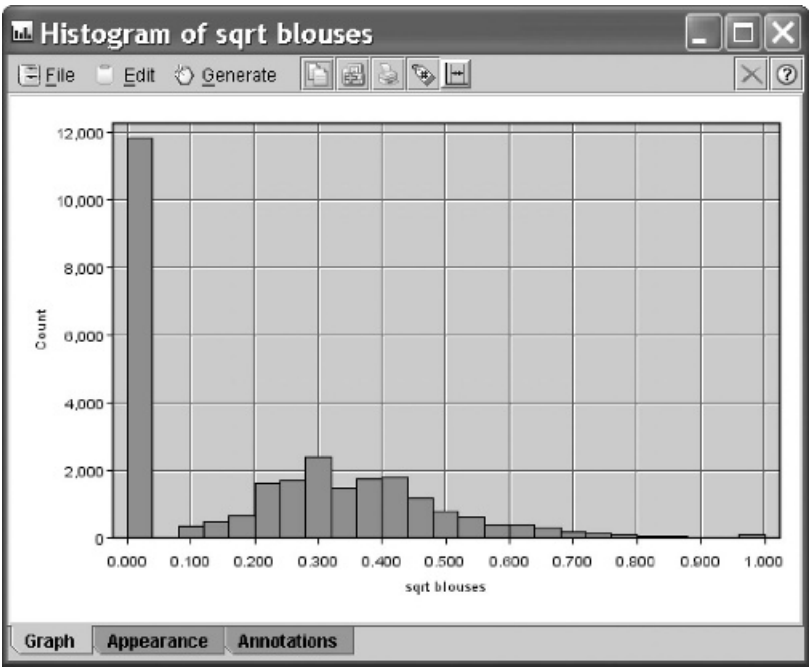


Figure 7.7 Distribution of *sqrt percentage spent on blouses*.

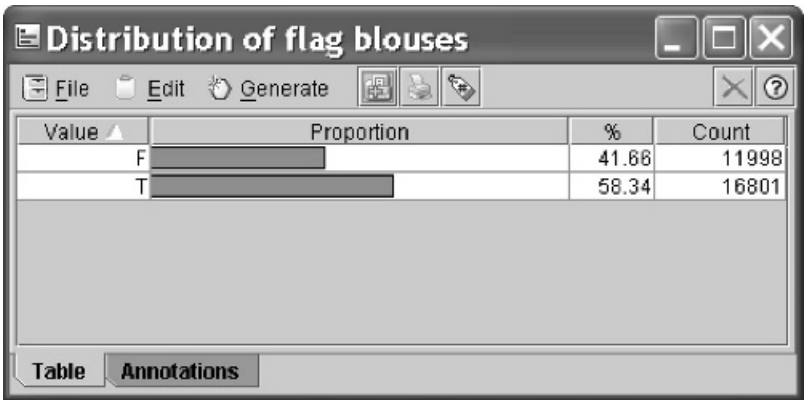


Figure 7.8 Distribution of *blouse purchasers flag variable*.

Standardization and Flag Variables

When there are large differences in variability among the numerical variables, the data analyst needs to apply standardization. The transformations already applied do help in part to reduce the difference in variability among the variables, but substantial differences still exist. For example, the standard deviation for the variable *sqrt spending in the last six months* is 10.02, while the standard deviation for the variable *sqrt # coupons used* is 0.735. To avoid the greater variability of the *sqrt spending in the last six months* variable overwhelming the *sqrt # coupons used* variable, the numeric fields should be normalized or standardized. Here, we choose to standardize the numeric fields, so that they all have a mean of zero and a standard deviation of 1. For each variable, this is done by subtracting the mean of the variable and dividing by the standard deviation, to arrive at the *z-score*. In this analysis, the resulting variable names are prefixed with a "*z*" (e.g., *z sqrt # coupons used*). Other normalization techniques, such as min-max normalization, may be substituted for *z*-score standardization if desired.

Figure 7.9 shows the histogram of the variable *z sqrt spending last one month*. Note the spike that represents the majority of customers who have not spent any money at the store in the past month. For this reason, flag (indicator) variables were constructed for *spending last one month*, as well as the following variables:

- Spending at the AM store (one of the four franchises), to indicate which customers spent money at this particular store
- Spending at the PS store
- Spending at the CC store
- Spending at the AX store
- Spending in the last three months
- Spending in the last six months
- Spending in the same period last year (SPLY)
- Returns, to indicate which customers have ever returned merchandise

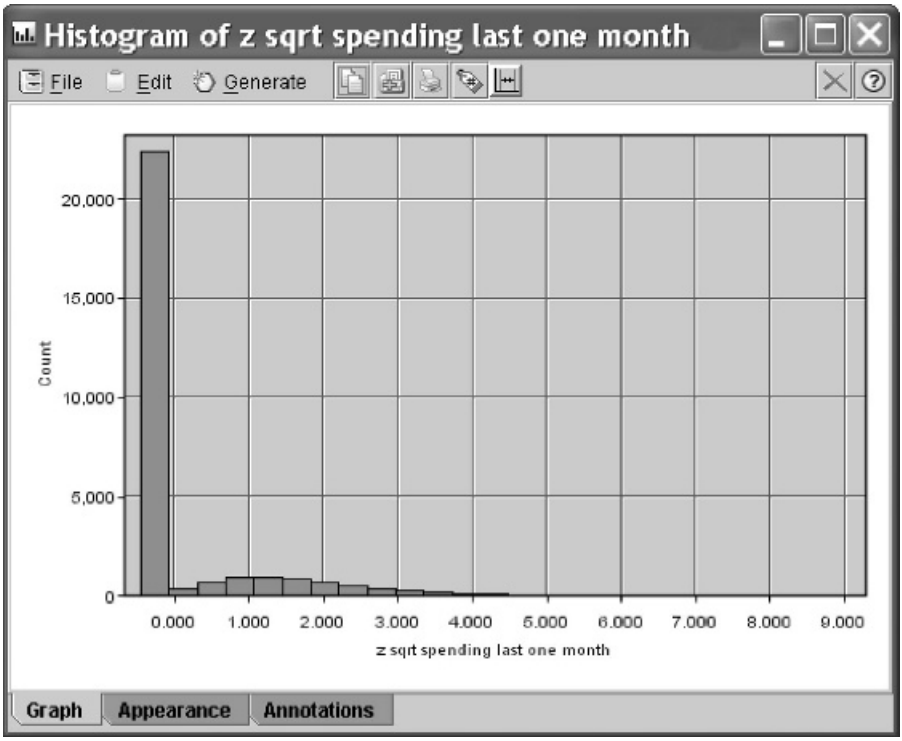


Figure 7.9 Histogram of *z sqrt spending last one month* motivates us to create a flag variable to indicate which customers spent money in the past month.

- Response rate, to indicate which customers have ever responded to a marketing promotion before
- Markdown, to indicate which customers have purchased merchandise that has been marked down

Deriving New Variables

The data preparation phase offers the data miner the opportunity to clarify relationships between variables and to derive new variables that may be useful for the analysis. For example, consider the following three variables: (1) amount spent (by customer) in the last month, (2) amount spent in the last three months, and (3) amount spent in the last six months. Clearly, the amount spent by the customer in the last month is also contained in the other two variables, the amount spent in the last three months and the last six months. Therefore, the amount spent in the last month is getting triple-counted. Now, the analyst may not wish for this most recent amount to be so heavily weighted. For example, in time-series models, the more recent measurements are the most heavily weighted. In this case, however, we prefer not to triple-count the most recent month and must therefore derive two new variables, as shown in Table 7.3.

TABLE 7.3 New Derived Spending Variables

Derived Variable	Formula
Amount spent in previous months 2 and 3	Amount spent in last three months – amount spent in last one month
Amount spent in previous months 4, 5, and 6	Amount spent in last six months – amount spent in last three months

By "amount spent in previous months 2 and 3" we mean the amount spent in the period 90 days to 30 days previous. We shall thus use the following three variables: (1) amount spent in the last month; (2) amount spent in previous months 2 and 3; and (3) amount spent in previous months 4, 5, and 6. We omit the following variables: amount spent in the last three months, and amount spent in the last six months.

Note that even with these derived variables, the most recent month's spending may still be considered to be weighted more heavily than any of the other months' spending. This is because the most recent month's spending has its own variable, while the previous two and three month's spending have to share a variable, as do the previous four, five, and 6 months spending.

The raw data set may have its own derived variables already defined. Consider the following variables: (1) number of purchase visits, (2) total net sales, and (3) average amount spent per visit. The *average amount spent per visit* represents the ratio

$$average = \frac{total\ net\ sales}{number\ of\ purchase\ visits}$$

Since the relationship among these variables is functionally defined, it may turn out that the derived variable is strongly correlated with the other variables. The analyst should check this. Figure 7.10 shows that there is only weak correlation between the derived variable *average* and either of the other variables. On the other hand, the correlation is strong between *total net sales* and *number of purchase visits*. This strong correlation bears watching; we return to this below. By the way, the correlation coefficients between the raw variables should be the same as the correlation coefficients obtained by the z-scores of those variables.

Exploring the Relationships Between the Predictors and the Response

We return to the correlation issue later, but first we would like to investigate the variable-by-variable association between the predictors and the target variable, *response to the marketing promotion*. Ideally, the analyst should examine graphs and statistics for every predictor variable, especially with respect to the relationship with the response. However, the huge data sets prevalent in most data mining applications make this a daunting task. Therefore, we would like to have some way to examine the most useful predictors in an exploratory framework.

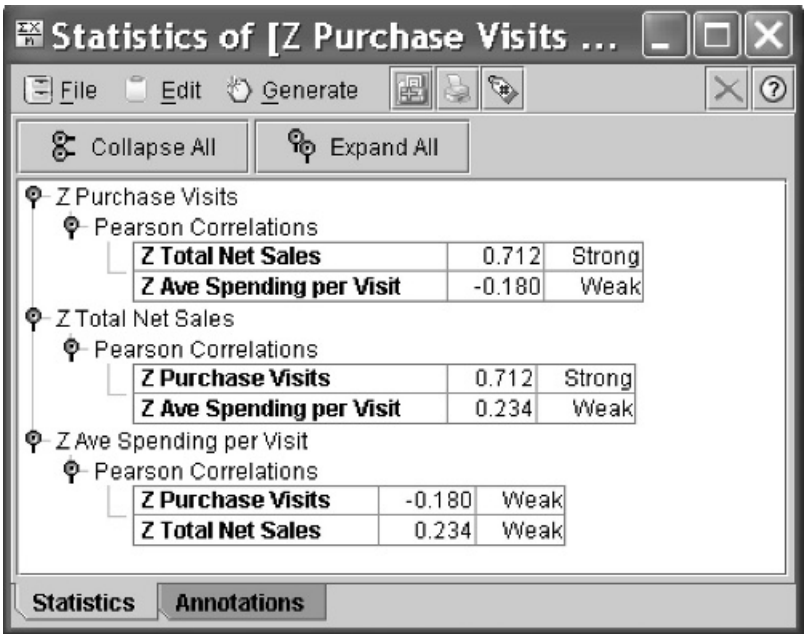


Figure 7.10 Check to make sure that the derived variable is not correlated with the original variables.

Of course, choosing the most useful variables is a modeling task, which lies downstream of our present phase, the EDA-flavored data understanding phase. However, a very rough tool for choosing some useful variables to examine at this early phase is correlation. That is, examine the correlation coefficients for each predictor with the response, and select for further examination those variables that have the largest absolute correlations, say, $|r| \geq 0.30$.

The data miner should, of course, be aware that this is simply a rough EDA tool, and linear correlation with a 0–1 response variable is not appropriate for inference or modeling at this stage. Nevertheless, this method can be useful for paring down the number of variables that would be helpful to examine at the EDA stage. Table 7.4 provides a list of the variables with the highest absolute correlation with the target variable, *response*.

We therefore examine the relationship between these selected predictors and the response variable. First, Figure 7.11 shows a histogram of *z ln lifetime average time between visits*, with an overlay of *response* (0 = no response to the promotion). It appears that records at the upper end of the distribution have lower response rates. To make the interpretation of overlay results more clearly, we turn to a normalized histogram, where each bin has the same height, shown in Figure 7.12.

Figure 7.12 makes it clear that the rate of response to the marketing promotion decreases as the lifetime average time between visits increases. This makes sense, since customers who visit the store more rarely will presumably be less likely to respond to the promotion. For the remaining variables from Table 7.4, we examine

TABLE 7.4 Variables with the Largest Absolute Correlation with the Target Variable, *Response*

Variable	Correlation Coefficient	Relationship
z ln lifetime ave time between visits	−0.431	Negative
z ln purchase visits	0.399	Positive
z ln # individual items purchased	0.368	Positive
z ln total net sales	0.336	Positive
z ln promotions responded in last year	0.333	Positive
z ln # different product classes	0.329	Positive
z ln # coupons used	0.322	Positive
z ln days between purchases	−0.321	Negative

the normalized histogram only, to save space. However, the analyst should not depend on the normalized histograms alone, since these do not display information about the differing densities in the distribution.

Figure 7.13 shows the normalized histograms for the following variables, *z ln purchase visits*, *z ln # individual items purchased*, *z ln total net sales*, and *z ln # different product classes*. All of the relationships show that as the variable increases,

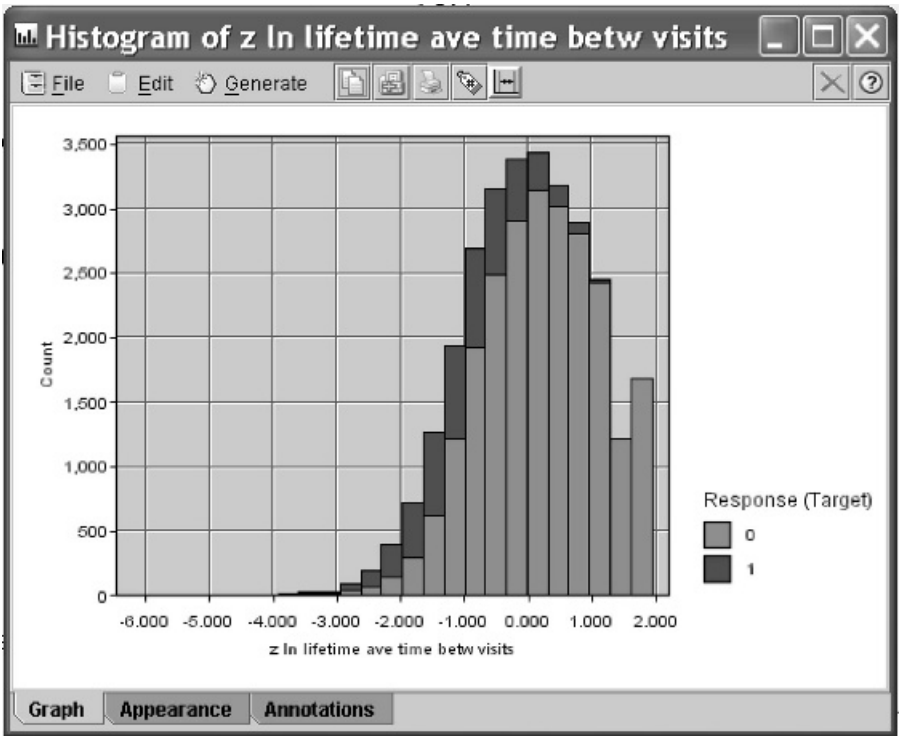


Figure 7.11 Histogram of *z ln lifetime average time between visits* with *response* overlay: may be difficult to interpret.

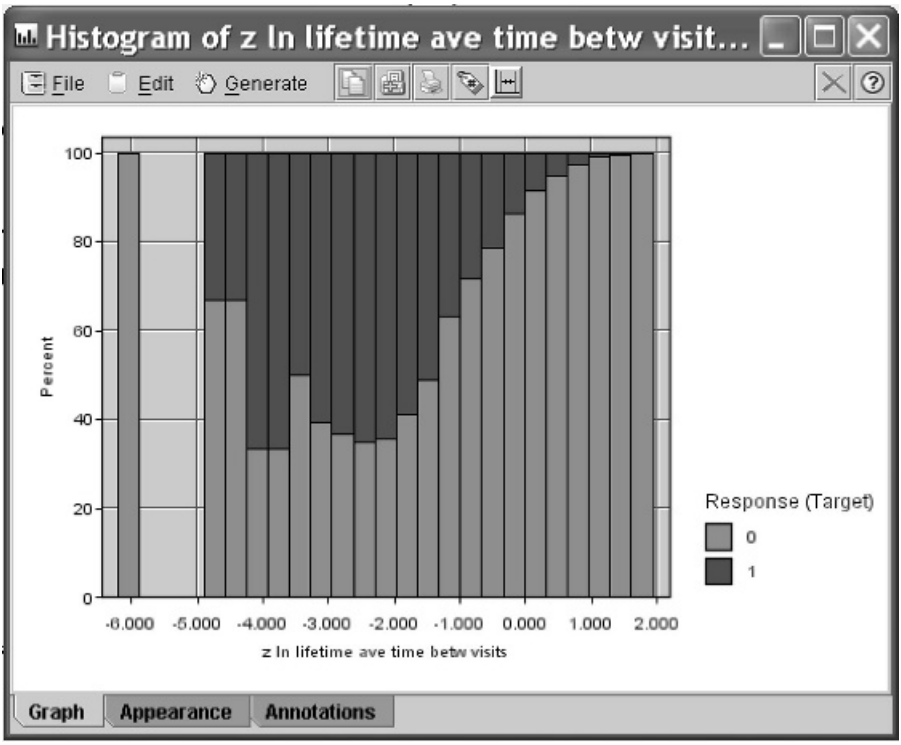


Figure 7.12 Normalized histogram of *z ln lifetime average time between visits* with *response* overlay: easier to discern a pattern.

the response rate increases as well. This is not surprising, since we might anticipate that customers who shop at our stores often, purchase many different items, spend a lot of money, and buy a lot of different types of clothes might be interested in responding to our marketing promotion.

Figure 7.14 shows the relationships between the response variable and the remaining three variables from Table 7.4, *z sqrt responded* (number of promotions responded to in the past year), *z sqrt # coupons used*, and *z ln days between purchases*. We see that the response rate increases as the number of promotions responded to increases, just as it does as the number of coupons used increases. However, the response rate decreases as the number of days between purchases increases. We might expect that the eight variables from Table 7.4 will turn out, in one form or another, to be among the best predictors of promotion response. This is investigated further in the modeling phase.

Next consider Figure 7.15, which shows the normalized version of Figure 7.7, the histogram of *sqrt percentage spent on blouses*, this time with an overlay of the *response* variable. Note from Figure 7.15 that apart from those who spend nothing on blouses (the leftmost bin), as the percentage spent on blouses increases, the response rate *decreases*. This behavior is not restricted to blouses, and is prevalent among all the clothing percentage variables (not shown). What this seems to indicate is that

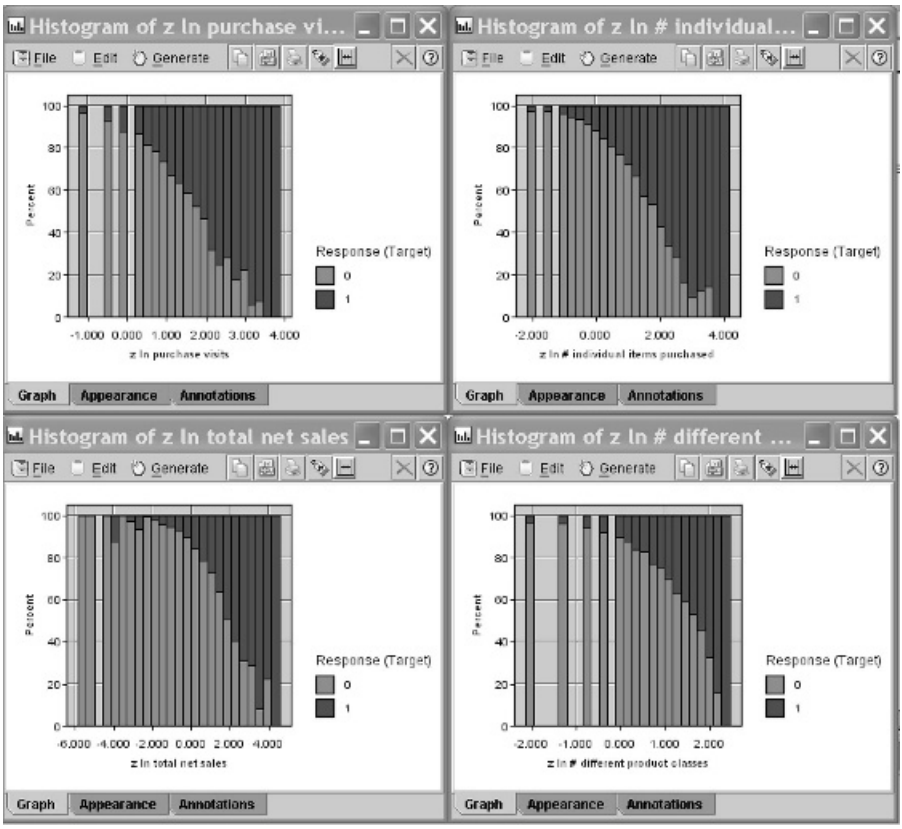


Figure 7.13 The response rate increases as the *z ln number of purchase visits, z ln number of individual items purchased, z ln total net sales*, and *z ln number of different product classes* increase.

customers who concentrate on a particular type of clothing, buying only one or two types of clothing (e.g., blouses), tend to have a lower response rate.

The raw data file contains a variable that measures product uniformity, and based on the behavior observed in Figure 7.15, we would expect the relationship between product uniformity and response to be negative. This is indeed the case, as shown by the normalized histogram in Figure 7.16. The highest response rate is shown by the customers with the lowest uniformity, that is, the highest diversity of purchasing habits, in other words, customers who purchase many different types of clothing.

Next, we turn to an examination of the relationship between the response and the many flag variables in the data set. Figure 7.17 provides a directed web graph of the relationship between the response (upper right) and the following indicator variables (counterclockwise from the response): credit card holder, spending months 4, 5, and 6, spending months 2 and 3, spending last one month, spending same period last year, returns, response rate, markdown, Web buyer, and valid phone number on file. Web graphs are exploratory tools for determining which categorical variables may be of interest for further study.

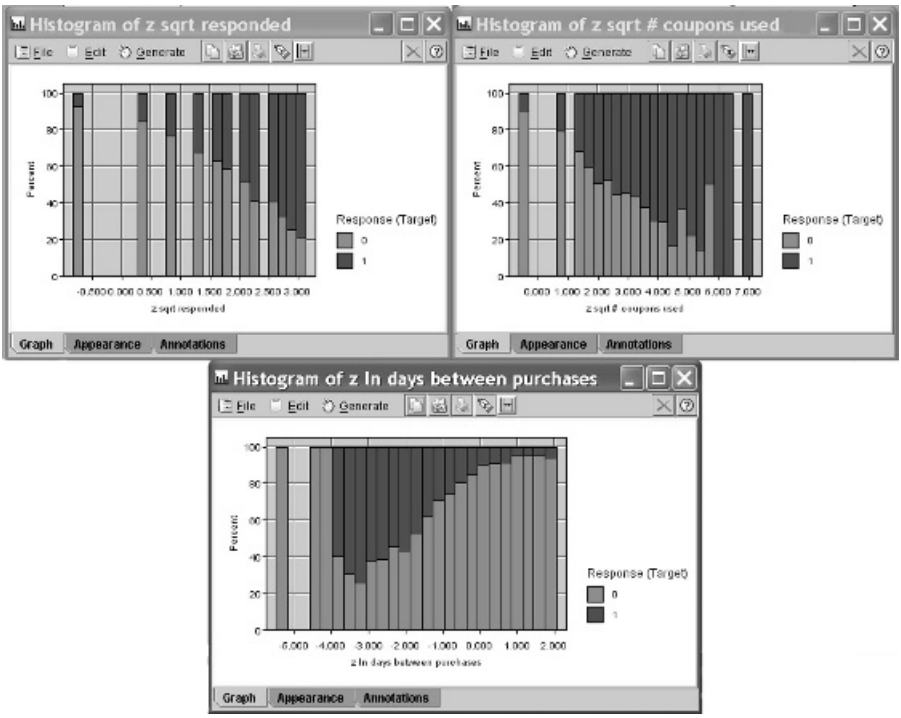


Figure 7.14 The response rate is positively related to the *z sqrt number of promotions responded to*, and the *z sqrt number of coupons used*, but negatively related to the *z ln number of days between purchases.*

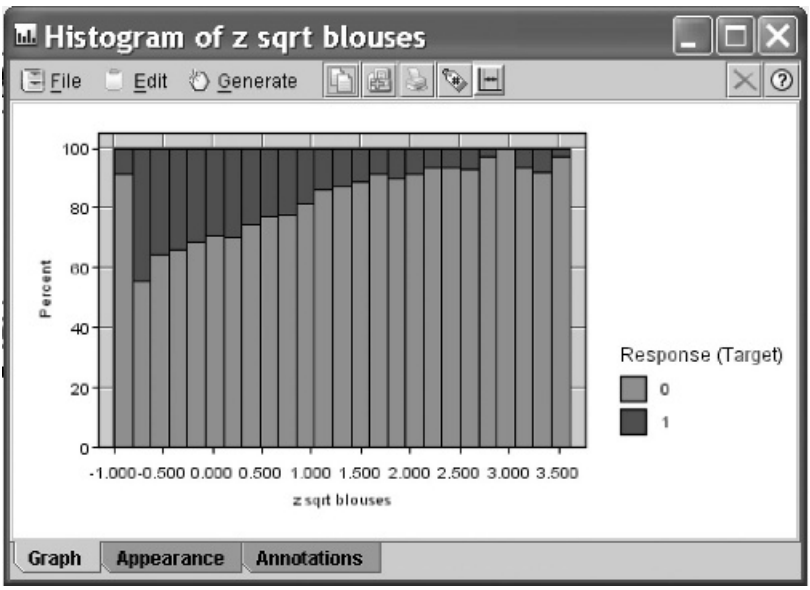


Figure 7.15 *z sqrt percentage spent on blouses*, with *response* overlay.

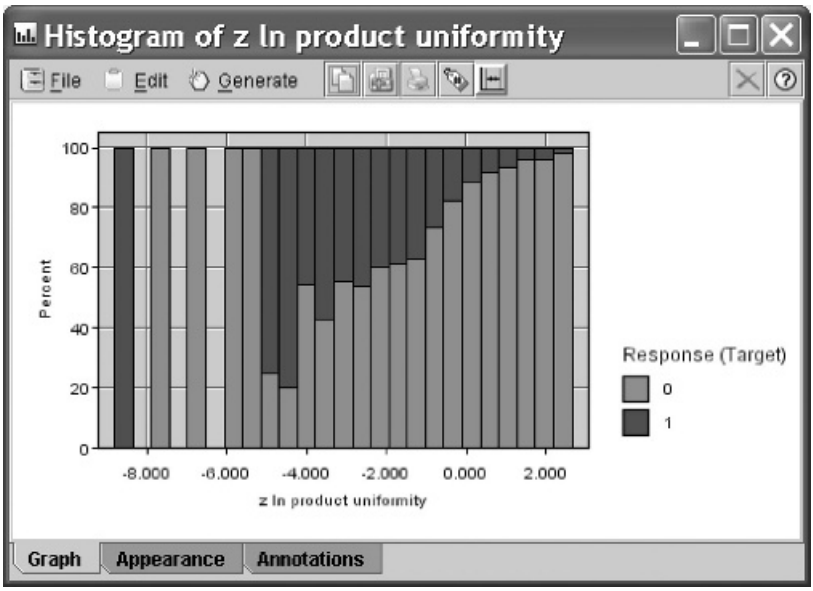


Figure 7.16 As customers concentrate on only one type of clothing, the response rate goes down.

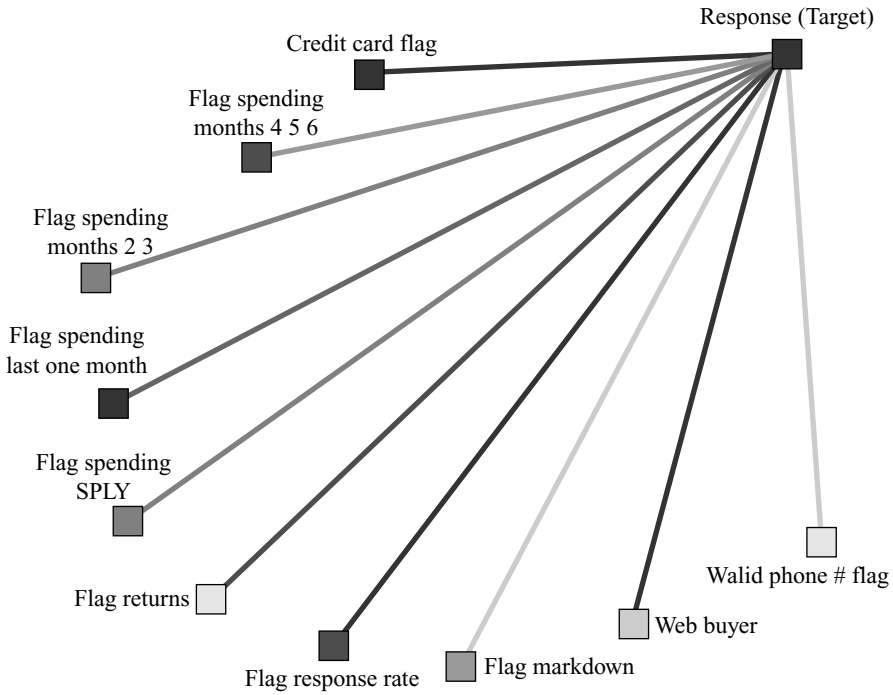


Figure 7.17 Directed web graph of the relationship between the *response* and several flag variables.

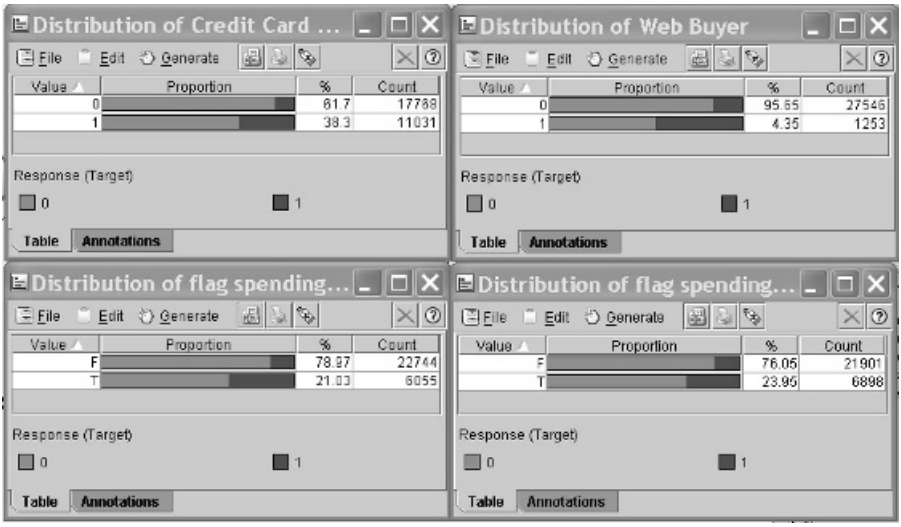


Figure 7.18 Higher response rates are associated with web buyers, credit card holders, customers who made a purchase within the past month (lower left), and customers who made a purchase in the same period last year (lower right).

In this graph, only the true values for the various flags are indicated. The darkness and solidity of the line connecting the flag variable with the response is a measure of the association of that variable with the response. In particular, these connections represent percentages of the *true* predictor flag values associated with the *true* value of the response. Therefore, more solid connections represent a greater association with responding to the promotion. Among the most solid connections in Figure 7.17 are the following: (1) Web buyer, (2) credit card holder, (3) spending last one month, and (4) spending same period last year. We therefore examine the normalized distribution of each of these indicator variables, with the *response* overlay, as shown in Figure 7.18. The counts (and percentages) shown in Figure 7.18 indicate the frequencies (and relative frequencies) of the predictor flag values and do not represent the proportions shown graphically. To examine these proportions, we turn to the set of results matrices (confusion matrices) in Figure 7.19.

Consider the highlighted cells in Figure 7.19, which indicate the proportions of customers who have responded to the promotion, conditioned on their flag values. Credit card holders are about three times as likely as non-credit card holders (28.066% versus 9.376%) to respond to the promotion. Web buyers (those who have made purchases via the company's Web shopping option) are also nearly three times as likely to respond compared to those who have not made a purchase via the Web (44.852% versus 15.247%). Customers who have made a purchase in the last month are nearly three times as likely to respond to the promotion (33.642% versus 11.981%). Finally, those who made a purchase in the same period last year are twice as likely to respond than those who did not make a purchase during the same period last year (27.312% versus 13.141%). We would therefore expect these flag variables to play some nontrivial role in the model-building phase downstream.

Figure 7.19 The statistics in these matrices describe the graphics from Figure 7.18.

Recall Figure 7.3, which showed the 20 most common Microvision lifestyle clusters. What is the relationship between these clusters and the probability of responding to the direct mail marketing promotion? Figure 7.20 shows the normalized distribution of the clusters, with response overlay. Somewhat surprisingly, there do not appear to be any substantial differences in response among the clusters. We return to this result later during the modeling phase.

Investigating the Correlation Structure Among the Predictors

Recall that depending on the objective of our analysis, we should be aware of the dangers of multicollinearity among the predictor variables. We therefore investigate the pairwise correlation coefficients among the predictors and note those correlations that are the strongest. Table 7.5 contains a listing of the pairwise correlations that are the strongest in absolute value among the predictors.

Figure 7.21 shows a scatter plot of *z ln total net sales* versus *z ln number of items purchased*, with a *response* overlay. The strong positive correlation is evident in that as the number of items purchased increases, the total net sales tends to increase.

TABLE 7.5 Strongest Absolute Pairwise Correlations Among the Predictors

Predictor	Predictor	Correlation
z ln purchase visits	*z ln # different product classes*	0.804
z ln purchase visits	*z ln # individual items purchased*	0.860
z ln # promotions on file	*z ln # promotions mailed in last year*	0.890
z ln total net sales	*z ln # different product classes*	0.859
z ln total net sales	*z ln # individual items purchased*	0.907
z ln days between purchase	*z ln lifetime ave time between visits*	0.847
z ln # different product classes	*z ln # individual Items purchased*	0.930

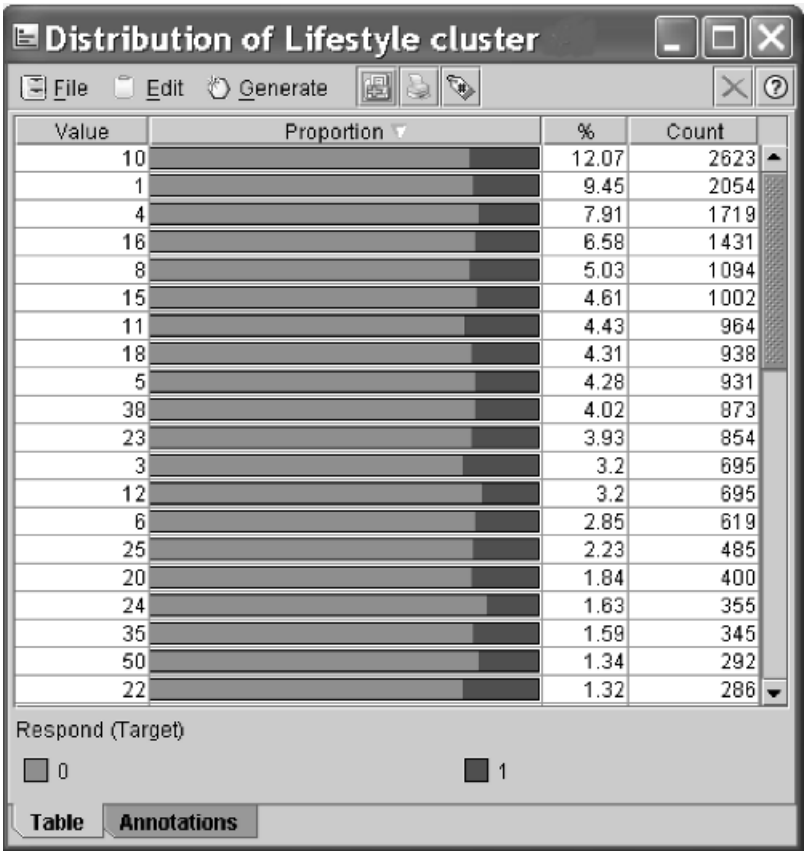


Figure 7.20 There are no substantial differences in promotion response among the 20 most prevalent microvision lifestyle cluster types.

Of course, such a relationship makes sense, since purchasing more items would presumably tend to result in spending more money. Also, at the high end of both variables (the upper right), responders tend to outnumber nonresponders, while at the lower end (the lower left), the opposite is true.

For an example of a negative relationship, we may turn to Figure 7.22, the scatter plot of *z gross margin percentage* versus *z markdown*, with *response* overlay. The correlation between these variables is −0.772, so they did not make the list in Table 7.5. In the scatter plot, it is clear that as markdown increases, the gross margin percentage tends to decrease. Note that the markdown variable seems to have a floor, presumably associated with customers who never buy anything on sale. The relationship with response is less clear in this scatter plot than in Figure 7.21.

A convenient method for examining the relationship between categorical variables and *response* is a cross-tabulation, using a function of the response instead of raw cell counts. For example, suppose that we are interested in the relationship between response to the promotion and two types of customers: those who have purchased sweaters and those who have made a purchase within the last month. Figure 7.23

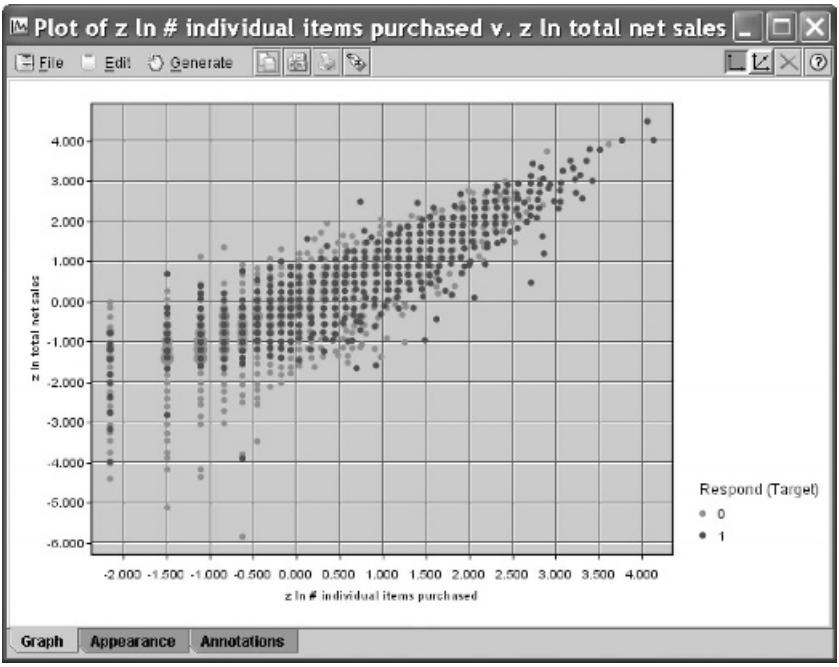


Figure 7.21 Scatter plot of *z ln total net sales* versus *z ln number of items purchased*, with *response* overlay.

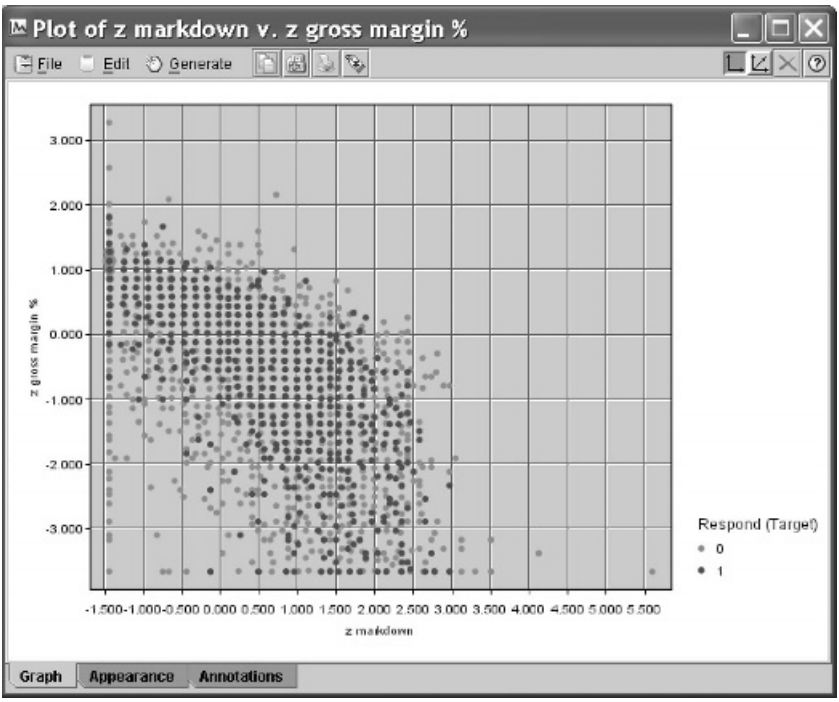


Figure 7.22 Negative relationship between *z gross margin percentage* and *z markdown*.

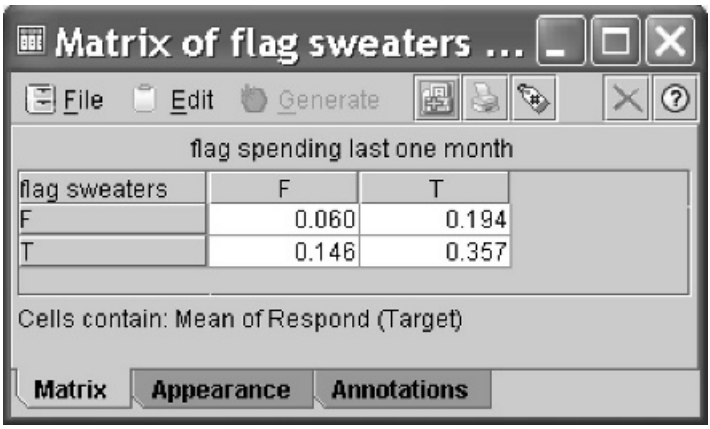


Figure 7.23 Cross-tabulation of spending within the last month versus sweater purchase, with cell values representing promotion response percentages.

contains such a cross-tabulation, with the cells representing the mean value of the target variable (response). Since the target represents a dichotomous variable, the means therefore represent proportions.

Thus, in the cross-tabulation, we see that customers who have neither bought sweaters nor made a purchase in the last month have only a 0.06 probability of responding to the direct-mail marketing promotion. On the other hand, customers who have both bought a sweater and made a purchase in the last month have a 0.357 probability of responding positively to the promotion. If a customer has a true flag value for exactly one of the two predictors, the *spending last one month* variable is slightly more indicative of promotion response than is the sweaters variable (0.194 versus 0.146 probability, respectively).

MODELING AND EVALUATION PHASES

Of course, exploratory data analysis is fun and can provide many useful insights. However, it is time now to move on to the formal modeling stage so that we may bring to bear on our promotion response problem the suite of data mining classification algorithms. An outline of our modeling strategy is as follows:

- Partition the data set into a training data set and a test data set.
- Provide a listing of the inputs to all models.
- Apply principal components analysis to address multicollinearity.
- Apply cluster analysis and briefly profile the resulting clusters.
- Balance the training data set to provide the algorithms with similar numbers of records for responders and nonresponders.
- Establish the baseline model performance in terms of expected profit per customer contacted, in order to calibrate the performance of candidate models.

- Apply the following classification algorithms to the training data set:
  - Classification and regression trees (CARTs)
  - C5.0 decision tree algorithm
  - Neural networks
  - Logistic regression
- Evaluate each of these models using the test data set.
- Apply misclassification costs in line with the cost–benefit table defined in the business understanding phase.
- Apply overbalancing as a surrogate for misclassification costs, and find the most efficacious overbalance mixture.
- Combine the predictions from the four classification models using model voting.
- Compare the performance of models that use principal components with models that do not use the components, and discuss the role of each type of model.

Because our strategy calls for applying many models that need to be evaluated and compared, we hence move fluidly back and forth between the modeling phase and the evaluation phase. First we partition the data set into a training data set and a test data set. Figure 7.24 shows one method of accomplishing this using Clementine 8.5. A new variable is defined, *training test*, which is distributed uniformly between zero and 1. The rectangle attached to the node indicates that the *data cache* has been set; this is necessary so that the same records will be assigned to each partition every time the process is run.

The data miner decides the proportional size of the training and test sets, with typical sizes ranging from 50% training/50% test, to 90% training/10% test. In this case study we choose a partition of approximately 75% training and 25% test. In Clementine this may be done by selecting those records whose *training test* value is at most 0.75 and outputting those records to a file, in this case called Case Study 1 Training Data Set. Similarly, the remaining records are output to the Case Study 1 Test Data Set.

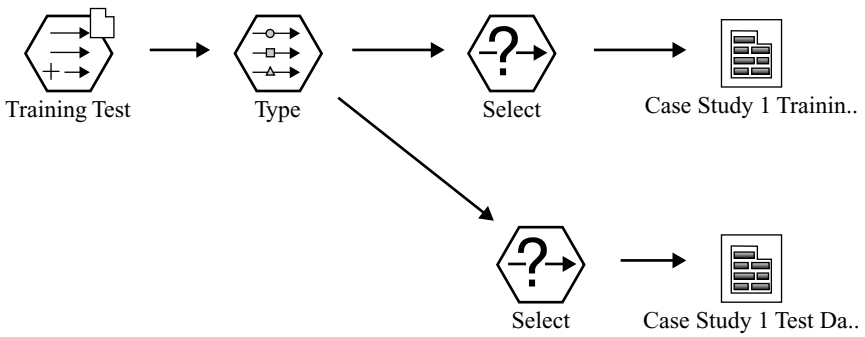


Figure 7.24 Partitioning the data set into a training data set and a test data set.

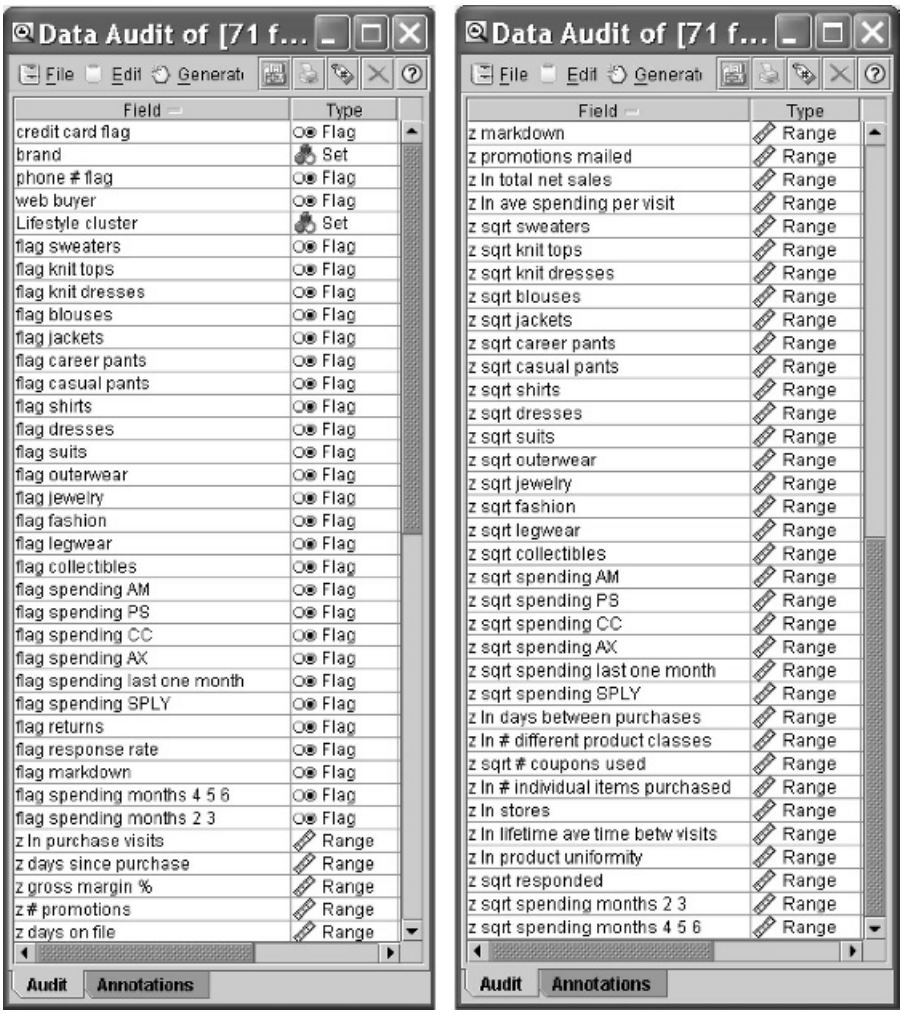


Figure 7.25 Input variables for classification models.

The analyst should always provide the client or end user with a comprehensive listing of the inputs to the models. These inputs should include derived variables, transformed variables, or raw variables, as well as principal components and cluster membership, where appropriate. Figure 7.25 contains a list of all the variables input to the classification models analyzed in this case study.

Note that all of the numeric (range) variables have been both transformed and standardized, that many flag variables have been derived, and that only two nonflag categorical variables remain, *brand*, and *lifestyle cluster*. In fact, only a handful of variables remain untouched by the data preparation phase, including the flag variables *Web buyer* and *credit card holder*. Later, when the principal components and clusters are developed, we shall indicate for which models these are to be used for input.

Principal Components Analysis

Recall Table 7.5, which listed the strongest pairwise correlations found among the predictors. Bear in mind that strongly correlated predictors lead to multicollinearity, as discussed earlier. *Depending on the primary objective of the business or research problem*, the researcher may decide to substitute the principal components for a particular collection of correlated predictors.

- If the primary objective of the business or research problem pertains *solely* to estimation, prediction, or classification of the target variable, with no interest whatsoever in the characteristics of the predictors (e.g., customer profiling), substitution of the principal components for the collection of correlated predictors is not strictly required. As noted in Chapter 3, multicollinearity does not significantly affect point or interval estimates of the target variable.
- However, if the primary (or secondary) objective of the analysis is to assess or interpret the effect of the individual predictors on the response or to develop a profile of likely responders based on their predictor characteristics, substitution of the principal components for the collection of correlated predictors is strongly recommended. Although it does not degrade prediction accuracy, multicollinearity nevertheless plays havoc with the individual predictor coefficients, such as those used in linear or logistic regression.

Therefore, part of our strategy will be to report two types of best models, one (containing no principal components) for use solely in target prediction, and the other (containing principal components) for all other purposes, including customer profiling. We thus proceed to derive the principal components for the collection of correlated variables listed in Table 7.5 using the training data set. The minimum communality was 0.739, indicating that all seven variables share a healthy portion of the common variability. Varimax rotation is used, and two components are extracted, using the eigenvalue criterion. The eigenvalues for these components are 3.9 and 2.2. These two components account for a solid 87% of the variability among the seven variables in Table 7.5. The component loadings are given in Table 7.6. Here follow brief profiles of these components.

- *Principal component 1*: *purchasing habits*. This component consists of the most important customer general purchasing habits. Included here are the total number of items purchased, the number of different types of clothing purchased, the number of different times that customers came to the store to purchase something, and the total amount of money spent by the customer. All of these variables are positively correlated to each other. Conversely, the variable lifetime average time between visits is also included in this component, but it is negatively correlated to the others, since longer times between visits would presumably be negatively correlated with the other purchasing habits. We would expect that this component would be strongly indicative of response to the direct mail marketing promotion.

TABLE 7.6 Component Loadings for the Two Principal Components Extracted from the Training Data Set[a]

	Component	
	1	2
z ln # individual items purchased	0.915	
z ln # different product classes	0.887	
z ln purchase visits	0.858	
z ln lifetime ave time between visits	−0.858	
z ln total net sales	0.833	
z promotions mailed		0.944
z # promotions		0.932

[a] Extaction method: principal component analysis; rotation method: varimax with Kaiser normalization. Rotation converged in three iterations.

- *Principal component 2: promotion contacts.* This component consists solely of two variables, the number of promotions mailed in the past year, and the total number of marketing promotions on file. Note that there is no information in this component about the response of the customer to these promotion contacts. Thus, it is unclear whether this component will be associated with response to the promotion.

As mentioned in Chapter 1, the principal components extracted from the training data set should be validated by comparison with principal components extracted from the test data set. Table 7.7 contains the component loadings for the principal components extracted from the seven correlated variables in Table 7.5, this time using the test data set. Once again, two components are extracted using the eigenvalue criterion and varimax rotation. The eigenvalues for these components are again 3.9 and 2.2. This time, 87.2% of the variability is explained, compared to 87% earlier. A comparison of Tables 7.6 and 7.7 shows that the component loadings, although not identical, are nevertheless sufficiently similar to confirm that the extracted components are valid.

TABLE 7.7 Component Loadings for the Two Principal Components Extracted from the Test Data Set[a]

	Component	
	1	2
z ln # individual items purchased	0.908	
z ln # different product classes	0.878	
z ln lifetime ave time betw visits	−0.867	
z ln purchase visits	0.858	
z ln total net sales	0.828	
z promotions mailed		0.942
z # promotions		0.928

[a] Extaction method: principal component analysis; rotation method: varimax with Kaiser normalization. Rotation converged in three iterations.

Cluster Analysis: BIRCH Clustering Algorithm

Next, we turn to cluster analysis. In *Discovering Knowledge in Data: An Introduction to Data Mining* [2] we demonstrated hierarchical clustering, k-means clustering, and Kohonen clustering. For this case study, however, we shall apply the BIRCH clustering algorithm [5]. The BIRCH algorithm requires only one pass through the data set and therefore represents a scalable solution for very large data sets. The algorithm contains two main steps and hence is termed *two-step clustering* in Clementine. In the first step, the algorithm preclusters the records into a large number of small subclusters by constructing a cluster feature tree. In the second step, the algorithm then combines these subclusters into higher-level clusters, which represent the algorithm's clustering solution.

One benefit of Clementine's implementation of the algorithm is that unlike k-means and Kohonen clustering, the analyst need not prespecify the desired number of clusters. Thus, two-step clustering represents a desirable exploratory tool. For this case study, two-step clustering was applied with no prespecified desired number of clusters. The algorithm returned $k = 3$ clusters. The two main advantages of clustering are (1) exploratory cluster profiling, and (2) the use of the clusters as inputs to downstream classification models.

Figure 7.26 provides an excerpt from Clementine's cluster viewer. Across the top are the clusters, ordered by number of records per cluster, so that cluster 2 (8183 records) comes first, followed by cluster 3 (7891 records) and cluster 1 (5666 records). Down the left side are found variable names, in this case all of which are flags. In each row are found bar charts for that particular variable for each cluster. Since all the variables in Figure 7.26 are flags, the first bar in each bar chart represents 0 (false) and the second bar represents 1 (true).

Note that the bars representing 1 (i.e., a true value for the flag) for cluster 3 are consistently higher than those for clusters 1 or 2. In other words, for every variable listed in Figure 7.26, the proportion of true flag values for cluster 3 is greater than that for the other two clusters. For example, the proportion of customers in cluster 3 who spent money at the AX store is larger than the proportions for the other clusters, and similarly for the other variables in Figure 7.26.

Continuing our exploration of these clusters, Table 7.8 contains the mean values, by cluster, for a select group of numeric variables. Table 7.9 displays the proportion of true flag values, by cluster, for a select group of flag variables. Armed with the information in Figure 7.26, Tables 7.8 and 7.9, and similar information, we now proceed to construct profiles of each cluster.

- *Cluster 1: moderate-spending career shoppers*
  - This cluster has the highest proportion of customers who have ever bought a suit.
  - The proportion who have ever bought career pants is six times higher than cluster 2.
  - The total net sales for this cluster is moderate, lying not far from the overall mean.

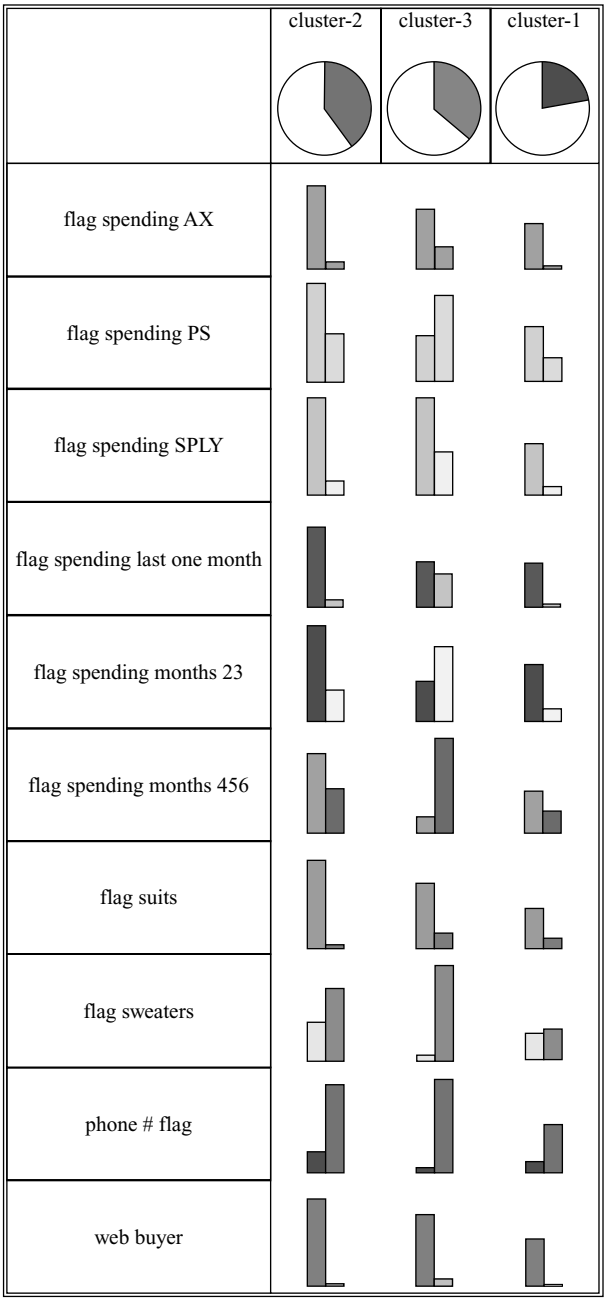


Figure 7.26 Bar charts by cluster for a set of flag variables: cluster 3 appears to be the most promising cluster.

TABLE 7.8 Mean Values by Cluster for a Select Group of Numeric Variables

	Cluster 1	Cluster 2	Cluster 3
z ln Purchase Visits	−0.575	−0.570	1.011
z ln Total Net Sales	−0.177	−0.804	0.971
z sqrt Spending Last One Month	−0.279	−0.314	0.523
z ln Lifetime Average Time Between Visits	0.455	0.484	−0.835
z ln Product Uniformity	0.493	0.447	−0.834
z sqrt # Promotion Responses in Past Year	−0.480	−0.573	0.950
z sqrt Spending on Sweaters	−0.486	0.261	0.116

  - Product uniformity is high, meaning that these shoppers tend to focus on particular types of clothing.
  - The overall shopping habits, however, do not indicate that these are the most loyal customers, since purchase visits and spending last month are low, whereas the time between visits is high. Also, this cluster has not tended to respond to promotions in the past year.
- *Cluster 2: low-spending casual shoppers*
  - This cluster has the lowest total net sales, with a mean nearly one standard deviation below the overall average.
  - Compared to cluster 1 (career clothes shoppers), this cluster tends to shop for more casual wear, having more than double the proportion of casual pants purchases and the highest overall amount spent on sweaters.
  - This cluster is not interested in suits, with only 0.1% ever had bought one.
  - This cluster is similar to cluster 1 in some respects, such as the low numbers of purchase visits, the low spending in the past month, the high product uniformity, the high time between visits, and the low response rate to past promotions.
- *Cluster 3: frequent, high-spending, responsive shoppers*
  - The mean purchase visits and the mean total net sales are each about one standard deviation above the overall average, meaning that this cluster represents frequent shoppers who tend to spend a lot.

TABLE 7.9 Proportion of True Values by Cluster for a Select Group of Flag Variables (%)

	Cluster 1	Cluster 2	Cluster 3
Credit card	27.6	16.7	68.6
Web buyer	1.9	1.2	8.9
Ever bought marked-down merchandise	77.6	81.7	99.8
Ever bought career pants	63.7	9.9	70.7
Ever responded to a promotion	26.2	19.4	88.2
Ever bought a suit	18.7	0.1	18.1
Ever bought casual pants	15.9	34.5	70.5

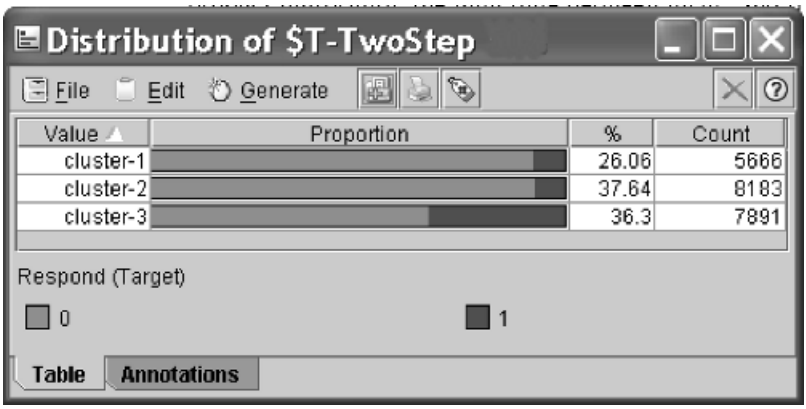


Figure 7.27 Cluster 3 shows a higher rate of response to the marketing promotion.

- These shoppers have low product uniformity, meaning that they are not focusing on any particular type of clothing. For example, they buy both career pants and casual pants in about the same proportions.
- These shoppers are responsive, since nearly 90% of them have responded to a marketing promotion in the past year.
- A majority of these shoppers have a credit card on file, as opposed to the other two clusters.
- This cluster buys online at a rate four times higher than either of the other clusters.

Based on the cluster profiles above, which cluster would you expect to be most responsive to the present direct mail marketing promotion? Clearly, we would expect cluster 3 to have a higher promotion response rate. Figure 7.27 shows the distribution of the clusters with a response overlay; indeed, cluster 3 is the most responsive. Figure 7.28 contains the cross-tabulation of cluster and target response.

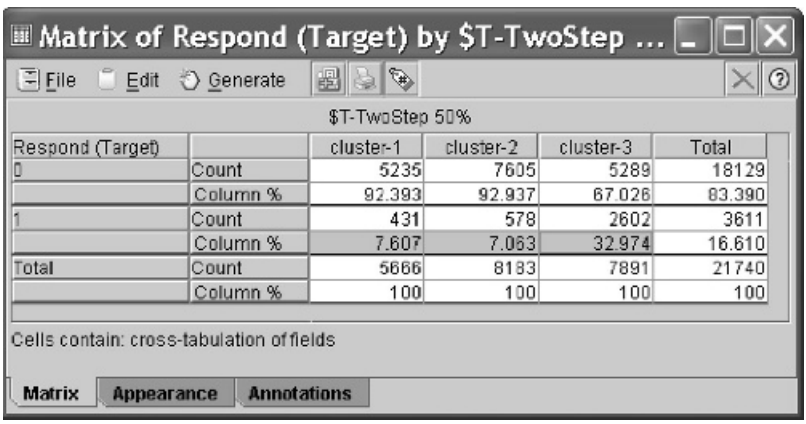


Figure 7.28 Cross-tabulation of cluster (two-step) and response.

Note that the proportion of positive responders to the direct mail marketing promotion is more than four times larger for cluster 3 (32.974%) than for the other clusters (7.607% and 7.063%). Based on this result, we shall include cluster membership as an input to the downstream classification models, and we will not be surprised if cluster membership turns out to play some role in helping to classify potential responders correctly.

The classification models discussed below will contain the following inputs:

- *Model collection A* (includes principal components analysis: models appropriate for customer profiling, variable analysis, or prediction)
  - The 71 variables listed in Figure 7.25, *minus* the seven variables from Table 7.6 used to construct the principal components
  - The two principal components constructed using the variables in Table 7.6
  - The clusters uncovered by the BIRCH two-step algorithm
- *Model collection B* (PCA not included): models to be used for target prediction only)
  - The 71 variables listed in Figure 7.25
  - The clusters uncovered by the BIRCH two-step algorithm.

Balancing the Training Data Set

For classification models in which one of the target variable classes has much lower relative frequency than the other classes, balancing is recommended. For example, suppose that we are running a fraud classification model and our training data set consists of 100,000 transactions, only 1000 of which are fraudulent. Then our classification model could simply predict "nonfraudulent" for all transactions and achieve 99% classification accuracy. However, clearly this model is useless.

Instead, the analyst should balance the training data set so that the relative frequency of fraudulent transactions is increased. It is not recommended that current fraudulent records be cloned to achieve this balance, since this amounts to fabricating data. Rather, a sufficient number of nonfraudulent transactions should be set aside, thereby increasing the proportion of fraudulent transactions. For example, suppose that we wanted our 1000 fraudulent records to represent 25% of the balanced training data set rather than the 1% represented by these records in the raw training data set. That would mean that we could retain only 3000 nonfraudulent records. We would then need to discard from the analysis 96,000 of the 99,000 nonfraudulent records, using random selection. Of course, one always balks at discarding data, but in the data mining world, data are abundant. Also, in most practical applications, the imbalance is not as severe as this 99-to-1 fraud example, so that relatively fewer records need be omitted.

Another benefit of balancing the data is to provide the classification algorithms with a rich balance of records for each classification outcome, so that the algorithms have a chance to learn about all types of records, not just those with high target frequency. In our case study, the training data set contains 18,129 (83.4%) customers who have not responded to the direct mail marketing promotion and 3611 (16.6%)

customers who have responded. Although it is possible to proceed directly with the classification algorithms with this degree of imbalance, it is nevertheless recommended that balancing be applied so that the minority class contain at least 25% of the records, and perhaps at most 50%, depending on the specific problem.

In our case study, we apply balancing to achieve an approximate 50%–50% distribution for the response/nonresponse classes. To do this, we set the Clementine balance node to retain approximately 20% of the nonresponse records, randomly selected. The resulting balance is as follows: 3686 (50.5%) nonresponse records, and all of the 3611 response records (49.5%).

The test data set should never be balanced. The test data set represents new data that the models have not yet seen. Certainly, the real world will not balance tomorrow's data for our classification models; therefore, the test data set itself should not be balanced. Note that all model evaluation will take place using the test data set, so that the evaluative measures will all be applied to unbalanced (real-world-like) data. In other words, tables showing comparative measures of candidate models are obtained using the data found in the test set.

Establishing the Baseline Model Performance

How will we know when our models are performing well? Is 80% classification accuracy good enough? 90%? 95%? To be able to calibrate the performance of our candidate models, we need to establish benchmarks against which these models can be compared. These benchmarks often come in the form of baseline model performance for some simple models. Two of these simple models are (1) the "don't send a marketing promotion to anyone" model, and (2) the "send a marketing promotion to everyone" model.

Clearly, the company does not need to employ data miners to use either of these two models. Therefore, if after arduous analysis, the performance of the models reported by the data miner is lower than the performance of either of the baseline models above, the data miner better try again. In other words, the models reported by the data miner absolutely need to outperform these baseline models, hopefully by a margin large enough to justify the project.

Recall Table 7.2, the cost/benefit decision table for this case study. Applying those costs and benefits to these two baseline models, we obtain for the test data set (5908 negative responses and 1151 positive responses) the performance measures shown in Table 7.10. Which of these two baseline models performs better? A comparison of the overall error rates would lead one to prefer the "send to everyone" model. However, as we discussed earlier, data mining models need to take into account real-world considerations such as costs and benefits, so that traditional model performance measures such as false positive rate, false negative rate, and overall error rate are deemphasized. Instead, we deal directly with the bottom line: What effect would the deployment of this model have on the profitability of the company?

Here, the "don't send to anyone" model is costing the company an estimated $4.63 per customer in lost profits. Of the customers in this data set, 16.3% would have responded positively to the direct mail marketing promotion had they only been given the chance. Therefore, this model must be considered a complete failure and shall

TABLE 7.10 Performance Measures for Two Baseline Models

Model	TN Cost $0	TP Cost −$26.4	FN Cost $28.40	FP Cost $2.00	Overall Error Rate	Overall Cost
Don't send to anyone	5908	0	1151	0	16.3%	$32,688.40 ($4.63 per customer)
Send to everyone	0	1151	0	5908	83.7%	−$18,570.40 (−$2.63 per customer)

no longer be discussed. On the other hand, the "send to everyone" model is actually making money for the company, to the tune of an estimated $2.63 per customer. This "per customer" statistic embraces all customers in the test data set, including nonresponders. The 83.7% error rate is initially shocking until we take into account the low cost of the type of error involved. Therefore, it is this "send to everyone" model that we shall define as our baseline model, and the profit of $2.63 per customer is defined as the benchmark profit that any candidate model should outperform.

Model Collection A: Using the Principal Components

We begin modeling by applying our four main classification model algorithms to the data set using the principal components, and using 50%–50% balancing for the target field *response*. The results are provided in Table 7.11. Note that the percentages indicated in the FN and FP columns represent the false negative rate and the false positive rate, respectively. That is, FP percentage = FP/FP + TP and FN percentage = FN/FN + TN. The logistic regression model outperforms the other three, with a mean estimated profit of $1.68 per customer. However, clearly this is a moot point since none of these models come close to the minimum benchmark of $2.63 profit per customer established by the "send to everyone" model.

TABLE 7.11 Performance Results from Classification Models Using 50%–50% Balancing and Principal Components

Model	TN Cost $0	TP Cost −$26.40	FN Cost $28.40	FP Cost $2.00	Overall Error Rate	Overall Cost per Customer
Neural network	4694	672	479	1214	24.0%	−$0.24
			9.3%	64.4%		
CART	4348	829	322	1560	26.7%	−$1.36
			6.9%	65.3%		
C5.0	4465	782	369	1443	25.7%	−$1.03
			7.6%	64.9%		
Logistic regression	4293	872	279	1615	26.8%	−$1.68
			6.1%	64.9%		

Why are these models performing so poorly? The answer is that we have not applied misclassification costs. To develop candidate models that we will evaluate using a strictly defined cost–benefit matrix, we should seek to embed these costs within the models themselves. In Clementine 8.5, two classification algorithms are equipped with explicit mechanisms for defining asymmetric misclassification costs: C5.0 and CART. Therefore, our next step is to develop decision tree models using misclassification costs in C5.0 and CART. We proceed to define the cost of making a false negative decision to be 28.4 and the cost of making a false positive decision to be 2.0; there is no mechanism for defining the benefit of a true positive to be 26.4, so it is left as 1.0. It should be noted that using these values to define the misclassification costs is equivalent to setting the false negative cost to 14.2 and the false positive cost to 1.0.

Unfortunately, the application of these costs resulted in both the CART model and the C5.0 model classifying all customers as responders (not shown) (i.e., similar to the "send to everyone" model). Evidently, the combination of 50% balancing with these strong misclassification costs made it too expensive for either model to predict negatively. Therefore, the misclassification costs were reduced from the 14.2–1.0 ratio down to a 10.0–1.0 ratio, with the false negative cost equal to 10 and the false positive cost equal to 1. Again, this is equivalent to a false negative cost of 20 and a false positive cost of 2. The resulting performance measures are provided in Table 7.12. Suddenly, with the application of misclassification costs at the model-building stage, the overall profit per customer has jumped by more than a dollar. Both the CART model and the C5.0 model have now outperformed the baseline "send to everyone" model.

Let's take a closer look at these models. Figure 7.29 shows the results from the C5.0 model in Table 7.12. Note the highlighted node. For the 447 records in this node, only 20.8% of them are responders. Yet, as indicated by the "**1**" to the right of the arrow, the model is predicting that the customers in this node are responders. Why is this happening? Because the high false negative misclassification cost makes the model very wary of making negative predictions. This phenomenon helps to illustrate why the C5.0 model with 14.2–1 misclassification costs returned not a single negative prediction.

Also note from Figure 7.29 the dominant role played by the first principal component, *purchasing habits* (Table 7.6), denoted as $*F-PCA-1* in the decision tree.

TABLE 7.12 Performance Results from CART and C5.0 Classification Models Using 10–1 Misclassification Costs

Model	TN Cost $0	TP Cost −$26.40	FN Cost $28.40	FP Cost $2.00	Overall Error Rate	Overall Cost per Customer
CART	754	1147	4	5154	73.1%	−$2.81
			0.5%	81.8%		
C5.0	858	1143	8	5050	71.7%	−$2.81
			0.9%	81.5%		

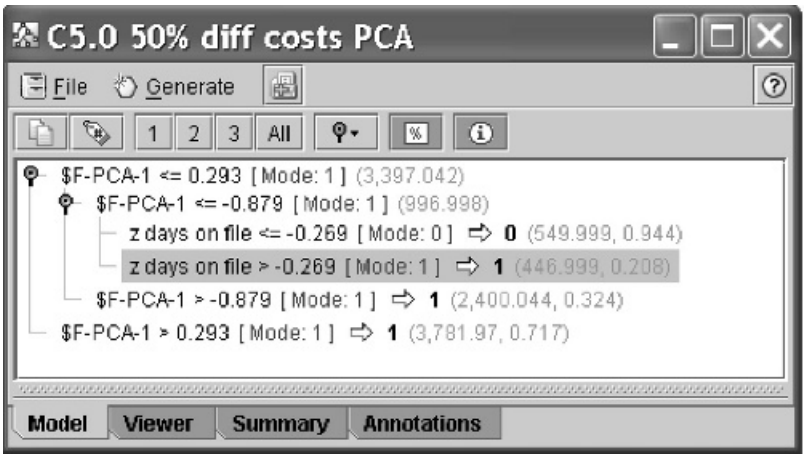


Figure 7.29 C5.0 decision tree using 10–1 misclassification costs.

This first principal component represents both the root node split and the secondary split, indicating that this component is easily the most important factor for predicting response.

We were able to apply misclassification costs for the CART and C5.0 models. But what about the algorithms that don't come with built-in misclassification cost options?

Overbalancing as a Surrogate for Misclassification Costs

Table 7.12 did not contain either a neural network model or a logistic regression model, since Clementine does not have an explicit method for applying misclassification costs for these algorithms. Nevertheless, there is an alternative method for achieving decision effects similar to those provided by the misclassification costs. This alternative method is *overbalancing*.

Table 7.13 contains the performance results for a series of neural network models run, using no principal components, for various levels of balancing. For the first model there is no balancing; for the second model the target variable is balanced 50%–50%; for the third model the target variable is overbalanced, about 65% responders and 35% nonresponders; for the fourth model the target variable is overbalanced, about 80% responders and 20% nonresponders; and for the fifth model the target variable is overbalanced, about 90% responders and 10% nonresponders. Note that the three models that have been overbalanced each outperform the baseline "send to everyone" model, even though none of these models applied misclassification costs directly. Thus, overbalancing, properly applied, may be used as a surrogate for misclassification costs.

The optimal performance by the neural network was obtained using the 80%–20% overbalancing ratio. Let's compare this performance against the other three algorithms using the same ratio. Table 7.14 shows the performance results for all four algorithms, using the 80%–20% overbalancing ratio.

TABLE 7.13 Performance Results from Neural Network Models for Various Levels of Balancing and Overbalancing

Model	TN Cost $0	TP Cost −$26.40	FN Cost $28.40	FP Cost $2.00	Overall Error Rate	Overall Cost per Customer
No balancing	5865	124	1027	43	15.2%	+$3.68
16.3%–83.7%			14.9%	25.7%		
Balancing	4694	672	479	1214	24.0%	−$0.24
50%–50%			9.3%	64.4%		
Overbalancing	1918	1092	59	3990	57.4%	−$2.72
65%–35%			3.0%	78.5%		
80%–20%	1032	1129	22	4876	69.4%	−$2.75
			2.1%	81.2%		
90%–10%	592	1141	10	5316	75.4%	−$2.72
			1.7%	82.3%		

The logistic regression model is the top performer in this group, though all four models outperform the baseline "send to everyone" model. A perusal of the output from the logistic regression model (not shown) shows that the logistic regression model deals with the inclusion of *lifestyle cluster* in the model by using 49 different indicator variables (representing the 50 different values for this single variable). This may be considered overparameterization of the model. If the field was of strong influence on the target response, we might consider keeping the field in the model, probably in binned form. However, because the different lifestyle clusters do not appear to be strongly associated with response or nonresponse, we should consider omitting the variable from the analysis. Retaining the variable to this point has led to an overparameterization of the neural network model; that is, the model has too many nodes for the amount of information represented by the variable. Therefore, we omit the variable and rerun the analysis from Table 7.13, with the results given in Table 7.15.

TABLE 7.14 Performance Results from the Four Algorithms Using the 80%–20% Overbalancing Ratio

Model	TN Cost $0	TP Cost −$26.40	FN Cost $28.40	FP Cost $2.00	Overall Error Rate	Overall Cost per Customer
Neural network	1032	1129	22	4876	69.4%	−$2.75
			2.1%	81.2%		
CART	1724	1111	40	4184	59.8%	−$2.81
			2.3%	79.0%		
C5.0	1195	1127	24	4713	67.1%	−$2.78
			2.0%	80.7%		
Logistic regression	2399	1098	53	3509	50.5%	−$2.90
			2.2%	76.2%		

TABLE 7.15 Performance Results from the Four Algorithms Using the 80%–20% Overbalancing Ratio After Omitting *Lifestyle Cluster*

Model	TN Cost $0	TP Cost −$26.40	FN Cost $28.40	FP Cost $2.00	Overall Error Rate	Overall Cost per Customer
Neural network	885	1132	19	5023	71.4%	−$2.73
			2.1%	81.6%		
CART	1724	1111	40	4184	59.8%	−$2.81
			2.3%	79.0%		
C5.0	1467	1116	35	4441	63.4%	−$2.77
			2.3%	79.9%		
Logistic regression	2389	1106	45	3519	50.5%	−$2.96
			1.8%	76.1%		

Note that the decision to discard the variable is made here in the modeling phase rather than the EDA phase. We should not discard variables in the EDA (data understanding) phase due simply to lack of apparent pairwise relationship with the response. One never knows what relationships exist in higher dimensions, and we should allow the models to decide which models should and should not be retained, as we have done here.

The exclusion of *lifestyle cluster* has improved the performance of the logistic regression model from $2.90 to an estimated $2.96 profit per customer. This $0.06 represents an improvement of 18% over the model that retained the variable, compared to the baseline model. Therefore, in this case, "less is more." On the other hand, the CART model is completely unaffected by the exclusion of *lifestyle cluster*, since the variable did not make it into the model to begin with. The performance of both the C5.0 model and the neural network model degraded slightly with the exclusion of the variable.

However, our best model so far remains the logistic regression model using 80%–20% balancing, no principal components, and excluding the *lifestyle cluster* field. Note that without the application of overbalancing as a surrogate for misclassification costs, we would not have had access to a helpful logistic regression model. This model provides an estimated profit per customer of $2.96, which represents a solid improvement of 45% over the models that applied misclassification costs ($2.81) directly as compared to the baseline benchmark of $2.63 [i.e., ($2.96 − $2.81)/($2.96 − $2.63) = 45%].

Combining Models: Voting

In Olympic figure skating, the champion skater is not decided by a single judge alone but by a panel of judges. The preferences of the individual judges are aggregated using some combination function, which then decides the winner. Data analysts may also be interested in combining classification models, so that the strengths and weaknesses of each model are smoothed out through combination with the other models.

One method of combining models is to use simple voting. For each record, each model supplies a prediction of either response (1) or nonresponse (0). We may then count the votes that each record obtains. For example, we are presently applying four classification algorithms to the promotion response problem. Hence, records may receive from 0 votes up to 4 votes predicting response. In this case, therefore, at the overall level, we may predict a positive response to the promotion based on any one of the following four criteria:

A. Mail a promotion only if all four models predict response.

B. Mail a promotion only if three or four models predict response.

C. Mail a promotion only if at least two models predict response.

D. Mail a promotion if any model predicts response.

Clearly, criterion A would tend to protect against false positives, since all four classification algorithms would have to agree on a positive prediction according to this criterion. Similarly, criterion D would tend to protect against false negatives, since only a single algorithm would need to predict a positive response. Each of these four criteria in effect defines a combination model whose performance may be evaluated, just as for any other model. Hence, Table 7.16 contains the performance results for each of these four combination models. The best combination model is the model defined by criterion B: *Mail a promotion only if three or four models predict response*. This criterion has an intuitive alternative representation: *Mail a promotion only if a majority of the models predict response.*

One disadvantage of using combination models is their lack of easy interpretability. We cannot simply point to a decision rule or p-value to explain why or

TABLE 7.16 Performance Results from Four Methods of Counting the Votes Using the 80%–20% Overbalancing Ratio After Omitting *Lifestyle Cluster*

Combination Model	TN Cost $0	TP Cost −$26.40	FN Cost $28.40	FP Cost $2.00	Overall Error Rate	Overall Cost per Customer
Mail a promotion only if all four models predict response	2772	1067	84 2.9%	3136 74.6%	45.6%	−$2.76
Mail a promotion only if three or four models predict response	1936	1115	36 1.8%	3972 78.1%	56.8%	−$2.90
Mail a promotion only if at least two models predict response	1207	1135	16 1.3%	4701 80.6%	66.8%	−$2.85
Mail a promotion if any model predicts response	550	1148	3 0.5%	5358 82.4%	75.9%	−$2.76

TABLE 7.17 Most Important Variables/Components and Their *p*-Values

Variable or Component	*p*-Value
Principal component 1: purchasing habits	0.000
Principal component 2: promotion contacts	0.000
z days on file	0.000
z ln average spending per visit	0.000
z ln days between purchases	0.000
z ln product uniformity	0.000
z sqrt spending CC	0.000
Web buyer flag	0.000
z sqrt knit dresses	0.001
z sqrt sweaters	0.001
z in stores	0.003
z sqrt career pants	0.004
z sqrt spending PS	0.005

why not a particular customer received a promotion. Recall that the most easily interpreted classification models are the decision trees, such as those produced by the CART or C5.0 algorithms. In our case, however, our best model was produced by logistic regression, which, for interpretability, lies midway between the decision trees and neural networks. Let us therefore take a closer look at this logistic regression model. Table 7.17 contains a list of the most important variables and components reported by the logistic regression model along with their *p*-values.

Much more modeling work could be done here; after all, most models are usually considered works in progress and few models are ever considered complete. Thus, in the interests of brevity, we move on to the other class of models that awaits us: the non-PCA models.

Model Collection B: Non-PCA Models

Finally, we examine the models that do not include the principal components. Instead, these models retain the set of correlated variables shown in Table 7.5, and thus should not be used for any purpose except prediction of the target variable, promotion response. On the other hand, since the set of correlated variables is highly predictive of the response, we would expect the non-PCA models to outperform the PCA models in terms of response prediction.

Our strategy in this section will mirror our work with the PCA models, with one special addition:

1. Apply CART and C5.0 models, using misclassification costs and 50% balancing.
2. Apply all four classification algorithms, using 80% overbalancing.
3. Combine the four classification algorithms, using voting.
4. Combine the four classification algorithms, using the *mean response probabilities*.

TABLE 7.18 Performance Results from CART and C5.0 Classification Models Using 14.2–1 Misclassification Costs

Model	TN Cost $0	TP Cost −$26.40	FN Cost $28.40	FP Cost $2.00	Overall Error Rate	Overall Cost per Customer
CART	1645	1140	11	4263	60.5%	−$3.01
			0.7%	78.9%		
C5.0	1562	1147	4	4346	61.6%	−$3.04
			0.3%	79.1%		

We begin by applying the decision trees algorithms, CART and C5.0, using 14.2–1 misclassification costs, and 50%–50% balancing. The results are provided in Table 7.18. Note that both models have already outperformed the best of the PCA models, with an estimated profit per customer of $3.04 and $3.01, compared to $2.96 for the logistic regression PCA model. Suppose, however, that we wished to enrich our pool of algorithms to include those without built-in misclassification costs. Then we can apply overbalancing as a surrogate for misclassification costs, just as we did for the PCA models. Table 7.19 contains the performance results from all four algorithms, using 80% overbalancing.

Note the wide disparity in model performance. Here, C5.0 is the winner, with a solid estimated profit of $3.15, representing the best overall prediction performance by a single model in this case study. The logistic regression model is not far behind, at $3.12. The neural network model, however, performs relatively poorly, at only $2.78. (It should be noted here that all neural network models run in this case study used Clementine's default settings and the *quick* option. Perhaps the neural network performance could be enhanced by tweaking the many settings and options available.)

Next, we combine the four models, first through the use of voting. Table 7.20 provides the performance metrics from the four methods of counting the votes,

TABLE 7.19 Performance Results from the Four Algorithms Using the 80%–20% Overbalancing Ratio

Model	TN Cost $0	TP Cost −$26.40	FN Cost $28.40	FP Cost $2.00	Overall Error Rate	Overall Cost per Customer
Neural network	1301	1123	28	4607	65.7%	−$2.78
			2.1%	80.4%		
CART	2780	1100	51	3128	45.0%	−$3.02
			1.8%	74.0%		
C5.0	2640	1121	30	3268	46.7%	−$3.15
			1.1%	74.5%		
Logistic regression	2853	1110	41	3055	43.9%	−$3.12
			1.4%	73.3%		

TABLE 7.20 Performance Results from Four Methods of Counting the Votes Using the 80%–20% Overbalancing Ratio for Non-PCA Models

Combination Model	TN Cost $0	TP Cost −$26.40	FN Cost $28.40	FP Cost $2.00	Overall Error Rate	Overall Cost per Customer
Mail a promotion only if all four models predict response	3307	1065	86 2.5%	2601 70.9%	38.1%	−$2.90
Mail a promotion only if three or four models predict response	2835	1111	40 1.4%	3073 73.4%	44.1%	−$3.12
Mail a promotion only if at least two models predict response	2357	1133	18 0.7%	3551 75.8%	50.6%	−$3.16
Mail a promotion if any model predicts response	1075	1145	6 0.6%	4833 80.8%	68.6%	−$2.89

where once again we use 80% overbalancing. The results from the combined models may be a bit surprising, since one combination method, mailing a promotion only if at least two models predict response, has outperformed all of the individual classification models, with a mean overall profit per customer of about $3.16. This represents the *synergy* of the combination model approach, where the combination of the models is in a sense greater than the sum of its parts. Here, the greatest profit is obtained when at least two models agree on sending a promotion to a potential recipient. The voting method of combining models has provided us with better results than we could have obtained from any of the individual models.

Combining Models Using the Mean Response Probabilities

Voting is not the only method for combining model results. The voting method represents, for each model, an up-or-down, black-and-white decision without regard for measuring the confidence in the decision. It would be nice if we could somehow combine the confidences that each model reports for its decisions, since such a method would allow finer tuning of the decision space.

Fortunately, such confidence measures are available in Clementine, with a bit of derivation. For each model's results Clementine reports not only the decision, but also a continuous field that is related to the confidence of the algorithm in its decision. When we use this continuous field, we derive a new variable that measures for each record the probability that this particular customer will respond positively to

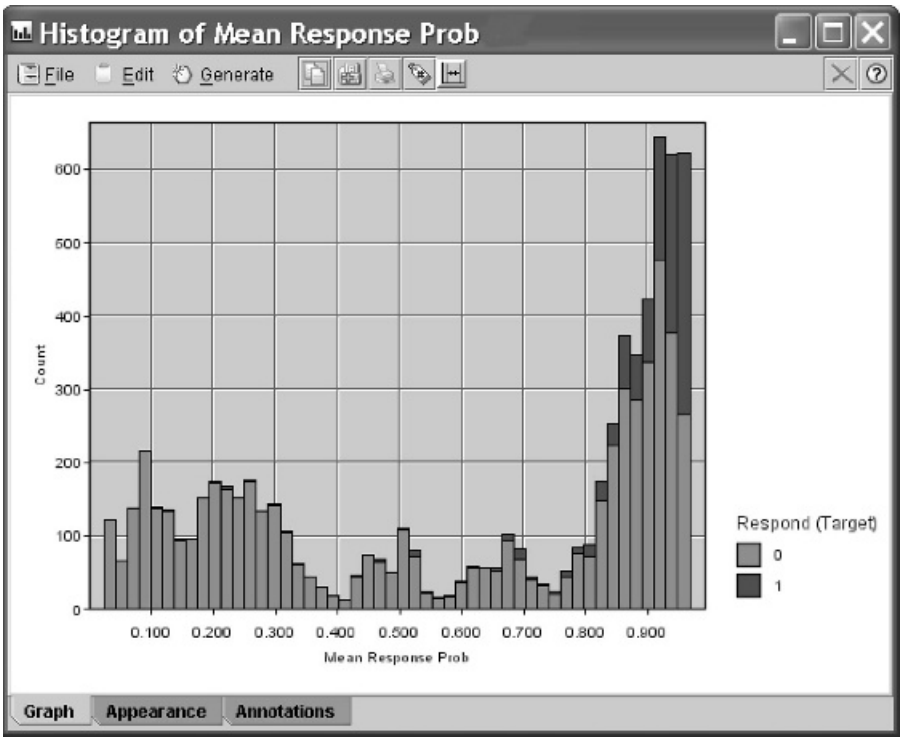


Figure 7.30 Distribution of *mean response probability*, with *response* overlay.

the promotion. This derivation is as follows:

If prediction = positive, then response probability = 0.5 + (confidence reported)/2
If prediction = negative, then response probability = 0.5 − (confidence reported)/2

For each model, the model response probabilities (MRPs) were calculated using this formula. Then the mean MRP was found by dividing the sum of the MRPs by 4. Figure 7.30 contains a histogram of the MRP with a promotion response overlay.

The multimodality of the distribution of MRP is due to the discontinuity of the transformation used in its derivation. To increase the contrast between responders and nonresponders, it is helpful to produce a normalized histogram with increased granularity, to enable finer tuning, obtained by increasing the number of bins. This normalized histogram is shown in Figure 7.31.

Next, based on this normalized histogram, the analyst may define bands that partition the data set according to various values of MRP. Recalling that the false negative error is 14.2 times worse than the false positive error, we should tend to set these partitions on the low side, so that fewer false negative decisions are made. For example, based on a perusal of Figure 7.31, we might be tempted to partition the records according to the criterion: MRP < 0.85 versus MRP ≥ 0.85, since it is near that value that the proportion of positive respondents begins to increase rapidly.

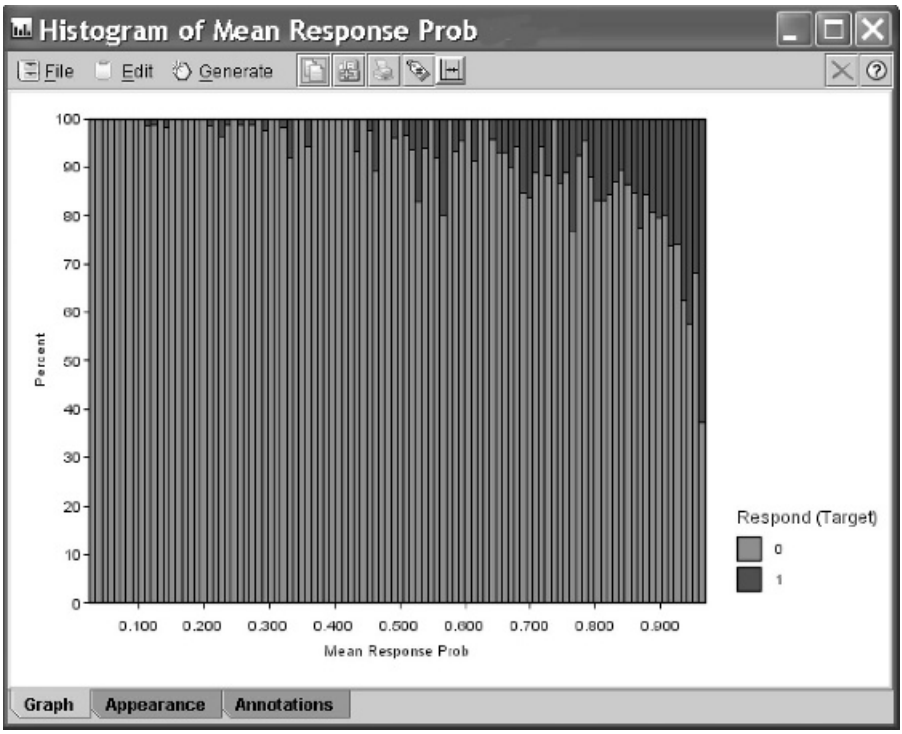


Figure 7.31 Normalized histogram of *mean response probability*, with *response* overlay showing finer granularity.

However, as shown in Table 7.21, the model based on such a partition is suboptimal since it allows so many false positives. As it turns out, the optimal partition is at or near 50% probability. In other words, suppose that we mail a promotion to a prospective customer under the following conditions:

- *Continuous combination model.* Mail a promotion only if the mean response probability reported by the four algorithms is at least 51%.

In other words, this continuous combination model will mail a promotion only if the mean probability of response reported by the four classification models is greater than half. This turns then out to be the optimal model uncovered by any of our methods in this case study, with an estimated profit per customer of $3.1744 (the extra decimal points help to discriminate small differences among the leading candidate models). Table 7.21 contains the performance metrics obtained by models defined by candidate partitions for various values of MRP. Note the minute differences in overall cost among several different candidate partitions. To avoid overfitting, the analyst may decide not to set in stone the winning partition value, but to retain the two or three leading candidates.

Thus, the continuous combination model defined on the partition at MRP $=$ 0.51 is our overall best model for predicting response to the direct mail marketing

TABLE 7.21 Performance Metrics for Models Defined by Partitions for Various Values of MRP

Combination Model	TN Cost $0	TP Cost −$26.40	FN Cost $28.40	FP Cost $2.00	Overall Error Rate	Overall Cost per Customer
Partition : { MRP < 0.95 / MRP ≥ 0.95 }	5648	353	798	260	15.0%	+$1.96
			12.4%	42.4%		
Partition : { MRP < 0.85 / MRP ≥ 0.85 }	3810	994	157	2098	31.9%	−$2.49
			4.0%	67.8%		
Partition : { MRP < 0.65 / MRP ≥ 0.65 }	2995	1104	47	2913	41.9%	−$3.11
			1.5%	72.5%		
Partition : { MRP < 0.54 / MRP ≥ 0.54 }	2796	1113	38	3112	44.6%	−$3.13
			1.3%	73.7%		
Partition : { MRP < 0.52 / MRP ≥ 0.52 }	2738	1121	30	3170	45.3%	−$3.1736
			1.1%	73.9%		
Partition : { MRP < 0.51 / MRP ≥ 0.51 }	2686	1123	28	3222	46.0%	−$3.1744
			1.0%	74.2%		
Partition : { MRP < 0.50 / MRP ≥ 0.50 }	2625	1125	26	3283	46.9%	−$3.1726
			1.0%	74.5%		
Partition : { MRP < 0.46 / MRP ≥ 0.46 }	2493	1129	22	3415	48.7%	−$3.166
			0.9%	75.2%		
Partition : { MRP < 0.42 / MRP ≥ 0.42 }	2369	1133	18	3539	50.4%	−$3.162
			0.8%	75.7%		

promotion. This model provides an estimated $3.1744 in profit to the company for every promotion mailed out. This is compared with the baseline performance, from the "send to everyone" model, of $2.63 per mailing. Thus, our model enhances the profitability of this direct mail marketing campaign by 20.7%, or 54.44 cents per customer. For example, if a mailing was to be made to 100,000 customers, the estimated increase in profits is $54,440. This increase in profits is due to the decrease in costs associated with mailing promotions to nonresponsive customers.

To illustrate, consider Figure 7.32, which presents a graph of the profits obtained by using the C5.0 model alone (not in combination). The darker line indicates the profits from the C5.0 model, after the records have been sorted, so that the most likely responders are first. The lighter line indicates the best possible model, which has perfect knowledge of who is and who isn't a responder. Note that the lighter line rises linearly to its maximum near the 16th percentile, since about 16% of the test data set records are positive responders; it then falls away linearly but more slowly as the costs of the remaining nonresponding 84% of the data set are incurred.

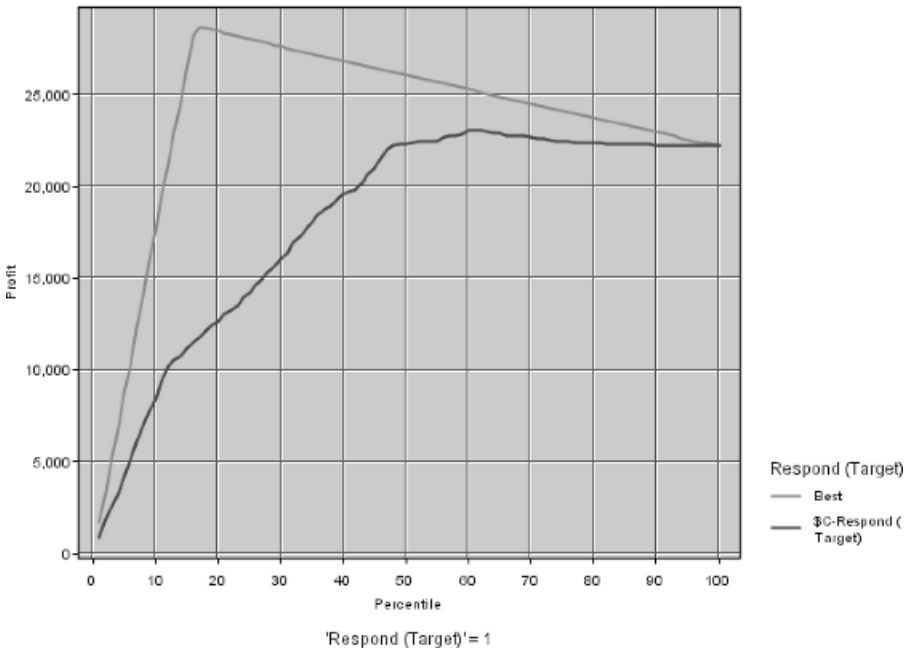


Figure 7.32 Profits graph for the C5.0 model.

On the other hand, the C5.0 model profit curve reaches a plateau near the 50th percentile. That is, the profit curve is, in general, no higher at the 99th percentile than it is near the 50th percentile. *This phenomenon illustrates the futility of the "send to everyone" model, since the same level of profits can be obtained by contacting merely half the prospective customers as would be obtained by contacting them all.*

Since the profit graph is based on the records sorted as to likelihood of response, it is in a sense therefore related to the continuous combination model above, which also sorted the records by likelihood of response according to each of the four models. Note that there is a "change point" near the 50th percentile in both the profit graph and the continuous combination model.

SUMMARY

The case study in this chapter, Modeling Response to Direct Mail Marketing, was carried out using the Cross-Industry Standard Process for Data Mining (CRISP–DM). This process consists of six phases: (1) the business understanding phase, (2) the data understanding phase, (3) the data preparation phase, (4) the modeling phase, (5) the evaluation phase, and (6) the deployment phase.

In this case study, our task was to predict which customers were most likely to respond to a direct mail marketing promotion. The *clothing-store* data set [3], located at the book series Web site, represents actual data provided by a clothing store chain

in New England. Data were collected on 51 fields for 28,799 customers. The objective of the classification model was to increase profits. A cost/benefit decision table was constructed, with false negatives penalized much more than false positives.

Most of the numeric fields were right-skewed and required a transformation to achieve normality or symmetry. After transformation, these numeric fields were standardized. Flag variables were derived for many of the clothing purchase variables. To flesh out the data set, new variables were derived based on other variables already in the data set.

EDA indicated that response to the marketing campaign was associated positively with the following variables, among others: *z ln purchase visits*, *z ln number of individual items purchase*, *z ln total net sales*, and *z ln promotions responded to in the last year*. *Response* was negatively correlated with *z ln lifetime average time between visits*. An interesting phenomenon uncovered at the EDA stage was the following: As customers concentrate on only one type of clothing purchase, the response rate goes down.

Strong pairwise associations were found among several predictors, with the strongest correlation between *z ln number of different product classes* and *z ln number of individual items purchased*.

The modeling and evaluation phases were combined and implemented using the following strategy:

- Partition the data set into a training data set and a test data set.
- Provide a listing of the inputs to all models.
- Apply principal components analysis to address multicollinearity.
- Apply cluster analysis and briefly profile the resulting clusters.
- Balance the training data set to provide the algorithms with similar numbers of records for responders and nonresponders.
- Establish the baseline model performance in terms of expected profit per customer contacted, in order to calibrate the performance of candidate models.
- Apply the following classification algorithms to the training data set:
  - Classification and regression trees (CARTs)
  - C5.0 decision tree algorithm
  - Neural networks
  - Logistic regression
- Evaluate each of these models using the test data set.
- Apply misclassification costs in line with the cost/benefit table defined in the business understanding phase.
- Apply overbalancing as a surrogate for misclassification costs, and find the most efficacious overbalance mixture.
- Combine the predictions from the four classification models using model voting.
- Compare the performance of models that use principal components with models that do not use the components, and discuss the role of each type of model.

Part of our strategy was to report two types of best models, one (containing no principal components) for use solely in target prediction, and the other (containing principal components) for all other purposes, including customer profiling. The subset of variables that were highly correlated with each other were shunted to a principal components analysis, which extracted two components from these seven correlated variables. *Principal component 1* represented purchasing habits and was expected to be highly indicative of promotion response.

Next, the BIRCH clustering algorithm was applied. Three clusters were uncovered: (1) moderate-spending career shoppers, (2) low-spending casual shoppers, and (3) frequent, high-spending, responsive shoppers. Cluster 3, as expected, had the highest promotion response rate.

Thus, the classification models contained the following inputs:

- *Model collection A* (included principal components analysis: models appropriate for customer profiling, variable analysis, or prediction
  - The 71 variables listed in Figure 7.25, *minus* the seven variables from Table 7.6 used to construct the principal components
  - The two principal components constructed using the variables in Table 7.6
  - The clusters uncovered by the BIRCH two-step algorithm
- *Model collection B* (PCA not included): models to be used for target prediction only
  - The 71 variables listed in Figure 7.25
  - The clusters uncovered by the BIRCH two-step algorithm

To be able to calibrate the performance of our candidate models, we established benchmark performance using two simple models:

- The "don't send a marketing promotion to anyone" model
- The "send a marketing promotion to everyone" model

Instead of using the overall error rate as the measure of model performance, the models were evaluated using the measure of overall cost derived from the cost–benefit decision table. The baseline overall cost for the "send a marketing promotion to everyone" model worked out to be –$2.63 per customer (i.e., negative cost = profit).

We began with the PCA models. Using 50% balancing and no misclassification costs, none of our classification models were able to outperform this baseline model. However, after applying 10–1 misclassification costs (available in Clementine only for the CART and C5.0 algorithms), both the CART and C5.0 algorithms outperformed the baseline model, with a mean cost of –$2.81 per customer. The most important predictor for these models was *principal component 1*, purchasing habits.

Overbalancing as a surrogate for misclassification costs was developed for those algorithms without the misclassification cost option. It was demonstrated that as the training data set becomes more overbalanced (fewer negative response records retained), the model performance improves, up to a certain point, when it again begins to degrade.

For this data set, the 80%–20% overbalancing ratio seemed optimal. The best classification model using this method was the logistic regression model, with a mean cost of –$2.90 per customer. This increased to –$2.96 per customer when the overparametrized variable *lifestyle cluster* was omitted.

Model voting was investigated. The best combination model mailed a promotion only if at least three of the four classification algorithms predicted positive response. However, the mean cost per customer for this combination model was only –$2.90 per customer. Thus, for the models including the principal components, the best model was the logistic regression model with 80%–20% overbalancing and a mean cost of –$2.96 per customer.

The best predictors using this model turned out to be the two principal components, purchasing habits and promotion contacts, along with the following variables: *z days on file*, *z ln average spending per visit*, *z ln days between purchases*, *z ln product uniformity*, *z sqrt spending CC*, *Web buyer*, *z sqrt knit dresses*, and *z sqrt sweaters*.

Next came the non-PCA models, which should be used for prediction of the response only, not for profiling. Because the original (correlated) variables are retained in the model, we expect the non-PCA models to outperform the PCA models with respect to overall cost per customer. This was immediately borne out in the results for the CART and C5.0 models using 50% balancing and 14.2–1 misclassification costs, which had mean costs per customer of –$3.01 and –$3.04, respectively. For the 80%–20% overbalancing ratio, C5.0 was the best model, with an overall mean cost of –$3.15 per customer, with logistic regression second with –$3.12 per customer.

Again, model combination using voting was applied. The best voting model mailed a promotion only if at least two models predicted positive response, for an overall mean cost of –$3.16 per customer. A second, continuous method for combining models was to work with the response probabilities reported by the software. The mean response probabilities were calculated, and partitions were assigned to optimize model performance. It was determined that the same level of profits obtained by the "send to everyone" model could also be obtained by contacting merely half of the prospective customers, as identified by this combination model.

As it turns out the optimal partition is at or near 50% probability. In other words, suppose that we mailed a promotion to a prospective customer under the following conditions: Mail a promotion only if the mean response probability reported by the four algorithms is at least 51%. In other words, this continuous combination model will mail a promotion only if the mean probability of response reported by the four classification models is greater than half. This turned out to be the optimal model uncovered by any of our methods in this case study, with an estimated profit per customer of $3.1744.

Compared with the baseline performance, from the "send to everyone" model, of $2.63 per mailing, this model enhances the profitability of this direct mail marketing campaign by 20.7%, or 54.44 cents per customer. For example, if a mailing was to be made to 100,000 customers, the estimated increase in profits is $54,440. This increase in profits is due to the decrease in costs associated with mailing promotions to nonresponsive customers.

REFERENCES

1. Peter Chapman, Julian Clinton, Randy Kerber, Thomas Khabaza, Thomas Reinart, Colin Shearer, and Rudiger Wirth, *CRISP-DM Step-by-Step Data Mining Guide*, http://www.crisp-dm.org/, 2000.
2. Daniel Larose, *Discovering Knowledge in Data: An Introduction to Data Mining*, Wiley, Hoboken, NJ, 2005.
3. Clothing store data set, compiled by Daniel Larose, 2005. Available at the book series Web site.
4. Claritas Demographics ®, http://www.tetrad.com/pcensus/usa/claritas.html.
5. Tian Zhang, Raghu Ramakrishnan, and Miron Livny, BIRCH: an effiecient data clustering method for very large databases, presented at Sigmod' 96, Montreal, Canada, 1996.