The Astrophysical Journal, 618:723–732, 2005 January 10

CLASS DISCOVERY IN GALAXY CLASSIFICATION

David Bazell
Eureka Scientific, Inc., 6509 Evensong Mews, Columbia, MD 21044; bazell@comcast.net

and

David J. Miller
Department of Electrical Engineering, Pennsylvania State University, University Park, PA 16802; millerdj@ee.psu.edu
Received 2004 June 7; accepted 2004 September 21

ABSTRACT

In recent years, automated, supervised classification techniques have been fruitfully applied to labeling and organizing large astronomical databases. These methods require off-line classifier training, based on labeled examples from each of the (known) object classes. In practice, only a small batch of labeled examples, hand-labeled by a human expert, may be available for training. Moreover, there may be *no* labeled examples for some classes present in the data; i.e., the database may contain several *unknown classes*. Unknown classes may be present because of (1) uncertainty in or lack of knowledge of the measurement process, (2) an inability to adequately "survey" a massive database to assess its content (classes), and/or (3) an incomplete scientific hypothesis. In recent work, the question of new class discovery in mixed labeled/unlabeled data was formally posed, with a proposed solution based on mixture models. In this work we investigate this approach, propose a competing technique suitable for class discovery in neural networks, and evaluate methods for both classification and class discovery in several astronomical data sets. Our results demonstrate up to a 57% reduction in classification error compared to a standard neural network classifier that uses only labeled data.

Subject headings: astronomical data bases: miscellaneous — galaxies: general — methods: data analysis — methods: statistical

1. INTRODUCTION

Finding patterns in large astronomical databases and grouping the data into different classes has become an important task in recent years, one that must be done in an automated fashion given the massive amounts of sky survey data currently being collected and stored. Traditional methods such as looking at large sky plates and identifying galaxies and clusters by eye are no longer feasible. Within statistical pattern recognition there are two traditional approaches to data classification: supervised statistical classification and unsupervised learning (clustering). In the supervised approach one is given a batch of training data containing labeled examples from each of the known classes of interest. These examples are used to learn a decision function that partitions the feature space into disjoint regions, each associated with one of the classes. Typical decision function structures used in practice are neural networks, decision trees, and prototype-based classifiers. Once the decision function is learned, it can be used to automatically classify new examples. The supervised learning of the decision function can be slow and generally requires off-line training. Moreover, enough labeled examples from each of the classes are required to learn an accurate decision function that adequately separates data into the different classes. However, extracting labeled examples from a large database is a time-consuming and expensive process, generally requiring hand labeling by human experts.

Alternatively, unsupervised learning, or clustering, techniques assign data to groups without any need for supervising examples. In these approaches, the grouping is chosen so that the data examples belonging to each cluster are "as similar as possible" and so that examples from different clusters are "as dissimilar as possible." This notion of similarity is quantified through a mathematical clustering objective function, one that relies on the choice of a distance measure defined on the feature space, e.g., the sum-of-squared-errors criterion. Mixture models (Duda et al. 2001; McLachlan & Peel 2000; Banfield & Raftery 1993) are one form of model-based clustering. They produce probabilistic, or soft, assignments of data points to each of the mixture components, or clusters. The nature and quality of the learned groupings obtained via unsupervised clustering critically depend on the choice of clustering distance measure and also on the number of clusters to be learned, which must be specified as part of the algorithm. There are currently no generally agreed on approaches for choosing these parameters in unsupervised learning. Furthermore, without supervising examples, there are no guarantees that the chosen parameters are consistent with the learning of clusters that correspond to the ground-truth classes in the data.

In recent years, to overcome the disadvantages of both supervised and unsupervised learning, "semisupervised learning" techniques have been proposed, e.g., Shashahani & Landgrebe (1994), Miller & Uyar (1997), and Nigam et al. (2000). These methods learn based on a batch of data that consists of both labeled *and* unlabeled examples. On the one hand, appropriate use of unlabeled examples, in addition to labeled ones, can help to better learn the "shapes" of each of the classes, i.e., the class-conditional density functions (Shashahani & Landgrebe 1994; Miller & Uyar 1997). On the other hand, use of some labeled examples can potentially help to guide unsupervised clustering methods toward solutions that capture the ground-truth classes in the data (Miller & Uyar 1997; Basu et al. 2002).

In nearly all prior semisupervised work, it has been assumed that the number of classes present in the database is known and that there are both labeled and unlabeled examples from each of these classes. However, for scientific domains, especially those with massive data collections, this assumption may not be

very reasonable. We use the term "unknown class" to refer to a ground-truth class that is present in a semisupervised data set but for which there are *no* labeled examples. If such classes are lying "latent" in semisupervised data, it will be scientifically interesting to identify these groups and to distinguish unlabeled known class samples from unlabeled unknown class samples.

There are several reasons why the presence of some classes within a given data set may be unknown. First, there may be uncertainty associated with the measurement process. As one example, suppose the data are measured by a new device or one whose operation (e.g., measurement sensitivity) is imprecisely known. If the device's sensitivity or dynamic range is greater than was supposed, it may record measurements corresponding to unanticipated events or objects. Second, in some cases, the set of known classes may be inferred by surveying or sampling a subset of the collected database. However, if there are millions of data samples, it is only practical to sample a very small data subset; if 99% of a database remains unsurveyed, it is quite possible that important content (such as some classes) will be missed. Finally, because the set of known classes reflects the currently accepted scientific hypotheses for a given domain, unknown classes may be present in the data if the current theory is wrong or incomplete. In fact, we can go so far as to say that the assumption that a collected database is composed of a fixed (known) set of classes is in some way inconsistent with the scientific method; one is guaranteed to find what one is looking for, i.e., known classes, rather than what may actually be present in the data.

In recent work (Miller & Browning 2003a, 2003b), the problem of new class discovery in mixed labeled/unlabeled data sets was formally proposed. The authors recognized that within a mixed labeled/unlabeled data set, unknown classes will consist of clusters or groups that are purely unlabeled. The authors proposed a special mixture-modeling technique tailored for discovering the cluster/group structure in the data and, in particular, the unknown classes. In their approach, either individual mixture components represent data from known classes, in which case they own both labeled and unlabeled samples, or they represent unknown classes, in which case they own purely unlabeled data subsets. Their learning approaches were demonstrated to be very effective at identifying purely unlabeled clusters (Miller & Browning 2003a) or *nearly* purely unlabeled ones (Miller & Browning 2003b) in partially labeled data sets. Such clusters represent *putative* unknown classes. Their approach was further demonstrated to improve the overall accuracy of the mixture-modeling solution.

In this work, we consider the problem of galaxy classification based on sky survey data, with several unknown classes present in the data. For this domain, we evaluate both the approach of Miller & Browning (2003b) and a new approach that we propose here, one that is applicable to class discovery for neural network–based classifiers. In § 2 we describe the data sets and data preparation. In § 3 we review Miller & Browning (2003a, 2003b) and also introduce a class discovery approach for neural network classifiers. In § 3 we also describe several performance criteria, each capturing different aspects of the class discovery problem. In § 4 we present our experimental results. Finally, the paper concludes with a summary and some discussion.

2. DATA PREPARATION

In our experiments we used two data sets, each with over 5000 data points. The first consisted of data from Storrie-Lombardi et al. (1992) (hereafter the ESOLV data, after the ESO-LV catalog of Lauberts & Valentijn [1989]), which have been used previously in several studies of automated classification methods (Storrie-Lombardi et al. 1992; Owens et al. 1996; Bazell & Aha 2001). The second data set consisted of Sloan Digital Sky Survey (SDSS) early release data (Stoughton et al. 2002) composed of over 50,000 objects of various types.

TABLE 1
ESOLV Data Set Summary

Class	Number	Portion of Whole (%)	Object
0	466	8.93	E
1	851	16.3	S0
2	2403	46.1	Sa+Sb
3	1132	21.7	Sc+Sd
4	365	7.00	Irregular

Storrie-Lombardi et al. (1992) performed one of the earliest attempts at morphological classification of galaxies using neural networks. Their data set consisted of 13 input features derived from images of galaxies that were then used to classify the galaxies into five classes: E, S0, Sa+Sb, Sc+Sd, and Irr. We used their input data set of 5217 galaxies. The features in this data set are described in Storrie-Lombardi et al. (1992). Bazell & Aha (2001) describe the use of this data set for galaxy classification using ensembles of neural networks. For our studies we eliminated one of the features, $E_{\rm Err}^{\rm Fit}$, which is the error in an ellipse fitted to B isophotes. This feature had very small variance and equaled zero for approximately 80% of the objects. Thus, we used 12 of the 13 features in the original data set.

The second data set has an order of magnitude more objects than the ESOLV data. The SDSS data consist of 54,007 objects drawn from seven different classes. Each object is described by a total of six features: photometric values in u, g, r, i, and z and the redshift of the object. For later reference, we denote the number of objects/data points by N and the number of features for each object by d, i.e., for the SDSS data $N = 54{,}007$, and $d = 6$. Each object is represented as a "feature vector," $x = (x_1, x_2, \ldots, x_d)$, with x_j the measured value for the jth feature. For the SDSS data, these six values are the five photometric values and the redshift. The data set of objects is thus represented as $\{x_{ij},\ i = 1, \ldots, N;\ j = 1, \ldots, d\}$.

Tables 1 and 2 summarize the properties of the data sets we used. For each class the tables show the number of objects in the class, the percentage of total objects that class represents, and the type of object in the class.

For our experiments, we treated one or two of the classes as being unknown, withholding from use during model learning the label for every data example from each of the unknown classes. For data from all other classes, we retained the labels for a randomly selected subset (roughly 10% of the points from

TABLE 2
SDSS Data Set Summary

Class	Number	Portion of Whole (%)	Object
0	229	0.42	Unknown spectrum
1	6049	11.2	Stellar spectrum
2	41930	77.6	Galaxy spectrum
3	4409	8.16	Quasar spectrum
4	237	0.44	High-z quasar spectrum
5	130	0.24	Sky spectrum
6	1023	1.89	Late-type star

these classes: 521 examples for the ESOLV data and 5400 examples for the SDSS data). The random selection was performed in a "stratified" fashion, ensuring that the number of labeled examples from each known class was in proportion to the mass, or frequency of occurrence, of the class. In this way, we obtained a data set containing both labeled and unlabeled examples, and with all labels missing from one or two classes. This is precisely the data scenario proposed and addressed in Miller & Browning (2003a, 2003b).

3. DESCRIPTION OF ALGORITHMS

We used two algorithms to classify the data and perform class discovery, a mixture model and a back-propagation algorithm. These approaches are described in detail below.

3.1. *Mixture-Modeling Approach*

This subsection concisely reviews the work in Miller & Browning (2003a, 2003b). There are three main contributions in these works: (1) The problem of new class discovery in mixed labeled/unlabeled data was proposed. (2) A mixture model was proposed, tailored for this scenario: one that incorporates a realistic statistical model for how data become labeled. This model has built into it the competing hypotheses that a data sample may come from a known or an unknown group. Thus, this model naturally yields a posteriori probabilities for these hypotheses, as well as the standard a posteriori probabilities for the known classes (now assuming a known class hypothesis). These probabilities are used to make several types of classification decisions, as is discussed below. (3) Methods for learning the mixture model from given data were proposed. In these approaches, individual mixture components learn to represent data of either known or unknown classes. We next give a descriptive summary of the procedure, followed by a more detailed review of Miller & Browning (2003a, 2003b).

We start with a data set in which each object is described by a feature vector. The elements, or dimensions, of the feature vector represent some measured or derived parameters of each object. In the case of the SDSS data, the feature vectors consist of the photometric values in u, g, r, i, and z of each object and the object's redshift, resulting in a six-dimensional vector.

The data set is then divided into two parts: objects that have a class label describing what we think the object is (S0 galaxy, star, quasar, etc.) and objects without a class label. Every object in the data set belongs to some class, but we may not know which class and in fact we may not have identified the existence of the proper class. Each labeled object is described by its feature vector, its class label, and a parameter indicating that the object is labeled. Each unlabeled object is described by its feature vector and a parameter indicating that the label is absent. This effectively labels all the data, either with a class or with a label of "unlabeled." A novelty of this method is to add the parameter indicating the presence or absence of a label and to require the mixture model—the probabilistic model of the data—to explain all the parameters describing the data, including the presence or absence of the label. This results in a more powerful model of the data, as we describe below.

The mixture model consists of a weighted sum of probability densities, which we take to be Gaussian densities but which could be any density function that suitably describes the data. It also includes a function that generates class labels (including the "unlabeled" class) given a particular mixture component. This approach contains another improvement over the standard mixture model in that it defines two types of mixture components, predefined and non-predefined. Predefined components generate data points that can be either labeled or unlabeled, for which the labels are assumed to be missing at random. The non-predefined components generate only unlabeled data points, i.e., the labels are always missing from the data. The non-predefined components describe data points that are purely unlabeled and may represent new classes of objects.

Along with the mixture model, we define a likelihood function that assumes all the data are labeled, perhaps with a class label and perhaps with a label of "unlabeled." The mixture model is then learned based on maximizing this likelihood function. In other words, the best-fit parameters that describe the mixture model, including Gaussian parameters (means and variances), the coefficients of the mixture components (which are prior probabilities), and the function that generates labels given a certain mixture component, are determined by maximizing the likelihood function. This produces a set of probabilities that each mixture component describes a given class, including unknown classes. If the probability is high (close to 1) that a specific mixture component describes an unknown class, then that component is effectively non-predefined and is likely to be describing a new class.

The remainder of this subsection describes in more detail the implementation of the mixture-model approach, following Miller & Browning (2003a, 2003b). Consider a data set with both labeled and unlabeled samples, i.e., $\mathcal{X}_m = \{\mathcal{X}_l, \mathcal{X}_u\}$, where $\mathcal{X}_l = \{(x_1, c_1), (x_2, c_2), \ldots, (x_{N_l}, c_{N_l})\}$ is the labeled subset and $\mathcal{X}_u = \{x_{N_l+1}, \ldots, x_N\}$ is the unlabeled subset. Here, $x_i \equiv (x_{i1}, x_{i2}, \ldots, x_{id})$ is a feature vector, and c_i is the associated class label from the set of known classes $\mathcal{P}_c$. This mixed data scenario was considered previously in, e.g., Shashahani & Landgrebe (1994), Miller & Uyar (1997), and Nigam et al. (2000). However, in these works, it was assumed that all unlabeled samples originate from known classes. Here we consider the case in which unlabeled samples may also originate from unknown classes.

If a sample is labeled, then it is known that the sample originates from one of the known classes. On the other hand, if the sample is unlabeled, then there are two sources of uncertainty. First, since there may be unknown classes present, it is unknown whether or not the given sample originates from a known class. Second, even if it does belong to one of the known classes, it may not be known which one. An example is shown in Figure 1, with labeled data denoted by a single number, the class, and with unlabeled data denoted by "U" followed by the ground-truth class of origin.

The two-dimensional data points were generated according to a Gaussian mixture with five components. For this example, all points originating from the same component come from the same class, i.e., classes "own" either one or multiple mixture components. Class 2 consists of two components, with the other classes consisting of single components. Classes 1, 2, and 3 are known classes, while class 4 is an unknown class. For the data from known classes, we randomly select a subset of samples to label. As indicated in Figure 1, based on this random selection, each component from a known class ends up owning both some labeled data and some unlabeled data. In contrast, for the unknown class 4, no samples are labeled. Accordingly, as indicated in the figure, the component from class 4 owns only unlabeled data (shown by "U4"). Thus, components from unknown classes are characterized by the fact that they own purely unlabeled data subsets, while components from known classes own both labeled and unlabeled data.

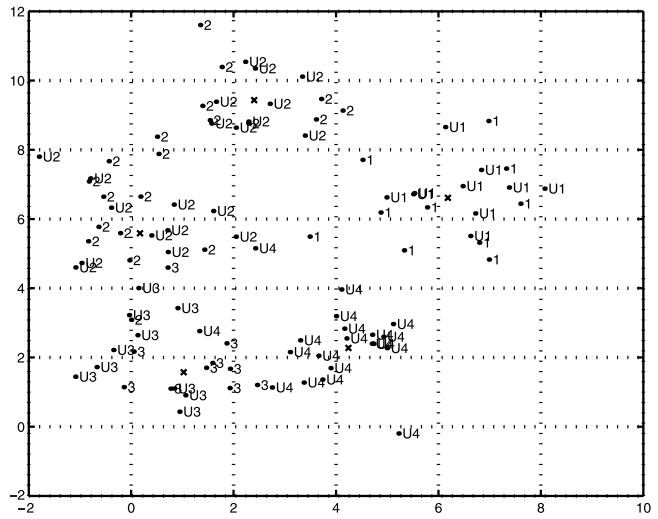

FIG. 1.—Example involving partially labeled data and an unknown class. Those labeled are denoted by a single class number, while unlabeled data are denoted by a "U," followed by the class. The crosses denote mixture component centers.

Since the presence or absence of a label for each sample appears to be helpful for distinguishing known class from unknown class groups, it was suggested in Miller & Browning (2003a, 2003b) to treat these presence/absence indications as data, to be explained by the mixture model, along with the feature vectors and the known class labels. In particular, the authors redefined the data set as consisting of $\mathcal{X}_m = \{\mathcal{X}_l,\ \mathcal{X}_u\}$, where now $\mathcal{X}_l = \{(x_1,\ l,\ c_1),\ (x_2,\ l,\ c_2),\ \ldots,\ (x_{N_l},\ l,\ c_{N_l})\}$ and $\mathcal{X}_u = \{(x_{N_l+1},\ m),\ \ldots,\ (x_N,\ m)\}$. Here the new random observation $\mathcal{L} \in \{l,\ m\}$ is introduced, taking on values indicating that a sample is either labeled or missing the label. The authors proposed a special mixture model to explain all the data, including the presence or absence of a label for each sample. Two types of mixture components were posited, differing in the mechanism they use for generating the label presence/absence data. "Predefined" components generate both labeled and unlabeled data and assume labels are missing at random. These components represent the known classes. "Non-predefined" components generate only unlabeled data; thus, in localized regions, they capture data subsets that are purely unlabeled. These components represent the unknown classes. Note that these two types of components are precisely matched to the data scenario depicted in Figure 1, in which, based on random selection of labels solely for the known class data, known class components (predefined ones) have labels missing at random, while for the unknown class (non-predefined) components, labels are deterministically (always) missing. We next briefly summarize the mathematical formulation in Miller & Browning (2003a, 2003b). More details can be found in these original papers.

3.1.1. *Notation*

Let $\mathcal{M}_k$, $k = 1, \ldots, M$, denote the kth mixture component. Let $\mathcal{C}_{\text{pre}}$ denote the subset of predefined components, with the remaining subset denoted $\bar{\mathcal{C}}_{\text{pre}}$. For example, in Figure 1 there are four predefined components and one non-predefined component. Let $C \in \mathcal{P}_c \equiv \{1,\ 2,\ \ldots,\ N_c\}$ be a random variable defined over the N_c known classes, with $c(x) \in \mathcal{P}_c$ the class label for sample x. Let α_k denote the prior probability for component k, θ_k denote the parameter set specifying the kth component's (component-conditional) joint feature density, and $f(x|\theta_k)$ denote this density. We also introduce a new class set $\tilde{\mathcal{P}}_c = \{1,\ 2,\ \ldots,\ N_c,\ u\}$, consisting of the set $\mathcal{P}_c$ plus a value u, used to indicate that a sample is unlabeled. With respect to the class set $\tilde{\mathcal{P}}_c$, every sample is now labeled, with the unlabeled samples taking on the values "u." We suppose a different random label generator, conditioned on each mixture component, i.e., $\text{Prob}(c|\mathcal{M}_k) \equiv \beta_{c|k}$, $c \in \tilde{\mathcal{P}}_c$, where $\sum_{c \in \tilde{\mathcal{P}}_c} \beta_{c|k} = 1$. Note that $\beta_{c|k}$ measures the fraction of samples from component k that belong to class c, with $\beta_{u|k}$ the fraction of unlabeled samples. For a non-predefined component, $\beta_{u|k} = 1$, i.e., all samples from the non-predefined component k are unlabeled. In summary, the mixture model is based on the parameter set $\Lambda = \{\{\alpha_k\},\ \{\theta_k\},\ \{\beta_{c|k}\}\}$.

Hypothesis for Random Generation of the Data.—This model of Miller & Browning (2003b) hypothesizes that each sample from $\mathcal{X}_m$ is generated independently, based on Λ, according to the following stochastic generation process:

1. Randomly select a component $\mathcal{M}_j$ according to $\{\alpha_k\}$.
2. Randomly select a sample x according to $P(x|\theta_j)$ and a label c according to $\{\beta_{c|k}\}$.

Joint Data Likelihood.—The log of the joint data likelihood associated with this model is

$$\mathcal{L} = \sum_{x \in \mathcal{X}_m} \log \sum_{k=1}^{M} \alpha_k f(x|\theta_k) \beta_{c(x)|k}, \tag{1}$$

where $c(x) \in \tilde{\mathcal{P}}_c$. The model parameters Λ can be chosen to maximize the log-likelihood (eq. [1]) via the expectation-maximization (EM) algorithm (e.g., Duda et al. 2001). Since the derivation of these EM equations is standard, their exposition is herein omitted.

This model does not explicitly discover new class components, i.e., mixture components that are purely unlabeled. However, suppose that for a given component $\mathcal{M}_j$, we have that $\beta_{u|j} \simeq 1$, and $\beta_{u|j}$ is also significantly greater than the *average* value $(1/M)\sum_{j'=1}^{M} \beta_{u|j'}$. In this case, the fraction of unlabeled data owned by the component is unusually high. We categorize these components as non-predefined, i.e., $\mathcal{M}_j \in \bar{\mathcal{C}}_{\text{pre}}$. Such components are putative unknown class components. All other components are categorized as predefined, representing known class data. To summarize, we have the following strategy for new class discovery in mixed labeled/unlabeled data: (1) learn a mixture model to maximize the log-likelihood (eq. [1]); (2) for each component, declare it non-predefined if $\beta_{u|j} - (1/M)\sum_{j'=1}^{M} \beta_{u|j'} > \delta$; otherwise, declare it predefined. Here, δ is a suitably chosen threshold. In practice, we declare a component non-predefined when its value $\beta_{u|j}$ is closer to 1.0 than to the average value; i.e., we choose $\delta = (1/2)[1 - (1/M)\sum_{j'=1}^{M} \beta_{u|j'}]$. We have found this choice for δ to give reasonable results for a variety of experimental conditions (for different data sets and for different fractions of labeled data).

3.1.2. *Statistical Inferences from the Model*

After applying this thresholding operation to each component, the resulting model is naturally applied to address several inference tasks: (1) standard classification of a given sample to one of the known classes and (2) known versus unknown class discrimination. For classification to known classes, for a given sample x, we compute the a posteriori probabilities

$$P(C = c|x;\ \Lambda) = \frac{\sum_{k \in \mathcal{C}_{\text{pre}}} \alpha_k f(x|\theta_k)[\beta_{c|k}/(1 - \beta_{u|k})]}{\sum_{k \in \mathcal{C}_{\text{pre}}} \alpha_k f(x|\theta_k)}, \quad c \in \mathcal{P}_c. \tag{2}$$

These can be used in a maximum a posteriori (MAP) class decision rule, i.e., $c^* = \arg\max_c P(C = c|x;\ \Lambda)$.

In order to discriminate between the hypotheses that an unlabeled sample originates from a known versus an unknown class, we need the a posteriori probability that the given feature vector is generated by a non-predefined component. This is given by

$$P(\mathcal{M}_{\text{np}}|x \in \mathcal{X}_u) = \frac{\sum_{k \in \tilde{C}_{\text{pre}}} \alpha_k f(x|\theta_k)\beta_{u|k}}{\sum_k \alpha_k f(x|\theta_k)\beta_{u|k}}. \quad (3)$$

If $P(\mathcal{M}_{\text{np}}|x \in \mathcal{X}_u) > \frac{1}{2}$, then the sample is declared to belong to an unknown class; otherwise, it is declared as a known class sample.

3.1.3. *New Class Discovery*

While the EM learning assumes that the number of mixture components M is fixed and known, in practice this size must be estimated. Model order selection is a difficult and pervasive problem, with several criteria proposed (Schwarz 1978; Wallace & Freeman 1987; McLachlan & Peel 2000) but no consensus on the right one. In our class discovery setting, the importance of accurate model order selection cannot be overstated: *the non-predefined components in the validated solution will be taken as candidates for new classes*, to be forwarded to a domain expert for further study. Accurate model order selection is thus important for successful new class discovery. Here, as in Miller & Browning (2003a, 2003b), the Bayesian information criterion (BIC; Schwarz 1978) is applied. The BIC model selection criterion is written in the form

$$\text{BIC}(M) = \frac{N_p(M)}{2}\log N - \mathcal{L}, \quad (4)$$

with $N_p(M)$ the number of free parameters in the M-component mixture model and N the data length. The first term is the penalty on model complexity, with the second term the negative log-likelihood. We apply BIC in a "wrapper-based" model selection approach; i.e., we build models for increasing M, evaluate BIC for each model, and then select the model with minimum BIC cost.

While any clustering/mixture-modeling technique can in principle be used to discover unknown classes, standard methods do not have any special impetus for finding label-free (or largely label-free) clusters. In contrast, the log-likelihood (eq. [1]) and the likelihood function used in Miller & Browning (2003a) both encourage solutions with non-predefined components, when such components are warranted by the presence of unknown classes in the data. In equation (1), it is the $\beta_{c|j}$ term that provides the impetus for forming these unknown classes, since this term approaches its maximum value ($\beta_{u|j} = 1$) in the non-predefined component case.

3.2. *Neural Network Approach*

The mixture-modeling approach provides several inference capabilities when dealing with mixed labeled/unlabeled data sets and possibly unknown classes: (1) it allows one to infer whether or not a given sample belongs to one of the known classes; (2) it identifies purely unlabeled mixture components/clusters, which are reasonably treated as putative unknown classes or, at any rate, components of unknown classes; (3) assuming a known class hypothesis for a given sample, the model can infer from which known class the sample originates (i.e., the usual classification inference capability).

While the mixture-modeling approach is naturally suited to new class discovery given mixed labeled/unlabeled data, neural network classifiers do not appear to be predisposed to making these inferences. Neural networks are generally trained using a purely supervised approach, with class labels provided for every example in the training set. Thus, in general, unlabeled samples play no role in the training; given a mixed labeled/unlabeled data set, the neural network training will either discard all the unlabeled examples, including those from unknown classes, or, perhaps worse, erroneously impute and use known class labels for these unknown class data. Accordingly, the neural network is only explicitly trained to discriminate between the known classes; it is not trained to distinguish known from unknown classes. While it thus appears that neural networks do not possess any class discovery inference capability, we next suggest an approach that gives neural networks at least a weak form of this capability.

The neural network algorithm we used was a basic back-propagation algorithm available with the WEKA machine learning package (Witten & Frank 2000). We used the default configuration, consisting of a three-layer network (input, hidden, and output). The number of input nodes was N_i, one node for each input feature. There were N_c output nodes, one for each known class. The number of hidden nodes was calculated according to $N_h = (N_i + N_c)/2$. For the ESOLV data we used 12 nodes in the input layer corresponding to the 12 input features, eight nodes in the hidden layer, and four in the output layer. For the SDSS data we used five input layer nodes, five hidden layer nodes, and six output layer nodes.

3.2.1. *Decision Confidence*

One possible neural network indicator that a given sample originates from an unknown class is if the neural network does not make a "confident" decision for this sample. The neural network produces an output $g_j(x)$ for each known class $j = 1, \ldots, N_c$ and decides on the class with the *largest* output. There are several ways of quantifying the degree of uncertainty in the neural network's decision. One measure motivated by information theory is Shannon's entropy function (Cover & Thomas 1991). Since entropy measures uncertainty in a probability mass function (pmf), it is necessary to convert the neural network class outputs $g_j(x), j = 1, \ldots, N_c$ into a pmf. This is achieved as follows: Let

$$\tilde{g}_j(x) = \frac{g_j(x) - \min_l g_l(x)}{\sum_k g_k(x) - \min_l g_l(x)}.$$

With this choice, we have $0 \le \tilde{g}_j(x) \le 1$ and $\sum_j \tilde{g}_j(x) = 1$, i.e., $\tilde{g}_j(x)$ is a pmf defined on the known classes. We can then measure the Shannon entropy as $H = -\sum_j \tilde{g}_j(x)\log\tilde{g}_j(x)$. Entropy is nonnegative, with greater entropy indicating greater uncertainty. If H is greater than a preset threshold, we can declare that the sample x does not convincingly belong to any of the known classes; i.e., it is declared an unknown class sample. This approach, based on a measure of the classifier's degree of indecision, is the one we have taken in imparting the neural network with some class discovery inference capability. Other measures of the classifier's degree of indecision are also possible.

3.3. *Error Measures*

There are three error measures that we have used to evaluate the class discovery approaches. They are defined as follows:

Criterion 1: Misclassification rate in deciding between known and unknown class hypotheses.—This criterion was

TABLE 3
MIXTURE MODEL AND NEURAL NETWORK RESULTS USING CRITERION 1 ERROR MODEL SELECTION FOR ESOLV DATA

	Mixture Model					Neural Network	Δ
Unknown Class	Ncomp	Nonpre	Criterion 1	Criterion 2	Criterion 3	Criterion 1	(%)
0	27	1	0.089	0.728	0.374	0.0992	−10.7
1	42	1	0.172	0.904	0.322	0.181	−5.1
2	70	16	0.380	0.551	0.308	0.488	−22.2
3	48	5	0.218	0.748	0.325	0.241	−9.3
4	14	1	0.051	0.518	0.422	0.078	−34.3
0, 1	16	4	0.140	0.559	0.328	0.281	−50.3
0, 2	13	3	0.455	0.647	0.214	0.611	−25.5
0, 3	39	4	0.318	0.848	0.252	0.340	−6.6
0, 4	34	2	0.146	0.770	0.357	0.177	−17.3
Average	33.7	4.1	0.219	0.698	0.323	0.277	−20.2

evaluated for both the mixture-model and neural network approaches. For the mixture model, the classification decisions were made using a MAP probability rule, based on equation (3). For the neural network, the decisions were made based on thresholding of the neural network's entropy measure, as discussed earlier. The error rate was measured over the unlabeled portion of the data set (which consisted of both known and unknown class data); i.e., it was estimated as the fraction of unlabeled samples that were misclassified.

Criterion 2: Misclassification rate within putative unknown classes.—The first criterion simply measures how effective an algorithm is at identifying the subset of (unlabeled) samples that come from new, i.e., unknown, classes. If there is a single unknown class present in the data, then this is all that is required. However, suppose that there are multiple unknown classes present. Then, in addition to identifying the subset of samples from unknown classes, one would also like to identify the individual classes that compose this unknown class subset. In other words, one would like to identify the underlying cluster (group) structure within the unknown class data. The mixture-modeling approach directly models the unknown and, separately, the known class data by a mixture of components (clusters). Each such cluster can be viewed as a putative unknown class. A measure of the accuracy of this clustering is the unknown class label purity of these clusters. In particular, suppose one of the learned non-predefined clusters owns (in a MAP sense) 20 samples that are ground-truth from unknown class A, 30 samples from unknown class B, and 35 unlabeled samples that in fact belong to known classes. The most populous unknown class in the cluster is B. All samples in the cluster that do not possess label B are reasonably counted as errors. One can sum these errors over all non-predefined clusters and divide by the total number of samples owned by all non-predefined clusters. This is the fraction of samples that in effect have been erroneously assigned to individual non-predefined clusters. Note that this criterion is only well defined when the model finds at least one non-predefined component.

Criterion 3: Known class error rate.—If a sample belongs to a known class, then we are interested in identifying to which known class it belongs. Accordingly, we can define an error fraction measured over the known class data. Several such criteria are possible. Here, we count an error if an unlabeled sample that is from a known class is assigned to the wrong known class. For the mixture model, a MAP classification rule based on the probabilities in equation (2) was used.

All three of the above criteria require various forms of ground-truth label information for the unlabeled data subset: criterion 1 requires a ground-truth known/unknown class indication for all the data, criterion 2 requires knowledge of unknown class labels for all the data, and criterion 3 requires knowledge of all known class labels. Since in practice one would not have this information (by definition, no labels are

TABLE 4
MIXTURE MODEL AND NEURAL NETWORK RESULTS USING CRITERION 1 ERROR MODEL SELECTION FOR SDSS DATA

	Mixture Model					Neural Network	Δ
Unknown Class	Ncomp	Nonpre	Criterion 1	Criterion 2	Criterion 3	Criterion 1	(%)
0	49	0	0.005	NA	0.061	0.005	0
1	47	5	0.033	0.209	0.046	0.124	−73.3
2	28	20	0.034	0.007	0.112	0.134	−74.9
3	69	4	0.022	0.170	0.007	0.091	−76.0
4	47	0	0.005	NA	0.064	0.005	0
5	68	0	0.003	NA	0.063	0.003	0
6	74	1	0.012	0.338	0.080	0.021	−43.6
0, 1	79	5	0.041	0.243	0.035	0.129	−68.5
0, 2	33	20	0.034	0.012	0.090	0.867	−96.1
0, 3	80	5	0.025	0.206	0.062	0.095	−73.9
0, 4	66	1	0.009	NA	0.074	0.010	−4.3
0, 5	77	1	0.006	0.652	0.096	0.007	−22.6
0, 6	76	1	0.0156	0.455	0.064	0.026	−39.5
Average	61	4.8	0.019	0.26	0.066	0.117	−57.3

TABLE 5
MIXTURE MODEL SELECTED BY BIC AND NEURAL NETWORK RESULTS ON THE ESOLV DATA

	Mixture Model					Neural Network	Δ
Unknown Class	Ncomp	Nonpre	Criterion 1	Criterion 2	Criterion 3	Criterion 1	(%)
0	60	7	0.153	0.921	0.399	0.099	54.2
1	53	10	0.262	0.867	0.335	0.181	44.7
2	63	11	0.433	0.743	0.315	0.488	−11.4
3	54	6	0.256	0.890	0.307	0.241	6.5
4	67	7	0.103	0.614	0.437	0.078	32.9
0, 1	69	13	0.190	0.620	0.318	0.281	−32.2
0, 2	57	8	0.525	0.799	0.209	0.611	−14.0
0, 3	62	11	0.339	0.854	0.237	0.340	−0.3
0, 4	62	7	0.200	0.773	0.368	0.177	13.2
Average	60.8	8.9	0.274	0.787	0.325	0.277	10.4

known for data from unknown classes), these criteria can only be used for model evaluation/validation. In practice, other information sources or expert knowledge would be needed in order to assess the quality of, or to confirm, the model inferences. We also note that criterion 2 can only be evaluated for the mixture model, since the neural network approach does not attempt to partition the estimated unknown class data into smaller groups.

4. RESULTS

We evaluated the mixture model and neural network on both the ESOLV and SDSS data sets. For SDSS, we constrained mixture component variances to be at least 0.1 in order to avoid an observed tendency for the learning to find singular solutions (zero variance), as well as solutions with very small variances along some dimensions. For both the mixture model and the neural network there are ''operating parameters'' whose choices affect the class discovery inference performance. For the mixture model, we need to select the model order, i.e., the number of components. This choice will clearly affect the ability to identify unknown class components. For example, if an unknown class has a small mass, one or more components will only be ''deployed'' for its representation if the model has many components. Likewise, if the unknown class has a very large mass (and significant within-class variation), quite a few components may be needed to represent it well. For the neural network, the choice of the entropy threshold affects performance. We have performed several experimental evaluations of the mixture model and neural network, based on different approaches for choosing these operating parameters. In one set of experiments, shown for the ESOLV and SDSS data sets in Tables 3 and 4, we picked both the mixture order (over the range 10–80) and the neural network's entropy threshold (by an exhaustive search) to maximize criterion 1 performance. Note that these approaches cannot be used in practice since, in performing the model selection, these methods require evaluating a cost (criterion 1) that depends on knowledge of the unknown class labels. However, this experiment does allow a comparison of best-case performances achieved by the mixture and neural network approaches. For the mixture model, we have also applied BIC-based selection as described earlier. This approach is wholly unsupervised and thus feasible in practice.

Tables 3 and 4 show the results for the ESOLV and the SDSS data, respectively, for the case in which both the neural network and mixture model were chosen to give the best criterion 1 performance. The first column shows the classes that were treated as unknown for that series of runs. For example, ''3'' indicates that class 3 was treated as unknown, while ''0, 3'' indicates that both classes 0 and 3 were treated as unknown. The value of ''Ncomp'' is the number of components used by the mixture model corresponding to the best value of criterion 1, while ''Nonpre'' is the number of non-predefined components

TABLE 6
MIXTURE MODEL SELECTED BY BIC AND NEURAL NETWORK RESULTS ON THE SDSS DATA

	Mixture Model					Neural Network	Δ
Unknown Class	Ncomp	Nonpre	Criterion 1	Criterion 2	Criterion 3	Criterion 1	(%)
0	70	0	0.005	NA	0.110	0.0047	0
1	63	5	0.104	0.535	0.038	0.124	−16.1
2	67	24	0.100	0.038	0.097	0.134	−25.7
3	74	5	0.023	0.060	0.105	0.091	−74.9
4	76	0	0.005	NA	0.106	0.005	0
5	80	1	0.003	0.539	0.118	0.003	9.2
6	71	3	0.027	0.666	0.107	0.021	29.8
0, 1	66	5	0.094	0.489	0.034	0.130	−27.0
0, 2	79	30	0.113	0.027	0.064	0.867	−87.0
0, 3	74	5	0.033	0.145	0.104	0.095	−65.4
0, 4	61	1	0.010	0.530	0.112	0.010	−0.9
0, 5	71	1	0.006	0.604	0.112	0.007	−15.0
0, 6	71	3	0.0273	0.666	0.107	0.026	6.1
Average	71	6.4	0.042	0.391	0.093	0.117	−20.5

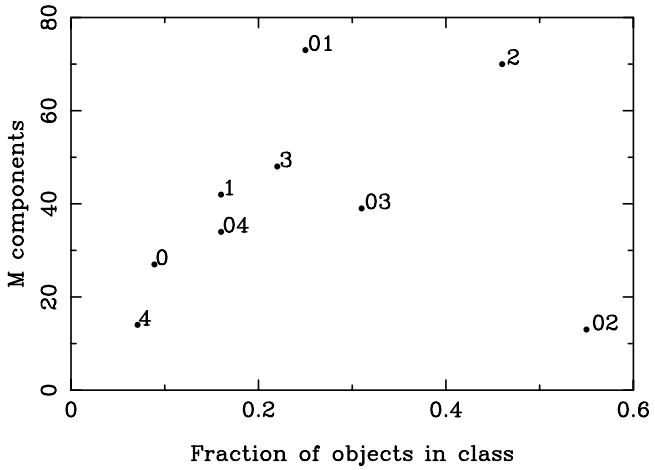

FIG. 2.—Number of components in the best mixture model, using criterion 1 model selection, as a function of the unknown class mass for ESOLV data. Points are labeled by the unknown class or classes. Thus, the point "3" means that class 3 was taken as the unknown class. A point "03" means that both class 0 and class 3 were taken as unknown classes.

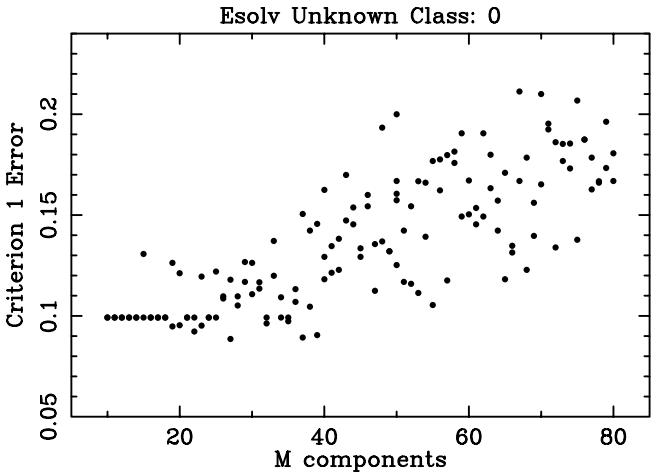

FIG. 4.—Criterion 1 error as a function of the number of components in the mixture model using ESOLV data with unknown class 0. Again, low error is achieved at a modest number of mixture components. The error remains approximately constant until non-predefined components are introduced.

in that model. The remaining columns under "Mixture Model" list error fractions for the three criteria discussed above. Under "Neural Network" we list the value of criterion 1, the only error measure evaluated for the neural network. The last column shows the percentage change in the criterion 1 value between the neural network and the mixture model, with a negative value indicating a lower criterion 1 error for the mixture model compared to the neural network. For the moment we restrict discussion to the criterion 1 performance. Tables 3 and 4 show that, with both methods optimized for criterion 1 performance, significantly better inference accuracy is achieved by the mixture model–based approach. For the ESOLV data we find an average decrease in the criterion 1 error of 20% and a maximum decrease of 50%. For the SDSS data we find an average decrease in the criterion 1 error of 57% and a maximum decrease of 96%. This is not especially surprising, since the mixture is learned using the unknown class data (but without use of the labels), while the neural network is only trained on labeled known class data.

In Tables 5 and 6 we again compare the neural network, with the threshold optimized for criterion 1, against the mixture model but with the order now selected based on the BIC criterion. Since the neural network decision-making threshold is optimized based on knowledge of the unknown class labels, while the mixture model and its order are chosen without use of this information, this comparison is not a fair one. However, in practice unsupervised order selection will be required. Thus, this comparison does give insight into the loss in accuracy attributable to the use of generally suboptimal but practically feasible model order–selection techniques. As Tables 5 and 6 show, the criterion 1 error for the mixture model is now higher in five of nine cases for the ESOLV data and is the same or higher in five of 13 cases for the SDSS data. For the ESOLV data we find an average 10% increase in error, while for the SDSS data we find an average 20% decrease in error, compared to the neural network.

Figure 2 shows a plot of the number of components in the best-performing mixture model (selected to optimize criterion 1) as a function of the fraction of objects in the unknown classes, for the ESOLV data. There is a clear trend toward a larger number of components selected when the unknown classes make up a larger mass fraction of the total data set. One point, with classes 0 and 2 as the unknown classes, is well described

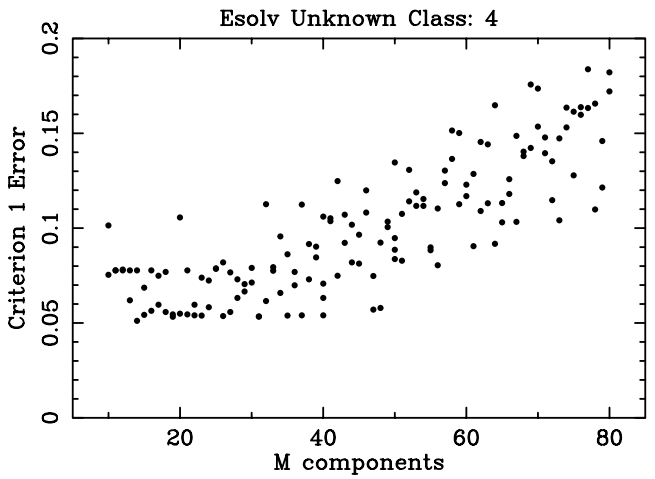

FIG. 3.—Criterion 1 error as a function of the number of components in the mixture model using ESOLV data with unknown class 4. The plot demonstrates that a low error can be achieved with a modest number of mixture components.

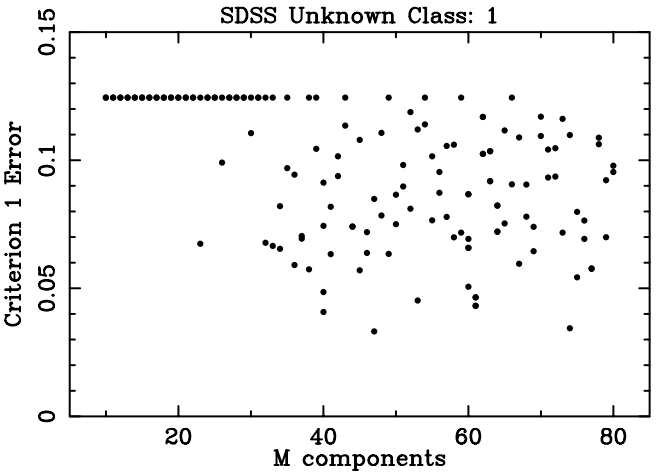

FIG. 5.—Criterion 1 error as a function of the number of components in the mixture model using SDSS data with unknown class 1.

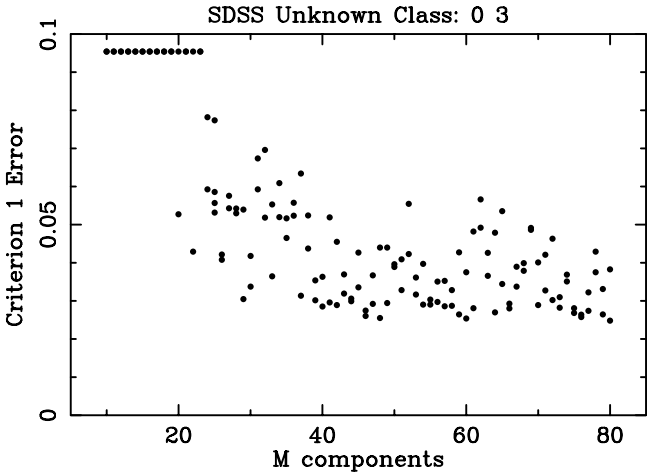

FIG. 6.—Criterion 1 error as a function of the number of components in the mixture model using SDSS data with unknown classes 0 and 3.

by an unexpectedly small number of components (13). This is discussed below. A similar trend is not evident in the SDSS data. The SDSS data are dominated by two classes (1 and 2), which together represent over 88% of the data.

Figures 3 and 4 show plots of the value of the criterion 1 error as a function of the number of components in the mixture model for two different unknown class combinations. These plots reflect the information in Figure 2 in a different way. When the unknown classes represent a relatively small mass fraction of the total number of objects in the data set, the minimum value of criterion 1 is found at a relatively moderate number of components. For example, this is evident in the figure for unknown class 4. Conversely, as seen in Figure 2, if the unknown classes comprise a large fraction of the total number of objects in the data set, then the minimum value of criterion 1 is found at a correspondingly higher number of components. Note that in Figure 4 criterion 1 remains static over a range of model orders (for unknown class 0, over the range of 10–20 components). This is due to the fact that the performance only changes at the discrete points where additional non-predefined components are introduced. In this case, no non-predefined components were chosen until M increased beyond 20.

Figures 5 and 6 show the criterion 1 error as a function of the number of mixture model components for the SDSS data. Again, we see the criterion 1 error remaining pretty much static until the model order M reaches 25–35 components. At this point, non-predefined components are added to the model, allowing further decrease in the criterion 1 error.

The SDSS data contain approximately 10 times the number of data points as the ESOLV data. The mixture-model approach generally requires a larger number of components to describe the SDSS data set. On average, 34 components were required to describe the ESOLV data using the best model by criterion 1, while 61 components were needed to describe the SDSS data. Using BIC to determine the best model required, on average, 61 components for the ESOLV data and 71 for the SDSS data.

5. DISCUSSION

As can be seen from the results presented above, we are, in general, able to achieve a significantly lower criterion 1 error value when using the mixture approach, which learns based on unlabeled data, as well as the labeled training data. Overall, we obtained a 22% (listed in Table 3 as the fraction 0.219, not as the percent) criterion 1 error for ESOLV data and a 2% error for SDSS data when using criterion 1 as the model-selection method. When using BIC for model selection we obtained a 27% error for ESOLV data and a 4% error for SDSS data. The percentage change column of Table 3 shows that, on average, the mixture models reduced the criterion 1 error for ESOLV data by 20%, but this reduction was as high as 50% when classes 0 and 1 were unknown. Table 4 similarly shows an average 57% reduction in criterion 1 error for SDSS data, with a 96% reduction when classes 0 and 2 were unknown.

Our study is the first to apply semisupervised learning to astronomical data and the first, to our knowledge, to use a data set as large as 50,000 points. This is an important test of the methodology because of the vast amount of astronomical data freely available today, most of which is unlabeled. Demonstrating that our methods work with large astronomical data sets was a primary goal of this work.

Nevertheless, there are a number of factors that influence the reliability of the proposed method and what level of error can be achieved. The results in Tables 3–6 demonstrate the importance of the model order–selection technique. BIC-based selection fares well on the SDSS data, achieving substantially better average criterion 1 results than the neural network optimized for criterion 1 and only modestly worse results than the mixture model optimized for criterion 1 (0.02 vs. 0.04 average error rates). However, there is a significant average performance gap between the two mixture approaches on the ESOLV data (0.22 vs. 0.27), and the BIC-selected mixture is only comparable to the neural network on ESOLV (0.273 vs. 0.277 average error rates). It is possible that a better model order–selection technique could improve the mixture results on ESOLV.

One artifact of optimizing the mixture for criterion 1 is that, on average, smaller models are selected, compared to when the mixtures are selected by BIC. For ESOLV, an average of 34 components were selected by the former approach, while an average of 61 components were selected by the latter. Furthermore, an average of 4.1 non-predefined components were chosen with criterion 1 model selection, while an average of 8.9 were chosen with BIC. For the SDSS data an average of 61 components, including 4.8 non-predefined components, were selected using criterion 1. Finding the best SDSS model by BIC produced an average of 71 components, including 6.4 non-predefined components. While we learned models with up to 80 components, in some cases for the SDSS data the best models were using close to all 80 components. This suggests that it may be reasonable to evaluate solutions with even more components. While the mean number of components selected by BIC is greater than that selected according to criterion 1, the variance in the number of components is much greater when selecting according to criterion 1. This is consistent with the results in Figure 2, which indicate that, for the best criterion 1 performance, the number of components is strongly correlated with the mass of the unknown classes (which varies greatly, since the classes are far from equally likely). This further means that, in some cases, when the mass of the unknown classes is large, using criterion 1 selects more components than using BIC. For example, for the ESOLV data, class 2 occurs 46% of the time. When this class is taken as unknown, BIC selects 63 components, while criterion 1 selects 70.

Another factor that influences model accuracy is the fact that the learning objective function $\mathcal{L}$ is multimodal, with significant potential for finding suboptimal local maxima rather than the global maximum. At each model order, we generated several solutions based on different initializations and picked the one with the greatest log-likelihood. However, there is anecdotal

evidence in our results that we may only be finding locally optimal solutions at each model order. Referring back to Figure 2 we see that the best model for unknown classes 0 and 2 contains only 13 mixture components. This clearly is not in keeping with the trend that more mixture components are needed to explain the data with a larger mass fraction of unknown classes. It appears in this case that a particularly good solution was found. This likewise suggests that, at other orders, suboptimal, local maximum solutions were found.

The criterion 2 error is a measure of how well the algorithm can classify objects within the newly found classes. This is a very difficult problem because we are asking the algorithm to do two things: first, to determine if some mixture components (the non-predefined components) are needed to describe objects that do not fit into the existing known class structure, and second, to partition these objects between the unknown class components, such that each component exclusively owns objects from a single unknown class. This second step is effectively looking for substructure in the newly discovered classes.

For the ESOLV data we find on average about a 70% criterion 2 error compared to about a 25% error for the SDSS data when using criterion 1 model selection. Similarly, we find about a 79% error for the ESOLV data and a 39% error for SDSS data when using BIC model selection. The values of "NA" for the criterion 2 error for some of the SDSS experiments reflect models in which there are no non-predefined components; in this case, the error measure is undefined. Note that, in all these cases, the unknown classes had very small mass, which explains why no non-predefined components were found. In particular, class 0 and class 4 collectively comprise less than 1% of the SDSS data. Thus, when these classes are missing, we would not expect to find non-predefined components in the solution unless both (1) there were more than 100 components in the model and (2) the model criterion selected a solution of this size.

The criterion 3 error measures how well the model can assign objects to known classes. For this measure we obtained about a 32% error for ESOLV data and a 7% error for SDSS data using criterion 1 model selection. When using BIC model selection we obtained a 33% error for ESOLV data and a 9% error for SDSS data.

We find the overall results presented here very promising. The tests done here have demonstrated the efficacy of the class discovery problem and approaches. However, more work will be required to develop a mature technology for highly reliable new class discovery.

We would like to thank the NASA Applied Information Systems Research Program for supporting us in this effort under contract NAS5-02098. One of the authors (D. B.) would like to thank Ofer Lahav for supplying the ESO-LV data. We also thank the referee for pointing out deficiencies in the text and encouraging a clearer exposition of the mixture-model approach.

REFERENCES

Banfield J. D., & Raftery, A. E. 1993, Biometrics, 39, 803
Basu, S., Banerjee, A., & Mooney, R. 2002, in Machine Learning, ed. C. Sammut & A. G. Hoffmann (San Francisco: Morgan Kaufmann), 19
Bazell, D., & Aha, D. W. 2001, ApJ, 548, 219
Cover, T. M., & Thomas, J. A. 1991, Elements of Information Theory (New York: Wiley)
Duda, R. O., Hart, P. E., & Stork, D. G. 2001, Pattern Classification (2nd ed.; New York: Wiley)
Lauberts, A., & Valentijn, E. A. 1989, The Surface Photometry Catalogue of the ESO-Uppsala Galaxies (Garching: ESO)
McLachlan, G., & Peel, D. 2000, Finite Mixture Models (New York: Wiley)
Miller, D. J., & Browning, J. 2003a, IEEE Trans. Pattern Anal. Machine Intell., 25, 1468
———. 2003b, in Neural Networks for Signal Processing, ed. C. Molina, T. Adali, J. Larsen, M. Van Hulle, & S. Douglas (New York: IEEE), 489
Miller, D. J., & Uyar, H. 1997, Neural Inf. Processing Syst., 9, 571
Nigam, K., McCallum, A., Thrun, S., & Mitchell, T. 2000, Machine Learning, 39, 1
Owens, E. A., Griffiths, R. E., & Ratnatunga, K. U. 1996, MNRAS, 281, 153
Schwarz, G. 1978, Ann Stat., 6, 461
Shashahani, B., & Landgrebe, D. 1994, IEEE Trans. Geo. Remote Sens., 32, 1087
Storrie-Lombardi, M. C., Lahav, O., Sodré, L., Jr., & Storrie-Lombardi, L. J. 1992, MNRAS, 259, 8
Stoughton, C., et al. 2002, AJ, 123, 485
Wallace, C. S., & Freeman, P. R. 1987, J. R. Stat. Soc. B, 49, 223
Witten, I. H., & Frank, E. 2000, Data Mining: Practical Machine Learning Tools with Java Implementations (San Francisco: Morgan Kaufman)