Chapter 11
Statistical Issues in Predictive Modeling

Abstract Whereas Chapter 10 describes the basic process of predictive modeling, this chapter goes into depth on three key issues: selection of variables, treatment of missing data, and evaluation of models. Topics covered include stepwise selection and principal components methods of variable selection; imputation methods, missing variable dummies, and data fusion techniques for missing data; and validation techniques and metrics for evaluating predictive models.

The word (scientific) "model" has several meanings. Here we focus on statistical models. A model is, in effect, a statement of reality or its approximation. Most phenomena in the social sciences are extremely complex. With a model we simplify the reality and focus on a manageable number of factors. For example, economists often assume a world where there are only two products available, apples and oranges, in order to understand the relationship between the price of an apple and the price of an orange. Economists know that the assumed world is far from the reality. However, they devise a model to allow them to answer the question of their interests within the assumed world. Once economists build knowledge from a simple world, they often extend their model to study more complex world that may be closer to the reality.

Managers build a statistical model to understand and predict variables of importance to their firms. Consider a manager in a bank who wants to determine to whom he or she should issue credit cards. Or the manager wants to know who will default and who will not. It is impossible to completely understand why consumers default and identify all the factors influencing customer's default behavior. Simplifying the reality, the bank manager sets up a statistical model that relates customer's default behavior to only two important factors, the income and the education. We know that this simple model is not very close to the reality. There surely are thousands of other variables that may influence customer's default behavior. The manager does not include them into the model because they are either minor factors or

they are not available in the database. Hopefully, the manager could reduce the default rate by 80% with the help of this simple model.

This chapter describes the fundamentals of a statistical model building. We do not discuss statistical basics that can be found in other textbooks on statistics or marketing research. Instead, we focus on issues that are important to database marketers but are not well-treated in other books. We begin our discussion on the managerial justification for building a statistical model. Then we discuss three important statistical issues that are of prime importance to database marketers: model/variable selection, treatment of missing data, and evaluation of the model.

11.1 Economic Justification for Building a Statistical Model

Why do we want to build a statistical model? What is the economic benefit from building a model? Database marketers have often used the decile analysis to evaluate the economic value of a model. The concept may best be explained by an example. Suppose that a bank has budget available to issue 2,000 credit cards and needs to determine who should receive a card among 10,000 credit card applicants. It will lose $400 if a customer defaults and gain $100 if the customer does not default. And assume that the market average default probability is 11.5%. Without a statistical model, the bank does not know who will default and will not. Hence, it may randomly select 2,000 customers among 10,000 applicants and issue credit cards. Given the market (average) default probability of 11.5%, 230 ($0.115 \times 2{,}000$) customers will default and 1,770 customers will not. Therefore, with randomly issuing credit cards to 2,000 applicants, the bank will collect total profits of $85,000 ($1{,}770 \times \$100 - 230 \times \$400$).

Now assume the bank develops a statistical model estimated using data from current customers. Based on the estimated model, it can predict the default probability for each of the 10,000 applicants. Customers are ranked in descending order in terms of their predicted default probabilities. The ranked customers are evenly divided into ten groups (or *deciles*), each with 1,000 customers as shown in Table 11.1. The second column in Table 11.1 represents the average of the predicted default probabilities over 1,000 customers. Instead of randomly selecting 2,000 customers to receive a credit card, it targets the 2,000 customers in decile 9 and 10, who are least likely to default.[1] The third column shows the percentage of actual defaulters. Eleven out of 1,000

[1] We could actually find the breakeven default rate to maximize the profit if we were not constrained to issue only 2,000 cards. The bank should issue the credit card if the expected profit is greater than zero. Therefore, the profit maximizing rule is "issue card if $\$100 \times (1-p) - \$400 \times p > 0$ where p is the default probability. Hence, the breakeven default probability is 0.2. That is, the bank should issue the credit card if the predicted

Table 11.1 Decile analysis demonstrating the economic value of a statistical model

Decile	Model predicted Default rate (%)	Actual Default rate (%)	Expected Profits ($)	Actual Profits ($)
1	25.5	26.7	−27,500	−33,500
2	21.4	22.3	−7,000	−11,500
3	18.0	17.8	10,000	11,000
4	13.3	12.9	33,500	35,500
5	12.6	12.5	37,000	37,500
6	10.8	10.4	46,000	48,000
7	8.1	7.6	59,500	62,000
8	3.7	3.3	81,500	83,500
9	1.2	1.1	94,000	94,500
10	0.1	0.1	99,500	99,500
Total	11.5	11.5	426,000	426,000

customers actually default in decile 9 while only one customer defaults in decile 10. The bank collects profits of \$194,000 ($1{,}988 \times \$100 - 12 \times \$400$) by employing the statistical model. As a result, the bank can increase the profit from \$85,000 to \$194,000 with the model. Therefore, the economic benefit of the statistical model is \$109,000.

The economic benefit of a statistical model can only be realized by building a good model that provides accurate predictions. The predictive model in Table 11.1 can be said to be a good model since its predicted default rates are very close to the corresponding actual default rates. However, we can easily think of more accurate model that perfectly forecasts who will default and who will not. In the following three sections, we discuss three key statistical (however, often ignored) issues that will help database marketers to develop more accurate models.

11.2 Selection of Variables and Models

11.2.1 Variable Selection

Most predictive models (e.g., regression, logistic regression, neural nets) can be stated in the following regression-type format:

$$Y = f(X_1, X_2, X_3, \ldots X_K) + \varepsilon \tag{11.1}$$

where Y is the variable being predicted (customer response, customer value, etc.), the X's are the potential predictor variables, and ε are other (random) variables that have not been observed by researchers. Note that "K" is

default probability is less than 0.2. Applied to the data given in Table 11.1, credit cards should be issued to 8,000 applicants (from decile 3 to 10) to maximize its profits.

the number of potential predictor variables (including the intercept). In real-world applications, the value for K can be very high, easily in the hundreds if not in the thousands. This is because there are often many demographic variables and other customer characteristics available, several measures of previous customer behavior (RFM, etc.), and several previous contact variables (e.g., marketing contacts). There are several reasons why all K variables cannot be included in the model: (1) Computation time – for example, a neural net would take an enormous amount of time to run with 300 predictors. (2) Feasibility – in a decision-tree model, one would run out of observations if all 300 predictors were used. (3) Overfitting – there is a danger that using 300 variables will result in "overfitting," whereby the model is able to find a unique idiosyncratic combination of variables that can predict an individual observation, but the relationship implied by this combination does not hold up in general. (4) Interpretation – it is often difficult to interpret a model with 300 variables. As a result, the model cannot be easily communicated to upper level management, and hence is less likely to be trusted and used.

The ideal approach to selecting which variables should be in the model would be theory. To the extent that theory is available for why certain variables should be in the model, these variables should be included (e.g., if data on customer complaints are available, this variable should certainly be included in a model of customer churn). However, very often there is not good theory available that we would be confident in relying on. In this case, we should rely on statistical methods to select variables to be included in the model. There are several techniques available for this: (1) all-subset regression, (2) step-wise techniques, (3) principal components regression, and (4) other advanced techniques. We discuss these methods in this section.

11.2.1.1 All-Possible Subset Regression

All possible subset regression is frequently used to determine the optimal set of independent variables. This procedure first requires the fitting of all possible combinations among the available independent variables. For example, if three independent variables are available, we need to fit eight regression equations, $\varnothing$, $\{X_1\}$, $\{X_2\}$, $\{X_3\}$, $\{X_1, X_2\}$, $\{X_1, X_3\}$, $\{X_2, X_3\}$, and $\{X_1, X_2, X_3\}$. Next, select the best regression equation using some statistical criteria such as adjusted R^2, AIC (Akaike Information Criteria) or BIC (Bayesian Information Criteria):

$$\text{Adjusted } R^2 = 1 - \left[\frac{n-1}{n-k}\right]\left(1 - R^2\right) \tag{11.2a}$$

$$AIC = -2\log\hat{L} + 2k \tag{11.2b}$$

$$BIC = -2\log\hat{L} + k\log n \tag{11.2c}$$

where n is the number of observations, k is the number of predictors including the intercept and $\hat{L}$ is the value of the likelihood function achieved by

the model. Different from R^2, these criteria penalize more complex models so that a simple model may often be chosen if the increase in fit by including additional variables is not large enough. We select the best model with the largest adjusted R^2 or the lowest *AIC* or *BIC*. The adjusted R^2 is used for linear regression models while *AIC* and *BIC* can be used for both linear and non-linear models. Assuming that the model errors (ε) are normally distributed, the *AIC* becomes $n \log \hat{\sigma}^2(k) + 2k$ where $\log \hat{\sigma}^2(k)$ is the variance of ε. *AIC* however still tends to select models that overfit the data. To overcome this difficulty, the *BIC* penalizes the number of estimated parameters, and hence the number of variables included in the model, more strongly (2 for *AIC* and $\log n$ for *BIC*) than the *AIC* (Schwarz 1978).

The major weakness of all-possible subset regression is that the method is not practical when there are a large number of independent variables. If there are 50 variables, we need to run $2^{50} = 1.16 \times 10^{15}$ regressions. Most of commercial statistical packages have all possible subset regressions but they cannot practically handle more than 30 independent variables.

11.2.1.2 Stepwise Selection

An alternative method to select an optimal set of independent variables is a stepwise selection method that combines "forward selection" and "backward elimination". The forward selection procedure begins with fitting a simple regression model for each of the $K - 1$ potential X variables. For each regression model, the F statistic for testing whether or not the slope is zero is obtained. The X variable with the largest F value is the candidate for first variable to be included. If this F value exceeds a predetermined level F_0, the X variable is added. Otherwise, the process terminates with no variables in the model. Suppose that X_1 is entered at step 1. Now the forward selection routine fits all regression models with two X variables, where X_1 is one of the two. For each such regression, we compute the partial F test statistic that will test whether or not $\beta_k = 0$ when X_1 and X_k are two variables in the model. The X variable with the largest partial F value is the candidate for addition. If this F value exceeds a predetermined level F_0, the second X variable is added. Otherwise, the process terminates. The forward selection continues until no further X variables can be added, at which point the routine terminates.

The backward elimination procedure is an attempt to achieve a similar conclusion working from the other direction. That is, it begins with the regression using all independent variables, and subsequently reduces the number of variables in the equation until a decision is reached on the equation to use. The order of deletion is determined by using the partial F value. The backward elimination begins with a regression containing all variables. The partial F value is calculated for every predictor variable treated as though it were the last variable to enter the regression equation. The variable for which

this F value is the smallest is the candidate for deletion. If this F value is smaller than the predetermined level F_1, the variable is dropped from the model. Otherwise, the routine terminates. The backward elimination routine continues until no further X variables can be deleted, at which point the routine terminates.

The most popular stepwise selection procedure combines forward selection and backward elimination. It begins with the forward search procedure. And assume that X_1 is entered at step 1 and X_2 is entered at step 2. Now the backward elimination routine comes in to determine whether or not any of these two X variables already in the model (X_1 and X_2 in this case) should be dropped. If a variable's partial F is smaller than the predetermined level F_1, the variable is dropped; otherwise, it is retained. The stepwise selection routine continues until no further X variables can be added or deleted, at which point the search terminates. The stepwise selection allows an X variable to be added into the model at an earlier stage and to be dropped subsequently, if it is no longer useful in conjunction with variables added at later stages.

The stepwise selection is computationally very efficient since it does not need to evaluate the full factorial. However, because of its algorithmic characteristics (e.g., sequential search), the stepwise selection often leads to the sub-optimal solution. The relative merits and drawbacks of stepwise procedures, lower computational costs versus sub-optimality, have been mainly discussed within the linear regression context (Hocking 1976; Miller 1989).

An issue often raised in conjunction with stepwise selection (although it applies to all-possible subset regression as well) is the difficulty it can create in interpreting the results. Stepwise will tend to eliminate a variable if (1) it has little predictive power, or (2) it has predictive power but *is highly correlated with* a variable that has better predictive power. In this latter case is where difficulty in interpretation arises. For example, assume income predicts customer profitability well, but age predicts even better, and income and age are positively correlated. It is possible that stepwise regression will include age in the final model, and eliminate income. But as a result, the estimated coefficient for age picks up not only the impact of age but the impact of income as well – income is not in the model explicitly, so age serves as its representative. How should we interpret an age coefficient of say $1,000? Taken literally, this means that every additional year of the customer's age makes her or him $1,000 more profitable. But implicitly, it's the additional age *plus* the extra income that comes with age that makes the customer more profitable.

From a practical standpoint, researchers should always ask themselves – is this variable we've included in the model serving to represent certain other variables besides itself? If so, we need to be careful not to assume that changing that variable alone will induce the change in the dependent variable indicated by its coefficient. An important example of this is if data on catalogs and emails were available but stepwise selected only catalogs for the final model. The coefficient for catalogs would reflect the impact of catalogs *and* emails combined. If we just increase the number of catalogs without a concomitant

increase in emails, we may not achieve the gain in sales predicted by the coefficient for catalog. The careful researcher needs to be savvy in interpreting coefficients when stepwise selection has been used.

11.2.1.3 Principal Components Regression

Massy (1965) developed principal components regression by combining principal components analysis and regression analysis. We first review the method of principal components. Principal components analysis is a technique for combining a large number of variables into a smaller number of variables, while retaining as much information as possible in the original variables. Suppose we have an $n \times k$ matrix of $\mathbf{X}$ of n observations on k variables, and Σ is its variance–covariance matrix. The objective of principal components analysis is to find a linear transformation of $\mathbf{X}$ into a new set of data denoted by $\mathbf{P}$, where $\mathbf{P}$ is $n \times p$ and $p \leq k$. The p variables in $\mathbf{P}$ are called "factors" and the n observations for each factor are called factor scores. The data matrix $\mathbf{P}$ has certain desirable properties: (i) the p variables (columns) of $\mathbf{P}$ are uncorrelated with each other (orthogonality), and (ii) each variable in $\mathbf{P}$, progressing from P_1 to P_2, etc., accounts for as much of the combined variance of the X's as possible, consistent with being orthogonal to the preceding P's. The new variables correspond to the principal axes of the ellipsoid formed by the scatter of sample points in the n dimensional space having the elements of $\mathbf{X}$ as a basis. Hence, the principal components transformation is a rotation from the original X coordinate system to the system defined by the principal axes of this ellipsoid.[2] Specifically, the transformation to principal components is given by

$$\mathbf{P} = \mathbf{M}'\mathbf{X} \tag{11.3}$$

To see how $\mathbf{M}$ $(p \times n)$ is determined, post-multiply Equation 11.3 by $\mathbf{P}'$. Then, $\mathbf{PP}' = \mathbf{M}'\mathbf{XX}'\mathbf{M}$. $\mathbf{XX}'$ is simply the variance–covariance matrix $\mathbf{\Sigma}$. The variance–covariance matrix for principal components $\mathbf{PP}' = \mathbf{\Lambda}$ should be diagonal by virtue of requirement (i) above. Hence, we have:

$$\mathbf{\Lambda} = \mathbf{M}'\mathbf{\Sigma}\mathbf{M} \tag{11.4}$$

Equation 11.4 is an orthogonal similarity transformation diagonalizing the symmetric matrix $\mathbf{\Sigma}$. The transformation matrix M has an orthonormal set of eigenvectors of $\mathbf{\Sigma}$ as its columns, and $\mathbf{PP}' = \mathbf{\Lambda}$ has the eigenvalues of $\mathbf{\Sigma}$ as its diagonal elements. If the columns of $\mathbf{M}$ are ordered so that the first diagonal element of $\mathbf{\Lambda}$ contains the largest eigenvalue of $\mathbf{\Sigma}$, the second the

[2] The principal axes spanned by the elements of $\mathbf{X}$ are not invariant to changes in the scales in which the variables are measured. Hence, $\mathbf{X}$ is usually standardized before the transformation to principal components.

next largest, etc., the principal components will be ordered as specified in requirement (ii).

Instead of fitting a linear regression $\mathbf{Y} = \mathbf{X}\beta + \varepsilon$, principal components regression fits the following regression:

$$\mathbf{Y} = \mathbf{P}\gamma + \varepsilon \tag{11.5}$$

where $\mathbf{P}$ is factor scores from Equation 11.3. Once the values of $\mathbf{P}$ are determined from principal components analysis, the parameters γ can be estimated by ordinary regression.

Massy (1965) suggested that only a few factors should be included – actually, data reduction is its main purpose – but did not provide formal guidelines to determine the number of factors. Marketers often employ heuristic methods. For example, a factor is selected if its eigenvalue is greater than one. Basilevsky (1994) proposed an alternative criterion to determine the number of factors using AIC, the Akaike's information criterion (Naik and Tsai 2004). Or the number of factors can be selected judgmentally according to which set of factors is easiest to interpret[3].

In summary, Principal Components Regression uses all the k original variables, but transforms them to a more manageable number of p factors. The value of this approach hinges on the interpretability of the p factors. Very often the factors generated by the transformation matrix $\boldsymbol{M}$ are difficult to interpret. It turns out however that $\boldsymbol{M}$ is not unique in that, for a given number of factors p, there are other "rotated" versions of $\boldsymbol{M}$ that can produce a factor score matrix $\boldsymbol{P}$ retaining the same amount of information as in the original $\boldsymbol{X}$ matrix (see Lehmann et al. 1998 for more discussion). Often, but not always, the rotated version of $\boldsymbol{M}$ produces factor scores that are easier to interpret. Hence, we recommend principal components regression only when the original independent variables are highly collinear with one another, when there are a great number of potential explanatory variables, and when the factors can be easily interpreted. Principal components regression can be implemented easily in SAS or SPSS by running the principal component analysis, interpreting the factors it yields, computing the factor scores, and then running a regression with these factor scores as independent variables.

11.2.1.4 Other Techniques

Recently, Naik et al. (2000) introduced a new reduction technique called sliced inverse regression to the marketing community. The method was originally developed by Li (1991). Similar to principal components regression, it attempts to extract important factors (a linear combination of all the original independent variables) to reduce dimension. But sliced inverse regression

[3] Principal components analysis generates a "loadings matrix," representing the correlation between each factor and each original $\boldsymbol{X}$ variable, that can be used to interpret the factors. See Lehmann et al. (1998) for more details.

provides simple tests for determining the number of factors to retain and for assessing the significance of factor-loading coefficients (the elements of $\boldsymbol{M}$). The composition of factors is determined objectively on the basis of t-values. Naik et al. (2000) demonstrated that sliced inverse regression performs better than principal components regression using Monte Carlo experiments and two real-world applications. However, sliced inverse regression is also not free from the interpretation problems. So we only recommend its usage when the derived factors are meaningful.

Finally, the variable selection problem has attracted the interest of statisticians interested in applying newly developed Markov Chain Monte Carlo (MCMC) estimation methods. In the previous section, we discussed the relative merits and drawbacks of stepwise procedures, lower computational costs versus sub-optimality. George and McCulloch (1993) proposed a stochastic search variable selection model (SSVS) to overcome the problems of all-possible subset regression (computational costs) and the stepwise selection (sub-optimality). Their procedure uses probabilistic considerations for selecting promising subsets of X's. SSVS is based on embedding the entire regression setup in a hierarchical Bayes normal mixture model, where latent variables are used to identify subset choices. The promising subsets of independent variables can be identified as those with higher posterior probabilities. The computational burden is then alleviated by using the Gibbs sampler to indirectly sample from this multinomial posterior distribution on the set of possible subset choices. Those subsets with higher probability can then be identified by their more frequent appearance in the Gibbs sample.

11.2.2 Variable Transformations

One of the most popular models used among database marketers is the classical linear regression model. It is easy to apply and its interpretation is clear. However, the classical linear regression model assumes that the relationships between a dependent variable and several independent variables are linear. For example, consider the case that we are predicting customer value with a linear regression model:

$$Customer\ \ Value(i) = Y_i = \sum_{k=1}^{K} \beta_k X_{ik} + \varepsilon_i \tag{11.6}$$

This linear regression assumes that the relationship between the customer value (Y_i) and the independent variables (X_{ik}) is linear. However, in many applications the straight-line assumption does not approximate the true relationship very well. For example, customer value will be minimal for small values of marketing contact (one of the X's), but once marketing expenditure passes a certain point, customer value increases dramatically. This is called

a threshold effect. Or customer value increases rapidly at first with increased marketing investment, but then levels off. This is called a saturation effect.

However, the linearity assumption in classical linear regression is not as narrow as it might first appear. In the regression context, linearity refers to the manner in which the parameters and the disturbance enter the equation, not necessarily to the relationship between variables (Greene 1997). Specifically, we are able to write the linear regression model in a very general form. Let $\mathbf{X} = \{x_1, x_2, \ldots, x_k\}$ be a set of K independent variables and let f_1, $f_2, \ldots, f_M$ be M independent functions. And let $g(Y)$ be a function of Y. Then the linear regression model is:

$$\begin{aligned} g(Y) &= \beta_1 f_1(\mathbf{Z}) + \beta_2 f_2(\mathbf{Z}) + \ldots\ldots + \beta_M f_M(\mathbf{Z}) + \varepsilon \\ &= \beta_1 x_1 + \beta_2 x_2 + \ldots\ldots + \beta_M x_m + \varepsilon \end{aligned} \tag{11.7}$$

Hence, the original linear regression can be tailored to various modeling situations by using logarithms, exponentials, reciprocals, transcendental functions, polynomials, products, ratios, and so on for $f(\bullet)$ or $g(\bullet)$. For example, the relationship between X and Y might be hypothesized as:

$$Y = X_1^{\beta_1} X_2^{\beta_2} \ldots X_K^{\beta_K} e^{\varepsilon} = \prod_{k=1}^{K} X_k^{\beta_k} e^{\varepsilon} \tag{11.8a}$$

In logs,

$$\ln Y = \beta_1 \ln X_1 + \beta_2 \ln X_2 + \cdots + \beta_K \ln X_K + \varepsilon = \sum_{k=1}^{K} \beta_k \ln X_k + \varepsilon \tag{11.8b}$$

This model is known as a multiplicative model or log-linear model, where $f(\bullet) = g(\bullet) = \ln(\bullet)$. It is also known as the constant elasticity model since the elasticity of Y with respect to changes in X_k does not vary with X_k (note: $\eta_k = \partial \ln Y / \partial \ln X_k = \beta_k$). This model has been widely used for various marketing problems. We can estimate parameters of Equation 11.8 using a standard least squares procedure since its functional form belongs to the class of Equation 11.7.

Another popular model among marketers is an exponential model in which the relationship between X and Y is hypothesized as:

$$Y = e^{\beta_1 X_1 + \beta_2 X_2 + \beta_K X_K + \varepsilon} = e^{\sum_{k=1}^{K} \beta_k X_k + \varepsilon} \tag{11.9a}$$

In logs,

$$\ln Y = \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_K X_K + \varepsilon = \sum_{k=1}^{K} \beta_k X_k + \varepsilon \tag{11.9b}$$

Here we just apply the transformation $g(y) = \ln(y)$. This model is also known as a semi-log model in which the relationship between Y and X is not linear (but $\ln Y$ and X is linear). Again we can estimate parameters of Equation 11.9 using a standard least squares procedure since its functional form belongs to the class of Equation 11.7.

Another useful model is the Box-Cox model that embodies many models as special cases. Suppose we consider a form of the linear model $Y = \alpha + \beta g(x) + \varepsilon$ in which $g(x)$ is defined as:

$$g^{(\lambda)}(x) = \begin{cases} (x^{\lambda} - 1)/\lambda & \text{if } \lambda \neq 0 \\ \ln(x) & \text{if } \lambda = 0 \end{cases} \tag{11.10}$$

The linear model results if λ equals 1, whereas a log-linear or semi-log model (depending on how Y is measured) results if λ equals 0. If λ equals -1, then the equation will involve the reciprocal of x. That is, depending on the value of λ, we can explain various forms of relationship between Y and X. If λ is known, we simply transform x into $g(x)$ by inserting λ into Equation 11.9. But λ typically is unknown *a priori*. So we may try different values of λ's (e.g., $-1, 0, 1$) and compare the performance of models with different λ's. Alternatively, we can treat λ as an additional unknown parameter in the model that will provide us with a tremendous amount of flexibility. The cost of doing so is that the model becomes non-linear in its parameters. That is, the model does not belong to the class in Equation 11.7 so that we cannot use ordinary least square for its estimation. We would have to use nonlinear regression (available in SAS).

Finally, if we are ready to sacrifice the simplicity of the classical linear regression, we can employ nonparametric regression in which no *a priori* relationship is assumed. Simply assuming the relationship is smooth, nonparametric regression overcomes the highly restrictive structure of linear model and flexibly determines the shape of the relationship. It is data-driven method and, in result, its estimation is computationally intensive. However, because of the recent explosion in the size and speed of computers, several nonparametric procedures can be run on personal computers. For more on nonparametric regression, see Härdle and Turlach (1992), and Hastie and Tibshirani (1990).

11.3 Treatment of Missing Variables

Missing variables is a fact of life for DBM applications. For example, demographics such as income and marital status are often missing because customers do not wish to divulge this information. Previous marketing efforts may be available for some customers but not for others. The question is how to handle this situation. One extreme solution is to eliminate a variable from the analysis if it is missing for any customer. This is obviously wasteful. For example, income could be an important predictor. There would appear to be a huge opportunity cost for omitting this variable from the analysis just because it is missing for say 20% of customers. The following are methods that have been proposed and used for dealing with missing data.

11.3.1 Casewise Deletion

In casewise deletion, the observation is discarded if any one of the variables in the observation is missing. This method is simple but is not desirable if the entire sample size is small. It is especially undesirable when each record has a lot of variables and, hence, there is a high probability that at least one variable is missing. In addition, casewise deletion may lead to the biased results if the characteristics of the discarded records are different from those of the remaining records.

11.3.2 Pairwise Deletion

Pairwise deletion can be used as an alternative to casewise deletion in situations where different samples can be used for different calculations. For example, consider a case of calculating pairwise correlations. The correlation between variable 1 and 2 is calculated across the remaining observations after deleting observations in which variable 1 and/or 2 are missing. And the correlation between variable 1 and 3 is based on the observations after deleting observations in which variable 1 and/or 3 are missing. Each pairwise correlation is computed over different sets of observations. For example, several SAS procedures (in default) employ pairwise deletion. "PROC CORR" in SAS estimates a correlation by using all cases with non-missing values for the pair of variables.

This procedure is also not appropriate when the sample size is small. Moreover, if there is a systematic pattern in generating missing data, the correlation coefficient matrix may be seriously biased because each pairwise correlation is calculated over different subsets of cases.

11.3.3 Single Imputation

In single imputation, one substitutes a single value for each missing value. Once all missing values are substituted, standard statistical procedures are applied to the filled-in complete data set. The simplest form of the single imputation is "mean substitution", in which all missing values of a variable are replaced by the mean value for that variable (computed across observed values). Or the patterns in the complete (non-missing) data can be used to impute a suitable value for the missing response. For example, household income may be related to the value of the car owned and the value of the house owned. Hence, one estimates a regression model with household income as a dependent variable and the value of the car owned and the value of the house owned as two independent variables. The missing values

for the household income can be predicted ("imputed") by the estimated regression.

A particular challenge in single imputation is when the analyst has available a variable at the aggregate level, but that variable is missing at the household level. Consider the case where the analyst has mean income for the census tract where the customer resides. It is common to use that mean income as the income value for that customer. However, Duan et al. (2007) show this leads to biased estimates of the income parameter in a predictive model if income is correlated with a variable observed at the individual level, e.g., age. Duan et al. then prescribe a Bayesian procedure for inferring an individual-level estimate of the income variable. The procedure relies on a survey or other source of data that contains individual-specific values of both age and income.

The most critical problem of the single imputation is that it ignores the uncertainty on the predictions of the unknown missing values. As a result, the variability in the variable for which observations are missing is misleadingly decreased in direct proportion to the number of missing data points.

11.3.4 Multiple Imputation

Multiple imputation is an advanced method of dealing with missing data that can solve the "over-certainty" problem of the single imputation method. Rather than replacing a single value for each missing data point, multiple imputation imputes multiple values. For example, we introduced the regression-based single imputation in which each missing value for the household income is predicted by the estimated regression. But we know that the predicted value is distributed as normal. In single imputation, we replace the missing value by the expected value of this normal distribution. In multiple imputation, we can impute the missing value several times by drawing from this normal distribution.

Statisticians have developed a general multiple imputation procedure that replaces each missing value by a *set* of plausible values to incorporate the uncertainty involved in imputing the missing value (Rubin 1987; Schafer 1997). The procedure consists of three steps. First, the missing data are generated m times, resulting in m sets of complete data. Second, each of the m complete data sets is analyzed using the predictive modeling technique being employed for this application. Finally, these intermediate results from the m complete data sets are combined to generate a final model.

There are several ways to implement the multiple imputation procedure. The choice depends on the type of missing data patterns. For monotone missing data patterns, either a regression method or propensity score method can be used. The data set is said to have monotone missing data pattern when a variable X_j is missing for the customer i implies that all subsequent

variables $X_k (k > j)$ are all missing for the customer i. For an arbitrary missing data pattern, a Markov chain Monte Carlo (MCMC) method is used (Schafer 1997). We will discuss MCMC method since missing data patterns in database marketing are more likely to be arbitrary.

Let $\mathbf{X}$ be the $n \times p$ matrix of complete data, which is not fully observed. Let the observed part of $\mathbf{X}$ be $\mathbf{X}_{\text{obs}}$ and the missing part by $\mathbf{X}_{\text{mis}}$. Schafer's imputation method uses a Bayesian approach, in which information about unknown parameters is expressed in the form of a posterior probability distribution. MCMC has been applied as a method for deriving posterior distributions in Bayesian inference. In addition, we need to assume the data model or a probability model for the complete data. Multivariate normal models are usually used for normally distributed data while a log-linear model is assumed for categorical data. Without loss of generality, we assume that the complete data are from a multivariate normal distribution with the unknown parameters θ (i.e., mean vector and variance–covariance matrix). Our goal is to derive the joint posterior distribution of X_{mis} and θ given X_{obs} that is $h(X_{mis}, \theta | X_{obs})$. For multiple imputations for missing data, the following two steps are repeated.

1. *The Imputation Step:* Generate X_{mis} from $f(X_{mis}|X_{obs}, \theta)$. That is, given the estimated mean vector and variance–covariance matrix of the multivariate normal distribution (θ), the imputation step generates the missing values (X_{mis}) from a conditional distribution f.
2. *The Posterior Step:* Generate θ from $g(\theta|X_{mis}, X_{obs})$. The posterior step generates the posterior parameters (mean vector and variance–covariance matrix) for the multivariate normal distribution from a conditional distribution g. These new estimates will then be used in the imputation step.

The two steps are iterated long enough for the iterated values to converge to their stationary distribution. That is, at the tth iteration, the imputation step generates $X_{mis}^{(t+1)}$ given $\theta^{(t)}$ and the posterior step generates $\theta^{(t+1)}$ given $X_{mis}^{(t+1)}$. As a result, we have a Markov chain $\{X_{mis}^{(1)}, \theta^{(1)}\}$, $\{X_{mis}^{(2)}, \theta^{(2)}\}$, $\{X_{mis}^{(3)}, \theta^{(3)}\} \ldots$ which converges to $h(X_{mis}, \theta | X_{obs})$. In practice, 50 to 100 burn-in iterations are used to make the iterations converge to the stationary joint distribution before imputing missing values. Then a set of missing values are independently generated m times from this joint distribution.

When the imputation step is finished, each of the m complete data sets is in turn analyzed with the predictive model. This yields m different sets of the point and the variance estimates for the predictive model parameters. Let $\hat{Q}_i$ be the point estimate from the ith imputed data set and $\hat{U}_i$ be the corresponding variance estimate. That is, we have $\{\hat{Q}_1, \hat{U}_1\}, \{\hat{Q}_2, \hat{U}_2\}, \ldots, \{\hat{Q}_m, \hat{U}_m\}$ from m applications of the predictive model. Then the point estimate for Q from m complete data sets is the average of the point estimates from m different data sets.

$$\bar{Q} = \sum_{i=1}^{m} \hat{Q}_i / m \tag{11.10a}$$

On the other hand, the variance estimate for $\bar{Q}$ should consider the between-imputation variance as well as the within-imputation variance.

$$Var(\bar{Q}) = \bar{U} + (1 + 1/m)B \tag{11.10b}$$

$$\bar{U} = \sum_{i=1}^{m} \bar{U}_i/m \text{ (within-imputation variance)}$$

$$B = \sum_{i=1}^{m} (\hat{Q}_i - \bar{Q})^2/(m-1) \text{(between-imputation variance)}$$

The total variance $Var(\bar{Q})$ is the weighted average of the within-imputation variance and the between-imputation variance. The within-imputation variance $\bar{U}$ is simply the average of the variance estimates from m different data sets. The between-imputation variance B is the key term which explains the uncertainty associated with missing values. Since single imputation does not consider this between-imputation variance for missing values, its variance estimates become underestimated.

Multiple imputation is becoming more popular in treating missing values among database marketers because of its theoretical attractiveness and the availability of commercial software. For example, SAS has a procedure PROC MI that can implement multiple imputations for an $n \times p$ matrix of incomplete data. Once the m complete data are generated and analyzed by using the predictive model of our choice, PROC MIANALYZE can be used to generate valid statistical inference (e.g., Equation 11.10) by combining results from the m applications of the predictive model.

11.3.5 Data Fusion

Kamakura and Wedel (1997) introduced a special type of missing data problem called data fusion. Figure 11.1 shows the structure of data set for data fusion. A marketing researcher conducts a survey and then attempts to relate its results to another survey conducted with a different sample of respondents. We conduct a survey to respondents in sample A to collect variables I and II, and conduct another survey to respondents in sample B to collect variable II and III. Combining these two survey responses, we have missing observations of variable III for sample A respondents and variable I for sample B respondents. The variables common to sample A and B can be demographic variables, whereas the variables unique to sample A or sample B can be brand choice behavior and media exposure, respectively.

It is not practical to apply the multiple imputation procedure to this type of data since there are too many missing variables to be imputed. Statisticians have traditionally developed a special technique called a file concatenation

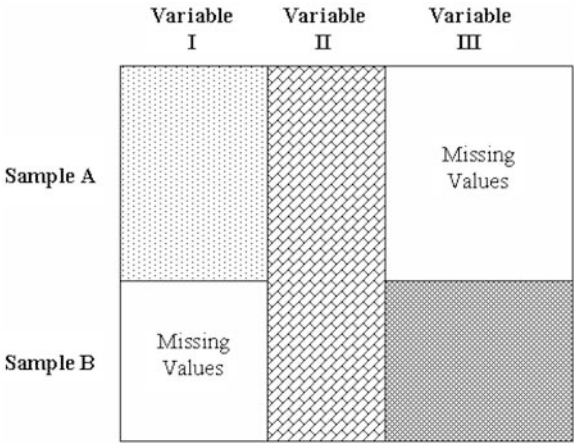

Fig. 11.1 Data structure for the data fusion problem (Modified from Kamakura and Wedel 1997).

method which is designed to combine data from two different sources. In a file concatenation approach, when a variable is missing from sample A (the recipient sample), its value is taken from sample B (the donor sample) to replace it (Ford 1983; Roberts 1994). Each recipient customer in sample A is linked to one or more donor customers in sample B on the basis of common variables. Similarity or distance between customers is measured in terms of common variables (e.g., demographic variables).

Kamakura and Wedel's data fusion method is different from previous file concatenation methods in several aspects. First, they assume the existence of a set of unobserved imputation groups in the combined sample of A and B. If some variables in sample A are missing, they can be replaced by values that are derived from the (mixture) model estimates obtained from customers from sample B belonging to that same latent imputation group. The fundamental idea is similar to the previous file concatenation methods. However, Kamakura and Wedel identify (latent) homogeneous groups based on statistical estimation, whereas the previous file concatenation methods identify similar customers in rather heuristic ways. Second, their procedure is specially tailored to discrete data and the problem of forming cross-classified tables with chi-square tests of significance among variables from independent studies obtained from separate samples. Third, previous methods concatenate two independent files by matching them on the basis of the information on the common variables (e.g., variable II in Fig. 11.1) only. In contrast, data fusion uses a mixture model that identifies homogeneous imputation groups on the basis of all information available from the two samples. Finally, their data fusion method overcomes problems of model selection encountered in previous approaches to modeling under missing data. Data fusion uses multiple

imputations to provide an assessment of the uncertainty caused by the data-fusion process.

As mentioned, there are several ways to implement multiple imputation procedures. The choice depends on the type of missing data patterns. Data fusion is a special type of multiple imputation technique designed to combine data from multiple sources. Hence, the multiple imputation method in previous section may be appropriate for general missing variable problems encountered by database marketers. However, when database marketers attempt to combine customer data from various sources, data fusion can be a very efficient method.

11.3.6 Missing Variable Dummies

Another simple approach to treat missing variables is to create a missing variable dummy per covariate to signify that the variable is missing for a given customer (Van den Poel and Larivière 2003). The extra dummy takes on the value of 1 or 0 depending on whether a variable for a particular customer is missing or complete. For example, suppose that one of the independent variables is *INCOME* that contains some missing values. We define:

$INCOMEM_i = 1$ if income is missing for customer i, and 0 if not missing
$INCOMEO_i =$ customer i's income if income is not missing, and 0 if missing
$Y_i =$ dependent variable to be predicted for customer i, e.g., customer value

The model is then $Y_i = \beta_0 + \beta_1 INCOMEM_i + \beta_2 INCOMEO_i + \varepsilon_i$. After estimation, we would have the following predictions:

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 \text{ if income is missing for customer} i \tag{11.11a}$$

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_2 INCOMEO_i \text{ if income is not missing for customer } i. \tag{11.11b}$$

This method allows us to learn about the relationship between income and customer value among the customers for whom we have such information. That relationship is quantified by $\hat{\beta}_2$. We also learn whether the fact that the customer has missing income data provides any insight on customer value. That insight is quantified by $\hat{\beta}_1$. For example, we might find $\hat{\beta}_1 > 0$ if wealthy people are reluctant to divulge their income. In summary, this method allows us to learn about the relationship between the missing variable and the dependent variable of interest, while at the same time providing information on the types of people for whom information is missing.

Missing variable dummies can actually be used after we impute missing observations from single or multiple imputations. The extra dummy takes on the value of 1 if the observation is imputed (previously missing) and 0 if

the observation is complete. For example, we apply the above regression with two income variables: $INCOMEM_i = 1$ if income is imputed for customer i, and 0 if complete; $INCOMEO_i$ = customer i's income if income is not imputed, and customer i's imputed income if missing. If missing values occur randomly and our imputation procedure is unbiased, the coefficient associated with *INCOMEM* will be estimated to be 0. If this coefficient is not zero and statistically significant, we conclude that our imputing method does not capture the pattern of missing data generation process appropriately.

11.4 Evaluation of Statistical Models

We often come up with several alternative models in a database marketing application. We then need to determine which one to use. This is the subject of model selection in statistics. In a typical database marketing application, we randomly partition the data into two mutually exclusive subsets, the estimation sample and the holdout (or test) sample. We estimate competing models on the estimation (also called calibration) sample, and test their predictive performance on the holdout (also called validation) sample.

The major drawback in comparing models in the estimation sample has been found to be the problem of overfitting (i.e., finding statistical parameters that predict idiosyncratic characteristics of the calibration data that do not hold up in the real world). For example, the estimation of classical regression model is designed to minimize its mean squared error (or sum of squared errors). As a result, a complex model is guaranteed to have a lower mean squared error (or higher R^2) than a simpler model. Taking an extreme case, using a polynomial of sufficiently high order, we can develop a regression model with zero mean squared error. However, this complex model may overfit the data by identifying random fluctuations as true patterns in the data. As mentioned in the section of all-possible subset regression, to overcome the problem of overfitting, statisticians have developed evaluation criteria such as adjusted R^2, *AIC* and *BIC* for model selection in the calibration sample. You select the model with the highest adjusted R^2 or the lowest *AIC* or *BIC*. Basically, these criteria avoid the problem of overfitting by panelizing the number of parameters to be estimated.

Even though model selection in the calibration sample has been widely studied in the statistical literature, database marketers rarely evaluate alternative models in the calibration sample. Hence, we limit our attention to model selection problems based on the validation sample. We first discuss various methods to divide the sample into calibration and validation samples, and then study evaluation criteria to compare alterative models in the validation sample.

11.4.1 Dividing the Sample into the Calibration and Validation Sample

How much of the data should be saved for the validation sample? The larger the calibration sample, the more accurate the model (hence, lower standard errors for the parameter estimates), although the returns would begin to diminish once the size of calibration data exceeds to a certain limit. And the larger the validation sample, the more discerning is the comparison between alternative models. There is a tradeoff.

11.4.1.1 The Holdout Method

The holdout method randomly partitions the data into two mutually exclusive subsets, the calibration sample and the validation (or holdout) sample. Models are estimated with the calibration sample and the prediction errors of the estimated models are compared using the validation sample.

The first important issue in the holdout method is what percentage of the data should be used for the calibration sample. More data in the calibration sample leads to more efficient estimates, while more in the validation sample leads to a more powerful validity test. That is, as more data are used for the calibration sample (so less data for the holdout sample), the model parameter estimates become more accurately estimated but the variance of the prediction errors for each competing model becomes larger. Alternatively, if you decrease the size of the calibration sample, the variance of the prediction errors becomes smaller but the parameter estimates become inaccurate.

In many database marketing applications, it is common to reserve one-third of the data for the holdout sample and use the remaining two-third for the estimation (Kohavi 1995). Steckel and Vanhonacker (1993) showed that the proportion of the sample optimally devoted to validation increases, levels off, and then decreases as the total sample size increases. Specifically, in small samples (e.g., $n < 100$), the one-quarter to one-third validation split was recommended. However, once the sample size gets larger, any reasonable split performed equally well. Hence, we may not need to worry about the optimal split between estimation and validation sample since the sample sizes in real database marketing applications are very large.[4]

A second issue in creating the calibration and holdout samples is the risk that the resulting calibration or holdout sample may not be representative despite the random partitioning. The overrepresented class in the calibration sample will be underrepresented in the holdout sample. For an example of credit scoring, suppose that the data have 50% of defaults and 50% of

[4] All the results from Steckel and Vanhonaker (1993) were based on a regression model containing two independent variables, which is a rather restrictive specification. Hence, more research may be required to generalize their results.

non-defaults. If the calibration sample happens to have more than 50% of defaults, then the percentage of defaults in the holdout sample will be less than 50%.

There are two ways of addressing this problem (Witten and Frank 2000). One is to employ stratification in partitioning the data. For the discrete dependent variable, a data is partitioned such that both the calibration and the holdout sample have the same proportion of each class. For the continuous dependent variable, the data are ranked in ascending order and is partitioned such that observations are evenly represented in both samples. An alternative way is to do random sub-sampling where the holdout method is repeated k times and the prediction error is derived by averaging the prediction errors from different iterations. Even though it is time consuming, random sub-sampling is a better way of minimizing the problem of sample misrepresentation.

A third issue is that selecting the best model from a single experiment (or partitioning) may be too naïve. The estimates of prediction errors are random variables so that they are expected to show random variations. Hence, in order to compare the true performances of alternative models, we require a set of prediction error estimates from multiple experiments. More specifically, we randomly divide our data into the calibration and the validation sample. Estimate two alternative models using the calibration sample, and derive the prediction errors of two models applied to the validation sample, $\{x_1, y_1\}$, where x_1 and y_1 are the prediction error estimate of the first model and the second model. We now repeat the procedure all over again. We divide the data, estimate models, and derive another set of prediction errors. This yields $\{x_2, y_2\}$. We repeat the same experiment k times resulting in k sets of prediction error estimates. Considering these k sets of estimates as a paired comparison data, we can design a formal statistical test for comparison, a paired t-test. The test statistic is $t = \bar{D}/\sqrt{\sigma_D^2/k}$ where $D_i = x_i - y_i$, $\bar{D}$ is the mean of D_i, and σ_D^2 is the variance of D_i. Given your choice of significance level, we reject or accept the null hypothesis that the performances of two models are the same.[5]

11.4.1.2 *K*-Fold Cross-Validation

As implied, the holdout method makes inefficient use of the data by reserving a large portion of the data for the validation sample. If the size of the data

[5] Once we find the best performing model, we are often interested in reporting its parameter estimates. Then we apply the best model to the entire sample and report its parameter estimates. This procedure is applied to the other calibration/validation methods such as k-fold cross-validation, leave-one-out and the bootstrap. That is, the goal of dividing the sample into the calibration and validation is to get accurate prediction error estimates in an efficient way. So if we are interested in parameter estimates, we do not need to divide the sample into two so that we can estimate the parameters more accurately.

is really big, this is not a significant problem. However, the size of your data in practice may often be smaller than you would like it to be. Database marketers frequently adopt K-fold cross-validation technique to use the data more efficiently.

In K-fold cross-validation, the data is randomly divided into K equal sized and mutually exclusive subsets (or folds). The model is estimated and validated k times; each subset in turn is reserved for the validation and the remaining data are used for estimation. The k prediction errors from different iterations are averaged to provide the overall prediction error estimate.

Similar to the holdout method, the problem of sample misrepresentation problem in K-fold cross-validation can be mitigated by stratification and/or repetition. If stratification is adopted to K-fold cross-validation, it is called stratified k-fold cross-validation. Repeating k-fold cross-validation multiple times using different partitions (or folds) and averaging the results will provide a better error estimate.

How many folds should be used? Various tests on a number of datasets have shown that ten is about the right number even though there are not any strong theoretical explanations as to why (Witten and Frank 2000). Kohavi (1995) has empirically shown that as k decreases (e.g., $k = 2$ and 5) and the estimation sample sizes get smaller, there is a variance due to the instability of the estimation sample, leading to an increase in variance. The k-fold cross-validation with 10 to 20 folds produced the best performance.

11.4.1.3 Leave-One-Out Method

Leave-one-out cross-validation is simply a type of k-fold cross-validation when k is equal to the size of the entire sample. Each observation in turn is reserved for validation and the remaining $(k - 1)$ observations are used for estimation. Upon estimation, the model is applied to the validation sample (consisted of one observation) and its prediction error is computed. The overall estimate of prediction error is the average of k error estimates from k iterations.

Leave-one-out cross-validation is an attractive method in using the data (Witten and Frank 2000). Since it uses a large amount of data for estimation, parameters are estimated more accurately. It is shown to work especially well when the dependent variable is continuous. However, it has not performed well for discrete dependent variable or for model selection problem (Shao 1993).

11.4.1.4 Bootstrap

Given a dataset of size n, the principle of the bootstrap is to select samples of size n with replacement from the original sample. Since the bootstrap samples

are selected with replacement, some cases are typically sampled more than once. Originally introduced by Efron (1983), bootstrapping has been shown to work better than other cross-validation techniques, especially in small samples.

There are various bootstrap methods that can be used for estimating prediction error and confidence bounds (Efron and Tibshirani 1993). One of the simplest is the 0.632 bootstrap in which a dataset of n observations is selected (with replacement) from an original sample of size n. Since some cases are sampled more than once, there are cases that are not picked. Those observations not included in the bootstrap sample are used as validation samples. The probability of any given observation not being chosen in the original sample is $(1 - 1/n)^n \approx e^{-1} = 0.368$. Therefore, the expected number of distinct observations from the original dataset appearing in the calibration set is 63.2% of the sample size n. Accordingly, we expect that the size of the validation set will be 36.8% of the original sample size n for a reasonably large dataset. The 0.632 bootstrap has been improved to the popular 0.632+ bootstrap that performs very well for estimating prediction error with discrete dependent variables (Efron and Tibshirani 1993).

The estimate of prediction error for 0.632 bootstrap is derived by combining the error from the validation sample and the error from the calibration sample. Since the model is estimated on the sample containing only 63.2% of distinct cases, the prediction error applied to the validation sample may overestimate the true prediction error. On the other hand, the error in the calibration sample underestimate the true prediction error. Hence, the estimate is given by the linear combination of these two errors, given by 0.632 × (prediction error in validation sample) plus 0.368 × (error in calibration sample). Given a bootstrap sample, prediction errors for alternative models are calculated and compared to find the best model.

11.4.2 Evaluation Criteria

Here we describe several evaluation criteria frequently employed by database marketers to choose the best model. Several alternative measures are available to evaluate the performance of the model applied to the validation sample. All of them measure "goodness-of-fit", which refers to how well the model can predict the dependent variable. In other words, these measures all assess the distance between what really happened and what the model predicts to happen. But they differ in ways of quantifying the distance. Depending on the purpose of models, one criterion is preferred to another. There is no dominating criterion. Database marketers employ different performance measures depending on the nature of dependent variables. We first discuss various measures when the dependent variable is continuous (sales, market

share, monthly shopping expenditure, etc.). Next we discuss discrete dependent variables (churn, response, etc.).

11.4.2.1 Continuous Dependent Variable

Suppose we have n observations for the validation sample on which we want to evaluate predictions. The actual values for the dependent variable are $Y_1, Y_2, \ldots, Y_n$ and the corresponding predicted values are $\hat{Y}_1, \hat{Y}_2, \ldots, \hat{Y}_n$. If the model predicts perfectly for the ith observation, $\hat{Y}_i$ should be the same as Y_i. There is no error. The distance between $\hat{Y}_i$ and Y_i indicates a prediction error. Table 11.2 summarizes the formulae of alternative performance measures frequently used for the continuous dependent variable. They are different in terms of defining this distance.

Mean squared error may be the most popular measure among statisticians partially because of its mathematical tractability. It is easier to make a statistical inference on the summation of the squared terms. An alternative measure is mean absolute error that measures the Euclidean distance between the predicted and the actual value. Mean squared error panelizes the larger errors more heavily by squaring them while mean absolue error treats all errors evenly. Hence, if your application accepts marginal prediction errors but tries to avoid large errors, you should employ mean squared error as the evaluation measure. On the other hand, note that mean absolute error is more robust to outliers than mean squared error.

Table 11.2 Various evaluation criteria for prediction error with a continuous dependent variable

Evaluation criteria	Formula
Mean squared error	$\sum_{i=1}^{n} e_i^2/n = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2/n$
Mean absolute error	$\sum_{i=1}^{n} \lvert e_i \rvert/n = \sum_{i=1}^{n} \lvert Y_i - \hat{Y}_i \rvert/n$
Root mean squared error	$\sqrt{\sum_{i=1}^{n} e_i^2/n} = \sqrt{\sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2/n}$
Mean absolute percentage error	$\left[\sum_{i=1}^{n} \left\lvert \frac{e_i}{Y_i} \right\rvert /n\right] \times 100 = \left[\sum_{i=1}^{n} \left\lvert \frac{Y_i - \hat{Y}_i}{Y_i} \right\rvert /n\right] \times 100$
Relative squared error	$\sum_{i=1}^{n} e_i^2 / \sum_{i=1}^{n} (Y_i - \bar{Y})^2 = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 / \sum_{i=1}^{n} (Y_i - \bar{Y})^2$
Relative absolute error	$\sum_{i=1}^{n} \lvert e_i \rvert / \sum_{i=1}^{n} \lvert Y_i - \bar{Y} \rvert = \sum_{i=1}^{n} \left\lvert Y_i - \hat{Y}_i \right\rvert / \sum_{i=1}^{n} \lvert Y_i - \bar{Y} \rvert$

e_i = the prediction error of the ith observation
Y_i = the actual value of the ith observation
$\hat{Y}_i$ = the (model) predicted value of the ith observation
$\bar{Y}$ = the mean of the actual values that is $\sum Y_i/n$

Sometimes it is more relevant to use a relative performance measure. Both mean squared error and mean absolute error are unit dependent. For example, we can increase or decrease mean squared error simply by multiplying our data by an arbitrary number (that is not zero). We cannot tell how good a model with mean squared error of 100 is. Relative squared error makes mean square error unit-free by normalization. Similar to R^2 in linear regression, total sum of squares is used for the normalizing factor. Similarly, relative absolute error normalizes mean absolute error. Mean absolute percentage error can also be interpreted as a kind of relative measure since it has normalization factor (actual value in the denominator) applied to each observation.

There is no dominant performance measure. As shown in the comparison between mean squared error and mean absolute error, each measure has its advantages and limitations. The choice will be determined after studying the research problem itself. For example, the cost associated with prediction errors helps us to select a prediction measure. But cost information is not always available. Hence, researchers report a number of measures to evaluate their model performance. Fortunately, a number of studies have shown that correlations among performance measures are strongly positive. We may not need to worry about much the selection among performance measures.

11.4.2.2 Discrete Dependent Variable

Hit Ratio

Different performance measures have been proposed when the dependent variable is discrete. We first describe two popular measures, the hit ratio and predictive log-likelihood, when the dependent variable takes on values of either 0 or 1. These measures can easily be generalized for the dependent variables with more than two discrete values. For a discrete dependent variable, the predictive value typically takes the form of probabilities. For example, suppose a bank wants to know who is going to default. Historical data includes both defaulters (coded 1) and no defaulters (coded 0) with their demographic characteristics. Upon estimation the model (e.g., logit) is applied to the test sample with n customers. With customer specific demographic information, the model provides the predictive default probability of each customer. If the probability is greater than a cut-off threshold 0.5, then we predict that she will default. Otherwise, she is predicted not to default. Hit ratio is calculated as

$$Hit\ ratio = \sum_{i=1}^{n} H_i/n \tag{11.12}$$

where H_i is 1 if the prediction is correct and 0 if the prediction is wrong. That is, hit ratio is the percentage of correct predictions.

One may ask why we use the cut-off of 0.5 for the hit ratio. The answer is that if the prediction is higher than 0.5, the event is more likely to occur than not occur, and so we predict that it occurs. However, this is somewhat arbitrary and we will generalize this notion of hit rate, across all thresholds, when we discuss ROC curves.

Predicted Log-Likelihood

Hit ratio employs a 0/1 loss function. The loss is 0 if the prediction is correct and 1 if it is not. This loss function is intuitive and easy to understand. However, the hit ratio is a lexicographic type of measure. It treats the case with the predicted probability of 0.51 to be the same as the case with 0.99 when the customer actually defaults. It ignores the distance between the actual and the predicted once it passes the threshold (i.e., 0.5). Adopting a loss function with continuous form, the predictive log-likelihood overcomes the problem associated with the lexicographic loss function of the hit ratio. The predicted likelihood of observing the data can be expressed as:

$$\text{Predicted Likelihood} = \prod_{i=1}^{n} \left[\hat{P}_i^{Y_i} \times (1 - \hat{P})_i^{(1-Y_i)} \right] \tag{11.13a}$$

Taking logs of this equation, the formula for the predictive log-likelihood is

$$\text{Predictive log-likelihood} = \sum_{i=1}^{n} \left[Y_i \log \hat{P}_i + (1 - Y_i) \log(1 - \hat{P}_i) \right] \tag{11.13b}$$

where $\hat{P}_i$ is the predicted probability of default, and Y_i represents the actual default value taking 1 if customer defaults, 0 otherwise. The larger the log-likelihood, the better the model. The perfect model in which the model predicts 0 when the actual is 0 and 1 when the actual is 1 will have the log-likelihood of zero. Imperfect models will have negative log-likelihoods; the more negative the value, the worse the prediction.

ROC Sensitivity

The concept of ROC (Receiver Operating Characteristic) curve originated in the field of signal detection to measure the diagnostic power of a model (Swets 1988). In order to understand its concept, let us take a look at a two-by-two contingency table shown in Table 11.3. A diagnostic system (or model) looks for a particular "signal" and ignores other events called "noise." The event is considered to be "positive" or "negative," and the diagnosis made is correspondingly positive or negative. For example, there are customers who will respond to the mailing offer ("positive") and who will not ("negative").

Table 11.3 True event versus diagnosis (From Swets 1988)

		Event		
		Positive	Negative	
Diagnosis	Positive	True positive (a)	False positive (b)	$a+b$
	Negative	False negative (c)	True negative (d)	$c+d$
		$a+c$	$b+d$	$a+b+c+d-N$

And using the predictive model we estimate customers' response probabilities, and assign them into responders or non-responders. There are two ways in which the actual event and the diagnosis can agree: "true-positive" and "true-negative" in Table 11.3. And there are two cases that diagnosis can be wrong: "false-positive" and "false-negative."

In a test of a diagnostic model, the true-positive proportion, $a/(a+c)$, and the false-positive proportion, $b/(b+d)$, can capture all of the relevant information on accuracy of the model. These two proportions are often called the proportion of "hits" and "false alarms." The true positive proportion is also called 'sensitivity' that is the probability of a randomly selected positive event being evaluated as positive by the model. In addition, the true negative proportion is often called specificity that is the probability of a randomly selected negative event being evaluated as negative by the model. Note that the false positive proportion is (1 – specificity). A good diagnostic model will provide many hits with few false alarms.

The ROC curve plots the proportion of hits versus false alarms for various settings of the decision criterion (see Fig. 11.2). Going back to the credit assessment example, we derived hit ratio based on the decision that if the predicted default probability of a customer is greater than a threshold value (e.g., 0.5), then we predict that she will default. Otherwise, she is predicted not to default. In an ROC curve, we initially set the threshold value high, say 0.9. We do not issue a credit card to a customer if her predicted default probability is higher than 0.9. We issue a credit card otherwise. Given the threshold, we can prepare the two-by-two contingency table. The proportions of hits and false alarms from the table will become a point of the ROC curve for the model. Now we set the threshold value a bit lower, say 0.8. And plot a point of the ROC curve. Changing the value of the threshold value to 0 will complete the ROC curve.

Note an ROC curve is generated for a particular model as a function of a critical decision criterion or parameter in the model, the cut-off threshold. The performance (or value) of the model is measured to be the area under the ROC curve. The area varies from 0.5 to 1. The major diagonal in Fig. 11.2 represents the case of the area equal to 0.5 when the proportions of hits and false alarms are the same. Random assignment will lead to the area of 0.5. On the other hand, a perfect model when the curve follows the left and upper axes has the area of 1. There are no false alarms with 100% hits. The realistic

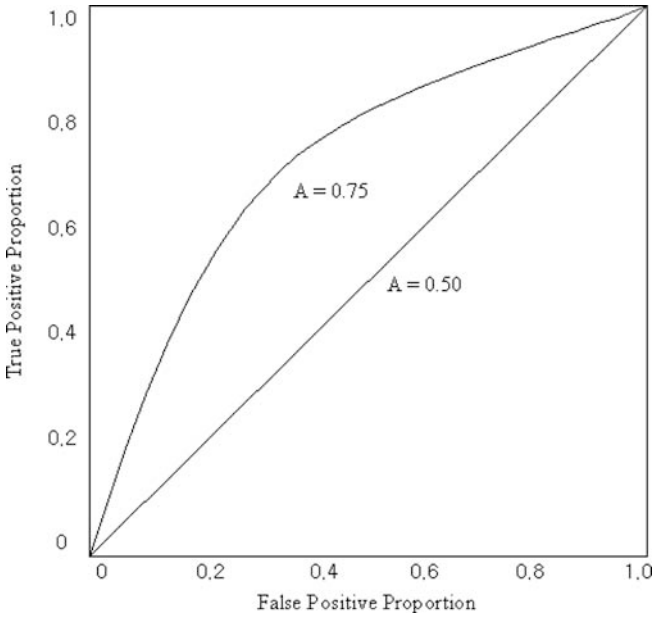

Fig. 11.2 The ROC curve*.

* "A" signifies the area under the ROC curve. The best model is the one that generates ROC curves with highest area (From Swets 1988).

model lies in between. The area under the curve increases as the model can increase more hits while reducing the number of false alarms. We want to select the model with the highest-area ROC curve, because this means that for a given threshold cut-off, it generates more true-positives relative to false-positives.

11.4.2.3 Evaluation Criteria to Assess Financial Performance

The evaluation criteria discussed so far measure the goodness-of-fit that refers to how well the model can predict the dependent variable. However, these evaluation criteria are not useful for assessing the financial performance of the predictive models. Models that do not fit well can still perform well (Malthouse 2002). There are several evaluation criteria to assess the financial performance of the models.

Lift (Gains) Chart

Direct marketers frequently evaluate their proposed models using a gains table (or chart) analysis. Gains table can be developed as follows (see Banslaben

Table 11.4 Gains and lift table

Decile	Response rate (%)	Lift	Cumulative lift (%)
1	6.00	3.75	37.5
2	3.50	2.19	59.4
3	2.50	2.56	75.0
4	1.50	0.94	84.4
5	1.00	0.63	90.6
6	0.65	0.41	94.7
7	0.50	0.31	97.8
8	0.19	0.12	99.0
9	0.12	0.08	99.8
10	0.04	0.03	100.0
Total	1.60	1.00	

(1992) and Chapter 10 more details). Once we estimate the response model, the model is applied to each customer in the validation sample to derive the corresponding response probability ($\hat{P}_i$). Then all customers in validation are ordered by their predicted response probabilities. In the final step, customers are sequentially divided in groups of equal size (usually ten groups), and the average actual response rate per group is calculated. The gains table describes the relationship between the ordered groups and the (cumulative) average response rate in these groups. Table 11.4 shows an example of gains table.

The response rate of the top decile is usually used to evaluate the performances of models. The response rate of the top decile in Table 11.4 is 6%, which is much higher than the overall response rate of 1.6%. The best performing model is the one which provides the highest response rate in the top decile. Alternatively, the variation among the response rates in each of 10 deciles can be used to evaluate the performances of competing models (Ratner 2002). The best model will show the greatest variation. That is, our goal here is to maximize the separation between top deciles and the bottom deciles.

Lift is a useful measure that can be calculated directly from the gains table, and also used to compare the performances among alternative models. Formally, we can define lift as $\lambda_k = r_k/\bar{r}$ where λ_k is lift for the kth tile, r_k is response rate for the kth tile and $\bar{r}$ is the average response rate across the entire sample. In words, λ_k is how much more likely customers in kth tile are to respond, compared to the average response rate for the entire sample. We want lift in the top tiles to be greater than 1, and correspondingly, lift in the lower tiles to be less than 1. For example, the average response rate across the entire sample is 1.60% while the response rate in the top decile 6.00%. Therefore, customers in the top decile are 3.75 times more likely to respond than average ($\lambda_k = 6.00/1.60 = 3.75$). We say, top decile lift is 3.75. Lift itself does not have direct managerial (or financial) significance. However,

the extent to which top tiles have higher lifts makes them more profitable, since lift is directly proportional to response rate. In addition, it is easy to compare lift across models or different applications. See more discussion in Chapter 10.

Another evaluation criteria frequently used in database marketing is the cumulative lift chart, which tabulates cumulative response rates from the top n-tile down. Continuing our example in Table 11.4, the cumulative lift for the kth decile is defined by the percentage of all responders accounted for by the first k deciles. For example, the top 3 deciles account for 75.0% of all responders. Obviously, the higher the cumulative lift is for a given decile, the better the model.

Gini Coefficient

The Gini coefficient is essentially the area between the model's cumulative lift curve and the lift curve that would result from random prediction. It was originally developed by the Italian statistician Corrado Gini. To understand the general concept, we need to define the Lorenz curve and the perfect equality line. The Lorenz curve is a graph representing the cumulative distribution function of a probability distribution. For example, it is frequently used to represent income distribution of a country, where it shows for the top $x\%$ of its population, what percentage ($y\%$) of the total income they have. The percentage of the population is plotted on the x-axis, and the percentage of the total income on the y-axis. To draw the Lorenz curve, all the elements (or customers) of a distribution must be ordered from the largest to the smallest (in terms of their predicted response probabilies). Then, each element (customer) is plotted according to its cumulative percentage of x and y. The Lorenz curve is compared with the perfect equality line, which represents a linear relationship between x and y. For example, if all the people in the population earn the same income, the Lorenz curve becomes the perfect equality line.

In database marketing applications, we would plot the percentage of customers on the X-axis ordered by their predicted likelihood of responding, and the cumulative percentage of responders (i.e., the cumulative lift curve) on the Y-axis (see Fig. 11.3). The perfect equality line would represent a model where predictions are made randomly, since then each customer would have an equal chance of being predicted to be a responder. The higher this curve relative to the perfect equality line, the better the model because our model can account for a large percentage of the responders by targeting a relatively small percentage of cutomers.

The Gini coefficient is defined graphically as a ratio of the summation of all vertical deviations between the Lorenz curve and the perfect equality line (A) divided by the total area above the perfect inequality line (A + B).

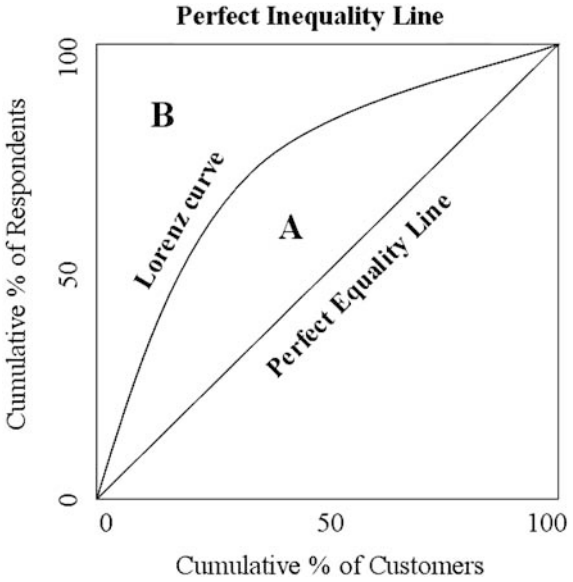

Fig. 11.3 Gini coefficient.

That is, the Gini coefficient is equal to A/(A + B) in Fig. 11.3. Its value lies between 0 and 1, where 0 corresponds to the perfect equality (i.e., everyone has the same income) and 1 corresponds to the perfect inequality (i.e., one person has all the income, while everyone else has zero income). In database marketing terms, a Gini coefficient of 0 means that the model is predicting no better than random, while a value of one corresponds to the (very rare case) that there is only one responder and that customer is identified as the customer with the highest probability of responding. In general, higher Gini Coefficients mean that more responders can be identified by targeting smaller numbers of customers.

The Gini coefficient for a given model can be calcualted as:

$$\text{Gini coefficient} = \sum_{i=1}^{N} (c_i - \hat{c}_i)/(1 - \hat{c}_i) \tag{11.14}$$

where $\hat{c}_i$ is the proportion of the customers who have a predicted probability of response equal or greater than customer i's and c_i is the proportion of actual responders who are ranked equal or higher than customer i in their response probability. That is, $\hat{c}_i$ is the locus of the cumulative lift curve and c_i is the locus of the perfect equality line. We choose the model with the highest Gini coefficient, that is, the Gini coefficient closest to 1.

11.5 Concluding Note: Evolutionary Model-Building

The scientific method for predicting the future is based on the assumption that the future repeats the past. For many applications, this assumption is reasonable. Suppose we try to predict monthly sales of color television. We may build forecasting models, (whether they are time-series models or regression models) based on historical sales of color television, and isolate patterns from random variations. The predicted sales of a color TV are based on the estimated model (or identified patterns). However, the future can be very different from the past especially the market conditions are changing. The model becomes useless. This is why we need to keep updating models. Sometimes it may be enough to re-estimate the model with additional data. Sometimes we need to change the model itself. Remember that the model cannot be static. (See discussion of model "shelf life" in Chapter 10 (Sect. 10.5.2).)