Abstract

The task of inferring a set of classes and class descriptions most likely to explain a given data set can be placed on a firm theoretical foundation using Bayesian statistics. Within this framework, and using various mathematical and algorithmic approximations, the AutoClass system searches for the most probable classifications, automatically choosing the number of classes and complexity of class descriptions. A simpler version of AutoClass has been applied to many large real data sets, have discovered new independently-verified phenomena, and have been released as a robust software package. Recent extensions allow attributes to be selectively correlated within particular classes, and allow classes to inherit, or share, model parameters though a class hierarchy. In this paper we summarize the mathematical foundations of AutoClass.

1 Introduction

The task of supervised classification - i.e., learning to predict class memberships of test cases given labeled training cases - is a familiar machine learning problem. A related problem is unsupervised classification, where training cases are also unlabeled. Here one tries to predict all features of new cases; the best classification is the least "surprised" by new cases. This type of classification, related to clustering, is often very useful in exploratory data analysis, where one has few preconceptions about what structures new data may hold.

We have previously developed and reported on AutoClass [Cheeseman et al., 1988a; Cheeseman et al., 1988b], an unsupervised classification system based on Bayesian theory. Rather than just partitioning cases, as most clustering techniques do, the Bayesian approach searches in a model space for the "best" class descriptions. A best classification optimally trades off predictive accuracy against the complexity of the classes, and so does not "overfit" the data. Such classes are also "fuzzy"; instead of each case being assigned to a class, a case has a probability of being a member of each of the different classes.

2 Bayesian Learning

Bayesian theory gives a mathematical calculus of degrees of belief, describing what it means for beliefs to be consistent and how they should change with evidence. This section briefly reviews that theory, describes an approach to making it tractable, and comments on the resulting tradeoffs. In general, a Bayesian agent uses a single real number to describe its degree of belief in each proposition of interest. This assumption, together with some other assumptions about how evidence should affect beliefs, leads to the standard probability axioms. This result was originally proved by Cox [Cox, 1946] and has been reformulated for an AI audience [Heckerman, 1990]. We now describe this theory.

2.1 Theory

Let E denote some evidence that is known or could potentially be known to an agent; let H denote a hypothesis specifying that the world is in some particular state; and let the sets of possible evidence E and possible states of the world H each be mutually exclusive and exhaustive sets. For example, if we had a coin that might be
two-headed the possible states of the world might be "ordinary coin", "two-headed coin". If we were to toss it once the possible evidence would be "lands heads", "lands tails".

In general, \( P(a|b,c,d) \) denotes a real number describing an agent's degree of belief in the conjunction of propositions \( a \) and \( b \), conditional on the assumption that propositions \( c \) and \( d \) are true. The propositions on either side of the conditioning bar "|" can be arbitrary Boolean expressions. More specifically, \( \pi(E) \) is a "prior" describing the agent's belief in \( E \) before, or in the absence of, seeing evidence \( E \). \( \pi(E|H) \) is a "posterior" describing the agent's belief after observing some particular evidence \( E \), and \( L(E|H) \) is a "likelihood" embodying the agent's theory of how likely it would be to see each possible evidence combination \( E \) in each possible world \( H \).

To be consistent, beliefs must be non-negative, \( 0 \leq P(a|b) \leq 1 \), and normalised, so that \( \sum_H \pi(H) = 1 \) and \( \sum_H L(E|H) = 1 \). That is, the agent is sure that the world is in some state and that some evidence will be observed. The likelihood and the prior together give a "joint" probability \( J(E|H) = L(E|H)\pi(H) \) of both \( E \) and \( H \). Normalising the joint gives Bayes' rule, which tells how beliefs should change with evidence:

\[
\pi(H|E) = \frac{J(E|H)}{\sum_H J(E|H)} = \frac{L(E|H)\pi(H)}{\sum_H L(E|H)\pi(H)}.
\]

When the set of possible \( H \)s is continuous, the prior \( \pi(H) \) becomes a differential \( d\pi(H) \), and the sums over \( H \) are replaced by integrals. Similarly, continuous \( E \)s have a differential likelihood \( dL(E|H) \), though any real evidence \( \Delta E \) will have a finite probability \( \Delta J(E|H) \approx dL(E|H) \Delta\pi(H) \).

In theory, all an agent needs to do in any given situation is to choose a set of states \( H \), an associated likelihood function describing what evidence is expected to be observed in those states, a set of prior expectations on the states, and then collect some relevant evidence. Bayes' rule then specifies the appropriate posterior beliefs about the state of the world, which can be used to answer most questions of interest. An agent can combine these posterior beliefs with its utility over states \( U(H) \), which says how much it prefers each possible state, to choose an action \( A \) which maximises its expected utility

\[
EU(A) = \sum_H U(H)\pi(H|EA).
\]

### 2.2 Practice

In practice this theory can be difficult to apply, as the sums and integrals involved are often mathematically intractable. So one must use approximations. Here is our approach.

Rather than consider all possible states of the world, we focus on some smaller space of models, and do all of our analysis conditional on an assumption \( S \) that the world really is described by one of the models in our space. As with most modeling, this assumption is almost certainly false, but it makes the analysis tractable. With time and effort we can make our models more complex, expanding our model space in order to reduce the effect of this simplification.

The parameters which specify a particular model are split into two sets. First, a set of discrete parameters \( T \) describe the general form of the model, usually by specifying some functional form for the likelihood function. For example, \( T \) might specify whether two variables are correlated or not, or how many classes are present in a classification. Second, free variables in this general form, such as the magnitude of the correlation or the relative sizes of the classes, constitute the remaining continuous model parameters \( V \).

We generally prefer a likelihood \( L(E|VTS) \) which is mathematically simple and yet still embodies the kinds of complexity we believe to be relevant. Similarly, we prefer a simple prior distribution \( \pi(VTS) \) over this model space, allowing the resulting \( V \) integrals, described below, to be at least approximated. A prior that predicts the different parameters in \( V \) independently, through a product of terms for each different parameter, often helps. We also prefer the prior to be as broad and uninformative as possible, so our software can be used in many different problem contexts, though in principal we could add specific domain knowledge through an appropriate prior. Finally we prefer a prior that gives nearly equal weight to different levels of model complexity, resulting in a "significance test". Adding more parameters to a model then induces a cost, which must be paid for by a significantly better fit to the data before the more complex model is preferred.

Sometimes the integrable priors are not broad enough, containing meta-parameters which specify some part of model space to focus on, even though we have no prior expectations about where to focus. In these cases we "cheat" and use simple statistics collected from the evidence we are going to use, to help set these priors. For example, see Sections 4.2, 4.5.

The joint can now be written as \( dJ(EVTS) = L(E|VTS)\pi(VTS) \) and, for a reasonably-complex problem, is usually a very rugged distribution in \( VT \), with an immense number of sharp peaks distributed widely over a huge high-dimensional space. Because of this we despair of directly normalising the joint, as required by Bayes' rule, or of communicating the detailed shape of the posterior distribution.

Instead we break the continuous \( V \) space into regions \( R \) surrounding each sharp peak, and search until we tire for combinations \( RT \) for which the "marginal" joint

\[
M(ERT|S) \equiv \int_{V \in R} dJ(EVTS)
\]

is as large as possible. The best few such "models" \( RT \) are then reported, even though it is usually almost certain that more probable models remain to be found.

Each model \( RT \) is reported by describing its marginal joint \( M(ERT|S) \), its discrete parameters \( T \), and estimates of typical values of \( V \) in the region \( R \), like the mean estimate of \( V \):

\[
E(V|ERTS) \equiv \frac{\int_{V \in R} V dJ(EVTS)}{M(ERT|S)}
\]

1Note that when a variable like \( V \) sits in a probability expression where a proposition should be, it stands for a proposition that the variable has a particular value.

2This is cheating because the prior is supposed to be independent of evidence.
or the \( V \) for which \( dJ(EVT|S) \) is maximum in \( R \). While these estimates are not invariant under reparameterizations of the \( V \) space, and hence depend on the syntax with which the likelihood was expressed, the peak is usually sharp enough that such differences don't matter.

Reporting only the best few models is usually justified, since the models weaker than this are usually many orders of magnitude less probable than the best one. The main reason for reporting models other than the best is to show the range of variation in the models, so that one can judge how different the better, not yet found, models might be.

The decision to stop searching for better models \( R \) than the current best can often be made in a principled way by using estimates of how much longer it would take to find a better model, and how much better than model would be. If the fact that a data value is unknown might be informative, one can model "unknown" as just another possible (discrete) data value; otherwise the likelihood for an unknown value is just a sum over the possible known values.

To make predictions with these resulting models, a reasonable approximation is to average the answer from the best few peaks, weighted by the relative marginal joints. Almost all of the weight is usually in the best few, justifying the neglect of the rest.

2.3 Tradeoffs

Bayesian theory offers the advantages of being theoretically well-founded and empirically well-tested [Berger, 1985]. It offers a clear procedure whereby one can almost "turn the crank", modulo doing integrals and search, to deal with any new problem. The machinery automatically trades off the complexity of a model against its fit to the evidence. Background knowledge can be included in the input, and the output is a flexible mixture of several different "answers," with a clear and well-founded decision theory [Berger, 1985] to help one use that output.

Disadvantages include being forced to be explicit about the space of models one is searching in, though this can be good discipline. One must deal with some difficult integrals and sums, although there is a huge literature to help one here. And one must often search large spaces, though most any technique will have to do this and the joint probability provides a good local evaluation function. Finally, it is not clear how one can take the computational cost of doing a Bayesian analysis into account without a crippling infinite regress.

Some often perceived disadvantages of Bayesian analysis are really not problems in practice. Any ambiguities in choosing a prior are generally not serious, since the various possible convenient priors usually do not disagree strongly within the regions of interest. Bayesian analysis is not limited to what is traditionally considered "statistical" data, but can be applied to any space of models about how the world might be. For a general discussion of these issues, see [Cheeseman, 1990].

We will now illustrate this general approach by applying it to the problem of unsupervised classification.

3 Model Spaces Overview

3.1 Conceptual Overview

In this paper we deal only with attribute-value, not relational, data. For example, medical cases might be described by medical forms with a standard set of entries or slots. Each slot could be filled only by elements of some known set of simple values, like numbers, colors, or blood-types. (In this paper, we will only deal with real and discrete attributes.)

We would like to explain this data as consisting of a number of classes, each of which corresponds to a differing underlying cause for the symptoms described on the form. For example, different patients might fall into classes corresponding to the different diseases they suffer from.

To do a Bayesian analysis of this, we need to make this vague notion more precise, choosing specific mathematical formulas which say how likely any particular combination of evidence would be. A natural way to do this is to say that there are a certain number of classes, that a random patient has a certain probability to come from each of them, and that the patients are distributed independently—once we know all about the underlying classes then learning about one patient doesn't help us learn what any other patient will be like.

In addition, we need to describe how each class is distributed. We need a "single class" model saying how likely any given evidence is, given that we know what class the patient comes from. Thus we build the multi-class model space from some other pre-existing model space, which can be arbitrarily complex. (In fact, much of this paper will be spend describing various single class models.) In general, the more complex each class can be, the less of a need there is to invoke multiple classes to explain the variation in the data.

The simplest way to build a single-class model is to predict each attribute independently, i.e., build it from attribute-specific models. A class has a distribution for each attribute and, if you know the class of a case, learning the values of one attribute doesn't help you predict the value of any other attributes. For real attributes one can use a standard normal distribution, characterized by some specific mean and a variance around that mean. For discrete attributes one can use the standard multinomial distribution, characterized by a specific probability for each possible discrete value.

Up to this point we have described the model space of Autoclass III. Autoclass IV goes beyond this by introducing correlation and inheritance. Correlation is introduced by removing the assumption that attributes are independent within each class. The simplest way to do this is to let all real attributes covary, and let all discrete attributes covary. The standard way for real attributes to covary is the multivariate normal, which basically says that there is some other set of attributes one could define, as linear combinations of the attributes given, which vary independently according to normal distributions. A simple way to let discrete attributes covary is to define one super-attribute whose possible values are all possible.
combinations of the values of the attributes given.

If there are many attributes, the above ways to add correlation introduce a great many parameters in the models, making them very complex and, under the usual priors, much less preferable than simpler independent models. What we really want are simpler models which only allow partial covariance. About the simplest way to do this is to say that, for a given class, the attributes clump together in blocks of inter-related attributes. All the attributes in a block covary with each other, but not with the attributes in other blocks. Thus we can build a block model space from the covariant model spaces.

Even this simpler form of covariance introduces more parameters that the independent case, and when each class must have its own set of parameters, multiple classes are penalised more strongly. Attributes which are irrelevant to the whole classification, like a medical patient's favorite color, can be particularly costly. To reduce this cost, one can allow classes to share the specification of parameters associated with some of their independent blocks. Irrelevant attributes can then be shared by all classes at a minimum cost.

Rather than allow arbitrary combinations of classes to share blocks, it is simpler to organise the classes as leaves of a tree. Each block can be placed at some node in this tree, to be shared by all the leaves below that node. In this way different attributes can be explained at different levels of an abstraction hierarchy. For medical patients the tree might have “viral infections” near the root, predicting fevers, and some more specific viral disease near the leaves, predicting more disease specific symptoms. Irrelevant attributes like favorite-color would go at the root.

3.2 Notation Summary

For all the models to be considered in this paper, the evidence $E$ will consist of a set of $I$ cases, an associated set $K$ of attributes, of size $K$, and case attribute values $X_{ik}$, which can include “unknown.” For example, medical case number 8, described as (age $= 23$, blood-type $= A$, ...), would have $X_{8k} = 23, X_{83} = A$, etc.

In the next two sections we will describe applications of Bayesian learning theory to various kinds of models which could explain this evidence, beginning with simple model spaces and building more complex spaces from them. We begin with a single class. First, a single attribute is considered, then multiple independent attributes, then fully covariant attributes, and finally selective covariance. In the next section we combine these single classes into class mixtures. Table 1 gives an overview of the various spaces.

For each space $S$ we will describe the continuous parameters $V$, any discrete model parameters $T$, normalised likelihoods $dL(E|VTS)$, and priors $dV(VT|S)$. Most spaces have no discrete parameters $T$, and only one region $R$, allowing us to usually ignore these parameters. Approximations to the resulting marginals $M(E|ERTS)$ and estimates $E(V|ERTS)$ will be given, but not derived. These will often be given in terms of general functions $F$, so that they may be reused later on. As appropriate, comments will be made about algorithms and computational complexity. All of the likelihood functions considered here assume the cases are independent, i.e.,

$$L(E|VTS) = \prod_i L(E_i|VTS)$$

so we need only give $L(E_i|VTS)$ for each case, where $E_i = \{X_{1i}, X_{2i}, X_{3i}, \ldots, X_{K_i}\}$.

4 Single Class Models

4.1 Single Discrete Attribute - $S_{D1}$

A discrete attribute $k$ allows only a finite number of possible values $l \in \{1, 2, \ldots, L\}$ for any $X_i$. “Unknown” is usually treated here as just another possible value. A set of independent coin tosses, for example, might have $L = 3$ with $l_1 = \text{heads}$, $l_2 = \text{tails}$, and $l_3 = \text{unknown}$. We make the assumption $S_{D1}$ that there is only one discrete attribute, and that the only parameters are the continuous parameters $V = q_1 \ldots q_L$ consisting of the likelihoods $L(X_i|VS_{D1}) = q_{il}X_i$ for each possible value $l$. In the coin example, $q_1 = .7$ would say that the coin was so “unbalanced” that it has a 70 percent chance of coming up heads each time.

There are only $L - 1$ free parameters since normalisation requires $\sum_l q_l = 1$. For this likelihood, all that matters from the data are the number of cases with each value $^3 I_l = \sum_i \delta_{xl}$. In the coin example, $I_1$ would be the number of heads. Such sums are called “sufficient statistics” since they summarise all the information relevant to a model.

We choose a prior

$$dV(V|S_{D1}) = dB(q_1 \ldots q_L|L) \equiv \frac{\Gamma(aL)}{\Gamma(a)^L} \prod_l q_l^{a-1} dq_l$$

which for $a > 0$ is a special case of a beta distribution [Berger, 1985] ($\Gamma(y)$ is the Gamma function [Spiegel, 1968]). This formula is parameterised by $a$, a “hyperparameter” which can be set to different values to specify different priors. Here we set $a = 1/L$. This simple problem has only one maximum, whose marginal is given by

$$M(V|S_{D1}) = F_1(I_1, \ldots, I_L, L) \equiv \frac{\Gamma(aL)}{\Gamma(aL + L)} \prod I_l^{a+1}$$

We have abstracted the function $F_1$, so we can refer to it later. The prior above was chosen because it has a form similar to the likelihood (and therefore a “conjugate” prior), and to make the following mean estimate of $q_l$ particularly simple

$$E(q_l|ES_{D1}) = F_2(I_l, I, L) \equiv \frac{I_l + \frac{1}{a}}{I + aL} = \frac{I_l + \frac{1}{a}}{I + 1}$$

for $a = 1/L$. $F_2$ is also abstracted out for use later. Note that while $F_3(I_l, I, L)$ is very similar to the classical estimate of $\hat{q}_l$, $F_2$ is defined even when $I = 0$. Using a hash table, these results can be computed in order $I$ numerical steps, independent of $L$.

Note that $\delta_{uv}$ denotes 1 when $u = v$ and 0 otherwise.
4.2 Single Real Attribute - $S_{R1}$

Real attribute values $X_i$ specify a small range of the real line, with a center $z_i$ and a precision, $\Delta z_i$, assumed to be much smaller than other scales of interest. For example, someone's weight might be measured as 70±1 kilograms. For scalar attributes, which can only be positive, like weight, it is best to use the logarithm of that variable [Aitchison and Brown, 1957].

For $S_{R1}$, where there is only one real attribute, we assume the standard normal distribution, where the sufficient statistics are the data mean $\bar{z} = \frac{1}{\alpha} \sum z_i$, the geometric mean precision $\Delta \bar{z} = \left( \prod \Delta z_i \right)^\frac{1}{\alpha}$ and the standard deviation $s$ given by $s^2 = \frac{1}{\alpha} \sum (z_i - \bar{z})^2$. $V$ consists of a model mean $\mu$ and deviation $\sigma$, and the likelihood is given by the standard normal distribution.

$$dL(z_i|VS_{R1}) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(z_i - \bar{z})^2}{2\sigma^2}} dz_i.$$  

For example, people's weight might be distributed with a mean of 80 kilograms and a deviation of 15. Since all real data have a finite width, we replace $dz_i$ with $\Delta z_i$ to approximate the likelihood $\Delta L(X_i|VS_{R1}) = \int_{\Delta z_i} dL(z_i|VS_{R1}) \approx \Delta z_i dz_i(dL(z_i|VS_{R1}))$.

As usual, we choose priors that treat the parameters in $V$ independently.

$$d\pi(V|S_{R1}) = d\pi(\mu|S_{R1}) d\pi(\sigma|S_{R1})$$  

We choose a prior on the mean to be flat in the range of the data,

$$d\pi(\mu|S_{R1}) = dR(\mu|\mu^+, \mu^-)$$  

where $\mu^+ = \max z_i$, $\mu^- = \min z_i$, by using the general uniform distribution

$$dR(y|y^+, y^-) \equiv \frac{dy}{y^+ - y} \text{ for } y \in [y^-, y^+]$$  

A flat prior is preferable because it is non-informative, but note that in order to make it normalizable we must cheat and use information from the data to cut it off at some point. In the single attribute case, we can similarly choose a flat prior in $\log(\sigma)$.

$$d\pi(\sigma|S_{R1}) = dR(\log(\sigma)|\log(\Delta \mu), \log(\min \Delta z_i))$$  

where $\Delta \mu = \mu^+ - \mu^-$. The posterior again has just one peak, so there is only one region $R$, and the resulting marginal is

$$M(E|S_{R1}) = \frac{\Gamma(\frac{L-1}{2})}{2(\pi)^{\frac{L}{2}}} \log(\Delta \mu/\min \Delta z_i)^{L-1} \Delta z_i^L.$$  

Note that this joint is dimensionless. The estimates are simply $E(\mu|ES_{R1}) = \bar{z}$, and $E(\sigma|E) = \sqrt{\frac{1}{\sum (z_i - \bar{z})^2}}$. Computation here takes order $I$ steps, used to compute the sufficient statistics.

4.3 Independent Attributes - $S_I$

We now introduce some notation for collecting sets of indexed terms like $X_{ik}$. A single term inside $\{\}$ will denote the set of all such indexed terms collected across all of the indices, like $i$ and $k$ in $E = \{X_{ik}\}$ such that $i \in [1, \ldots, I, k \in K]$. To collect across only some of the indices we use $\cup_{i}$ as in $E_i = \cup_{i} X_{ik} \equiv \{X_{i1}, X_{i2}, \ldots\}$, all the evidence for a single case $i$.

The simplest way to deal with cases having multiple attributes is to assume $S_I$ that they are all independent, i.e., treating each attribute as if it were a separate problem. In this case, the parameter set $V$ partitions into parameter sets $V_R = \cup_{E_R} q_{R}$ or $[\mu_R, \sigma_R]$, depending on whether $k$ is discrete or real. The likelihood, prior, and joint for multiple attributes are all simple products of the results above for one attribute: $S_I = S_{D1}$ or $S_{R1}$ — i.e.,

$$L(E_i|VS_I) = \prod_k L(X_{ik}|V_kS_i),$$  

$$d\pi(V|S_I) = \prod_k d\pi(V_k|S_i),$$  

and

$$M(E|S_I) = \prod_k J(E(k)|S_i)$$  

where $E(k) \equiv \cup_{i} X_{ik}$, all the evidence associated with attribute $k$. The estimates $E(V_k|ES_I) = E(V_k|E(k)|S_I)$ are exactly the same. Computation takes order $IK$ steps here.

4.4 Fully Covariant Discrete - $S_D$

A model space $S_D$ which allows a set $K$ of discrete attributes to fully covariate (i.e., contribute to a likelihood in non-trivial combinations) can be obtained by treating all combinations of base attribute values as particular values of one super attribute, which then has $L' = \prod_k L_k$ values — so $L'$ can be a very large number! $V$ consists

<table>
<thead>
<tr>
<th>Space</th>
<th>Description</th>
<th>$V$</th>
<th>$T$</th>
<th>$R$</th>
<th>Subspaces</th>
<th>Compute Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_{D1}$</td>
<td>Single Discrete</td>
<td>$q_i$</td>
<td></td>
<td></td>
<td>$I$</td>
<td>$I$</td>
</tr>
<tr>
<td>$S_{R1}$</td>
<td>Single Real</td>
<td>$\mu \sigma$</td>
<td></td>
<td></td>
<td>$S_1 \equiv S_{D1} \text{ or } S_{R1}$</td>
<td>$IK$</td>
</tr>
<tr>
<td>$S_I$</td>
<td>Independent Attrs</td>
<td>$V_k$</td>
<td></td>
<td></td>
<td>$I$</td>
<td>$I$</td>
</tr>
<tr>
<td>$S_D$</td>
<td>Covariant Discrete</td>
<td>$q_{i1}l_{i2} \ldots$</td>
<td></td>
<td></td>
<td>$IK$</td>
<td></td>
</tr>
<tr>
<td>$S_R$</td>
<td>Covariant Real</td>
<td>$\mu \Sigma_{kk'}$</td>
<td></td>
<td></td>
<td>$S_R \equiv S_{D1} \text{ or } S_{R1}$</td>
<td>$(I + K)K^2$</td>
</tr>
<tr>
<td>$S_V$</td>
<td>Block Covariance</td>
<td>$V_0$</td>
<td>$Bk_0$</td>
<td></td>
<td>$R$ $S_R \equiv S_{D1} \text{ or } S_{R1}$</td>
<td>$N(K(K_0 + K_1))$</td>
</tr>
<tr>
<td>$S_M$</td>
<td>Flat Class Mixture</td>
<td>$\alpha V_c$</td>
<td>$C$</td>
<td>$R$</td>
<td>$S_{C} \equiv S_I \text{ or } S_V$</td>
<td>$N(KK(C_0K_1) + K_1^2)$</td>
</tr>
<tr>
<td>$S_H$</td>
<td>Tree Class Mixture</td>
<td>$\alpha V_c$</td>
<td>$J_k K_j T_c$</td>
<td>$R$</td>
<td>$S_{C} \equiv S_I \text{ or } S_V$</td>
<td>$N(KK(C_0K_1) + K_1^2)$</td>
</tr>
</tbody>
</table>
of terms like \( q_{l_1, l_2, \ldots, l_K} \), indexed by all the attributes. \( I_l \) generalises to

\[
I_{l_1, l_2, \ldots, l_K} = \sum_{i=1}^{K} \delta_{s_{l_i} a_i}.
\]

Given this transformation, the likelihoods, etc. look the same as before:

\[
L(E_l \mid VSD) = q_{l_1, l_2, \ldots, l_K},
\]

where each \( l_k \) is \( X_{ik} \),

\[
d\pi(V \mid S_D) = dB(\{q_{l_1, l_2, \ldots, l_K}\} \mid L'),
\]

\[
M(E \mid S_D) = \pi_l(\{I_{l_1, l_2, \ldots, l_K}\}, I, L'),
\]

and

\[
\mathcal{E}(q_{l_1, l_2, \ldots, l_K} \mid ESD) = F_2(I_{l_1, l_2, \ldots, l_K}, I, L').
\]

Computation takes order \( IK \) steps here. This model could, for example, use a single combined hair-color-eye-color attribute to allow a correlation between people being blond and blue-eyed.

4.5 Fully Covariant Reals - \( S_R \)

If we assume \( S_R \) that a set \( K \) of real-valued attributes follow the multivariate normal distribution, we replace the \( \phi_T \) above with a model covariance matrix \( \Sigma_{kk} \) and \( s_i^2 \) with a data covariance matrix

\[
S_{kk'} = \frac{1}{I} \sum (z_{i_k} - z_k)(z_{i_{k'}} - z_{k'})
\]

The \( \Sigma_{kk'} \) must be symmetric, with \( \Sigma_{kk'} = \Sigma_{k'k} \), and "positive definite", satisfying \( \sum_{k'k} y_k \Sigma_{kk'} y_{k'} > 0 \) for any vector \( y_k \). The likelihood for a set of attributes \( K \) is

\[
dL(E_l \mid VS_R) = dN(E_l, \{\mu_k\}, \{\Sigma_{kk'}\}, K)
\]

\[
e^{-\frac{1}{2} \sum_{k} G_k(\phi_k-\mu_k)\Sigma_{kk'}(\phi_k-\mu_k)}
\]

\[
\frac{(2\pi)^{\frac{K}{2}} |\Sigma_{kk'}|^\frac{1}{2}}{\prod dx_k}
\]

is the multivariate normal in \( K \) dimensions.

As before, we choose a prior that takes the means to be independent of each other, and independent of the covariance

\[
d\pi(V \mid S_R) = d\pi(\{\Sigma_{kk'}\} \mid S_R) \prod \pi(\mu_k \mid S_{R1}),
\]

so the estimates of the means remain the same, \( E(\mu_k \mid S_R) = \bar{z}_k \). We choose the prior on \( \Sigma_{kk'} \) to use an inverse Wishart distribution [Mardia et al., 1979]

\[
d\pi(\{\Sigma_{kk'}\} \mid S_R) = dW_{\Sigma}((\Sigma_{kk'}), \{G_{kk'}\}, h) \equiv
\]

\[
\frac{|\Sigma_{kk'}|^{-\frac{1}{2}} \Gamma\left(K-1\right)^{-1} \prod G_{kk'} \Gamma\left(K-h-1\right)}{2^h \pi^{\frac{h}{2}}} \prod_{k < k'} \Sigma_{kk'}^{h-1} \Sigma_{k'k'}^{h-1} \prod d\Sigma_{kk'}
\]

which is normalised (integrates to 1) for \( h \geq K \) and \( \Sigma_{kk'} \) symmetric positive definite. This is a conjugate prior, meaning that it makes the resulting posterior \( d\pi(\{\Sigma_{kk'}\} \mid ESD) \) take the same mathematical form

\[\delta_{s_{l_i} a_i}\]

\[F_1\] and \( F_2 \) are defined on page 4.

\[\Sigma_{kk'}^{-1}\] denotes the matrix inverse of \( \Sigma_{kk'} \) satisfying \( \sum_{k} \Sigma_{kk'} \Sigma_{kk'} = \delta_{aa} \), and \( |\Sigma_{kk'}| \) denotes components of the matrix determinant of \( \{\Sigma_{kk'}\} \).

as the prior. This choice makes the resulting integrals manageable, but requires us to choose an \( h \) and all the components of \( G_{kk'} \). We choose \( h = K \) to make the prior as broad as possible, and for \( G_{kk'} \) we "cheat" and choose \( G_{kk'} = S_{kk} \delta_{aa} \), in order to avoid overly distorting the resulting marginal

\[
M(E \mid S_R) = \prod_{k} \frac{\Gamma\left(K-1\right)}{\Gamma\left(K-h-1\right)} \prod dx_k
\]

and estimates

\[
E(\Sigma_{kk'} \mid ESD) = \frac{IS_{kk'} + G_{kk'}}{I + h - K - 2} = \frac{I + \delta_{kk'}}{I - 2}
\]

If we choose \( G_{kk'} \) too large it dominates the estimates, and if \( G_{kk'} \) is too small the marginal is too small. The compromise above should only over estimate the marginal somewhat, since it in effect pretends to have seen previous data which agrees with the data given. Note that the estimates are undefined unless \( I > 2 \). Computation here takes order \( (I + K)K^2 \) steps. At present, we lack a satisfactory way to approximate the above marginal when some values are unknown.

4.6 Block Covariance - \( S_V \)

Rather than just having either full independence or full dependence of attributes, we prefer a model space \( S_V \) where some combinations of attributes may covary while others remain independent. This allows us to avoid paying the cost of specifying covariance parameters when they cannot buy us a significantly better fit to the data.

We partition the attributes \( K \) into \( B \) blocks \( K_b \), with full covariance within each block and full independence between blocks. Since we presently lack a model allowing different types of attributes to covary, all the attributes in a block must be of the same type. Thus real and discrete types may not mutually covary.

We are away from other models of partial dependence, such as the trees of Chow and Liu described in [Pearl, 1988], but choose this approach because it includes the limiting cases of full dependence and full independence.

The evidence \( E \) partitions block-wise into \( E(K_b) \) (using \( E(K_b) \equiv \bigcup_{\{k \in K \} \in K_b} E_k(K_b) \)), each with its own sufficient statistics; and the parameters \( V \) partition into parameters \( V_b \) (or \( \{\Sigma_{kk'}\}, \{\mu_k\} \)). Each block is treated as a different problem, except that we now also have discrete parameters \( T \) to specify which attributes covary, by specifying \( B \) blocks and \( \{K_b\} \) attributes in each block. Thus the likelihood

\[
L(E_l \mid VTS_V) = \prod_{b} L(E_l(K_b) \mid V_bS_B)
\]

is a simple product of block terms \( S_B = S_D + S_R \) assuming full covariance within each block, and the estimates \( E(V_b \mid ETS_V) = E(V_b \mid E(K_b)S_B) \) are the same as before.

We choose a prior which predicts the block structure \( B \{K_b\} \) independently of the parameters \( V_b \) within each independent block

\[
d\pi(V \mid TS_V) = \pi(B \{K_b\} \mid S_V) \prod d\pi(V_b \mid S_B)
\]
which results in a similarly decomposed marginal
\[ M(ET|SV) = \pi(B|K_a|SV) \prod_i M(E_i|K_B). \]

We choose a block structure prior
\[ \pi(B|K_a|SV) = 1/K_B Z(K_a,B_B) Z(D,B_D), \]
where \( K_a \) is the set of real attributes and \( B_B \) is the number of real blocks (and similarly for \( K_D \) and \( B_D \)). This says that it is equally likely that there will be one or two or three, etc. blocks, and, given the number of blocks, each possible way to group attributes is equally likely. This is formalized using \( Z(A,U) \), given by
\[ Z(A,U) = \sum_u (-1)^{u-1} \frac{(U-u+1)^A}{(U-u+1)(u-1)!}, \]
which gives the number of ways one can partition a set with \( A \) elements into \( U \) subsets. This prior prefers the special cases of full covariance and full independence, since there are fewer ways to make these block combinations. For example, in comparing the hypothesis that each attribute is in a separate block (i.e., all independent) with the hypothesis that only one particular pair of attributes covaries together in a block of size two, this prior will penalize the covariance hypothesis in proportion to the number of such pairs possible. Thus this prior includes a "significance test", so that a covariance hypothesis will only be chosen if the added fit to the data from the extra covariance is enough to overcome this penalty.

Computation here takes order \( NK((K_a + K_B)^2) \) steps, where \( N \) is the number of search trials done before quitting, which would be around \((K-1)!\) for a complete search of the space. \( K_a \) is an average, over both the search trials and the attributes, of the block size of real attributes (and unity for discrete attributes).

5 Class Mixtures

5.1 Flat Mixtures - \( S_M \)

The above model spaces \( S_C = SV \) or \( S_I \) can be thought of as describing a single class, and so can be extended by considering a space \( S_M \) of simple mixtures of such classes [D.M.Titterington et al., 1985]. Figure 1 shows how this model, with \( S_C = S_I \), can fit a set of artificial real-valued data in five dimensions.

In this model space the likelihood
\[ L(E|VTS_M) = \sum_c \alpha_c L(E|VcTS_c) \]
sums over products of "class weights" \( \alpha_c \), that give the probability that any case would belong to class \( c \) of the \( C \) classes, and class likelihoods describing how members of each class are distributed. In the limit of large \( C \) this model space is general enough to be able to fit any distribution arbitrarily closely, and hence is "asymptotically correct".

The parameters \( T = [C, \{T_c\}] \) and \( V = [\{\alpha_c\}, \{V_c\}] \) combine parameters for each class and parameters describing the mixture. The prior is similarly broken down as
\[ d\pi(VTS_M) = F_3(C)C! \, dB(\{\alpha_c\} | C) \prod_c d\pi(VcTS_c) \]
give the probability that a particular case \( i \) is a member of class \( c \). These weights satisfy \( \sum_c \omega_{ic} = 1 \), since every case must really belong to one of the classes. Using these weights we can break each case into "fractional cases", assign these to their respective classes, and create new "class data" \( E_c = \bigcup_{ic} [X_{ic}, \omega_{ic}] \) with new weighted-class sufficient statistics obtained by using weighted sums \( \sum_i \omega_{ic} \) instead of sums \( \sum_i \). For example \( I_t = \sum_i \omega_{ic}, \sum_i \omega_{ic} \bar{x}_{ic} = \bar{x}_{ic} \sum_i \omega_{ic}, \sum_i \omega_{ic} \bar{i}_{ic} = \sum_i \omega_{ic} \bar{i}_{ic} \sum_{i=1}^t \bar{i}_{ic} \) and \( \sum_i \omega_{ic} \Delta x_{ic} = \sum_i \omega_{ic} \Delta x_{ic} \sum_{i=1}^t \Delta x_{ic} \) . Substituting these statistics into any previous class likelihood function \( L(E|VTS) \) gives a weighted likelihood \( L(E|VcTS_c) \) and associated new estimates and marginals.

At the maxima, the weights \( \omega_{ic} \) should be consistent with estimates of \( V = [\{\alpha_c\}, C] \) from \( E(V|ERS) = E'(V|E'SC) \) and \( E(\alpha_c|ERS) = F_3(I_c, I, C) \) . To reach a maxima we start out at a random seed and repeatedly use our current best estimates of \( V \) to compute the \( \omega_{ic} \), and then use the \( \omega_{ic} \) to re-estimate the \( V \), stopping when they both predict each other. Typically this takes 10–100 iterations. This procedure will converge from any starting point, but converges more slowly near the peak than second-order methods.

Integrating the joint in \( R \) can't be done directly because the product of a sum in the full likelihood is hard to decompose, but if we use fractional cases to approxi-
mate the likelihood

\[ L(E_i|\text{VTRS}_m) = \sum_c \alpha_c L(E_i|V_i,T_i,SC_c) \]

\[ \approx \prod_c (\alpha_c L(E_i|V_i,T_i,SC_c))^{w_c} \]

holding the \( w_c \) fixed, we get an approximate joint:

\[ M(ERT|S_m) \approx F_3(C)C_1 F_3(\{I_c\}, I, C) \prod_c M'(E'T|SC) \]

Our standard search procedure combines an explicit search in \( C \) with a random search in all the other parameters. Each trial begins converging from classes built around \( C \) random case pairs. The \( C \) is chosen randomly from a log-normal distribution fit to the \( C_s \) of the 6-10 best trials seen so far, after trying a fixed range of \( C_s \) to start. We also have developed alternative search procedures which selectively merge and split classes according to various heuristics. While these usually do better, they sometimes do much worse.

The marginal joints of the different trials generally follow a log-normal distribution, allowing us to estimate during the search how much longer it will take on average to find a better peak, and how much better it is likely to be.

In the simpler model space \( S_M \) where \( SC = ST \) the computation is order \( NTCK_6 \), where \( C \) averages over the search trials. \( N \) is the number of possible peaks, out of the immense number usually present, that a computation actually examines. In the covariant space \( S_MV \) where \( SC = SV \) this becomes \( NKC(K + K_6) \).

5.2 Class Hierarchy and Inheritance - \( S_H \)

The above class mixture model space \( S_M \) can be generalised to a hierarchical space \( S_H \) by replacing the above set of classes with a tree of classes. Leaves of the tree, corresponding to the previous classes, can now inherit specifications of class parameters from “higher” (closer to the root) classes. For the purposes of the parameters specified at a class, all of the classes below that class pool their weight into one big class. Parameters associated with “irrelevant” attributes are specified independently at the root. Figure 2 shows how a class tree, this time with \( SC = SV \), can better fit the same data as in Figure 1. See [Hanson et al., 1991] for more about this comparison.

The tree of classes has one root class \( r \). Every other class has one parent class \( P_c \), and every class has \( J_c \) child classes given by \( C_{cj} \), where the index \( j \) ranges over the children of \( c \). Each child class has a weight \( \alpha_{cj} \) relative to its siblings, with \( \sum_j \alpha_{cj} = 1 \), and an absolute weight \( \alpha_{ccj} = \alpha_{cj} \alpha_c \), with \( \alpha_c = 1 \).

While other approaches to inheritance are possible, here each class is given an associated set of attributes \( K_c \), which it predicts independently through a likelihood \( L(E_i(K_c)|V_i,T_i,SC_c) \) and which no class above or below it predicts. To avoid having redundant trees which describe the same likelihood function, only \( K_r \) can be empty, and non-leaves must have \( J_c \geq 2 \).

We need to ensure that all attributes are predicted somewhere at or above each leaf class. So we call \( A_r \) the set of attributes which are predicted at or below each class, with \( A_r = K_r \), and then recursively partition each \( A_r \) into attributes \( K_c \) “kept” at that class, and hence predicted directly by it, and the remaining attributes to be predicted at or below each child \( A_{cj} \).

For leaves \( A_r = K_r \).

Expressed in terms of the leaves the likelihood is again a mixture:

\[ L(E_i|\text{VTS}_M) = \sum_c \alpha_c \prod e \alpha_{cje} L(E_i(K_c)|V_i,T_i,SC_c) \]

allowing the same EM procedure as before to find local maxima. The case weights here \( w_{cr} = \sum_{cje} w_{cje}( \text{with} \ w_{cje} = 1 \text{sum like in the flat mixture case and define class statistics} E'(K_c) = \bigcup_k \{ X_{ki} \} \text{for leaves} K_c \).

We also choose a similar prior, though it must now specify the \( K_c \) as well:

\[ d\sigma(\text{VT}|S_H) = \prod_c d\sigma(J_c, K_c | A_r, S_H) J_c! \text{d}B(\bigcup \alpha_{cje} | J_c) d\sigma(\text{VTS}| K_c SC) \]

\[ d\sigma(J_c, K_c | A_r, S_H) = F_3(J_c - 1) \frac{K_c!(A_r - K_c)!}{A_r + \delta_K A_r} \]

for all subsets \( K_c \) of \( A_r \) of size in the range \([1 - \delta_K, A_r]\) except that \( F_3(J_c - 1) \) is replaced by \( \delta_{Kc} \) when \( A_r = K_r \). Note that this prior is recursive, as the prior for each class depends on the what attributes have been chosen for its parent class.

This prior says that each possible number of attributes kept is equally likely, and given the number to be kept each particular combination is equally likely. This prior prefers the simpler cases of \( K_r = A_r \) and \( K_c = 1 \) and so again offers a significance test. In comparing the hypothesis that all attributes are kept at class with a hypothesis that all but one particular attribute will be kept at that class, this prior penalises the all—but—one hypothesis in proportion to the number of attributes that could have been kept instead.

The marginal joint becomes

\[ M(ERT|S_H) \approx \prod_c d\sigma(J_c K_c | A_r S_H) J_c! F_3(\bigcup I_{A_{cj}}, J_c) M'(E'(K_c) T_c | SC) \]

Figure 2: AutoClass IV Finds Class Tree ×10180 Better Lists of attribute numbers denote covariant blocks within each class, and the ovals now indicate the leaf classes.
and estimates are again $E(V_c|ERS_H) = E'(V_c|E(v(K_c)S_C)$
and $E(\alpha_c|ERS_H) = P_I(I_c, I, J_c)$.

In the general case of $S_{RV}$, where $S_C = S_V$, computation again takes $NK(\{K_B + K^2\}$, except that the $J$ is now an average of, for each $k$, the number of classes in the hierarchy which use that $k$ (i.e., have $k \in K_c$). Since this is usually less than the number of leaves, the model $S_B$ is typically cheaper to compute than $S_M$ for the same number of leaves.

Searching in this most complex space $S_{HV}$ is challenging. There are a great many search dimensions where one can trade off simplicity and fit to the data, and we have only begun to explore possible heuristics. Blocks can be merged or split, classes can be merged or split, blocks can be promoted or demoted in the class tree, EM iterations can be continued farther, and one can try a random restart to seek a new peak. But even the simplest approaches to searching a more general model space seem to do better than smarter searches of simpler spaces.

6 Conclusion

The Bayesian approach to unsupervised classification describes each class by a likelihood function with some free parameters, and then adds in a few more parameters to describe how those classes are combined. Prior expectations on those parameters $V^T$ combine with the evidence $E$ to produce a marginal joint $M(EVT|S)$ which is used as an evaluation function for classifications in a region $R$ near some local maxima of the continuous parameters $V$ and with some choice of discrete model parameters $T$. This evaluation function optimally trades off the complexity of the model with its fit to the data, and is used to guide an open-ended search for the best classification.

In this paper we have applied this theory to model spaces of varying complexity in unsupervised classification. For each space we provides a likelihood, prior, marginal joint, and estimates. This should provide enough information to allow anyone to reproduce AutoClass, or to use the same evaluation functions in other contexts where these models might be relevant.

References


