Large-scale Bayesian Multi-label Learning via Positive Labels Only

Piyush Rai PIYUSH.RAI@DUKE.EDU
Changwei Hu CH237@DUKE.EDU
Ricardo Henao RICARDO.HENAO@DUKE.EDU
Lawrence Carin LCARIN@DUKE.EDU
Department of Electrical and Computer Engineering, Duke University, Durham 27708, USA

Abstract

We present a scalable Bayesian multi-label prediction framework that depends only on the positive labels in the label matrix. Dependence only on the positive labels (of a possibly very massive and *incompletely* observed label matrix) makes our model especially attractive for settings where only positive labels can be obtained in practice, or where the label matrix contains both observed zeros and ones but is highly *sparse* (i.e., has very few ones in it). Moreover, in contrast to the commonly used logistic or probit likelihood, our model uses an *asymmetric* inverse link function for the binary labels, which naturally also provides robustness against the label imbalance. Furthermore, using a data-augmentation strategy leads to a fully locally conjugate model, facilitating simple and very efficient Gibbs sampling, as well as an Expectation Maximization (EM) algorithm for our model; the computational cost for both algorithms scales in the number of ones in the label matrix. We report experimental results on several benchmark data sets comparing our model with various state-of-the art methods.

1. Introduction

Multi-label prediction refers to the problem setting where the goal is to assign to an object (e.g., a video, image, or webpage) a *subset* of labels (e.g., tags) from a (possibly very large) *set* of labels (e.g., a tag vocabulary). Despite a huge amount of prior work, multi-label prediction (Gibaja & Ventura, 2015; 2014) continues to be an active area of research, with a recent surge of interest (Agrawal et al., 2013; Yu et al., 2014; Prabhu & Varma, 2014; Kong et al., 2014; Kapoor et al., 2012) in designing scalable multi-label learning methods to address the challenges posed by modern applications such image/video/webpage annotation (Prabhu & Varma, 2014), computational advertising (Agrawal et al., 2013; Prabhu & Varma, 2014), medical coding (Yan et al., 2010), etc., where not only the number of examples and data dimensionality is large but the number of labels can also be massive (several thousands to even millions).

In multi-label learning problems, each example is associated with a binary *label vector*. Often, there may exist correlations among some of the labels. To leverage the label correlations and also handle the possibly massive number of labels, a common approach is to reduce the dimensionality of the label space, e.g., by projecting the label vectors to a lower dimensional space (Kapoor et al., 2012; Tai & Lin, 2012; Yu et al., 2014), learning in that space, and then projecting back to the original label space. However, as the label space dimensionality increases and/or the sparsity in the label matrix becomes more and more pronounced (i.e., massive number of zeros and very few ones), and/or if the label matrix is only partially observed, such methods tend to suffer (Yu et al., 2014) and can also become computationally prohibitive. A somewhat more subtle problem is the question of how to treat the zeros in the label matrix because a zero in the label matrix could mean a negative label or an *unknown* label (due to a missing annotation).

Motivated by these issues, we present a Bayesian framework for multi-label learning. Our framework is in the same spirit as methods that try to reduce the label space dimensionality (Kapoor et al., 2012; Tai & Lin, 2012; Yu et al., 2014). However, in contrast to these methods, our framework depends *only on the positive labels* in the label matrix. This property is appealing due to two reasons: (1) computational cost of the proposed model scales in the number of ones in the label matrix; and (2) the proposed model does not suffer from the label imbalance problem (massive number of zeros and very few ones) or the issue of whether to treat a zero as a true zero or an unknown.

Our Bayesian framework is based on a generative model for the binary label matrix, using a novel data augmentation scheme that allows us to accomplish the aforementioned desiderata. In addition to the modeling flexibility that leads to a robust and scalable model, our framework enjoy full local conjugacy which allows us to develop simple Gibbs sampling, as well as an Expectation Maximization (EM) algorithm for the proposed model, both of which are simple to implement in practice (and amenable for parallelization).

Under review at ICML 2015 - Extreme Classification: Learning with a Very Large Number of Labels Workshop, Lille, France, 2015.

2. The Model

We assume that the training data are given in form of N objects represented by a feature matrix $\mathbf{X} \in \mathbb{R}^{D\times N}$, along with their labels in a (possibly *incomplete*) label matrix $\mathbf{Y} \in \{0,1\}^{L\times N}$. The goal is to learn a model that can predict labels $\boldsymbol{y}_* \in \{0,1\}^L$ for a test object $\boldsymbol{x}_* \in \mathbb{R}^D$.

We model the label vector for the n^{th} object as

$$\boldsymbol{y}_n = \mathbb{1}(\boldsymbol{m}_n \geq 1) \tag{1}$$

which for each individual label $y_{ln} \in \boldsymbol{y}_n$, $l = 1,\ldots,L$, can also be written as $y_{ln} = \mathbb{1}(m_{ln} \geq 1)$. In Eq. (1), $\boldsymbol{m}_n = [m_{1n},\ldots,m_{Ln}] \in \mathbb{Z}^L$ denotes a *latent* count vector of size L and is assumed drawn from a Poisson

$$\boldsymbol{m}_n \sim \text{Poisson}(\boldsymbol{\lambda}_n) \tag{2}$$

with the rate parameters $\boldsymbol{\lambda}_n \in \mathbb{R}_+^L$ in Eq. (2)defined as

$$\boldsymbol{\lambda}_n = \mathbf{V}\boldsymbol{u}_n \tag{3}$$

Here $\mathbf{V} \in \mathbb{R}_+^{L\times K}$, $\boldsymbol{u}_n \in \mathbb{R}_+^K$ (typically $K \ll L$). Note that Eq. (1)-(3) can also be combined and written succinctly as

$$\boldsymbol{y}_n = f(\boldsymbol{\lambda}_n) = f(\mathbf{V}\boldsymbol{u}_n) \tag{4}$$

where the function f jointly denotes drawing the latent counts $\boldsymbol{m}_n$ from a Poisson (Eq. 2) with rate $\boldsymbol{\lambda}_n = \mathbf{V}\boldsymbol{u}_n$, followed by thresholding $\boldsymbol{m}_n$ at 1 (Eq. 1).

Note that, in Eq. (4), expressing the label vector $\boldsymbol{y}_n \in \{0,1\}^L$ in terms of $\mathbf{V}\boldsymbol{u}_n$ is equivalent to a low-rank assumption on the $L \times N$ label matrix $\mathbf{Y} = [\boldsymbol{y}_1 \ldots \boldsymbol{y}_N]$

$$\mathbf{Y} = f(\mathbf{V}\mathbf{U}) \tag{5}$$

where $\mathbf{V} = [\boldsymbol{v}_1 \ldots \boldsymbol{v}_K] \in \mathbb{R}_+^{L\times K}$ and $\mathbf{U} = [\boldsymbol{u}_1 \ldots \boldsymbol{u}_N] \in \mathbb{R}_+^{K\times N}$, which are modeled as follows

$$\begin{aligned}
\boldsymbol{v}_k &\sim \text{Dirichlet}(\eta \mathbf{1}_L) && (6)\\
u_{kn} &\sim \text{Gamma}(r_k, p_{kn}(1-p_{kn})^{-1}) && (7)\\
p_{kn} &= \sigma(\boldsymbol{w}_k^\top \boldsymbol{x}_n) && (8)\\
\boldsymbol{w}_k &\sim \mathcal{N}or(0, \text{diag}(\tau_1^{-1},\ldots,\tau_D^{-1})) && (9)
\end{aligned}$$

where $\sigma(z) = 1/(1+\exp(-z))$ and the hyperparameters $r_k, \tau_1, \ldots, \tau_D$ are given improper gamma priors.

Intuitively, we can think of the model as follows: the label vector $\boldsymbol{y}_n \in \{0,1\}^L$ is assumed to have a latent representation $\boldsymbol{u}_n \in \mathbb{R}_+^K$ which is conditioned on the *observed* feature vector $\boldsymbol{x}_n$ via the weights $\mathbf{W} = \{\boldsymbol{w}_k\}_{k=1}^K$. The representation $\boldsymbol{u}_n$ (a function of $\sigma(\mathbf{W}\boldsymbol{x}_n)$) is then mapped to the label vector $\boldsymbol{y}_n$ via $\mathbf{V}$ and the link function f (Eq. 4). In the end, we get the following generative model for the label vector

$$\boldsymbol{y}_n = f(\mathbf{V}g(\sigma(\mathbf{W}\boldsymbol{x}_n))) \tag{10}$$

Note that Eq. (10) can also be seen as performing a *nonlinear* feature extraction to learn $\boldsymbol{u}_n$ from the observed feature vector $\boldsymbol{x}_n$ (via the sigmoid $\sigma(\mathbf{W}\boldsymbol{x}_n)$), and then using these extracted features to *generate* the label vector $\boldsymbol{y}_n$ via $\mathbf{V}$.

2.1. Dependence Only on Positive Labels

Using (1)-(3), we can write the conditional posterior of the latent count vector $\boldsymbol{m}_n$, given $\boldsymbol{y}_n$, as

$$\boldsymbol{m}_n|\boldsymbol{y}_n \sim [\boldsymbol{y}_n \odot \text{Poisson}_+(\mathbf{V}\boldsymbol{u}_n)] \tag{11}$$

where Poisson_+ denotes the zero-truncated Poisson distribution with support only on the positive integers, and $\odot$ denotes the element-wise product.

Eq. 11 shows the dependence of the model only on the positive labels, because the zeros in $\boldsymbol{y}_n$ will result in the corresponding elements of the latent count vector $\boldsymbol{m}_n$ being zero, almost surely (i.e., with probability one). As shown in Section 3 on inference, the sufficient statistics of the parameters in the model do not depend on latent counts that are equal to zero; such latent counts can be simply ignored during the inference. This aspect leads to substantial computational savings in our model, making it scale only in the number of positive labels in the label matrix. In the rest of the exposition, we will refer to our model as **BMLPL** to denote **B**ayesian **M**ulti-label **L**earning via **P**ositive **L**abels.

2.2. Asymmetric Link Function

Another key observation to note here is that, after integrating out $\boldsymbol{m}_n$ from Eq. (1), we get the following

$$p(\boldsymbol{y}_n = 1|\boldsymbol{\lambda}_n) = 1 - \exp(-\boldsymbol{\lambda}_n) \tag{12}$$

Assuming the non-negative rate vector $\boldsymbol{\lambda}_n = \exp(\boldsymbol{s}_n)$ for some real-valued vector $\boldsymbol{s}_n$, the above can be written as $p(\boldsymbol{y}_n = 1) = 1 - \exp(-\exp(\boldsymbol{s}_n))$, which is the inverse of the complementary log-log link function (Piegorsch, 1992; Collett, 2002). In contrast to the logistic models with inverse link function $p(\boldsymbol{y}_n = 1) = 1/(1+\exp(-\boldsymbol{s}_n))$ which is symmetric around zero, the inverse link function in Eq. (12) is *asymmetric* (see Fig 1 where, along the y-axis, the red curve approaches zero slowly and one faster, as compared to the blue curve), which makes it a better choice for modeling binary-valued data where ones are rare.

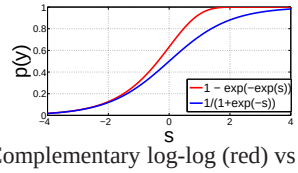

Figure 1. Complementary log-log (red) vs logistic (blue)

Therefore, in contrast to models based on logistic/probit likelihood function or standard loss functions such as the hinge-loss (Yu et al., 2014; Li et al., 2015) for the binary labels, our proposed model provides better robustness against label imbalance, as well as its dependence only on the ones in the label matrix leads to significant computational speedups (Sec 2.1) during inference. Moreover, since our model does not depend on the zeros in the label matrix, it does not have to make any assumptions as to whether the zeros are *true* zeros or not, as it only depends on the ones.

3. Inference

Inference in our model involves estimating $\mathbf{V} \in \mathbb{R}_+^{L\times K}$, $\mathbf{W} \in \mathbb{R}^{D\times K}$, $\mathbf{U} \in \mathbb{R}_+^{K\times N}$, and the hyperparameters. As we will see below, the latent count vectors $\{\boldsymbol{m}_n\}_{n=1}^N$ (which are functions of $\mathbf{V}$ and $\mathbf{U}$) provide sufficient statistics for the model parameters. Each element of $\boldsymbol{m}_n$ (*if* the corresponding element in $\boldsymbol{y}_n$ is one) is drawn from a truncated Poisson distribution

$$m_{ln} \sim \text{Poisson}_+(\mathbf{V}_{l,:}\boldsymbol{u}_n) = \text{Poisson}_+(\lambda_{ln}) \tag{13}$$

$\mathbf{V}_{l,:}$ denotes the l^{th} row of $\mathbf{V}$ and $\lambda_{ln} = \sum_{k=1}^K \lambda_{kln} = \sum_{k=1}^K v_{lk}u_{kn}$. Thus we can also write $m_{ln} = \sum_{k=1}^K m_{lkn}$ where $m_{lkn} \sim \text{Poisson}_+(\lambda_{kln}) = \text{Poisson}_+(v_{lk}u_{kn})$.

On the other hand, if $y_{ln} = 0$ then $m_{ln} = 0$ with probability one (Eq. (11)), and therefore need not be sampled because it does not affect the sufficient statistics needed for updating the model parameters.

Using the equivalence of Poisson and multinomial distribution (Zhou et al., 2012), we can express the decomposition $m_{ln} = \sum_{k=1}^K m_{lkn}$ as a draw from a multinomial

$$[m_{l1n}, \ldots, m_{lKn}] \sim \mathcal{M}ult(m_{ln}; \zeta_{l1n}, \ldots, \zeta_{lKn}) \tag{14}$$

where $\zeta_{lkn} = \frac{v_{lk}u_{kn}}{\sum_{k=1}^K v_{lk}u_{kn}}$.

We develop both a Gibbs sampler as well as an Expectation Maximization algorithm for inference in our model.

3.1. Gibbs Sampling

Gibbs sampling for the proposed model proceeds as follows

Sampling V: Using the Dirichlet-multinomial conjugacy, each column of $\mathbf{V} \in \mathbb{R}_+^{L\times K}$ can be sampled as

$$\boldsymbol{v}_k \sim \text{Dirichlet}(\eta + m_{1k}, \ldots, \eta + m_{Lk}) \tag{15}$$

where $m_{lk} = \sum_n m_{lnk}, \forall l = 1, \ldots, L$.

Sampling U: Using the gamma-Poisson conjugacy, each entry of $\mathbf{U} \in \mathbb{R}_+^{K\times N}$ can be sampled as

$$u_{kn} \sim \text{Gamma}(r_k + m_{kn}, p_{kn}) \tag{16}$$

where $m_{kn} = \sum_l m_{lnk}$ and $p_{kn} = \sigma(\boldsymbol{w}_k^\top \boldsymbol{x}_n)$.

Sampling W: Since $m_{kn} = \sum_l m_{lnk}$ and $m_{lnk} \sim \text{Poisson}_+(v_{lk}u_{kn})$, $p(m_{kn}|u_{kn})$ is also Poisson. Further, since $p(u_{kn}|r, p_{kn})$ is gamma, we can integrate out u_{kn} and get $m_{kn} = \text{NegBin}(r_k, p_{kn})$, where NegBin(.,.) denotes the negative Binomial distribution. This, combined with the fact that the prior on $\boldsymbol{w}_k$ is Gaussian, allows us to use the Pólya-Gamma strategy (Polson et al., 2013) to derive close form Gibbs sampling updates for $\mathbf{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_K]$. The Pólya-Gamma strategy is based on sampling a set of auxiliary variables (in our model, equal to the number of ones in the label matrix). Given the Pólya-Gamma variables, the conditional distribution of $\mathbf{W}$ becomes Gaussian (Polson et al., 2013), which can easily be sampled from. We skip the details here for brevity.

Sampling the hyperparameters: The other hyperparameters in our model, r_k and $\tau_1, \ldots, \tau_D$, also have gamma posteriors and can be sampled easily.

3.2. Expectation Maximization

Conjugacy of our model also allows us to develop an efficient Expectation Maximization (EM) inference algorithm. The E-step involves computing the expectations of the local variables $\mathbf{U}$ and the Pólya-Gamma variables (Scott & Sun, 2013). All these expectations are available in closed form and are therefore easy to compute. The M-step involves a maximization w.r.t. $\mathbf{V}$ and $\mathbf{W}$, which essentially involves solving for their *maximum-a-posteriori* (MAP) estimates, which are available in closed form. We skip the details here for brevity.

3.3. Predicting Labels for Test Examples

Predicting the label vector $\boldsymbol{y}_* \in \{0, 1\}^L$ for a new test example $\boldsymbol{x}_* \in \mathbb{R}^D$ can be done as follows

$$p(\boldsymbol{y}_* = 1|\boldsymbol{x}_*) = \int_{\boldsymbol{u}_*} (1 - \exp(-\mathbf{V}\boldsymbol{u}_*))p(\boldsymbol{u}_*)d\boldsymbol{u}_*$$

If using Gibbs sampling, the integral above can be approximated using samples $\{\boldsymbol{u}_*^{(m)}\}_{m=1}^M$ from the posterior of $\boldsymbol{u}_*$. It is actually also possible to integrate out $\boldsymbol{u}_*$ (details skipped due to brevity) and get closed form estimates of probability of each label y_{l*} in terms of the model parameters $\mathbf{V}$ and $\mathbf{W}$, and it is given by

$$p(y_{l*} = 1|\boldsymbol{x}_*) = 1 - \prod_{k=1}^K \frac{1}{[V_{lk}\exp(\boldsymbol{w}_k^\top \boldsymbol{x}_*) + 1]^{r_k}} \tag{17}$$

4. Computational Complexity

Computing the latent count m_{ln} for each nonzero entry y_{ln} in $\mathbf{Y}$ requires computing $[m_{l1n}, \ldots, m_{lKn}]$, which takes $O(K)$ time; therefore computing all the latent counts takes $O(\text{nnz}(\mathbf{Y})K)$ time, which is very efficient if $\mathbf{Y}$ has very few nonzeros (which is true of most real-world multi-label learning problems). Estimating $\mathbf{V}$, $\mathbf{U}$, and the hyperparameters is relatively cheap and can be done very efficiently. The other dominant cost comes from sampling the Pólya-Gamma variables which takes $O(\text{nnz}(\mathbf{Y}))$ time when doing Gibbs sampling but is relatively cheaper when doing EM because the Pólya-Gamma expectations are very cheaper to compute (Scott & Sun, 2013). The most dominant step is estimating $\mathbf{W}$ - when estimated *naïvely*, it would $O(DK^3)$ time if done row-wise, and $O(KD^3)$ time if done column-wise. However, if using the EM algorithm, this can be done much more efficiently; in fact, it it not even required to solved for $\mathbf{W}$ *exactly* in each iteration of the EM algorithm (Scott & Sun, 2013). Also

note that since most of the parameters updates for different $k = 1, \ldots, K, n = 1, \ldots, N$ are all independent of each other, our Gibbs sampler and the EM algorithms can be easily parallelized/block-updated.

5. Connection: Topic Models with Meta-Data

The proposed generative model, interestingly, also bears resemblance to topic models (Blei et al., 2003; Zhou et al., 2012) if each document $\boldsymbol{y}_n \in \{0,1\}^L$ (in a bag-of-words representation with vocabulary of length L) also has meta-data $\boldsymbol{x}_n \in \mathbb{R}^D$ associated with it. Note that, in our model, each Dirichlet-drawn column $\boldsymbol{v}_k$ of the matrix $\mathbf{V} \in \mathbb{R}_+^{L \times K}$ can be seen as representing a "topic". Our model can therefore also be used to perform topic modeling of text documents with associated meta-data (Mimno & McCallum, 2008; Kim & Sudderth, 2011; Zhu et al., 2011; Rabinovich & Blei, 2014) in a more robust and scalable manner.

In the more general multi-label learning setting considered here, $\boldsymbol{v}_k$ represents a distribution or "topic" over a *label vocabulary*, and each example n with label vector $\boldsymbol{y}_n \in \{0,1\}^L$ and feature vector $\boldsymbol{x}_n \in \mathbf{R}^D$ can be thought of as a combination of the K topics $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_K$, with the combination weights given by $\boldsymbol{u}_n \in \mathbb{R}_+^K$, which in turn depend on the features $\boldsymbol{x}_n$.

6. Experiments

We evaluate the proposed model on two benchmark multi-label prediction data sets - bibtex and delicious (Yu et al., 2014), with their statistics summarized in Table 1.

			Training set		Test set	
Data set	D	L	N_{train}	$\bar{L}$	N_{test}	$\bar{L}$
bibtex	1836	159	4880	2.40	2515	2.40
delicious	500	983	12920	19.03	3185	19.00

Table 1. Statistics of the data sets. $\bar{L}$ denotes average number of positive labels per example.

We compare the proposed model **B**ayesian **M**ulti-label **L**earning via **P**ositive **L**abels (abbreviated **BMLPL**) with four state-of-the-art methods. All these methods, just like our method, are based on the assumption that the label vectors live in a low dimensional space.

- CPLST: Conditional Principal Label Space Transformation (Tai & Lin, 2012).
- BCS: Bayesian Compressed Sensing for multi-label learning (Kapoor et al., 2012).
- WSABIE: This model assumes that the feature as well as the label vectors live in a low dimensional space. The model is based on optimizing a weighted approximate ranking loss (Weston et al., 2011).
- LEML: Low rank Empirical risk minimization for multi-label learning (Yu et al., 2014). For LEML, we report the best results across the three loss functions (squared, logistic, hinge) they propose.

Table 2 shows the results where we report the Area Under the ROC Curve (AUC) for each method on both the data sets. For each method, as done in (Yu et al., 2014), we vary the label space dimensionality from 20% - 100% of L, and report the best results. For BMLPL, both Gibbs sampling and EM based inference perform comparably (though EM runs much faster than Gibbs); here we report results obtained with EM inference only.

As shown in the results in Table 2, on both data sets, BMLPL performs better than the other methods. The better performance of our model justifies the flexible Bayesian formulation and also shows the evidence of the robustness provided by the asymmetric link function against sparsity and label imbalance in the label matrix (note that the data sets we use have very sparse label matrices).

	CPLST	BCS	WSABIE	LEML	BMLPL
bibtex	0.8882	0.8614	0.9182	0.9040	**0.9210**
delicious	0.8834	0.8000	0.8561	0.8894	**0.8950**

Table 2. Comparison of the various methods in terms of AUC scores on bibtex and delicious data sets.

6.1. Results with Missing Labels

Our generative model for the label matrix can also handle missing labels in the label matrix (the missing labels may include both zeros or ones). We perform an experiment on the bibtex data where only 20% of the labels from the label matrix are revealed, and compare our model with LEML and BCS (both are capable of handling missing labels). The results are shown in Table 3. As the results show, our model yields better results as compared to the competing methods even in the presence of missing labels.

	BCS	LEML	BMLPL
bibtex	0.7871	0.8332	**0.8420**

Table 3. AUC scores with only 20% labels observed.

7. Discussion and Conclusion

We have presented a scalable Bayesian framework for multi-label learning. In addition to providing a flexible model for sparse label matrices, our framework is also computationally attractive and can scale to massive data sets. The model is easy to implement and easy to parallelize. Both full Bayesian inference via simple Gibbs sampling and EM based inference can be carried out in this model in a computationally efficient way. Possible future work includes developing online Gibbs and online EM algorithms to further enhance the scalability of the proposed framework to handle even bigger data sets. Another possible extension could be to additionally impose label correlations more explicitly (in addition to the low-rank structure already imposed by the current model), e.g., by replacing the Dirichlet distribution on the columns of $\mathbf{V}$ with logistic normal distributions (Chen et al., 2013).

References

Agrawal, Rahul, Gupta, Archit, Prabhu, Yashoteja, and Varma, Manik. Multi-label learning with millions of labels: Recommending advertiser bid phrases for web pages. In *WWW*, 2013.

Blei, David M, Ng, Andrew Y, and Jordan, Michael I. Latent dirichlet allocation. *JMLR*, 2003.

Chen, Jianfei, Zhu, Jun, Wang, Zi, Zheng, Xun, and Zhang, Bo. Scalable inference for logistic-normal topic models. In *NIPS*, 2013.

Collett, David. *Modelling binary data*. CRC press, 2002.

Gibaja, Eva and Ventura, Sebastián. Multilabel learning: A review of the state of the art and ongoing research. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2014.

Gibaja, Eva and Ventura, Sebastián. A tutorial on multilabel learning. *ACM Comput. Surv.*, 2015.

Kapoor, Ashish, Viswanathan, Raajay, and Jain, Prateek. Multilabel classification using bayesian compressed sensing. In *NIPS*, 2012.

Kim, Dae I and Sudderth, Erik B. The doubly correlated nonparametric topic model. In *NIPS*, 2011.

Kong, Xiangnan, Wu, Zhaoming, Li, Li-Jia, Zhang, Ruofei, Yu, Philip S, Wu, Hang, and Fan, Wei. Large-scale multi-label learning with incomplete label assignments. In *SDM*, 2014.

Li, Xin, Zhao, Feipeng, and Guo, Yuhong. Conditional restricted boltzmann machines for multi-label learning with incomplete labels. In *AISTATS*, 2015.

Mimno, David and McCallum, Andrew. Topic models conditioned on arbitrary features with dirichlet-multinomial regression. In *UAI*, 2008.

Piegorsch, Walter W. Complementary log regression for generalized linear models. *The American Statistician*, 1992.

Polson, Nicholas G, Scott, James G, and Windle, Jesse. Bayesian inference for logistic models using pólya–gamma latent variables. *Journal of the American Statistical Association*, 108(504):1339–1349, 2013.

Prabhu, Yashoteja and Varma, Manik. FastXML: a fast, accurate and stable tree-classifier for extreme multi-label learning. In *KDD*, 2014.

Rabinovich, Maxim and Blei, David. The inverse regression topic model. In *ICML*, 2014.

Scott, James G and Sun, Liang. Expectation-maximization for logistic regression. *arXiv preprint arXiv:1306.0040*, 2013.

Tai, Farbound and Lin, Hsuan-Tien. Multilabel classification with principal label space transformation. *Neural Computation*, 2012.

Weston, Jason, Bengio, Samy, and Usunier, Nicolas. WSABIE: Scaling up to large vocabulary image annotation. In *IJCAI*, 2011.

Yan, Yan, Fung, Glenn, Dy, Jennifer G, and Rosales, Romer. Medical coding classification by leveraging inter-code relationships. In *KDD*, 2010.

Yu, Hsiang-Fu, Jain, Prateek, Kar, Purushottam, and Dhillon, Inderjit S. Large-scale multi-label learning with missing labels. In *ICML*, 2014.

Zhou, M., Hannah, L. A., Dunson, D., and Carin, L. Beta-negative binomial process and poisson factor analysis. In *AISTATS*, 2012.

Zhu, Jun, Lao, Ni, Chen, Ning, and Xing, Eric P. Conditional topical coding: an efficient topic model conditioned on rich features. In *KDD*, 2011.