Classification of Multi-labeled Data: A Generative Approach

Andreas P. Streich and Joachim M. Buhmann

Institute of Computational Science
ETH Zurich
8092 Zurich, Switzerland
{andreas.streich,jbuhmann}@inf.ethz.ch

Abstract. Multi-label classification assigns a data item to one or several classes. This problem of multiple labels arises in fields like acoustic and visual scene analysis, news reports and medical diagnosis. In a generative framework, data with multiple labels can be interpreted as additive mixtures of emissions of the individual sources. We propose a deconvolution approach to estimate the individual contributions of each source to a given data item. Similarly, the distributions of multi-label data are computed based on the source distributions. In experiments with synthetic data, the novel approach is compared to existing models and yields more accurate parameter estimates, higher classification accuracy and ameliorated generalization to previously unseen label sets. These improvements are most pronounced on small training data sets. Also on real world acoustic data, the algorithm outperforms other generative models, in particular on small training data sets.

1 Introduction

Data classification, the problem of assigning each data point to a set of categories or classes, is the presumably best studied but still challenging machine learning problem. Dichotomies or binary classifications distinguish between two classes, whereas multi-class classification denotes the case of several class choices.

Multi-label classification characterizes pattern recognition settings where each data point may belong to more than one category. Typical situations where multi-labeled data are encountered are classification of acoustic and visual scenes, text categorization and medical diagnosis. Examples are the well-known Cocktail-Party problem [1], where several signals are mixed together and the objective is to detect the original signal, or a news report about Sir Edmund Hillary, which would probably belong to the categories *Sports* as well as to *New Zealand*. The label set for such an article would thus be $\{Sports, NewZealand\}$.

In this paper, we restrict ourselves to generative models, where each data item is assumed to be generated by one (in the single-label case) or several (in the multilabel case) sources. In a probabilistic setting, the goal is to determine which set of sources is most likely to have produced the given data.

Despite its significance for a large number of application areas, multi-label classification has received comparatively little attention. All current approaches

W. Daelemans et al. (Eds.): ECML PKDD 2008, Part II, LNAI 5212, pp. 390–405, 2008.
© Springer-Verlag Berlin Heidelberg 2008

we are aware of reduce the problem to a single-label classification task. The trivial approach for this conceptual simplification either ignores data with multiple labels or considers those items with multiple labels as a new class [2]. Such a modeling strategy generates models which we will denote by $\mathcal{M}_{New}$. More advanced approaches decompose the task into a series of independent binary classification problems, deciding for each of the K classes whether the data at hand belongs to it, and then combine the K classifier outputs to a solution of the original problem. We review these approaches in Sect. 2.

All approaches have significant drawbacks. The trivial approach mainly suffers from data sparsity, as the number of possible label sets is in $\mathcal{O}(K^{d_{\max}})$, where $d_{\max}$ is the maximal size of the sets. Even for moderate K or $d_{\max}$, this is typically intractable. Furthermore, these methods can only assign label sets that already were present in the training data.

The main criticism on the reduction of the multi-label task to a series of binary decision tasks is the confusion between frequent co-occurrence and similar source statistics – in all approaches we are aware of, the more often two sources occur together, the more similar their statistics will be. In this way, these methods neglect the information which multi-labeled data contains about all classes in its label set, which deteriorates the source estimates and leads to poor recall rates. Dependencies between multi-labels are considered in [3], but the approach remains limited to the correlation of sources in the label sets.

In this paper, we propose a novel approach for multi-labeled data, which is inspired by the fundamental physical principle of superposition. We assume a source for each class and consider data with multiple labels as an additive mixture of independent samples of the respective classes. A deconvolution enables us to estimate the contributions of each source to the observed data point and thus to use multi-labeled data for inference of the class distributions. Similarly, the distributions of multi-label data are computed based on the source distributions. Doing so, this approach allows us to consistently model jointly occurring single- and multi-label data with a small amount of parameters. Such a deconvolutive learning technique is only possible for generative models. We therefore exclude purely discriminative classifiers from the further analysis.

In the following, we assume that data is generated by a set $\mathcal{S}$ of K sources. For convenience, we assume $\mathcal{S} = \{1, \ldots, K\}$. For each data item $x_i \in \mathbb{R}^D$, $\mathcal{L}_i = \{\lambda_i^{(1)}, \ldots, \lambda_i^{(d_i)}\}$ denotes the set of sources involved in the generation of x_i. $d_i = \deg \mathcal{L}_i = |\mathcal{L}_i|$ will be called the *degree* of the label set $\mathcal{L}_i$, and $\lambda_i^{(j)} \in \{1, \ldots, K\}$ for all $j = 1, \ldots, d_i$. Label sets with $d_i = 1$ will be called single labels. We denote by $\mathbb{L}$ the set of all admissible label sets – in the most general case, this is simply the power set of the classes except the empty set $\emptyset$, i.e. $\mathbb{L} = 2^{\mathcal{S}} \setminus \{\emptyset\}$, but restrictions to simplify the learning task often are available from the application area. Finally, $\mathcal{X} = (x_1, \ldots, x_N)$ will denote a tuple of data items, and $\mathcal{L} = (\mathcal{L}_1, \ldots, \mathcal{L}_N)$ the corresponding label sets.

The remainder of this paper is structured as follows: Sect. 2 reviews related word, and Sect. 3 then presents the underlying assumption of our method. In Sect. 4, we present the training and classification phase of the proposed

algorithm, both in general and for the special case of Gaussian distributions. Sect. 5 reports results of both synthetic and real world data. A summary and outlook in Sect. 6 concludes the paper.

2 Related Work

Multi-label classification has attracted an increasing research interest in the last decade. It was originally proposed in [4] when introducing error-correcting output codes for solving multiclass learning problems. Later on, a modified entropy formula was employed in [5] to adapt the C4.5-algorithm for knowledge discovery in multi-label phenotype data. Given the modified entropy formula, frequently co-occurring classes are distinguished only on the bottom of the decision tree. Support vector machines as a further type of discriminative classifiers are employed to solve multi-label problems in [6].

An important application area for the problem at hand is text mining. Support vector machines were introduced for this task in [7] and were shown to outperform competing algorithms such as nearest neighbor and C4.5 algorithms. A mixture model for text classification has been presented a year later in [8], where the word distribution of a document is represented as a mixture of the source distributions of the categories the document belongs to.

More recent work includes an application of several multi-label learning techniques to scene classification [2], where it was shown that taking data items with multiple labels as samples of each of the classes yields to discriminative classifiers with higher performance. This approach is called *cross-training* ($\mathcal{M}_{Cross}$).

A similar idea is used in probabilistic learning: Each data item has the same weight, which is then equally distributed among all classes in the label set of the data. $\mathcal{M}_{Prob}$ denotes models which are generated by this technique, which are comprehensively reviewed in [9].

The vaguely related topic of multitask learning is treated in [10]. We understand multitask learning mainly as a method to classify with respect to several different criteria (e.g. street direction and type of the markings, an example in the mentioned paper), the multilabel classification task can be formulated as multitask problem when class membership is coded with binary indicator variables. Each task is then to decide whether a given data item belongs to a class or not. Insofar, we confirm the result that joint training increases the performance. Additionally, our model provides a clearly interpretable generative model for the data at hand, which is often not or only partially true for neural networks.

3 Generative Models for Multi-label Data

The nature of multi-labeled data is best understood by studying how such data are generated. In the following, we contrast our view of multi-labeled data with the standard parametric model of classification where data are generated by one unique source, i.e., data of one specific source do not contain any information on parameters of any other source.

3.1 Standard Generative Model for Classification

In a standard generative model for classification, each data item x_i is a sample of a single source. The source of x_i is identified by the label $\lambda_i \in \mathcal{S}$, and the source distribution will be denoted by P_{λ_i}. Formally, we thus have $x_i \sim P_{\lambda_i}$.

In the learning phase, a set of data points along with corresponding labels is given. Based on this training sample, the class distributions are usually learned such that the likelihood of the observed data, given the class labels, is maximized. Class priors $\Pi = (\pi_1, \ldots, \pi_K)$ can also be learned based on the labels of the training set.

When classifying a new data item x_{new}, the estimated label $\hat{\lambda}_{new}$ is the one with maximal likelihood:

$$\hat{\lambda}_{new} = \arg\max_{\lambda \in \mathcal{S}} L(\lambda | x_{new}, P_\lambda) = \arg\max_{\lambda \in \mathcal{S}} \pi_\lambda L(x_{new} | P_\lambda) \tag{1}$$

This corresponds to a search over the set of possible labels.

3.2 A Generative Model of Multi-labeled Data

We propose an approach to classification of multi-labeled data which extends the generative model for single-label data by interpreting multi-labeled data as a superposition of the emissions of the individual sources. A data item x_j with label set $\mathcal{L}_j = \{\lambda_j^{(1)}, \ldots, \lambda_j^{(d_j)}\}$ of degree d_j is assumed to be the sum of one sample from each of the contributing sources, i.e.

$$x_j = \sum_{s=1}^{d_j} \chi_{\lambda_j^{(s)}} \qquad \text{with} \qquad \chi_{\lambda_j^{(s)}} \sim P_{\lambda_j^{(s)}} \tag{2}$$

The distribution of x_j is thus given by the convolution of all contributing sources:

$$x_j \sim P_{\lambda_j^{(1)}} * \ldots * P_{\lambda_j^{(d_j)}} =: P_{\mathcal{L}_j} \tag{3}$$

Thus, unlike in the single-label model, the distribution of data with multiple labels is traced back to the distribution of the contributing sources. We therefore propose the name *Additive-Generative Multi-Label* Model ($\mathcal{M}_{AdGen}$).

Note that it is possible to explicitly give the distribution $P_{\mathcal{L}_j}$ for data with label set $\mathcal{L}_j$. In contrast to $\mathcal{M}_{New}$, which would estimate $P_{\mathcal{L}_j}$ based solely on the data with this label set, we propose to compute $P_{\mathcal{L}_j}$ based on the distribution of all sources contained in $\mathcal{L}_j$. On the other hand, the estimation of each source distribution is based on all data items which contain the respective source in their label sets.

4 Learning a Generative Model for Multi-labeled Data

In the following, we will first describe the learning and classification steps in general and then we give explicit formula for the special case of Gaussian distributions. In order to simplify the notation, we will limit ourselves to the case of data generated by at most two sources. The generalization to label sets of higher degree is straightforward.

4.1 Learning and Classification in the General Case

The probability distribution of multi-labeled data is given by (3). The likelihood of a data item x_i given a label set $\mathcal{L}_i = \{\lambda_i^{(1)}, \lambda_i^{(2)}\}$, is

$$\begin{aligned} P_{\{\lambda_1^{(1)},\lambda_1^{(2)}\}}(x_i) &= \left(P_{\lambda_i^{(1)}} * P_{\lambda_i^{(2)}}\right)(x_i) \\ &= \int P_{\lambda_i^{(1)}}(\chi) P_{\lambda_i^{(2)}}(x_i - \chi)\,\mathrm{d}\chi \qquad (4) \\ &= \mathbb{E}_{\chi \sim P_{\lambda_i^{(1)}}}\left[P_{\lambda_i^{(2)}}(x_i - \chi)\right]. \qquad (5) \end{aligned}$$

In general, it may not be possible to solve this convolution integral analytically. In such cases, the formulation as an expected value renders Monte Carlo sampling possible to compute a numerical estimate of the data likelihood.

In the training phase, the optimal parameters θ_s of the distribution P_s are chosen such that they fulfil the condition

$$\frac{\partial}{\partial \theta_s}\left\{\prod_{\mathcal{L}\in\mathbb{L}}\prod_{i:\mathcal{L}_i=\mathcal{L}} P_{\mathcal{L}}(x)\right\} \stackrel{!}{=} 0 \qquad \text{for } s = 1,\ldots,K \qquad (6)$$

If the convolution integral (4) can not be expressed analytically, the formulation as expected value ((5), and similar terms for superpositions of three and more sources) can be used to estimate the optimal parameter set $(\theta_1, \ldots, \theta_K)$.

When classifying new data, label sets are assigned according to (1). Again, if the probability distribution of a data item x_i with label sets $\{\lambda_i^{(1)}, \lambda_i^{(2)}\}$ of degree 2 can not be expressed in closed form, (5) might be used to get an estimate of $P_{(\lambda_i^{(1)},\lambda_i^{(2)})}(x_i)$ by sampling χ from $P_{\lambda_i^{(1)}}$. The generalization to label sets of degree larger than 2 is straight forward.

The methods presented here are very general and they are applicable to all parametric distributions. For specific distributions, a closed form expression for the convolution integral and then analytically solving (6) for optimal parameter values will lead to much faster training and classification. The following subsection exemplifies this claim for the Gaussian distributions. Similar explicit convolution formulae can also be computed for, e.g., the chi-square or the Poisson distribution, and approximations exist for many distributions and convolutions of different distributions.

4.2 Gaussian Distributions

Let us assume for the remainder of this section that all source distributions are Gaussians, i.e. $P_s = \mathcal{N}(\mu_s, \Sigma_s)$ for $s = 1,\ldots,K$. The convolution of Gaussian distributions is again a Gaussian distribution, where the mean vectors and the covariance matrices are added:

$$\mathcal{N}(\mu_1, \Sigma_1) * \mathcal{N}(\mu_2, \Sigma_2) = \mathcal{N}(\mu_1 + \mu_2, \Sigma_1 + \Sigma_2)\,. \qquad (7)$$

A corresponding rule holds for convolutions of more than two Gaussians. This property drastically simplifies the algebraic expressions in our model.

Training for Gaussian Distributions. To find the optimal values for the means and the covariance matrices, we have to solve the ML equations

$$\frac{\partial}{\partial \mu_s}\left\{\prod_{\mathcal{L}\in\mathbb{L}}\prod_{i:\mathcal{L}_i=\mathcal{L}} P_{\mathcal{L}}(x)\right\} \stackrel{!}{=} 0 \qquad \frac{\partial}{\partial \Sigma_s}\left\{\prod_{\mathcal{L}\in\mathbb{L}}\prod_{i:\mathcal{L}_i=\mathcal{L}} P_{\mathcal{L}}(x)\right\} \stackrel{!}{=} 0 \tag{8}$$

for $s = 1, \ldots, K$. These conditions yield a set of coupled nonlinear equations, which can be decoupled by proceeding iteratively. As initial values, we choose the sample mean and variance of the single-labeled training data:

$$\mu_s^{(0)} = \frac{\sum_{i:\mathcal{L}_i=\{s\}} x_i}{|\{i:\mathcal{L}_i=\{s\}\}|} \qquad \Sigma_s^{(0)} = \frac{\sum_{i:\mathcal{L}_i=\{s\}} (x_i-\mu_s^{(0)})(x_i-\mu_s^{(0)})^T}{|\{i:\mathcal{L}_i=\{s\}\}|}. \tag{9}$$

For simpler notation, we define the following intermediate values:

$$m^{(t)}_{\mathcal{L}_i\setminus\{s\}} = \sum_{\substack{j\in\mathcal{L}_i\\ j\neq s}} \mu_j^{(t)} \qquad S^{(t)}_{\mathcal{L}_i\setminus\{s\}} = \sum_{\substack{j\in\mathcal{L}_i\\ j\neq s}} \Sigma_j^{(t)} \qquad V^{(t)}_{i\mathcal{L}_i} = (x_i-\mu^{(t)}_{\mathcal{L}_i})(x_i-\mu^{(t)}_{\mathcal{L}_i})^T,$$

where upper indices indicate the iteration steps. Using an iterative approach, the condition for the mean values yields the following update formula for μ_s, $s = 1, \ldots, K$:

$$\mu_s^{(t)} = \left(\sum_{i:\mathcal{L}_i\ni s} (x_i - m^{(t-1)}_{\mathcal{L}_i\setminus\{s\}}) \left(\Sigma^{(t-1)}_{\mathcal{L}_i}\right)^{-1}\right)\left(\sum_{i:\mathcal{L}_i\ni s}\left(\Sigma^{(t-1)}_{\mathcal{L}_i}\right)^{-1}\right)^{-1}.$$

Deriving the data likelihood with respect to the covariance matrix Σ_s yields the following condition:

$$\frac{1}{2}\sum_{i:\mathcal{L}_i\ni s}\left(\left(\mathbb{I}_d - (x_i-\mu_{\mathcal{L}_i})(x_i-\mu_{\mathcal{L}_i})^T\Sigma^{-1}_{\mathcal{L}_i}\right)\Sigma^{-1}_{\mathcal{L}_i}\right) \stackrel{!}{=} 0,$$

where $\mathbb{I}_d$ denotes the identity matrix in d dimensions. With $\Sigma_{\mathcal{L}_i} = \Sigma_s + S_{\mathcal{L}_i\setminus\{s\}}$, the left hand side of the condition can be rewritten as

$$\sum_{i:\mathcal{L}_i=\{s\}}\left(\left(\mathbb{I}_d - V_{i\mathcal{L}_i}\Sigma_s^{-1}\right)\Sigma_s^{-1}\right) + \sum_{\substack{i:\mathcal{L}_i\ni s\\ |\mathcal{L}_i|>1}}\left(\left(\mathbb{I}_d - V_{i\mathcal{L}_i}(S_{\mathcal{L}_i\setminus s}+\Sigma_s)^{-1}\right)(S_{\mathcal{L}_i\setminus s}+\Sigma_s)^{-1}\right)$$

Note that for a training set containing only single label data, the second sum vanishes, and the condition implies estimating Σ_s by the sample variance. If the training set does contain data with multiple labels, the optimality condition can – in general – not be solved analytically, as the condition for Σ_s corresponds to a polynomial which degree is twice the number of allowed label sets in $\mathbb{L}$

containing s. In this case, the optimal value of $\Sigma_s^{(t)}$ can either be determined numerically, or the Taylor approximation

$$(S_{\mathcal{L}_i \setminus s} + \Sigma_s)^{-1} = \Sigma_s^{-1}(S_{\mathcal{L}_i \setminus s}\Sigma_s^{-1} + \mathbb{I}_d)^{-1} \approx \Sigma_s^{-1}(\mathbb{I}_d - \Sigma_s S_{\mathcal{L}_i \setminus s}^{-1}) = \Sigma_s^{-1} - S_{\mathcal{L}_i \setminus s}^{-1}$$

can be used. The approximation is typically quite crude, we therefore prefer using a numerical solver to determine $\Sigma_s^{(t)}$ for all sources s after having determined the mean values $\mu_s^{(t)}$. Whenever a sufficient number of data is available, the covariance matrix of a source s might also be estimated purely based on the data with s as single label.

In spite of the rather complicated optimization procedure for the covariance matrices, we observed that the estimator for the mean values is quite robust with respect to changes in the covariance matrix. Furthermore, the relative importance per data item for the estimation of $\mu_s^{(t)}$ decreases as the degree of its label increases. If enough data with low degree label sets is available in the training phase, the convergence of the training step can be increased by discarding data items with high label degrees with only minor changes in the accuracy of the parameter estimates.

Classification for Gaussian Distributions. Recall the explicit formula for the convolution of two Gaussians (7). This relation yields a simple expression for the likelihood of the data x_{new} given a particular candidate label set $\mathcal{L}_{new} = \{\lambda_{new}^{(1)}, \lambda_{new}^{(2)}\}$:

$$P_{\mathcal{L}_{new}}(x_{new}) = \mathcal{N}(x_{new}; \mu_{\lambda_{new}^{(1)}} + \mu_{\lambda_{new}^{(2)}}, \Sigma_{\lambda_{new}^{(1)}} + \Sigma_{\lambda_{new}^{(2)}})$$

Again, the assignment of the label set for the new data item is done according to (1). As the density functions for data with multiple labels are computed based on the single source densities, this yields more accurate density estimates namely for data with medium to large label degree. This is the second major advantage of the proposed algorithm.

The task of finding the most likely label set, given the data and the source parameters, may become prohibitively expensive if a large number of sources are observed, or if the allowed label degree is large. In the following section, we present an approximation technique that leads to drastically reduced computation costs, while incurring a computable error probability.

4.3 Efficient Classification

In the proposed model, the classification tasks consist of choosing a subset of given sources such that the observed data item has maximal likelihood. The classification task thus comprises a combinatorial optimization problem. While this type of problem is NP-hard in most cases, good approximations are possible in the present case, as we exemplify in the following for Gaussian sources.

For Gaussian distributions with equal spherical covariance matrix $\Sigma_s = \sigma^2 I_D$ for all sources $s = 1, \ldots, K$, maximum likelihood classification of a new data item $x_{new} \in \mathbb{R}^D$ can be reduced to

$$\hat{\mathcal{L}}_{new} = \arg\max_{\mathcal{L}\in\mathbb{L}} \left\{ \frac{\pi_{\mathcal{L}}}{\sigma^D (2\pi d_{\mathcal{L}})^{D/2}} \exp\left(-\frac{||x_{new} - \mu_{\mathcal{L}}||_2^2}{2 d_{\mathcal{L}} \sigma^2} \right) \right\}$$
$$= \arg\min_{\mathcal{L}\in\mathbb{L}} \left\{ ||x_{new} - \mu_{\mathcal{L}}||_2^2 + d_{\mathcal{L}}\sigma^2 (D \log(d_{\mathcal{L}}) - 2\log(\pi_{\mathcal{L}})) \right\}, \quad (10)$$

where $\pi_{\mathcal{L}}$ is the prior probability of the label set $\mathcal{L}$, and $d_{\mathcal{L}} = |\mathcal{L}|$ is its degree.

In cases where the set $\mathbb{L}$ of admissible label sets is relatively small, label set $\hat{\mathcal{L}}_{new}$ with maximal likelihood can be found directly within reasonable computation time. Such a case e.g. arises when the new data can only be assigned to a label set that was also present in the training set, i.e. if $\mathbb{L}$ is the set of all label sets contained in the training sample.

However, in a more general setting, there are no such constraints, and the classifier should also be able to assign a label set that was not seen during the training phase. In this case, $\mathbb{L}$ contains $|2^{\mathcal{S}}| - 1 = 2^K - 1$ possible label sets. The time for direct search thus grows exponentially with the number of sources K.

Our goal is therefore to determine a subset of sources $\mathcal{S}^- \subset \mathcal{S}$ which – with high probability – have not contributed to x_{new}. This constraint to $\mathcal{S}^-$ will limit the search space for the arg min operation and consequently will speed up data processing.

Note that all terms in (10) are positive. The label set prior typically decreases as the degree increases, and the second term grows logarithmically in the size of the label set. The later term thus tends to privilege smaller label sets, and neglecting these two terms might thus yield larger label sets. This is a type of regularization which we omit in the following, as we approximate (10) by the following subset selection problem:

$$\hat{\mathcal{L}}_{new} = \arg\min_{\mathcal{L}\in\mathbb{L}} \left\{ ||x_{new} - \mu_{\mathcal{L}}||_2^2 \right\}, \quad (11)$$

Defining the indicator vector $\hat{\beta}_{new} \in \{0,1\}^K$, with $\hat{\beta}^{(s)}_{new} = 1$ if $s \in \hat{\mathcal{L}}_{new}$ and $\hat{\beta}^{(s)}_{new} = 0$ otherwise, for all sources s, this can be written as

$$\hat{\beta}_{new} = \arg\min_{\beta\in\{0,1\}^K} \left\{ \sum_{s=1}^{K} \beta^{(s)} \mu_s - x_{new} \right\}.$$

Relaxing the constraints on $\hat{\beta}_{new}$, we get the following regression problem:

$$\tilde{\beta}_{new} = \arg\min_{\tilde{\beta}\in\mathbb{R}^K} \left\{ \sum_{s=1}^{K} \tilde{\beta}^{(s)} \mu_s - x_{new} \right\}.$$

Defining the matrix M of mean vectors as $M = [\mu_1, \ldots, \mu_K]$, we obtain the least-squares solution for the regression problem:

$$\tilde{\beta}_{new} = (M^T M)^{-1} M^T x_{new} \quad (12)$$

In order to reduce the size of the search space for the label set, we propose to compute a threshold τ for the components of $\tilde{\beta}_{new}$. Only sources s with $\tilde{\beta}^{(s)}_{new} > \tau$ will be considered further as potential members of the label set $\tilde{\mathcal{L}}_{new}$.

As we have omitted the constraints favoring small label sets, it may happen that single sources with mean close to x_{new} are discarded. This effect can be compensated by adding label sets of small degree (up to 2 is mostly sufficient) containing only discarded classes to the reduced label set. Formally, we have $\mathcal{S}^+ = \{s \in \mathcal{S} | \tilde{\beta}^{(s)}_{new} > \tau\}$, $\mathcal{S}^- = \mathcal{S} \setminus \mathcal{S}^+$, $\mathbb{L}^+ = \left(2^{\mathcal{S}^+} \setminus \{\emptyset\}\right) \cup \mathcal{S}^- \cup \mathcal{S}^- \times \mathcal{S}^-$, and $\mathbb{L}$ replaced by $\mathbb{L}^+$ in (10).

In our experiments, we found that this heuristic can drastically reduce computation times in the classification task. The error probability introduced by this technique is discussed in the following.

We assume the true label set of x_{new} is $\mathcal{L}_{new}$, with the corresponding indicator vector β_{new} and degree $d_{new} = |\mathcal{L}_{new}|$. The heuristic introduces an error whenever $\tilde{\beta}^{(s)}_{new} < \tau$ but $\beta^{(s)}_{new} = 1$ for any $s \in \mathcal{L}_{new}$. Thus,

$$P[\text{error}] = 1 - \prod_{s \in \mathcal{L}_{new}} P[(\tilde{\beta}^{(s)}_{new} > \tau) \wedge (\beta^{(s)}_{new} = 1)].$$

For the analysis, we assume that all source distributions have the same variance $\sigma^2 \cdot \mathbb{I}_d$. Then, we have $x_{new} = M\beta_{new} + \epsilon$, with $\epsilon \sim \mathcal{N}(0, d_{new} \cdot \sigma^2 \mathbb{I}_d)$. Inserting this into (12), we derive

$$\tilde{\beta}_{new} = \beta_{new} + (M^T M)^{-1} M^T \epsilon =: \beta_{new} + \epsilon',$$

where we have defined $\epsilon' = (M^T M)^{-1} M^T \epsilon$, with $\epsilon' \sim \mathcal{N}(0, d_{new}\sigma^2 (M^T M)^{-1})$. Using the eigendecomposition of the symmetric matrix $M^T M$, $M^T M = U \Lambda U^T$, the distribution of ϵ' can be rewritten as $\epsilon' \sim U\mathcal{N}(0, d_{new}\sigma^2 \Lambda^{-1})$. Note that Λ scales with the squared 2-norm of the mean vectors μ, which typically scales with the number of dimensions D.

For the special case when $U = \mathbb{I}_D$, we then have

$$P[\text{error}] = 1 - \prod_{s \in \mathcal{L}_{new}} \left(1 - \Phi\left(\frac{\tau - 1}{\sigma \sqrt{d_{new} \Lambda^{-1}_{ss}}}\right)\right)$$

where $\Phi(\cdot)$ is the cumulative distribution function of the standardized Gaussian. Summing up, the probability of an error due to the heuristic decreases whenever the dimensionality grows (Λ_{ss} grows), sources become more concentrated (σ gets smaller), or the degree of the true label set decreases (d_{new} grows).

For a given classification task, $\mathcal{L}_{new}$ will not be known. In our experiments, we derived an upper limit $d_{\max}$ for the label degree from the distribution of the label set degrees in the training set. For Λ_{ss}, we used the average eigenvalue $\bar{\lambda}$ of the eigendecomposition of $M^T M$. Finally, σ can be estimated from the variance of the single labeled data.

With these estimates, we finally get

$$P[\text{error}] \leq 1 - \left(1 - \Phi\left(\frac{\tau - 1}{\sigma \sqrt{d_{\max} \bar{\lambda}^{-1}}}\right)\right)^{d_{\max}} \tag{13}$$

Given an acceptable error probability, this allows us to choose an appropriate value for the threshold τ. Note that the bound is typically quite pessimistic, as most of the real-world data samples have a large number of data with label sets of small degree. For these data items, the effective error probability is much lower than indicated by (13). Keeping this in mind, (13) provides a reasonable error bound also in the general case where $U \neq \mathbb{I}_D$.

5 Experimental Evaluation

The experiments include artificial and real-world data with multiple labels. In the following, we first introduce a series of quality measures and then present the results.

5.1 Performance Measures

Precision and recall are common quality measures in information retrieval and multi-label classification. These measures are defined on each source. For a source s and a data set $\mathcal{X} = (x_1, \ldots, x_N)$, let tp_s, fn_s, fp_s and tn_s denote the number of true positives, true negatives, false positives and false negatives as defined in Table 1. Then, *precision* and *recall* on source s are defined as follows:

$$Precision_s = \frac{tp_s}{tp_s + fp_s} \qquad Recall_s = \frac{tp_s}{tp_s + fn_s}$$

Intuitively speaking, recall is the fraction of true instances of a base class correctly recognized as such, while precision is the fraction of classified instances that are correct. The F-score is the harmonic mean of the two:

$$F_s = \frac{2 \cdot Recall_s \cdot Precision_s}{Recall_s + Precision_s}$$

All these measures take values between 0 (worst) and 1 (best).

Furthermore, we define the *Balanced Error Rate* (BER) as the average of the ratio of incorrectly classified samples per label set over all label sets:

$$BER = \frac{1}{|\mathbb{L}|} \sum_{\mathcal{L} \in \mathbb{L}} \frac{|\{i | \hat{\mathcal{L}}_i \neq \mathcal{L}_i = \mathcal{L}\}|}{|\{i | \mathcal{L}_i = \mathcal{L}\}|}$$

Table 1. Definition of true positives, true negatives, false positives and false negatives for a base class s

true classification	estimated classification: $x_i \in s$	estimated classification: $x_i \notin s$
$x_i \in s$	tp_C (true positive)	fn_s (false negative)
$x_i \notin s$	fp_s (false positive)	tn_s (true negative)

Note that the BER is a quality measure computed on an entire data set, while $Precision_s$, $Recall_s$ and the F_s-score are determined for each source s.

5.2 Artificial Data

We use artificial data sampled from multivariate Gaussian distributions to compute the accuracy of the source parameter estimates of different models.

The artificial data scenario consisted of 10 sources denoted by $\{1, \ldots, 10\}$. In order to avoid hidden assumptions or effects of hand-chosen parameters, the mean values of the sources were uniformly chosen in the 10-dimensional hypercube $[-2; 2]^{10}$. The covariance matrix was diagonal with diagonal elements uniformly sampled from $[0; 1]$. 25 different subsets of $\{1, \ldots, 10\}$ were randomly chosen and used as label sets. Training sets of different sizes as well as a test set were sampled based on the label sets and the additivity assumption (2). This procedure was repeated 10 times for cross-validation.

Figure 1 shows the average deviation of the mean vectors and the average deviation of the largest eigenvalue from the corresponding true values. For the estimates of the source means, it can be clearly seen that the proposed model is the most accurate. The deviation of the parameters of $\mathcal{M}_{New}$ is explained by the small effective sample size available to estimate each of the mean vectors: As $\mathcal{M}_{New}$ learns a separate source for each label set, there are only two samples per

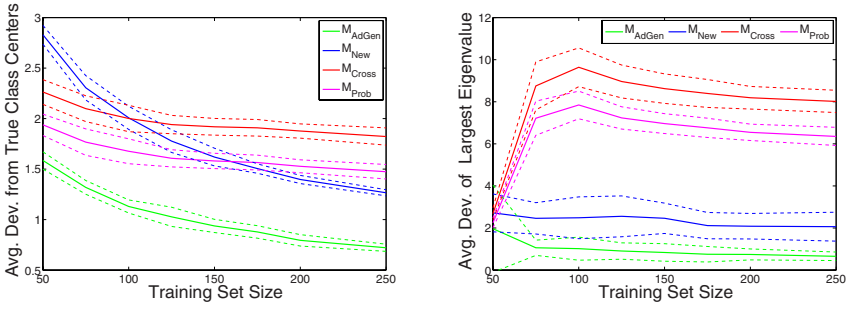

Fig. 1. Accuracy of the parameter estimation of different models. The left panel shows the deviation of the mean estimate, the right one shows the relative deviation between the true and the estimated value of the largest eigenvalue of the covariance matrix. For each model, the average (continuous bold line) over all classes and the standard deviation based on 10-fold cross-validation (dashed lines) is plotted.

We used a setting with 10 sources in 10 dimensions. The mean of each source was chosen uniformly in $[-1, 1]^{10}$. The sources were randomly combined to 25 label sets. Training data sets of different sizes were then sampled according to the generative model.

The generative multi-label model clearly yields the most accurate parameter estimates. The $\mathcal{M}_{New}$ suffers from the small sample size problem, while $\mathcal{M}_{Cross}$ and $\mathcal{M}_{Prob}$ can not clearly improve the estimates of the source parameters.

For the training on sample size 50, we used a default starting value for the covariance matrices. All models have therefore covariance estimates of roughly the same quality.

source when the training set size is 50. $\mathcal{M}_{AdGen}$, on the other hand, decomposes the contributions of each source to every data item. On the average, $\mathcal{M}_{AdGen}$ has thus 2.5 times more training samples per parameter than $\mathcal{M}_{New}$. Furthermore, and the samples used by $\mathcal{M}_{New}$ to estimate the density distribution of multi-labeled data have higher variance than the single label data.

For the estimation of the covariance, $\mathcal{M}_{AdGen}$ still yields distinctly more precise values, but the difference to $\mathcal{M}_{New}$ is not as large as in the estimation of the mean values. This is due to the more complicated optimization problem that has to be solved to estimate the covariance matrix.

The estimates of $\mathcal{M}_{New}$ and $\mathcal{M}_{Prob}$ for both the mean and the covariance are clearly less accurate. Using a data item with multiple label as a training sample independently for each class brings the source parameters closer to each other – and away from their true values. As multi-labeled data have a reduced weight for the estimation of the single sources, this effect is less pronounced in $\mathcal{M}_{Prob}$ than in $\mathcal{M}_{Cross}$.

As in many other machine learning problems, the estimation of the covariance matrix is a hard problem. As no analytic solution of the optimality condition exists and numerical methods have to be used, the computational effort to estimate the covariance grows linearly or even quadratically in the number of dimensions (depending on whether a diagonal or a full covariance matrix is assumed).

Only for spherical covariances, the conditions can be solved to get a coupled set of equations, which can be used for an iterative solution scheme. A possible remedy is to estimate the source covariances based on single label data only, and to use the deconvolution approach only for estimating the mean values. For classification, the proposed method yields considerably more accurate parameter estimates for the distributions of all label sets and therefore performs clearly better.

The estimation of the source means is much more stable and it performs independently of the dimensionality of the data. As expected, the amelioration due to $\mathcal{M}_{AdGen}$ is larger if the covariance matrix does not have to be estimated, and also the improvements in the classification are more pronounced.

5.3 Acoustic Data

For the experiments on real data, we used the research database provided by a collaborating hearing instrument company. This challenging data set serves as benchmark for next generation hearing instruments and captures the large variety of acoustic environments that are typically encountered by a hearing aid user. It contains audio streams of every day acoustic scenes recorded with state of the art hearing instruments. Given the typically difficult acoustic situations in day to day scenes, the recordings have significant artefacts.

Each sound clip is assigned to one of the four classes *Speech* (SP), *Speech in Noise* (SN), *Noise* (NO) and *Music* (MU). While $\mathcal{M}_{New}$ learns a separate source for each of the four label sets, $\mathcal{M}_{Cross}$, $\mathcal{M}_{Prob}$ and $\mathcal{M}_{AdGen}$ interpret SN as a mixture of SP and NO. SN is the only multi-label in our real data setting.

It should be noted that intra-class variance is very high – just consider various genres of music, or different sources of noise! Additionally, mixtures arise in different proportions, i.e. the noise level in the mixture class varies strongly between different sound clips. All these factors render the classification problem a difficult challenge: Even with specially designed features and a large training data set, we have been unable to train a classifier that is able to reach an accuracy of more than 0.75. Precision, recall and the F-score are around 0.80 for all three sources.

Mel Frequency Cepstral Coefficients (MFCCs) [11] have been extracted from the sound clips at a rate of about 100Hz, yielding a 12-dimensional feature vector per time window. As classification is expected to be independent of the signal volume, we have used normalized coefficients. Thus, the additivity assumption (2) has been changed to

$$x_{SP,NO} = \frac{x_{SP} + x_{NO}}{2} \tag{14}$$

Since the extraction of MFCCs is nonlinear, this modified additivity property in the signal space has been transformed into the feature space. A sequence of 10 MFCC feature sets is used as feature vector, describing also the short-time evolution of the signal. Features for the training and test sets have been extracted from different sound clips.

Hidden Markov models (HMM) are widely used in signal processing and speech recognition [12]. We use a HMM with Gaussian output and two states per sound source a simple generative model. In the training phase, we use the approximations

$$\begin{aligned} \mathbb{E}_{\chi\sim P_{NO}}\left[P_{SP}(x_i - \chi)\right] &\approx P_{SP}(x_i - \mathbb{E}_{\chi\sim P_{NO}}\left[\chi\right]) \\ \mathbb{E}_{\chi\sim P_{SP}}\left[P_{NO}(x_i - \chi)\right] &\approx P_{NO}(x_i - \mathbb{E}_{\chi\sim P_{SP}}\left[\chi\right]) \end{aligned}$$

to get a rough estimate of the individual source contributions to a data item x_i with label $\mathcal{L}_i = SN = \{SP, NO\}$. In the classification phase, the formulation of the convolution as expected value (5) is used to estimate the probability of the binary label by sampling from one of the two contributing sources.

Experiments are cross-validated 10 times. In every cross validation set, the number of training samples has been gradually increased from 4 (i.e. one per label set) to 60. The differences in F-score and BER are depicted in Fig. 2. The test sets consist of 255 data items.

Comparing the results of the four algorithms on the test data set, we observe only minor differences in the precision, with $\mathcal{M}_{AdGen}$ tending to yield slightly less precise results. The recall rate of $\mathcal{M}_{AdGen}$, however, is consistently higher than the corresponding results of its three competitors. The F-score obtained by the generic multi-label algorithm is consistently above the F-scores obtained by $\mathcal{M}_{New}$, $\mathcal{M}_{Cross}$ and $\mathcal{M}_{Prob}$. As can be observed in the plots, $\mathcal{M}_{New}$ approaches $\mathcal{M}_{AdGen}$ as the size of the training set increases. The difference between $\mathcal{M}_{AdGen}$ and the two other models does not shows a clear dependency on the size of the training set.

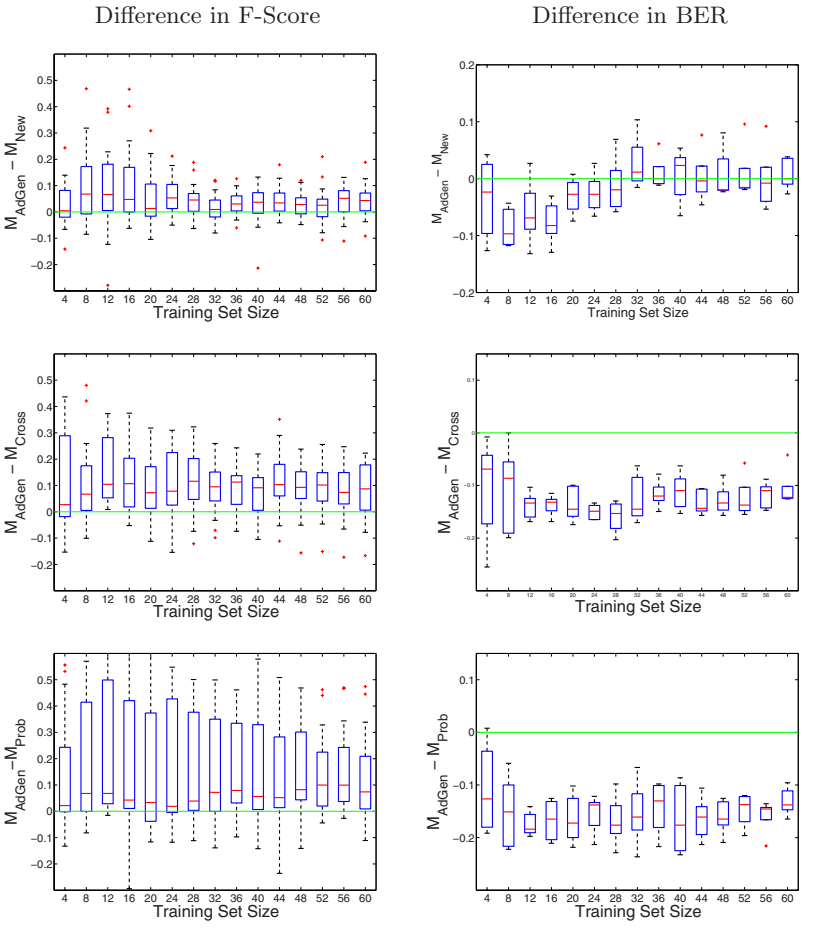

Fig. 2. Difference of quality measures between the proposed method and the three mentioned competing methods. The left column shows the differences in F-Score (higher is better), the right one the differences in BER (lower is better). The differences in F-score are less pronounced and have higher variance than the differences in BER. $\mathcal{M}_{New}$ is the strongest competitor to $\mathcal{M}_{AdGen}$. As the training set grows, $\mathcal{M}_{New}$ comes very close to $\mathcal{M}_{AdGen}$ in terms of F-score and occasionally gets slightly lower BER values. $\mathcal{M}_{Cross}$ and $\mathcal{M}_{Prob}$ are clearly lagging behind $\mathcal{M}_{AdGen}$ in terms of BER and also yield consistently lower F-scores. The absolute values are around 0.6 for the F-score and around 0.4 for the BER at the very small sample sizes.

In all plots, the green horizontal line at 0 indicates equal performance of the two compared algorithms. Note the difference in scale between the two columns.

Differences are more pronounced in terms of the BER. $\mathcal{M}_{New}$ is clearly outperformed on small training sets, but it is able to perform competitively as more training data are available. For larger training sets, learning a separate, independent class for the multi-labeled data as $\mathcal{M}_{New}$ does, sometimes even performs slightly

better, as multi-label data might not fulfill the additivity condition exactly. Independently of the training set size, both $\mathcal{M}_{Cross}$ and $\mathcal{M}_{Prob}$ are clearly performing worse than $\mathcal{M}_{AdGen}$. To our understanding, this is due to a model assumption which does not accurately enough match the nature of the true data source.

6 Conclusion and Outlook

We have presented a generative model to represent multi-labeled data in supervised learning. On synthetic data, this algorithm yields more accurate estimates of the distribution parameters than other generative models and outperforms other approaches for classification of multi-labeled data.

The comparison with other methods on challenging real world data shows that our approach yields consistently higher F-scores on all training set sizes and lower BER values on small training sets. We attribute this finding to the fact that extra regularization renders inference more stable when the training data set is small. This stabilization is observed even in situations where the assumed structure does not exactly match the distribution of the noisy real world data. We conjecture that this mismatch causes a performance drop of the proposed generic multi-label classifier below the performance of the single-class classifier when more data is available for training.

In order to handle recording artefacts common to all sound files, we propose to introduce an extra source to model these effects – similar to the class "English" introduced in [8] to automatically find a task-specific stop list. Thus separating noise due to recording artefacts from the signal should increase precision in recognizing the sources of a given acoustic stream.

Our model used for the acoustic data actually corresponds to a mixture of supervised and unsupervised learning, as for each time frame, one of the two states in the hidden Markov model is selected as a source. This is similar to the mixture discriminant analysis [13], where an unsupervised grouping among all data of one class yield several prototypes within each class. Tracking down these prototypes for multi-label data might yield the distinction between, say, different types of background noise in a mixture with speech.

Furthermore, we have not yet taken into account the fact that several noise sources might be mixed together at different intensities. For example, we might have 70% speech and 30% noise in a conversation situation with moderate background noise, or the opposite of only 30% speech and 70% noise in a very loud environment. In the presented model, both situations are treated equally and lead to a difficult learning and classification task. Modeling mixtures at different intensities is subject to future work.

Finally, the proposed model for data generation is also applicable to unsupervised learning. We expect more precise parameter estimations also in this scenario and thus more stable clustering. The binary assignments in the learning phase of data to its generating classes would be replaced by an estimated responsibility for each (data, class) pair, and the model could then be learned by expectation maximization.

Acknowledgement. This work was in part funded by CTI grant Nr. 8539.2;2 EPSS-ES.

References

1. Arons, B.: A review of the cocktail party effect. Journal of the American Voice I/O Society 12, 35–50 (1992)
2. Boutell, M., Luo, J., Shen, X., Brown, C.: Learning multi-label scene classification. Pattern Recognition, 1757–1771 (2004)
3. Zhu, S., Ji, X., Xu, W., Gong, Y.: Multi-labelled classification using maximum entropy method. In: Proceedings of SIGIR 2005 (2005)
4. Dietterich, T.G., Bakiri, G.: Solving multiclass learning problems via error-correcting output codes. J. of Articificial Intelligence Research 2, 263–286 (1995)
5. Clare, A., King, R.D.: Knowledge discovery in multi-label phenotype data. In: Siebes, A., De Raedt, L. (eds.) PKDD 2001. LNCS (LNAI), vol. 2168, pp. 42–53. Springer, Heidelberg (2001)
6. Elisseeff, A., Weston, J.: Kernel methods for multi-labelled classification and categorical regression problems. In: Proceedings of NIPS 2002 (2002)
7. Joachims, T.: Text categorization with support vector machines: learning with many relevant features. In: Nédellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398. Springer, Heidelberg (1998)
8. McCallum, A.K.: Multi-label text classification with a mixture model trained by EM. In: Proceedings of NIPS 1999 (1999)
9. Tsoumakas, G., Katakis, I.: Multi label classification: An Overview. Int. J. of Data Warehousing and Mining 3(3), 1–13 (2007)
10. Caruana, R.: Multitask learning. Machine Learning 28(1), 41–75 (1997)
11. Pols, L.: Spectral analysis and identification of Dutch vowels in monosyllabic words. PhD thesis, Free University of Amsterdam (1966)
12. Rabiner, L.R.: A tutorial on hidden markov models and selected applications in speech recognition. In: Readings in speech recognition, pp. 267–296 (1990)
13. Hastie, T., Tibshirani, R.: Discriminant analysis by Gaussian Mixtures. J. of the Royal Statist. Soc. B 58, 155–176 (1996)
14. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm. J. of the Royal Statist. Soc. B 39(1), 138 (1977)