Springer Series in Statistics

Trevor Hastie
Robert Tibshirani
Jerome Friedman

The Elements of Statistical Learning

Data Mining, Inference, and Prediction

Second Edition

Springer

14
Unsupervised Learning

14.1 Introduction

The previous chapters have been concerned with predicting the values of one or more outputs or response variables $Y = (Y_1, \ldots, Y_m)$ for a given set of input or predictor variables $X^T = (X_1, \ldots, X_p)$. Denote by $x_i^T = (x_{i1}, \ldots, x_{ip})$ the inputs for the ith training case, and let y_i be a response measurement. The predictions are based on the training sample $(x_1, y_1), \ldots, (x_N, y_N)$ of previously solved cases, where the joint values of all of the variables are known. This is called *supervised learning* or "learning with a teacher." Under this metaphor the "student" presents an answer $\hat{y}_i$ for each x_i in the training sample, and the supervisor or "teacher" provides either the correct answer and/or an error associated with the student's answer. This is usually characterized by some loss function $L(y, \hat{y})$, for example, $L(y, \hat{y}) = (y - \hat{y})^2$.

If one supposes that (X, Y) are random variables represented by some joint probability density $\Pr(X, Y)$, then supervised learning can be formally characterized as a density estimation problem where one is concerned with determining properties of the conditional density $\Pr(Y|X)$. Usually the properties of interest are the "location" parameters μ that minimize the expected error at each x,

$$\mu(x) = \underset{\theta}{\operatorname{argmin}}\, E_{Y|X} L(Y, \theta). \tag{14.1}$$

Conditioning one has

$$\Pr(X, Y) = \Pr(Y|X) \cdot \Pr(X),$$

where $\Pr(X)$ is the joint marginal density of the X values alone. In supervised learning $\Pr(X)$ is typically of no direct concern. One is interested mainly in the properties of the conditional density $\Pr(Y|X)$. Since Y is often of low dimension (usually one), and only its location $\mu(x)$ is of interest, the problem is greatly simplified. As discussed in the previous chapters, there are many approaches for successfully addressing supervised learning in a variety of contexts.

In this chapter we address *unsupervised learning* or "learning without a teacher." In this case one has a set of N observations $(x_1, x_2, \ldots, x_N)$ of a random p-vector X having joint density $\Pr(X)$. The goal is to directly infer the properties of this probability density without the help of a supervisor or teacher providing correct answers or degree-of-error for each observation. The dimension of X is sometimes much higher than in supervised learning, and the properties of interest are often more complicated than simple location estimates. These factors are somewhat mitigated by the fact that X represents all of the variables under consideration; one is not required to infer how the properties of $\Pr(X)$ change, conditioned on the changing values of another set of variables.

In low-dimensional problems (say $p \leq 3$), there are a variety of effective nonparametric methods for directly estimating the density $\Pr(X)$ itself at all X-values, and representing it graphically (Silverman, 1986, e.g.). Owing to the curse of dimensionality, these methods fail in high dimensions. One must settle for estimating rather crude global models, such as Gaussian mixtures or various simple descriptive statistics that characterize $\Pr(X)$.

Generally, these descriptive statistics attempt to characterize X-values, or collections of such values, where $\Pr(X)$ is relatively large. Principal components, multidimensional scaling, self-organizing maps, and principal curves, for example, attempt to identify low-dimensional manifolds within the X-space that represent high data density. This provides information about the associations among the variables and whether or not they can be considered as functions of a smaller set of "latent" variables. Cluster analysis attempts to find multiple convex regions of the X-space that contain modes of $\Pr(X)$. This can tell whether or not $\Pr(X)$ can be represented by a mixture of simpler densities representing distinct types or classes of observations. Mixture modeling has a similar goal. Association rules attempt to construct simple descriptions (conjunctive rules) that describe regions of high density in the special case of very high dimensional binary-valued data.

With supervised learning there is a clear measure of success, or lack thereof, that can be used to judge adequacy in particular situations and to compare the effectiveness of different methods over various situations.

Lack of success is directly measured by expected loss over the joint distribution $\Pr(X, Y)$. This can be estimated in a variety of ways including cross-validation. In the context of unsupervised learning, there is no such direct measure of success. It is difficult to ascertain the validity of inferences drawn from the output of most unsupervised learning algorithms. One must resort to heuristic arguments not only for motivating the algorithms, as is often the case in supervised learning as well, but also for judgments as to the quality of the results. This uncomfortable situation has led to heavy proliferation of proposed methods, since effectiveness is a matter of opinion and cannot be verified directly.

In this chapter we present those unsupervised learning techniques that are among the most commonly used in practice, and additionally, a few others that are favored by the authors.

14.2 Association Rules

Association rule analysis has emerged as a popular tool for mining commercial data bases. The goal is to find joint values of the variables $X = (X_1, X_2, \ldots, X_p)$ that appear most frequently in the data base. It is most often applied to binary-valued data $X_j \in \{0, 1\}$, where it is referred to as "market basket" analysis. In this context the observations are sales transactions, such as those occurring at the checkout counter of a store. The variables represent all of the items sold in the store. For observation i, each variable X_j is assigned one of two values; $x_{ij} = 1$ if the jth item is purchased as part of the transaction, whereas $x_{ij} = 0$ if it was not purchased. Those variables that frequently have joint values of one represent items that are frequently purchased together. This information can be quite useful for stocking shelves, cross-marketing in sales promotions, catalog design, and consumer segmentation based on buying patterns.

More generally, the basic goal of association rule analysis is to find a collection of prototype X-values $v_1, \ldots, v_L$ for the feature vector X, such that the probability density $\Pr(v_l)$ evaluated at each of those values is relatively large. In this general framework, the problem can be viewed as "mode finding" or "bump hunting." As formulated, this problem is impossibly difficult. A natural estimator for each $\Pr(v_l)$ is the fraction of observations for which $X = v_l$. For problems that involve more than a small number of variables, each of which can assume more than a small number of values, the number of observations for which $X = v_l$ will nearly always be too small for reliable estimation. In order to have a tractable problem, both the goals of the analysis and the generality of the data to which it is applied must be greatly simplified.

The first simplification modifies the goal. Instead of seeking *values* x where $\Pr(x)$ is large, one seeks *regions* of the X-space with high probability

content relative to their size or support. Let $\mathcal{S}_j$ represent the set of all possible values of the jth variable (its *support*), and let $s_j \subseteq \mathcal{S}_j$ be a subset of these values. The modified goal can be stated as attempting to find subsets of variable values $s_1, \ldots, s_p$ such that the probability of each of the variables simultaneously assuming a value within its respective subset,

$$\Pr\left[\bigcap_{j=1}^{p}(X_j \in s_j)\right], \tag{14.2}$$

is relatively large. The intersection of subsets $\cap_{j=1}^{p}(X_j \in s_j)$ is called a *conjunctive rule*. For quantitative variables the subsets s_j are contiguous intervals; for categorical variables the subsets are delineated explicitly. Note that if the subset s_j is in fact the entire set of values $s_j = \mathcal{S}_j$, as is often the case, the variable X_j is said *not* to appear in the rule (14.2).

14.2.1 Market Basket Analysis

General approaches to solving (14.2) are discussed in Section 14.2.5. These can be quite useful in many applications. However, they are not feasible for the very large ($p \approx 10^4$, $N \approx 10^8$) commercial data bases to which market basket analysis is often applied. Several further simplifications of (14.2) are required. First, only two types of subsets are considered; either s_j consists of a *single* value of X_j, $s_j = v_{0j}$, or it consists of the entire set of values that X_j can assume, $s_j = \mathcal{S}_j$. This simplifies the problem (14.2) to finding subsets of the integers $\mathcal{J} \subset \{1, \ldots, p\}$, and corresponding values v_{0j}, $j \in \mathcal{J}$, such that

$$\Pr\left[\bigcap_{j\in\mathcal{J}}(X_j = v_{0j})\right] \tag{14.3}$$

is large. Figure 14.1 illustrates this assumption.

One can apply the technique of *dummy variables* to turn (14.3) into a problem involving only binary-valued variables. Here we assume that the support $\mathcal{S}_j$ is finite for each variable X_j. Specifically, a new set of variables $Z_1, \ldots, Z_K$ is created, one such variable for each of the values v_{lj} attainable by each of the original variables $X_1, \ldots, X_p$. The number of dummy variables K is

$$K = \sum_{j=1}^{p} |\mathcal{S}_j|,$$

where $|\mathcal{S}_j|$ is the number of distinct values attainable by X_j. Each dummy variable is assigned the value $Z_k = 1$ if the variable with which it is associated takes on the corresponding value to which Z_k is assigned, and $Z_k = 0$ otherwise. This transforms (14.3) to finding a subset of the integers $\mathcal{K} \subset \{1, \ldots, K\}$ such that

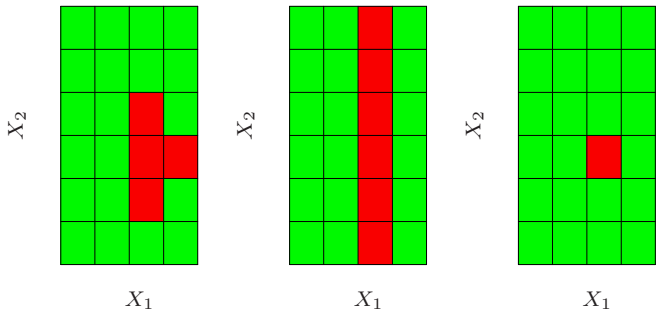

FIGURE 14.1. *Simplifications for association rules. Here there are two inputs X_1 and X_2, taking four and six distinct values, respectively. The red squares indicate areas of high density. To simplify the computations, we assume that the derived subset corresponds to either a single value of an input or all values. With this assumption we could find either the middle or right pattern, but not the left one.*

$$\Pr\left[\bigcap_{k\in\mathcal{K}}(Z_k = 1)\right] = \Pr\left[\prod_{k\in\mathcal{K}} Z_k = 1\right] \tag{14.4}$$

is large. This is the standard formulation of the market basket problem. The set $\mathcal{K}$ is called an "item set." The number of variables Z_k in the item set is called its "size" (note that the size is no bigger than p). The estimated value of (14.4) is taken to be the fraction of observations in the data base for which the conjunction in (14.4) is true:

$$\widehat{\Pr}\left[\prod_{k\in\mathcal{K}}(Z_k = 1)\right] = \frac{1}{N}\sum_{i=1}^{N}\prod_{k\in\mathcal{K}} z_{ik}. \tag{14.5}$$

Here z_{ik} is the value of Z_k for this ith case. This is called the "support" or "prevalence" $T(\mathcal{K})$ of the item set $\mathcal{K}$. An observation i for which $\prod_{k\in\mathcal{K}} z_{ik} = 1$ is said to "contain" the item set $\mathcal{K}$.

In association rule mining a lower support bound t is specified, and one seeks *all* item sets $\mathcal{K}_l$ that can be formed from the variables $Z_1, \ldots, Z_K$ with support in the data base greater than this lower bound t

$$\{\mathcal{K}_l | T(\mathcal{K}_l) > t\}. \tag{14.6}$$

14.2.2 The Apriori Algorithm

The solution to this problem (14.6) can be obtained with feasible computation for very large data bases provided the threshold t is adjusted so that (14.6) consists of only a small fraction of all 2^K possible item sets. The "Apriori" algorithm (Agrawal et al., 1995) exploits several aspects of the

curse of dimensionality to solve (14.6) with a small number of passes over the data. Specifically, for a given support threshold t:

- The cardinality $|\{\mathcal{K}|\, T(\mathcal{K}) > t\}|$ is relatively small.
- Any item set $\mathcal{L}$ consisting of a subset of the items in $\mathcal{K}$ must have support greater than or equal to that of $\mathcal{K}$, $\mathcal{L} \subseteq \mathcal{K} \Rightarrow T(\mathcal{L}) \geq T(\mathcal{K})$.

The first pass over the data computes the support of all single-item sets. Those whose support is less than the threshold are discarded. The second pass computes the support of all item sets of size two that can be formed from pairs of the single items surviving the first pass. In other words, to generate all frequent itemsets with $|\mathcal{K}| = m$, we need to consider only candidates such that *all* of their m ancestral item sets of size $m-1$ are frequent. Those size-two item sets with support less than the threshold are discarded. Each successive pass over the data considers only those item sets that can be formed by combining those that survived the previous pass with those retained from the first pass. Passes over the data continue until all candidate rules from the previous pass have support less than the specified threshold. The Apriori algorithm requires only one pass over the data for each value of $|\mathcal{K}|$, which is crucial since we assume the data cannot be fitted into a computer's main memory. If the data are sufficiently sparse (or if the threshold t is high enough), then the process will terminate in reasonable time even for huge data sets.

There are many additional tricks that can be used as part of this strategy to increase speed and convergence (Agrawal et al., 1995). The Apriori algorithm represents one of the major advances in data mining technology.

Each high support item set $\mathcal{K}$ (14.6) returned by the Apriori algorithm is cast into a set of "association rules." The items Z_k, $k \in \mathcal{K}$, are partitioned into two disjoint subsets, $A \cup B = \mathcal{K}$, and written

$$A \Rightarrow B. \tag{14.7}$$

The first item subset A is called the "antecedent" and the second B the "consequent." Association rules are defined to have several properties based on the prevalence of the antecedent and consequent item sets in the data base. The "support" of the rule $T(A \Rightarrow B)$ is the fraction of observations in the union of the antecedent and consequent, which is just the support of the item set $\mathcal{K}$ from which they were derived. It can be viewed as an estimate (14.5) of the probability of simultaneously observing both item sets $\Pr(A \text{ and } B)$ in a randomly selected market basket. The "confidence" or "predictability" $C(A \Rightarrow B)$ of the rule is its support divided by the support of the antecedent

$$C(A \Rightarrow B) = \frac{T(A \Rightarrow B)}{T(A)}, \tag{14.8}$$

which can be viewed as an estimate of $\Pr(B \,|\, A)$. The notation $\Pr(A)$, the probability of an item set A occurring in a basket, is an abbreviation for

$\Pr(\prod_{k\in A} Z_k = 1)$. The "expected confidence" is defined as the support of the consequent $T(B)$, which is an estimate of the unconditional probability $\Pr(B)$. Finally, the "lift" of the rule is defined as the confidence divided by the expected confidence

$$L(A \Rightarrow B) = \frac{C(A \Rightarrow B)}{T(B)}.$$

This is an estimate of the association measure $\Pr(A \text{ and } B)/\Pr(A)\Pr(B)$.

As an example, suppose the item set $\mathcal{K} = \{\texttt{peanut butter, jelly, bread}\}$ and consider the rule $\{\texttt{peanut butter, jelly}\} \Rightarrow \{\texttt{bread}\}$. A support value of 0.03 for this rule means that `peanut butter`, `jelly`, and `bread` appeared together in 3% of the market baskets. A confidence of 0.82 for this rule implies that when `peanut butter` and `jelly` were purchased, 82% of the time `bread` was also purchased. If bread appeared in 43% of all market baskets then the rule $\{\texttt{peanut butter, jelly}\} \Rightarrow \{\texttt{bread}\}$ would have a lift of 1.95.

The goal of this analysis is to produce association rules (14.7) with both high values of support and confidence (14.8). The Apriori algorithm returns all item sets with high support as defined by the support threshold t (14.6). A confidence threshold c is set, and all rules that can be formed from those item sets (14.6) with confidence greater than this value

$$\{A \Rightarrow B \,|\, C(A \Rightarrow B) > c\} \tag{14.9}$$

are reported. For each item set $\mathcal{K}$ of size $|\mathcal{K}|$ there are $2^{|\mathcal{K}|-1} - 1$ rules of the form $A \Rightarrow (\mathcal{K} - A)$, $A \subset \mathcal{K}$. Agrawal et al. (1995) present a variant of the Apriori algorithm that can rapidly determine which rules survive the confidence threshold (14.9) from all possible rules that can be formed from the solution item sets (14.6).

The output of the entire analysis is a collection of association rules (14.7) that satisfy the constraints

$$T(A \Rightarrow B) > t \quad \text{and} \quad C(A \Rightarrow B) > c.$$

These are generally stored in a data base that can be queried by the user. Typical requests might be to display the rules in sorted order of confidence, lift or support. More specifically, one might request such a list conditioned on particular items in the antecedent or especially the consequent. For example, a request might be the following:

> *Display all transactions in which ice skates are the consequent that have confidence over* 80% *and support of more than* 2%.

This could provide information on those items (antecedent) that predicate sales of ice skates. Focusing on a particular consequent casts the problem into the framework of supervised learning.

Association rules have become a popular tool for analyzing very large commercial data bases in settings where market basket is relevant. That is

when the data can be cast in the form of a multidimensional contingency table. The output is in the form of conjunctive rules (14.4) that are easily understood and interpreted. The Apriori algorithm allows this analysis to be applied to huge data bases, much larger that are amenable to other types of analyses. Association rules are among data mining's biggest successes.

Besides the restrictive form of the data to which they can be applied, association rules have other limitations. Critical to computational feasibility is the support threshold (14.6). The number of solution item sets, their size, and the number of passes required over the data can grow exponentially with decreasing size of this lower bound. Thus, rules with high confidence or lift, but low support, will not be discovered. For example, a high confidence rule such as `vodka` $\Rightarrow$ `caviar` will not be uncovered owing to the low sales volume of the consequent `caviar`.

14.2.3 Example: Market Basket Analysis

We illustrate the use of Apriori on a moderately sized demographics data base. This data set consists of $N = 9409$ questionnaires filled out by shopping mall customers in the San Francisco Bay Area (Impact Resources, Inc., Columbus OH, 1987). Here we use answers to the first 14 questions, relating to demographics, for illustration. These questions are listed in Table 14.1. The data are seen to consist of a mixture of ordinal and (unordered) categorical variables, many of the latter having more than a few values. There are many missing values.

We used a freeware implementation of the Apriori algorithm due to Christian Borgelt[1]. After removing observations with missing values, each ordinal predictor was cut at its median and coded by two dummy variables; each categorical predictor with k categories was coded by k dummy variables. This resulted in a 6876×50 matrix of 6876 observations on 50 dummy variables.

The algorithm found a total of 6288 association rules, involving ≤ 5 predictors, with support of at least 10%. Understanding this large set of rules is itself a challenging data analysis task. We will not attempt this here, but only illustrate in Figure 14.2 the relative frequency of each dummy variable in the data (top) and the association rules (bottom). Prevalent categories tend to appear more often in the rules, for example, the first category in language (English). However, others such as occupation are under-represented, with the exception of the first and fifth level.

Here are three examples of association rules found by the Apriori algorithm:

Association rule 1: Support 25%, confidence 99.7% and lift 1.03.

[1] See `http://fuzzy.cs.uni-magdeburg.de/~borgelt`.

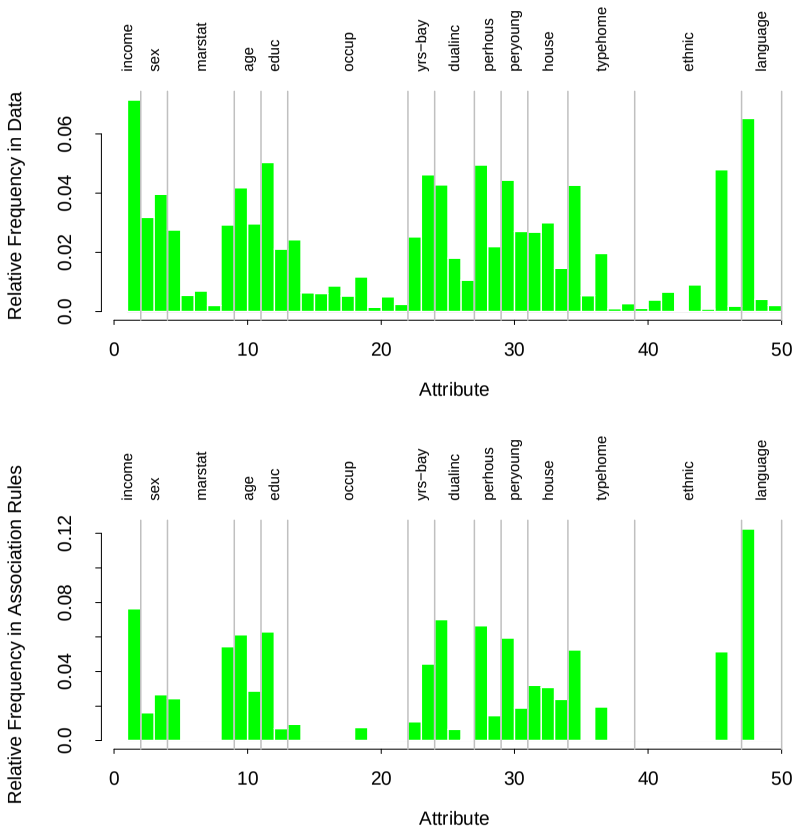

FIGURE 14.2. *Market basket analysis: relative frequency of each dummy variable (coding an input category) in the data (top), and the association rules found by the Apriori algorithm (bottom).*

TABLE 14.1. *Inputs for the demographic data.*

Feature	Demographic	# Values	Type
1	Sex	2	Categorical
2	Marital status	5	Categorical
3	Age	7	Ordinal
4	Education	6	Ordinal
5	Occupation	9	Categorical
6	Income	9	Ordinal
7	Years in Bay Area	5	Ordinal
8	Dual incomes	3	Categorical
9	Number in household	9	Ordinal
10	Number of children	9	Ordinal
11	Householder status	3	Categorical
12	Type of home	5	Categorical
13	Ethnic classification	8	Categorical
14	Language in home	3	Categorical

$$\begin{bmatrix} \text{number in household} & = & 1 \\ \text{number of children} & = & 0 \end{bmatrix}$$
$$\Downarrow$$
$$\text{language in home} = \textit{English}$$

Association rule 2: Support 13.4%, confidence 80.8%, and lift 2.13.

$$\begin{bmatrix} \text{language in home} & = & \textit{English} \\ \text{householder status} & = & \textit{own} \\ \text{occupation} & = & \{\textit{professional/managerial}\} \end{bmatrix}$$
$$\Downarrow$$
$$\text{income} \geq \textit{\$40,000}$$

Association rule 3: Support 26.5%, confidence 82.8% and lift 2.15.

$$\begin{bmatrix} \text{language in home} & = & \textit{English} \\ \text{income} & < & \textit{\$40,000} \\ \text{marital status} & = & \textit{not married} \\ \text{number of children} & = & 0 \end{bmatrix}$$
$$\Downarrow$$
$$\text{education} \notin \{\textit{college graduate, graduate study}\}$$

We chose the first and third rules based on their high support. The second rule is an association rule with a high-income consequent, and could be used to try to target high-income individuals.

As stated above, we created dummy variables for each category of the input predictors, for example, $Z_1 = I(\text{income} < \$40,000)$ and $Z_2 = I(\text{income} \geq \$40,000)$ for below and above the median income. If we were interested only in finding associations with the high-income category, we would include Z_2 but not Z_1. This is often the case in actual market basket problems, where we are interested in finding associations with the presence of a relatively rare item, but not associations with its absence.

14.2.4 Unsupervised as Supervised Learning

Here we discuss a technique for transforming the density estimation problem into one of supervised function approximation. This forms the basis for the generalized association rules described in the next section.

Let $g(x)$ be the unknown data probability density to be estimated, and $g_0(x)$ be a specified probability density function used for reference. For example, $g_0(x)$ might be the uniform density over the range of the variables. Other possibilities are discussed below. The data set $x_1, x_2, \ldots, x_N$ is presumed to be an *i.i.d.* random sample drawn from $g(x)$. A sample of size N_0 can be drawn from $g_0(x)$ using Monte Carlo methods. Pooling these two data sets, and assigning mass $w = N_0/(N + N_0)$ to those drawn from $g(x)$, and $w_0 = N/(N + N_0)$ to those drawn from $g_0(x)$, results in a random sample drawn from the mixture density $(g(x) + g_0(x))/2$. If one assigns the value $Y = 1$ to each sample point drawn from $g(x)$ and $Y = 0$ those drawn from $g_0(x)$, then

$$\begin{aligned} \mu(x) = E(Y \mid x) &= \frac{g(x)}{g(x) + g_0(x)} \\ &= \frac{g(x)/g_0(x)}{1 + g(x)/g_0(x)} \end{aligned} \tag{14.10}$$

can be estimated by supervised learning using the combined sample

$$(y_1, x_1), (y_2, x_2), \ldots, (y_{N+N_0}, x_{N+N_0}) \tag{14.11}$$

as training data. The resulting estimate $\hat{\mu}(x)$ can be inverted to provide an estimate for $g(x)$

$$\hat{g}(x) = g_0(x)\frac{\hat{\mu}(x)}{1 - \hat{\mu}(x)}. \tag{14.12}$$

Generalized versions of logistic regression (Section 4.4) are especially well suited for this application since the log-odds,

$$f(x) = \log \frac{g(x)}{g_0(x)}, \tag{14.13}$$

are estimated directly. In this case one has