Multi-Label Structure Learning with Ising Model Selection

André R Gonçalves
CPqD Foundation /
University of Campinas, Brazil
andrerg@cpqd.com.br

Fernando J Von Zuben
FEEC/Unicamp
University of Campinas, Brazil
vonzuben@dca.fee.unicamp.br

Arindam Banerjee
Computer Science Dept.
University of Minnesota, USA
banerjee@cs.umn.edu

Abstract

A common way of attacking multi-label classification problems is by splitting it into a set of binary classification problems, then solving each problem independently using traditional single-label methods. Nevertheless, by learning classifiers separately the information about the relationship between labels tends to be neglected. Built on recent advances in structure learning in Ising Markov Random Fields (I-MRF), we propose a multi-label classification algorithm that explicitly estimate and incorporate label dependence into the classifiers learning process by means of a sparse convex multi-task learning formulation. Extensive experiments considering several existing multi-label algorithms indicate that the proposed method, while conceptually simple, outperforms the contenders in several datasets and performance metrics. Besides that, the conditional dependence graph encoded in the I-MRF provides a useful information that can be used in a posterior investigation regarding the reasons behind the relationship between labels.

1 Introduction

In Multi-Label (ML) classification a single data sample may belong to many classes at the same time, as opposed to an exclusive single label usually adopted in traditional multi-class classification problems. For example, an image which contains trees, sky, and mountain may belong to *landscape* and *mountains* categories simultaneously; a single gene may be related to a set of diseases; a music/movie may belong to a set of genres/categories; and so on. One can see that multi-label learning includes both binary and multi-class classification problems as specific cases. Thus, such general aspect makes it more challenging then traditional classification problems.

Common strategies to attack ML classification problems are [Madjarov *et al.*, 2012]: (i) algorithm adaptation, and (ii) problem transformation. In the former, well-known learning algorithms such as SVM, neural networks, and decision trees are extended to deal with ML problems. In the latter strategy, the ML problem is decomposed into Q binary classification problems and each one is solved independently using traditional classification methods. This is known as Binary Relevance [Tsoumakas and Katakis, 2007] in ML literature. However, when solving each binary classification problem independently potential dependence among the labels are neglected. And this dependence tends to be very helpful particularly when a limited amount of training data is available.

There have been a number of attempts to incorporate label dependency information in ML algorithms, and they will be properly discussed in Section 4. We anticipate that in most of them, graphical models are used to model label dependence. However, these graphical models usually rely on inference procedures which are either intractable for general graphs or very slow in high-dimensional problems.

Building upon recent advances in structure learn in Ising-Markov Random Fields (I-MRF) [Ravikumar *et al.*, 2010], we propose a multi-label classification method capable of estimating and incorporating the hidden label dependence structure into the classifier learning process. The method involves two steps: (i) label dependence modelling using I-MRF; and (ii) joint learning of all binary classifiers in a regularized sparse convex multi-task learning (MTL) [Caruana, 1997] formulation, where classifiers corresponding to dependent labels in I-MRF are encouraged to share information.

Class labels are modeled as binary random variables and the interaction structure as an I-MRF, so that the I-MRF captures the conditional dependence graph among the labels. The problem of learning the labels (tasks) dependence reduces to the problem of structure learning in the Ising model, on which considerable progress has been made in recent years [Ravikumar *et al.*, 2010; Jalali *et al.*, 2011; Bresler, 2015]. The conditional dependence undirected graph is plugged into a convex sparse MTL formulation, for which efficient optimization methods can be applied [Beck and Teboulle, 2009; Boyd *et al.*, 2011]. The key contributions of this paper are:

- we propose a framework for multi-label classification problems that explicitly capture labels dependence employing a probabilistic graphical model (I-MRF) for which efficient inference procedures are available;
- we employed a stability selection procedure to identify persistent label dependencies (connections) in undirected graph associated with I-MRF;
- we impose sparsity in the coefficient vectors of the binary classifiers, so that the most important discriminative features are automatically selected;

- we have conducted extensive experiments on eight multi-label classification datasets and compared the effectiveness of the proposed formulation in terms of six performance measures.

The remaining of the paper is organized as follows. In Section 2, we cover background material, including the Ising model selection problem and the multi-task learning problems. Section 3 discusses the proposed method to multi-label classification with Ising model selection. Section 4 comments on related work in the literature. Section 5 outlines the experimental setup, describing the multi-label datasets, baseline algorithms, and performance metrics. We discuss experimental results in Section 6, and conclude in Section 7.

2 Background

In this section we provide some background on the relevant topics. We start describing the Ising model selection procedure for structure learning on binary data, followed by an introduction to the multi-task learning formulation.

2.1 Ising model selection

Ising model is a mathematical model originally proposed to study the behavior of atoms in ferro-magnetism. Each atom has a magnetic moment pointing either up or down, called *spin*. The atoms are arranged in a d-dimensional lattice, allowing only direct neighbors atoms to interact to each other.

From a probabilistic graphical model perspective, we can see atoms as binary random variables $x_i \in \{-1, +1\}$. The interaction structure among the atoms can be seen as an undirected graphical model. Let $\mathcal{G} = (V, E)$ be a graph with vertex set $V = \{1, 2, ..., p\}$ and edge set $E \subset V \times V$, and a parameter $\theta_{rs} \in \mathbb{R}$. The Ising model on $\mathcal{G}$ is a Markov random field with distribution given by

$$\mathcal{P}_\Theta(\mathbf{x}) = \frac{1}{\Phi(\mathbf{\Theta})} \exp\left\{ \sum_{(r,s)\in E} \theta_{rs} x_r x_s \right\}. \qquad (1)$$

where $\mathbf{\Theta}$ is a matrix with all parameters for each variable r, $\boldsymbol{\theta}_r$, as columns. Thus, the graphical model selection problem becomes: *Given n i.i.d samples $\mathbf{x} = (x_1, x_2, ..., x_p)$ with distribution given by (1), estimate the edge set E.* Such structural learning problem is difficult to solve due to computational intractability of the partition function $\Phi(\cdot)$ [Welsh, 1993].

Recently, [Ravikumar *et al.*, 2010] proposed a simple and efficient method for the graphical model selection problem. Basically, it involves performing an ℓ_1-regularized logistic regression on each variable while considering the remaining variables as covariates. The sparsity pattern of the regression vector is then used to infer the underlying graphical structure. For all variables $r = 1, ..., p$, the corresponding parameter $\boldsymbol{\theta}_r$ is obtained by

$$\boldsymbol{\theta}_r = \arg\min_{\boldsymbol{\theta}_r} \left\{ \text{logloss}(\mathbf{X}_{\setminus r}, \mathbf{X}_r, \boldsymbol{\theta}_r) + \lambda \|\boldsymbol{\theta}_r\|_1 \right\}. \qquad (2)$$

where logloss$(\cdot)$ is the logistic loss function and $\lambda > 0$ is a trade-off parameter. Note that it can be run in parallel for each label and can scale to problems with large number of labels.

To show the structure recovery capability of the method, a slightly different notion of edge recovery is studied, called *signed edge recovery*, where given a graphical model with parameter Θ, the signed edge set $\bar{E}$ is

$$\bar{E} := \begin{cases} \text{sign}(\theta_{rs}), & \text{if } (r,s) \in E \\ 0, & \text{otherwise.} \end{cases} \qquad (3)$$

The signed edge set $\bar{E}$ can be represented in terms of *neighborhood sets*. For a given vertex r, its neighborhood set is given by $\mathcal{N}(r) := \{s \in V | (r,s) \in E\}$ along with the correct signs $\texttt{sign}(\theta_{rs}), \forall s \in \mathcal{N}(r)$. In other words, the neighborhood set of a vertex r will be those vertices s corresponding to variables whose parameter θ_{rs} is non-zero in the regularized logistic regression. [Ravikumar *et al.*, 2010] showed that recovering the signed edge set $\bar{E}$ of an undirected graph $\mathcal{G}$ is equivalent to recovering the neighborhood set for each vertex.

It is noteworthy that the method in [Ravikumar *et al.*, 2010] can only handle pairwise interactions (clique factors of size $c = 2$). [Jalali *et al.*, 2011] presented a structure learning method for a more general class of discrete graphical models (clique factors of size $c \geq 2$). Block-ℓ_1 regularization is used to select clique factors. [Ding *et al.*, 2011] also considered high-order interactions ($c \geq 2$) among random variables, but conditioned to another random vector (e.g. observed features), similar to the ideas of conditional random fields.

Since the local dependencies are stronger, these can be predominant when estimating the graph. Then the neighborhood dependence (short-range) possibly will hide other long-range dependencies. Most of the methods just mentioned can not get provable recovery under long-range dependencies [Montanari and Pereira, 2009]. Very recently, [Bresler, 2015] presented an algorithm for Ising model with pairwise dependencies and bounded node degree which can also capture long-range dependencies. However, while theoretically proven to be polynomial time, the constants associated with sample complexity and runtime can be quite large.

2.2 Multi-task learning

Multitask Learning (MTL) [Caruana, 1997] is a machine learning paradigm which seeks to improve the generalization capability of a learning task by using information from other related tasks. Suppose we are given a set of Q supervised learning tasks, such that all data for the q-th task come from the space $\mathbf{X} \times \mathbf{Y}$, where $\mathbf{x}_q^i \in \mathbb{R}^d$ and $y_q^i \in \mathbb{R}$, $i = 1, ..., n_q$. So, for each task q a set of n_q data samples are available. The goal is to learn Q parameter vectors $\mathbf{w}_1, ..., \mathbf{w}_Q \in \mathbb{R}^d$ such that $f(\mathbf{x}_q^i, \mathbf{w}_q) \approx y_q^i$, $q = 1, ..., Q$, and $i = 1, ..., n_q$. Learning all tasks together we have the following cost function:

$$\mathcal{L}(\mathbf{X}, \mathbf{Y}, \mathbf{W}) = \sum_{q=1}^{Q} \frac{1}{n_q} \left(\sum_{i=1}^{n_q} \ell\left(f(\mathbf{x}_i^q, y_i^q, \mathbf{w}_q)\right) \right) + \mathcal{R}(\mathbf{W}) \qquad (4)$$

where $\ell(\cdot)$ is the loss function corresponding to the task we are dealing with, which includes squared, logistic, and hinge loss as examples; $\mathbf{W} \in \mathbb{R}^{d \times Q}$ is the parameter matrix, where columns are vector parameters $\mathbf{w}_q$, $q = 1, ..., Q$, for the tasks; $\mathcal{R}(\mathbf{W})$ is a regularization function of $\mathbf{W}$ designed to allow information sharing between tasks. Hence, exploiting

the underlying dependence structure may be advantageous. It is now clear that a fundamental step is to estimate the relationship structure among tasks, thus promoting a proper information sharing among related tasks while avoiding using information from unrelated tasks [Zhang and Yeung, 2010; Gonçalves *et al.*, 2014].

3 Multi-task learning with Ising model selection

This section contains a technical exposition of the proposed I-MTSL (*Ising Multi-task Structure Learning*) algorithm, which consists of two main steps: (i) estimation of the graph representing the dependence among labels, and (ii) estimation of the parameters for all single-label classifier, where the problem is posed as a convex multi-task learning problem.

3.1 Label dependence estimation

Let the conditional random variables representing the labels given the input data, $Z_i = \mathbf{y}_i|\mathbf{X}, i = 1, ..., Q$, be binary random variables. We then assume that the joint probability distribution of $\mathbf{Z} = (Z_1, Z_2, ..., Z_Q)$ follows an Ising Markov random field. So, given a collection of n i.i.d. samples $\{z^{(1)}, ..., z^{(n)}\}$, where each Q-dimensional vector $z^{(i)} \in \{-1, +1\}^Q$ drawn from the distribution $\mathcal{P}_\Theta$ (Eq. (1)), the problem is to learn the undirected graph $\mathcal{G} = (V, E)$ associated with the binary Ising Markov random field. We then use the method described Section 2.1 to infer the edge set E. Recall that, in fact, the method estimates the signed set of edges $\bar{E}$, i.e., each edge takes either "-1" or "$+1$" value.

The undirected graph $\mathcal{G}$ encodes the conditional dependencies among the labels. The edge absence between two nodes indicates that the corresponding labels are conditionally independent. Such information is crucial in the multi-task learning, which tells with whom each task shares information.

Stability selection

To find stable connections in the undirected graph associated with the I-MRF, a stability selection procedure is applied. We used the sub-sampling based technique proposed by [Meinshausen and Bühlmann, 2010]. With it, we eliminate possible spurious label dependencies due to noise and/or random data fluctuation. If such spurious connections are incorporated directly into the multi-task learning formulation, it can mislead the algorithm to share information among non-related tasks, which may adversely affect the performance of the classifiers.

The stability selection algorithm proceeds as follows: (1) sub-samples of size $\lfloor n/2 \rfloor$ are generated without replacement from the training data; (2) for each sub-sample the structure learning algorithm is applied; and (3) we then select the persistent connections, which are those that appeared in a large fraction of the resulting selection sets. For this, a cutoff threshold $0 < \pi_{thr} < 1$ is needed. In our experiments we set $\pi_{thr} = 0.8$, then a connection is said to be consistent if it appears in 80% of graphs constructed from the sub-samples.

To the best of authors' knowledge, the use of stability selection procedure to obtain the undirected graph of label dependence is a novelty of our paper.

3.2 Task parameters estimation

Once estimated the graph $\mathcal{G}$, we turn our attention now to the joint learning of all single-label classifiers.

In I-MTSL, we use the learned dependence structure among labels in an inductive bias regularization term which enforces related tasks to have similar task parameters $\mathbf{w}$. Tasks coefficients in I-MTSL are estimated by solving:

$$\min_{\mathbf{W}} \sum_{q=1}^{Q} \frac{1}{n_q} \sum_{i=1}^{n_q} \ell\left(\mathbf{x}_q^i, y_q^i, \mathbf{w}_q\right) + tr(\mathbf{W}\bar{\mathbf{L}}\mathbf{W}^T) + \gamma\|\mathbf{W}\|_1 \tag{5}$$

where $\bar{\mathbf{L}}$ is the signed Laplacian matrix computed from the signed edge set $\bar{\mathbf{E}}$ [Kunegis *et al.*, 2010], $tr(\cdot)$ is the trace operator, and $\gamma > 0$ is a penalization parameter. $\bar{L}$ is computed as $\bar{\mathbf{L}} = \bar{\mathbf{D}} - \bar{\mathbf{E}}$, where $\bar{\mathbf{D}} \in \mathbb{R}^{Q\times Q}$ is a diagonal matrix $\bar{D}_{ii} = \sum_{i\sim j} |\bar{E}_{ij}|$. As matrix $\bar{\mathbf{L}}$ is positive semi-definite (see [Kunegis *et al.*, 2010]) the problem (5) is (non-smooth) convex. The signed Laplacian is an extension of the ordinary Laplacian matrix when negative edges are present. Allowing negative edges, the multi-task learning method is then capable of modeling positive and negative tasks relationships, which is not always the case in existing MTL formulations [Argyriou *et al.*, 2008; Obozinski *et al.*, 2010].

The first term in the minimization problem (5) refers to any binary classification loss function, such as logistic and hinge loss. The second term is the bias inductive term which favors related tasks' weights to be similar. The third term induces sparsity on $\mathbf{W}$ matrix, which automatically selects the most relevant features, setting to zero weights of non-relevant ones.

The I-MTSL algorithm is outlined in 1. Note that no iterative process is required.

Algorithm 1 I-MTSL algorithm.

Require: $\mathbf{X}, \mathbf{Y}, \lambda > 0, \gamma > 0$
1: **for** $q = 1$ to Q **do**
2: $\quad \boldsymbol{\Theta}_{(q,:)} = \underset{\theta_q}{\text{argmin}} \left\{\text{logloss}(\bar{\mathbf{Y}}_{\setminus k}, \mathbf{y}_k, \boldsymbol{\theta}_q) + \lambda\|\boldsymbol{\theta}_q\|_1\right\}$;
3: **end for**
4: $\bar{\mathbf{L}} = \bar{\mathbf{D}} - \bar{\mathbf{E}}$;
5: Compute $\mathbf{W}$ by solving (5);
6: **return** $\mathbf{W}, \bar{\mathbf{E}}$.

3.3 Optimization

For both optimization problems (2) and (5), an accelerated proximal gradient method was used. In such class of algorithms the cost function $h(x)$ is decomposed as $h(x) = f(x) + g(x)$, where $f(x)$ is a convex and differentiable function and $g(x)$ is convex and typically non-differentiable. Thus, the accelerated proximal gradient iterates as follows

$$\begin{aligned} \mathbf{z}^{t+1} &:= \mathbf{x}^t + \omega^t\left(\mathbf{x}^t - \mathbf{x}^{t-1}\right) \\ \mathbf{x}^{t+1} &:= \mathbf{prox}_{\rho^t g}\left(\mathbf{z}^{t+1} - \rho^t \nabla f\left(\mathbf{z}^{t+1}\right)\right) \end{aligned} \tag{6}$$

where $\omega^t \in [0, 1)$ is an extrapolation parameter and ρ^t is the step size. The ω^t parameter is chosen as $\omega^t = (\eta_t - 1)/\eta_{t+1}$,

with $\eta_{t+1} = (1+\sqrt{1+4\eta_t^2})/2$ in [Beck and Teboulle, 2009] and ρ^t can be computed by a line search.

The $g(x)$ term in both problems corresponds to the ℓ_1-norm, which has a cheap proximity operator defined as

$$\mathbf{prox}_{\rho^t g}(\mathbf{x})_i = (|x_i| - \alpha)_+ \text{sgn}(x_i) \tag{7}$$

which is known as soft-threshold operator and is interpreted element-wise. It is a simple application of a function that can even be done in parallel. The convergence rate of the algorithm is $\mathcal{O}(1/k^2)$ [Beck and Teboulle, 2009]. Defining logistic loss as the cost function $\ell(\cdot)$ and writing (5) in the form of $\mathbf{vec}(\mathbf{W}) \in \mathbb{R}^{dQ\times 1}$, the gradient of the function $f(\cdot)$ is computed as

$$\nabla f\left(\mathbf{vec}(\mathbf{W})\right) = \bar{\mathbf{X}}^T\left[\mathbf{vec}(\mathbf{Y}) - \sigma\left(\bar{\mathbf{X}}\mathbf{vec}(\mathbf{W})\right)\right] + \mathbf{P}\left(\bar{\mathbf{L}} \otimes \mathbb{I}_d\right)\mathbf{P}^T\mathbf{vec}(\mathbf{W}) \tag{8}$$

where $\sigma(\cdot)$ is the sigmoid function, $\mathbf{P}$ is a permutation matrix that converts the column stacked arrangement of $\mathbf{vec}(\mathbf{W})$ to a row stacked arrangement, and $\otimes$ is the Kronecker product. $\bar{\mathbf{X}}$ is a block diagonal matrix where the main diagonal blocks are the task input data matrices $\mathbf{X}_q, \forall q = 1, ..., Q$, and the off-diagonal blocks are zero matrices. The gradient of (2) is simply the derivative of the logistic loss function w.r.t. $\boldsymbol{\theta}_r$.

4 Related work

A number of papers have explored ways of incorporating label dependence into ML algorithms. The early work of [McCallum, 1999] used a mixture model trained via Expectation-Maximization to represent the correlations between class labels. In [Read *et al.*, 2011] information from other labels are stacked as features, in a chain fashion, for the binary classifier corresponding to a specific label. Then, high importance will be given to those features associated with correlated labels. None of these, however, explicitly model labels dependence.

Among the explicit modeling approaches, [Rai and Daume, 2009] present a sparse infinite cannonical correlation analysis to capture label dependence, where a non-parametric Bayesian prior is used to automatically determine the number of correlation components. Due to the model complexity, the parameters estimation relies on sampling techniques which may be very slow for high-dimensional problems.

Somewhat similar in spirit to our approach, many papers have employed probabilistic graphical models to explicitly capture label dependence. Bayesian networks were used to model label dependence in [de Waal and van der Gaag, 2007; Zhang and Zhang, 2010], and [Bielza *et al.*, 2011]. However, the structure learning problem associated with Bayesian networks is known to be NP-hard [Chickering, 1996]. Markov networks formed from random spanning trees are considered in [Marchand *et al.*, 2014]. Conditional random field (CRF) was used in [Ghamrawi and McCallum, 2005], where the binary classifier for a given label not only considered its own data, but also information from neighboring labels determined by the undirected graphical model encoded into the CRF model. [Bradley and Guestrin, 2010] also proposed a method for efficiently learning tree structures for CRFs. For general graphs, however, the inference problems in CRF are intractable, and efficient exact inference is only possible for more restricted graph structures such as chains and trees [Sutton and McCallum, 2011]. [Shahaf and Guestrin, 2009] also present a structure learning method for a more restrict class of models known as low-treewidth junction trees. We use a more general undirected graph, I-MRF, that can capture any pairwise label dependence and for which efficient (and highly parallelizable) structure learn procedures have been recently proposed. These new approaches including [Ravikumar *et al.*, 2010] avoid the explicit reliance of the classical structure learning approaches on inference in the graphical model, making them computationally efficient and statistically accurate. We have discussed recent developments on Ising model structure learning in Section 2.1. Here, the dependence graph is plugged into a regularized MTL formulation, a paradigm which has shown improvements in predictive performance relative to traditional machine learning methods.

5 Experimental design

In this section we present a set of experiments on multi-label classification to assess the performance of the proposed I-MTSL algorithm.

5.1 Datasets

For the experiments we have chosen eight well-known datasets in the multi-label classification literature. Those datasets are from different application domains and have different number of samples, labels, and dimensions. Table 1 shows a basic description of the datasets. For a detailed characterization, refer to [Madjarov *et al.*, 2012]. All datasets were downloaded from Mulan webpage[1].

Dataset	Domain	# samples	Features	# labels
Emotions	music	593	72	6
Scene	image	2407	294	6
Yeast	biology	2417	103	14
Birds	audio	645	260	19
Genbase	biology	662	1186	27
Enron	text	1702	1001	53
Medical	text	978	1449	45
CAL500	music	502	68	174

Table 1: Description of the multi-label classification datasets.

5.2 Baselines

Five well known methods were considered in the comparison: three state-of-the-art MTL algorithms: CMTL [Zhou *et al.*, 2011a], *Low rank MTL* (LowRank) [Ji and Ye, 2009], and MTL-FEAT [Argyriou *et al.*, 2008]; besides two popular ML algorithms: Binary Relevance (BR) [Tsoumakas and Katakis, 2007] and Classifier Chain (CC) [Read *et al.*, 2011]. CC algorithm incorporates other labels information as covariates in a chain fashion, then label dependence information is explored in the classifier parameter estimation process. For CMTL, LowRank, and MTL-FEAT we used the MALSAR [Zhou *et al.*, 2011b] package. The remaining methods were implemented by the authors (I-MTSL code will be released).

[1] http://mulan.sourceforge.net/datasets-mlc.html

5.3 Experimental setup

Logistic regression was used as the base classifier for all algorithms. Z-score normalization was applied to all datasets, then covariates have zero mean and standard deviation one.

For all methods, parameters were chosen following the same procedure. We selected 20% of the training set to act as validation set (holdout cross-validation) and tested the parameters on a grid containing ten equally spaced values in the interval [0,5]. The parameter with the best average accuracy over all binary classification problems in the validation set was used in the test set. The results presented in the next sections are based on ten independent runs of each algorithm.

5.4 Evaluation measures

To assess the performance of multi-label classifiers is essential to consider multiple and contrasting evaluation measures due to the additional degrees of freedom that the multi-label setting introduces [Madjarov *et al.*, 2012]. Thus, six different measures were used: *accuracy*, *1 - Hamming loss* (1-HL), *macro-F1*, *precision*, *recall*, and *F1-score*. For a detailed description of those measures, refer to [Madjarov *et al.*, 2012]. In essence, all the measures produce a number in the interval [0,1], with higher values indicating better performance. We show the complement of HL for easy of exposition.

6 Results and Discussion

The results for all datasets and evaluation measures are shown in Figure 1. As expected, the performance of the algorithms varies as we look at different evaluation measures.

BR shows the worst performance among the algorithms for almost all datasets/metrics, except for *Emotions* dataset, which has the smallest number of labels/attributes. However, the difference is more pronounced as we have more labels, low performance seen on *Medical*, *Enron*, and *CAL500* datasets. It indicates that the information regarding dependence among labels, indeed, helps to improve performance.

The use of several performance measures is intended to show the distinct characteristics of the algorithms. As we can see in the plots, many algorithms do well for some metrics, while do poorly in others. To have an overall performance investigation we propose the use of a metric to compare all algorithms' relative performance. To do so, we use a measure inspired in a well-known metric in the literature of learn to rank, *Discounted Cumulative Gain* (DCG) [Järvelin and Kekäläinen, 2002]. Such measure we referred to here as *relative performance* (RP).

To obtain RP, first we compute the ranking r of all algorithms for a specific dataset/metric, then for a given algorithm a, $RP(a)$ is obtained as: $RP(a) = 1$ if $r_a = 1$ and $RP(a) = 1/\log_2 r(a)$, otherwise. It basically gives higher values to algorithms at the top with a logarithm discount as the rank goes down. Similar to the definition DCG [Järvelin and Kekäläinen, 2002], RP also gives equal importance to the first and second best algorithms. RP can be seen as a special case of DCG metric used to measure learn to rank algorithms, where only one relevant (1) document is returned at position r and all other are non-relevant (0), given a query. RP value ranges from 0 to 1, with 1 representing that the algorithm figured at the top. The logarithm discount in RP induces a smoother penalization to algorithm's rank than when considering the ranks directly. Table 2 shows the RP values computed over all datasets for all pairs algorithm/metric.

	1-HL	Acc	MacF1	Rec	Prec	F1
BR	0.39 ±.02	0.54 ±.09	0.46 ±.09	0.41 ±.04	0.70 ±.21	0.54 ±.09
CC	0.65 ±.25	0.77 ±.22	0.69 ±.29	0.55 ±.21	0.95 ±.14	0.77 ±.22
CMTL	0.65 ±.25	0.80 ±.25	0.74 ±.25	0.63 ±.26	0.54 ±.06	0.80 ±.25
Trace	0.64 ±.26	0.41 ±.02	0.51 ±.22	0.86 ±.25	0.41 ±.02	0.41 ±.02
MTL-F.	0.79 ±.26	0.43 ±.04	0.73 ±.27	**0.95** **±.14**	0.41 ±.02	0.43 ±.04
I-MTSL	**0.82** **±.22**	**1.00** **±0**	**0.81** **±.24**	0.55 ±.08	**0.95** **±.14**	**1.00** **±0**

Table 2: Mean and standard deviation of RP values. I-MTSL has a better balanced performance and is among the best algorithms for the majority of the metrics.

I-MTSL obtained better *accuracy* when compared to the remaining methods, as it figures at the top of all the datasets. In essence, the *accuracy* computes the Jaccard similarity coefficient between the set of labels predicted by the algorithm and the observed labels. The algorithm is also at the top for the majority of the datasets regarding *1-HL*, *macro-F1*, *precision*, and *F1-score*. Thus, I-MTSL obtained more balanced solutions, figuring at the top for the most of the analyzed metrics, except for *recall*. CMTL, LowRank, and MTF-FEAT, on the other hand, yielded the highest *recall*, but the lowest *precision*. Notice that it is easy to increase *recall* by predicting more 1's, however it may hurt *accuracy*, *precision* and *F1-score*. As the class imbalance problem is recurrent in multi-label classification, it may deceive the algorithm to polarize the prediction to a certain class.

In terms of *macro-F1*, I-MTSL also outperforms the contenders. *Macro-F1* evaluates the algorithm performance across the labels not across samples. It shows how good is an algorithm to classify labels independently. BR clearly has the worst result, which was expected, as BR is the only algorithm that does not use label dependence information.

Figure 2 presents examples of signed Laplacian matrices computed from the graph associated with the Ising model structure learned by I-MTSL for four of the datasets considered in the experiments. It is interesting to note the high sparsity of the matrices, showing that only a few tasks are conditionally dependent on each other and that structure led to a better classification performance. For some datasets, such as *Enron*, *Medical*, and *Genbase* we can clearly see a group of labels which are mutual dependent. Such matrix can be very useful in a posterior investigation regarding the reasons underlying those dependent labels.

7 Conclusion

We presented a method for multi-label classification problems that is capable of estimating the inherent label depen-

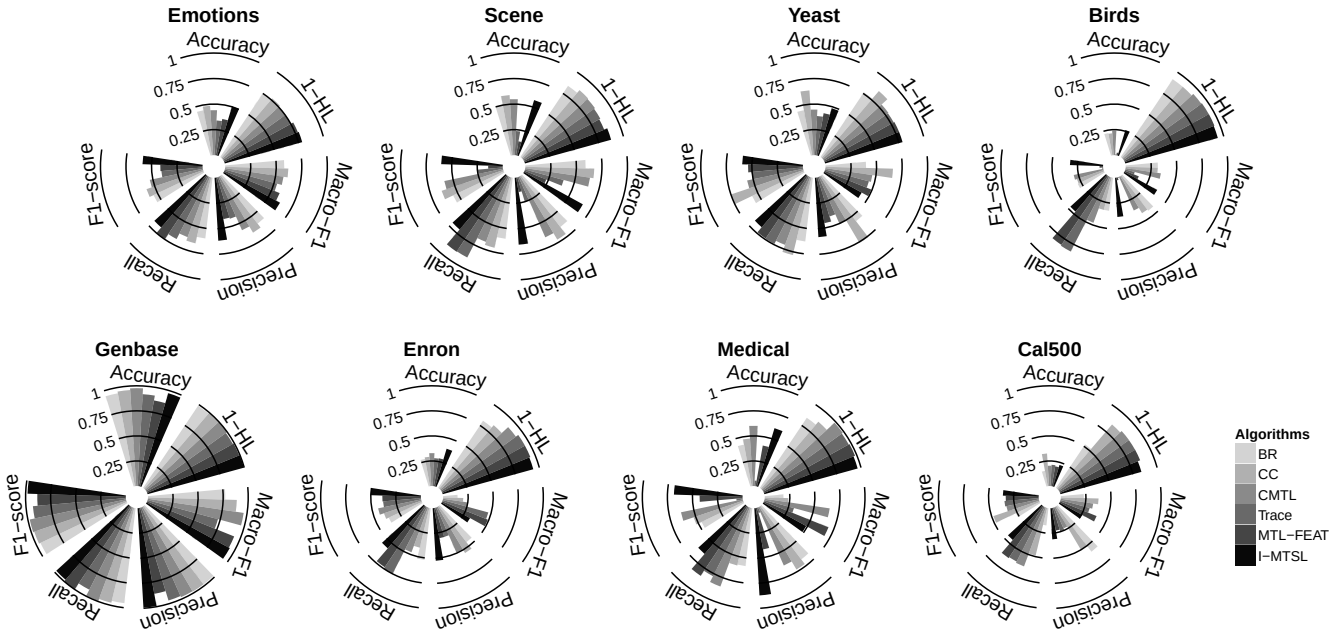

Figure 1: Algorithms' performance on multi-label classification problems in terms of six distinct evaluation measures. The performance significantly varies as we look at different metrics. However, I-MTSL figures at top for the majority of the datasets and metrics.

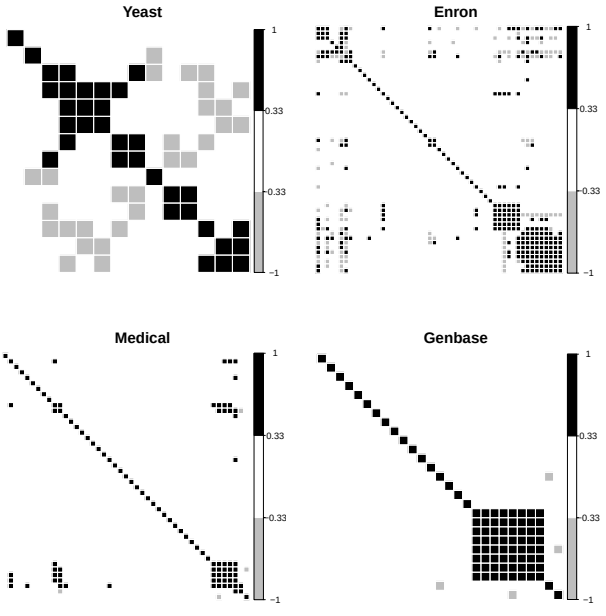

Figure 2: Signed Laplacian matrix of the undirected graph associated with I-MTSL using stability selection procedure. Black and gray squares mean positive and negative relationship respectively. Note the high sparsity and the clear group structure among labels.

dence structure. Such information is incorporated into the classifier learning process through a convex multi-task learning formulation. We model class labels as binary random variables and the interaction among the labels as an Ising Markov Random Field (I-MRF), so that the structure of the I-MRF captures the conditional dependence graph among the labels. We propose the use of a stability selection procedure to choose only stable label dependencies (graph connections). The problem of learning the labels dependence then reduces to the problem of structure learning in the Ising model, to which efficient methods have been recently proposed.

A comprehensive set of experiments on multi-label classification were carried out to demonstrate the effectiveness of the algorithm. Results showed its superior performance in several datasets and multiple evaluation metrics, when compared to already proposed multi-label and MTL algorithms. The algorithm exhibits the best compromise considering all performance metrics. Also, the learned graph associated with the I-MRF can be used in a posterior investigation regarding the reasons behind the relationship between labels.

Learning label dependence using more general graphical models (such as the ones described in Section 2.1) and embedding it into the binary relevance classifiers learning process will be the subject of future work.

Acknowledgments. ARG and FJVZ thank CNPq for the financial support. AB was supported by NSF grants IIS-1447566, IIS-1422557,CCF-1451986, CNS-1314560, IIS-0953274, IIS-1029711, NASA grant NNX12AQ39A, and gifts from Yahoo and IBM. Access to computing facilities were provided by University of Minnesota Supercomputing Institute (MSI).

References

[Argyriou *et al.*, 2008] A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Machine Learning*, 73(5):243–272, 2008.

[Beck and Teboulle, 2009] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. on Imaging Sciences*, 2(1):183–202, 2009.

[Bielza *et al.*, 2011] C. Bielza, G. Li, and P. Larranaga. Multi-dimensional classification with bayesian networks. *International Journal of Approximate Reasoning*, 52(6):705–727, 2011.

[Boyd *et al.*, 2011] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 2011.

[Bradley and Guestrin, 2010] J.K. Bradley and C. Guestrin. Learning tree conditional random fields. In *ICML*, pages 127–134, 2010.

[Bresler, 2015] Guy Bresler. Efficiently learning ising models on high degree graphs. *STOC*, 2015.

[Caruana, 1997] R. Caruana. Multitask learning. *Machine Learning*, pages 41–75, 1997.

[Chickering, 1996] D.M. Chickering. Learning Bayesian networks is NP-complete. In *Learning from data*. Springer, 1996.

[de Waal and van der Gaag, 2007] Peter R de Waal and Linda C van der Gaag. Inference and learning in multi-dimensional bayesian network classifiers. In *Symb. and Quant. Appr. to Reas. with Uncert.*, pages 501–511. 2007.

[Ding *et al.*, 2011] Shilin Ding, Grace Wahba, and Xiaojin Zhu. Learning higher-order graph structure with features by structure penalty. In *NIPS*, pages 253–261, 2011.

[Ghamrawi and McCallum, 2005] N. Ghamrawi and A McCallum. Collective multi-label classification. In *CIKM*, pages 195–200, 2005.

[Gonçalves *et al.*, 2014] A.R. Gonçalves, P. Das, S. Chatterjee, V. Sivakumar, F.J. Von Zuben, and A. Banerjee. Multi-task Sparse Structure Learning. In *CIKM*, pages 451–460, 2014.

[Jalali *et al.*, 2011] A. Jalali, P. Ravikumar, V. Vasuki, and S. Sanghavi. On learning discrete graphical models using group-sparse regularization. In *AISTATS*, pages 378–387, 2011.

[Järvelin and Kekäläinen, 2002] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Trans. on Inf. Sys.*, 20(2):422–446, 2002.

[Ji and Ye, 2009] S. Ji and J. Ye. An accelerated gradient method for trace norm minimization. In *ICML*, 2009.

[Kunegis *et al.*, 2010] J. Kunegis, S. Schmidt, A. Lommatzsch, J. Lerner, E.W. De Luca, and S. Albayrak. Spectral analysis of signed graphs for clustering, prediction and visualization. In *SDM*, 2010.

[Madjarov *et al.*, 2012] G. Madjarov, D. Kocev, D. Gjorgjevikj, and S. Džeroski. An extensive experimental comparison of methods for multi-label learning. *Pattern Recognition*, 45(9):3084–3104, 2012.

[Marchand *et al.*, 2014] M. Marchand, H. Su, E. Morvant, J. Rousu, and J.S. Shawe-Taylor. Multilabel structured output learning with random spanning trees of max-margin markov networks. In *NIPS*, pages 873–881, 2014.

[McCallum, 1999] A.K. McCallum. Multi-label text classification with a mixture model trained by EM. In *AAAI Workshop on Text Learning*, pages 1–7, 1999.

[Meinshausen and Bühlmann, 2010] N. Meinshausen and P. Bühlmann. Stability selection. *Journal of the Royal Statistical Society: Series B*, 72(4):417–473, 2010.

[Montanari and Pereira, 2009] A Montanari and J.A. Pereira. Which graphical models are difficult to learn? In *NIPS*, pages 1303–1311. 2009.

[Obozinski *et al.*, 2010] G. Obozinski, B. Taskar, and M. Jordan. Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, 20(2):231–252, 2010.

[Rai and Daume, 2009] P. Rai and H. Daume. Multi-label prediction via sparse infinite CCA. In *NIPS*, pages 1518–1526, 2009.

[Ravikumar *et al.*, 2010] P. Ravikumar, M.J. Wainwright, and J.D. Lafferty. High-dimensional Ising model selection using ℓ_1-regularized logistic regression. *The Annals of Statistics*, 38(3):1287–1319, 2010.

[Read *et al.*, 2011] J Read, B. Pfahringer, G. Holmes, and E. Frank. Classifier chains for multi-label classification. *Machine learning*, 85(3):333–359, 2011.

[Shahaf and Guestrin, 2009] D. Shahaf and C. Guestrin. Learning thin junction trees via graph cuts. In *AISTATS*, pages 113–120, 2009.

[Sutton and McCallum, 2011] C. Sutton and A. McCallum. An introduction to conditional random fields. *Machine Learning*, 4(4):267–373, 2011.

[Tsoumakas and Katakis, 2007] G. Tsoumakas and I. Katakis. Multi-label classification: An overview. *Int. Journal of Data Warehousing and Mining*, 2007.

[Welsh, 1993] D. Welsh. *Complexity: knots, colourings and countings*. Cambridge University Press, 1993.

[Zhang and Yeung, 2010] Y. Zhang and D.-Y. Yeung. A convex formulation for learning task relationships in multi-task learning. In *UAI*, 2010.

[Zhang and Zhang, 2010] Min-Ling Zhang and Kun Zhang. Multi-label learning by exploiting label dependency. In *KDD*, pages 999–1008. ACM, 2010.

[Zhou *et al.*, 2011a] J. Zhou, J. Chen, and J. Ye. Clustered Multi-Task learning via alternating structure optimization. In *NIPS*, 2011.

[Zhou *et al.*, 2011b] J. Zhou, J. Chen, and J. Ye. *MALSAR: Multi-tAsk Learning via StructurAl Regularization*. Arizona State University, 2011.