Solving $\mathbf{W}$ and $\mathbf{b}$ with Others Fixed The optimization subproblem is

$$\min_{\mathbf{W},\mathbf{b}} \ \mathrm{tr}(\mathbf{Z}-\mathbf{W}^T\mathbf{X}-\mathbf{b}\mathbf{1}^T)^T\mathbf{\Omega}^{-1}(\mathbf{Z}-\mathbf{W}^T\mathbf{X}-\mathbf{b}\mathbf{1}^T) + \sum_{j=1}^{d}\mathbf{W}_{(j,:)}\mathbf{\Sigma}_j^{-1}\mathbf{W}_{(j,:)}^T. \quad (12)$$

On setting the derivative of $\mathbf{W}$ to $\mathbf{0}$, we obtain its closed-form solution as

$$\mathrm{vec}(\mathbf{W}) = \left(\mathbf{\Omega}^{-1}\otimes(\mathbf{X}\mathbf{X}^T)+\sum_{j=1}^{d}\mathbf{\Sigma}_j^{-1}\otimes\mathbf{E}_j\right)^{-1}\mathrm{vec}(\mathbf{C}), \quad (13)$$

where $\mathbf{C}=\mathbf{X}(\mathbf{Z}-\mathbf{b}\mathbf{1})^T\mathbf{\Omega}^{-1}$. For $\mathbf{b}$, on setting its derivative to $\mathbf{0}$, we obtain

$$\mathbf{b}=\frac{1}{N}(\mathbf{Z}-\mathbf{W}^T\mathbf{X})\mathbf{1}. \quad (14)$$

Solving $\mathbf{\Omega}^{-1}$ with Others Fixed With the prior in (6), the optimization subproblem is

$$\min_{\mathbf{\Omega}^{-1}} \ \mathrm{tr}(\mathbf{Z}-\mathbf{W}^T\mathbf{X}-\mathbf{b}\mathbf{1}^T)^T\mathbf{\Omega}^{-1}(\mathbf{Z}-\mathbf{W}^T\mathbf{X}-\mathbf{b}\mathbf{1}^T) - N\log|\mathbf{\Omega}^{-1}| + \lambda_1\mathrm{tr}(\mathbf{\Omega}^{-1}) + \lambda_2\|\mathbf{\Omega}^{-1}\|_1.$$

$\mathbf{\Omega}^{-1}$ can be solved using standard sparse inverse covariance estimation algorithms, such as the graphical Lasso in (Friedman, Hastie, and Tibshirani 2008).

Solving $\mathbf{\Sigma}_j^{-1}$'s with Others Fixed With the prior in (9), the optimization subproblem for each $\mathbf{\Sigma}_j$ is

$$\min_{\mathbf{\Sigma}_j^{-1}} \ \mathbf{W}_{(j,:)}\mathbf{\Sigma}_j^{-1}\mathbf{W}_{(j,:)}^T - \log|\mathbf{\Sigma}_j^{-1}| + \beta_1\mathrm{tr}(\mathbf{\Sigma}_j^{-1}) + \beta_2\|\mathbf{\Sigma}_j^{-1}\|_1. \quad (15)$$

Again, $\mathbf{\Sigma}_j^{-1}$ can be obtained by sparse inverse covariance estimation.

Handling Missing Labels

As discussed in the introduction, the label vectors may have missing entries. Assume that sample $\mathbf{x}^{(i)}$ has l_i observed labels and $u_i = m - l_i$ missing labels. We reorder $\mathbf{y}^{(i)}$ (and, similarly, $\mathbf{z}^{(i)}$) as $[(\mathbf{y}_l^{(i)})^T, (\mathbf{y}_u^{(i)})^T]^T$ (where $\mathbf{y}_l^{(i)}\in\mathbb{R}^{l_i}$, and $\mathbf{y}_u^{(i)}\in\mathbb{R}^{u_i}$). Similarly, for each i, we reorder $\mathbf{\Omega}^{-1}$ by putting the l_i rows/columns corresponding to the observed labels first as $\begin{bmatrix}\mathbf{U}_i & \mathbf{V}_i\\ \mathbf{V}_i^T & \mathbf{Q}_i\end{bmatrix}$, where $\mathbf{U}_i\in\mathbb{R}^{l_i\times l_i}$, $\mathbf{V}_i\in\mathbb{R}^{l_i\times u_i}$ and $\mathbf{Q}_i\in\mathbb{R}^{u_i\times u_i}$.

Instead of estimating the values for the missing labels as in (Bucak, Jin, and Jain 2011; Chen, Zheng, and Weinberger 2013), we directly derive the posterior w.r.t. the observed labels. Analogous to (11), we have

$$p(\{\mathbf{z}_{l_i}^{(i)}\}_{i=1}^N,\mathbf{W},\mathbf{\Omega},\{\mathbf{\Sigma}_j\}_{j=1}^d|\mathbf{X},\{\mathbf{y}_l^{(i)}\}_{i=1}^N,\mathbf{b}) \propto p(\mathbf{\Omega})\prod_{j=1}^{d}p(\mathbf{W}_{(j,:)}|\mathbf{\Sigma}_j)p(\mathbf{\Sigma}_j) \cdot\prod_{i=1}^{N}p(\mathbf{y}_l^{(i)}|\mathbf{z}_l^{(i)})p(\mathbf{z}_l^{(i)}|\mathbf{x}^{(i)},\mathbf{W},\mathbf{\Omega},\mathbf{b}). \quad (16)$$

Note that $p(\mathbf{y}_l^{(i)}|\mathbf{z}_l^{(i)}) = \prod_{j\in l_i}p(y_j^{(i)}|z_j^{(i)})$ and so can be easily obtained as in $p(\mathbf{y}^{(i)}|\mathbf{z}^{(i)})$. However, this is not the case for $p(\mathbf{z}_l^{(i)}|\mathbf{x}^{(i)},\mathbf{W},\mathbf{\Omega},\mathbf{b})$. Instead, we marginalize the missing elements from $p(\mathbf{z}^{(i)}|\mathbf{W},\mathbf{x}^{(i)},\mathbf{\Omega},\mathbf{b})$, as

$$p(\mathbf{z}_l^{(i)}|\mathbf{W},\mathbf{x}^{(i)},\mathbf{\Omega},\mathbf{b}) = \int p([(\mathbf{z}_l^{(i)})^T,(\mathbf{z}_u^{(i)})^T]^T|\mathbf{W},\mathbf{b},\mathbf{x}^{(i)},\mathbf{\Omega})d\mathbf{z}_u^{(i)}. \quad (17)$$

From (Bishop 2006), this is still normally distributed, as

$$\mathbf{z}_l^{(i)}|\mathbf{W},\mathbf{x}^{(i)},\mathbf{\Omega}\sim\mathcal{N}(\mathbf{W}_{(:,l_i)}^T\mathbf{x}^{(i)}+\mathbf{b}_{l_i},\tilde{\mathbf{U}}_i), \quad (18)$$

where $\tilde{\mathbf{U}}_i=\mathbf{U}_i-\mathbf{V}_i\mathbf{Q}_i^{-1}\mathbf{V}_i^T$, and $\mathbf{W}_{(:,l_i)}$ is the submatrix of $\mathbf{W}$ with columns corresponding to the l_i observed labels. Note that each $\mathbf{z}_l^{(i)}$ is dependent on the whole $\mathbf{\Omega}$ matrix (via $\tilde{\mathbf{U}}_i$). Thus, even in the presence of missing labels, the inference procedure can still utilize label correlation information.

As in the previous section, we will use alternating maximization to maximize the posterior in (16). Note that the optimization subproblems for $\mathbf{\Sigma}_j^{-1}$'s are the same as before, and thus the updates remain unchanged.

Solving $\{\mathbf{z}_{l_i}^{(i)}\}_{i=1}^N$ with Others Fixed The optimization subproblem is

$$\min_{\mathbf{z}_l^{(i)}} \ \sum_{i=1}^{N}\|\mathbf{z}_{l_i}^{(i)}-\mathbf{y}_{l_i}^{(i)}\|^2 + (\mathbf{z}_l^{(i)}-\mathbf{W}_{(:,l_i)}^T\mathbf{x}^{(i)}-\mathbf{b}_{l_i})^T\tilde{\mathbf{U}}_i \cdot(\mathbf{z}_l^{(i)}-\mathbf{W}_{(:,l_i)}^T\mathbf{x}^{(i)}-\mathbf{b}_{l_i}).$$

Setting the derivative w.r.t. each $\mathbf{z}_l^{(i)}$ to $\mathbf{0}$, we have

$$\mathbf{z}_l^{(i)}=\mathbf{W}_{(:,l_i)}^T\mathbf{x}^{(i)}+\mathbf{b}_{l_i}-\tilde{\mathbf{U}}_i^{-1}\mathbf{y}_l^{(i)}.$$

Solving $\mathbf{W}$ and $\mathbf{b}$ with Others Fixed The optimization subproblem is

$$\min_{\mathbf{W},\mathbf{b}}\sum_{i=1}^{N}(\mathbf{z}_l^{(i)}-\mathbf{W}_{(:,l_i)}^T\mathbf{x}^{(i)}-\mathbf{b}_{l_i})^T\tilde{\mathbf{U}}_i(\mathbf{z}_l^{(i)}-\mathbf{W}_{(:,l_i)}^T\mathbf{x}^{(i)}-\mathbf{b}_{l_i}) + \sum_{j=1}^{d}\mathbf{W}_{(j,:)}\mathbf{\Sigma}_j^{-1}\mathbf{W}_{(j,:)}^T. \quad (19)$$

Unlike (12), a closed-form solution cannot be obtained for this convex problem. Thus, we optimize $\mathbf{W}$ by gradient descent. As for $\mathbf{b}$, we have the closed-form solution

$$\mathbf{b}=\left(\sum_{i=1}^{N}\Xi_i(\tilde{\mathbf{U}}_i)\right)^{-1}\sum_{i=1}^{N}\Xi_i\left(\tilde{\mathbf{U}}_i(\mathbf{z}_l^{(i)}-\mathbf{W}_{(:,l_i)}^T\mathbf{x}^{(i)})\right),$$

where Ξ_i is an operator that "expands" a matrix $\mathbf{A}\in\mathbb{R}^{l_i\times l_i}$ to $\begin{bmatrix}\mathbf{A} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\in\mathbb{R}^{m\times m}$.

Solving $\mathbf{\Omega}^{-1}$ with Others Fixed The optimization subproblem is

$$\min_{\mathbf{\Omega}^{-1}}\sum_{i=1}^{N}\Big((\mathbf{y}_l^{(i)}-\mathbf{W}_{(:,l_i)}^T\mathbf{x}^{(i)}-\mathbf{b}_{l_i})^T\tilde{\mathbf{U}}_i(\mathbf{y}_l^{(i)}-\mathbf{W}_{(:,l_i)}^T\mathbf{x}^{(i)}-\mathbf{b}_{l_i}) - \ln|\tilde{\mathbf{U}}_i|\Big) + \lambda_1\mathrm{tr}(\mathbf{\Omega}^{-1}) + \lambda_2\|\mathbf{\Omega}^{-1}\|_1.$$

This can be solved by iterative soft thresholding (Beck and Teboulle 2009). Specifically, we decompose the objective into two parts, as:

$$f(\mathbf{\Omega}) = \Big(\mathbf{y}_l^{(i)} - \mathbf{W}_{(:,l_i)}^T \mathbf{x}^{(i)} - \mathbf{b}_{l_i}\Big)^T \tilde{\mathbf{U}}_i \Big(\mathbf{y}_l^{(i)} - \mathbf{W}_{(:,l_i)}^T \mathbf{x}^{(i)} - \mathbf{b}_{l_i}\Big) - \ln|\tilde{\mathbf{U}}_i|\Big) + \lambda_1 \mathrm{tr}(\mathbf{\Omega}^{-1}),$$

$$g(\mathbf{\Omega}) = \lambda_2 \|\mathbf{\Omega}^{-1}\|_1.$$

Since $\mathbf{\Omega}$ is positive semidefinite (psd), instead of performing projected gradient descent (which requires the potentially expensive projection onto the psd cone in every iteration), we update $\mathbf{\Omega}^{-1}$ based on its factorization. In each iteration,

1. We factorize $\mathbf{\Omega}^{-1}$ as $\mathbf{G}\mathbf{G}^T$, and perform a one-step gradient descent of $f(\mathbf{\Omega})$ on $\mathbf{G}$;
2. recompute $\mathbf{\Omega}^{-1}$ from the updated $\mathbf{G}$;
3. sparsify $\mathbf{\Omega}^{-1}$ by shrinking each of its elements as $(|(\mathbf{\Omega}^{-1})_{ij}| - \tau)_+ \mathrm{sign}((\mathbf{\Omega}^{-1})_{ij})$, where $\tau = \lambda_2\eta$, η is the stepsize for gradient descent, and $(a)_+ = \max\{a, 0\}$.

Experiments

Setup

In this section, experiments are performed on five image annotation data sets[3] (Table 1) used in (Guillaumin et al. 2009). For each image, 1000 SIFT features are extracted.

Table 1: Data sets used.

data set	#labels	#samples	avg #positive labels per sample	max #negative labels per sample
pascal07	20	9,963	1.5	6
mirflickr	38	25,000	4.7	17
corel5k	260	4,999	3.4	5
espgame	268	23,641	4.7	15
iaprtc12	291	19,627	5.7	23

The proposed method, called "multilabel classification with label correlations and missing labels" (LCML), is compared with the following methods:

1. Multiple-output regression with output and task structures (MROTS) (Rai, Kumar, and Iii 2012): It leverages both the task structure on $\mathbf{W}$ and output structure on $\mathbf{Y}$. However, MROTS assumes the $\mathbf{\Sigma}_i$'s for all features are the same. Moreover, it is for regression problems and does not consider rounding its output to a binary prediction.
2. Conditional principal label space transformation (CPLST) (Chen and Lin 2012);
3. Max-margin multilabel classifier (M3L)[4] (Hariharan et al. 2010);
4. Multilabel ranking with group lasso (MLRGL)[5] (Bucak, Jin, and Jain 2011);
5. Fast image tagging (FastTag)[6] (Chen, Zheng, and Weinberger 2013);
6. Classifier chain (CC) (Read et al. 2009);
7. Binary relevance (BR) (Tsoumakas and Katakis 2007): It serves as a baseline that trains each label independently.

As the transformed labels in CPLST are real-valued, we use ridge regression as its base learner. For consistency, it is also used for CC and BR. Parameter tuning for all the methods is based on a validation set obtained by randomly sampling 30% of the training data. Moreover, some of the above methods (such as MROTS, CPLST, CC and BR) rely on a threshold to decide how many labels are to be predicted for each sample. However, this threshold setting depends heavily on the application. As in (Guillaumin et al. 2009), we avoid this problem by predicting as positive the five labels with the largest prediction scores. Performance is then evaluated by

$$\text{macro-F1} = \frac{1}{m}\sum_{j=1}^{m} \frac{2\sum_{i=1}^{N} \hat{y}_j^{(i)} y_j^{(i)}}{\sum_{i=1}^{N} \hat{y}_j^{(i)} + \sum_{i=1}^{N} y_j^{(i)}},$$

$$\text{micro-F1} = \frac{1}{N}\sum_{i=1}^{N} \frac{2\sum_{j=1}^{m} \hat{y}_j^{(i)} y_j^{(i)}}{\sum_{j=1}^{m} \hat{y}_j^{(i)} + \sum_{j=1}^{m} y_j^{(i)}},$$

which are commonly used in multilabel classification (Read et al. 2009; Petterson and Caetano 2011; Tai and Lin 2012). The higher the F1 value, the better the performance.

Performance Comparison

Results based on 5-fold cross-validation are shown in Table 2. As can be seen, LCML performs significantly better than the others on all data sets. Note that M3L and CC are even outperformed by BR. For M3L, this may be due to that the provided label correlations are too crude. This has also been noted in (Hariharan et al. 2010) that different label priors can greatly affect the performance. For CC, it is also known that the chain's order is important. Read et al. (2009) recommended the use of an ensemble of CC, but this can be very expensive. Thus, the label correlations, if inaccurately specified, may hurt performance.

Experiments on Data Sets with Missing Labels

We generate the missing labels as follows. Recall that there are m labels. For each training sample, we choose half of them as observed and the rest as missing. However, as each sample typically has very few positive labels, a random label splitting is likely to result in only negative labels being observed. Thus, for each sample, we make sure that there are $k = 1, 2, 3$ positive observed labels (if the sample have fewer than k positive labels, all its positive labels are selected). We compare LCML with MLRGL and FastTag, which are also capable of handling missing labels.[7] BR is also included as a baseline. In each binary label classification task, it simply removes samples with labels.

Table 3 shows the results based on 5-fold cross-validation. As can be seen, LCML still significantly outperforms the others (except for $k = 2$ on mirflickr). Moreover, the performance does not always improve with k, as the number of missing labels is much larger than the maximum value of k. Nevertheless, even for $k = 1$, the F1 values obtained by LCML are very close to those obtained on the complete labels in Table 2. On corel5k, LCML even performs better when labels are missing. One possible reason is that with complete labels, we need to learn the $m \times n$ label matrix $\mathbf{Z}$. When labels are missing, they are integrated out in (17) and only the submatrix of $\mathbf{Z}$ corresponding to the observed labels needs to be learned. Thus, the number of free parameters is reduced,

[3] http://lear.inrialpes.fr/people/guillaumin/data.php

[4] Code is from http://www.cs.berkeley.edu/~bharath2/codes/M3L/download.html

[5] Code is from http://www.cse.msu.edu/~bucakser/software.html

[6] Code is from http://www.cse.wustl.edu/~mchen/

[7] Recall that MLRGL and FastTag assume the missing labels are negative.

Table 2: Results on data sets with complete labels. The best and comparable results (according to the pairwise t-test with 95% confidence) are highlighted.

macro-F1								
data set	LCML	MROTS	CPLST	M3L	MLRGL	FastTag	CC	BR
pascal07	**0.3591** ± 0.0055	0.3587 ± 0.0061	0.3268 ± 0.0055	0.1539 ± 0.0641	0.3486 ± 0.0075	0.2942 ± 0.0133	0.2037 ± 0.0051	0.3267 ± 0.0055
mirflickr	**0.4992** ± 0.0052	0.4928 ± 0.0043	0.4930 ± 0.0046	0.2418 ± 0.0025	0.4958 ± 0.0078	0.4681± 0.0057	0.2479 ± 0.0032	0.4930 ± 0.0046
corel5k	**0.2077** ± 0.0562	0.2046 ± 0.0571	0.2005 ± 0.0493	0.0084 ± 0.0012	0.1321 ± 0.0443	0.2038 ± 0.0378	0.0223 ± 0.0071	0.2005 ± 0.0493
espgame	**0.2380** ± 0.0130	0.2312 ± 0.0164	0.2328 ± 0.0167	0.0153 ± 0.0100	0.1254 ± 0.0076	0.2287 ± 0.0150	0.0455 ± 0.0070	0.2328 ± 0.0167
iaprtc12	**0.2463** ± 0.0409	0.2455 ± 0.0469	0.2376 ± 0.0429	0.0276 ± 0.0074	0.1273 ± 0.0407	0.2285 ± 0.0423	0.0275 ± 0.0041	0.2377 ± 0.0429
micro-F1								
data set	LCML	MROTS	CPLST	M3L	MLRGL	FastTag	CC	BR
pascal07	**0.3493** ± 0.0050	0.3489 ± 0.0054	0.3181 ± 0.0051	0.1481 ± 0.0603	0.3386 ± 0.0067	0.2813 ± 0.0141	0.2039 ± 0.0052	0.3181 ± 0.0051
mirflickr	**0.4665** ± 0.0043	0.4605 ± 0.0053	0.4608 ± 0.0055	0.2131 ± 0.0029	**0.4668** ± 0.0096	0.4365 ± 0.0068	0.2476 ± 0.0030	0.4608 ± 0.0055
corel5k	**0.2071** ± 0.0554	0.2038 ± 0.0559	0.1980 ± 0.0484	0.0081 ± 0.0012	0.1303 ± 0.0434	0.2012 ± 0.0367	0.0220 ± 0.0070	0.1980 ± 0.0484
espgame	**0.2273** ± 0.0141	0.2203 ± 0.0181	0.2219 ± 0.0183	0.0146 ± 0.0090	0.1185 ± 0.0008	0.2179 ± 0.0169	0.0448 ± 0.0070	0.2219 ± 0.0183
iaprtc12	**0.2405** ± 0.0453	0.2397 ± 0.0474	0.2304 ± 0.0433	0.0259 ± 0.0068	0.1244 ± 0.0413	0.2209 ± 0.0432	0.0280 ± 0.0036	0.2305 ± 0.0433

though (17) depends on the quality of the estimated distribution $p([(\mathbf{z}_l^{(i)})^T,(\mathbf{z}_u^{(i)})^T]^T|\mathbf{W},\mathbf{x}^{(i)},\mathbf{\Omega})$ and may introduce error. The final performance thus depends on which factor is more important.

Conclusion

In this paper, we proposed a probabilistic model for multilabel classification. While inspired by the label transformation approach, the model is expressed in the original label space instead of the transformed label space. This allows flexibility in the handling of both label dependencies and missing labels, while still maintaining a simple inference procedure. Experimental results on data sets with both complete and missing labels demonstrate that the proposed algorithm can consistently outperform the state-of-the-art.

Acknowledgment

This research was supported in part by the Research Grants Council of the Hong Kong Special Administrative Region (Grant 614012).

References

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

Bertsekas, D. P. 1999. *Nonlinear Programming*. Athena Scientific.

Bishop, C. M. 2006. *Pattern Recognition and Machine Learning*. Springer-Verlag.

Bucak, S.; Jin, R.; and Jain, A. 2011. Multi-label learning with incomplete class assignments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2801 –2808.

Chen, Y.-N., and Lin, H.-T. 2012. Feature-aware label space dimension reduction for multi-label classification. In *Advances in Neural Information Processing Systems 25*, 1538–1546.

Chen, M.; Zheng, A.; and Weinberger, K. Q. 2013. Fast image tagging. In *Proceedings of the 30th International Conference on Machine Learning*, 1274–1282.

Duygulu, P.; Barnard, K.; Freitas, N.; and Forsyth, D. 2002. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Proceedings of the 7th European Conference on Computer Vision*, 97–112.

Friedman, J.; Hastie, T.; and Tibshirani, R. 2008. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* 9(3):432–441.

Goldberg, A. B.; Zhu, X.; Recht, B.; Xu, J.-M.; and Nowak, R. 2010. Transduction with matrix completion: Three birds with one

Table 3: Results on data sets with 50% missing labels and k positive observed labels.

macro-F1 ($k = 1$)				
data set	LCML	MLRGL	FastTag	BR
pascal07	**0.3480** ± 0.0069	0.3451 ± 0.0080	0.2491 ± 0.0115	0.3050 ± 0.0054
mirflickr	**0.4670** ± 0.0019	0.4579 ± 0.0093	0.4601 ± 0.0112	0.4549 ± 0.0024
corel5k	**0.2403** ± 0.0544	0.1100 ± 0.0156	0.1998 ± 0.0107	0.1417 ± 0.0281
espgame	**0.2327** ± 0.0196	0.1462 ± 0.0118	0.1790 ± 0.0082	0.1936 ± 0.0146
iaprtc12	**0.2362** ± 0.0458	0.0599 ± 0.0393	0.1920 ± 0.0051	0.2097 ± 0.0345
micro-F1 ($k = 1$)				
data set	LCML	MLRGL	FastTag	BR
pascal07	**0.3396** ± 0.0064	0.3363 ± 0.0074	0.2455 ± 0.0108	0.2981 ± 0.0050
mirflickr	**0.4453** ± 0.0032	0.4393 ± 0.0083	0.4170 ± 0.0118	0.4320 ± 0.0039
corel5k	**0.2375** ± 0.0536	0.1087 ± 0.0163	0.1950 ± 0.0108	0.1403 ± 0.0275
espgame	**0.2235** ± 0.0212	0.1401 ± 0.0118	0.1701 ± 0.0080	0.1858 ± 0.0157
iaprtc12	**0.2312** ± 0.0451	0.0587 ± 0.0383	0.1901 ± 0.0050	0.2053 ± 0.0345
macro-F1 ($k = 2$)				
data set	LCML	MLRGL	FastTag	BR
pascal07	**0.3509** ± 0.0057	0.3451 ± 0.0080	0.2484 ± 0.0146	0.2806 ± 0.0065
mirflickr	0.4537 ± 0.0077	**0.4579** ± 0.0093	0.4352 ± 0.0500	0.4442 ± 0.0043
corel5k	**0.2407** ± 0.0503	0.1100 ± 0.0156	0.1808 ± 0.0109	0.0180 ± 0.0101
espgame	**0.2302** ± 0.0203	0.1462 ± 0.0115	0.2222 ± 0.0045	0.1842 ± 0.0131
iaprtc12	**0.2355** ± 0.0451	0.0599 ± 0.0393	0.1800 ± 0.0071	0.1976 ± 0.0325
micro-F1 ($k = 2$)				
data set	LCML	MLRGL	FastTag	BR
pascal07	**0.3426** ± 0.0053	0.3363 ± 0.0074	0.2449 ± 0.0137	0.2739 ± 0.0058
mirflickr	0.4367 ± 0.0076	**0.4393** ± 0.0083	0.4156 ± 0.0471	0.4208 ± 0.0051
corel5k	**0.2380** ± 0.0495	0.1087 ± 0.0163	0.1746 ± 0.0110	0.0176 ± 0.0098
espgame	**0.2212** ± 0.0220	0.1400 ± 0.0118	0.2130 ± 0.0044	0.1769 ± 0.0145
iaprtc12	**0.2309** ± 0.0444	0.0587 ± 0.0383	0.1712 ± 0.0071	0.1934 ± 0.0326
macro-F1 ($k = 3$)				
data set	LCML	MLRGL	FastTag	BR
pascal07	**0.3501** ± 0.0063	0.3451 ± 0.0080	0.2437 ± 0.0061	0.1436 ± 0.0083
mirflickr	**0.4613** ± 0.0030	0.4579 ± 0.0093	0.4552 ± 0.0192	0.4128 ± 0.0021
corel5k	**0.2397** ± 0.0534	0.1100 ± 0.0156	0.1808 ± 0.0080	0.0268 ± 0.0169
espgame	**0.2228** ± 0.0240	0.1462 ± 0.0115	0.2150 ± 0.0084	0.1613 ± 0.0113
iaprtc12	**0.2300** ± 0.0370	0.0599 ± 0.0393	0.1852 ± 0.0060	0.1659 ± 0.0241
micro-F1 ($k = 3$)				
data set	LCML	MLRGL	FastTag	BR
pascal07	**0.3418** ± 0.0058	0.3363 ± 0.0074	0.2404 ± 0.0055	0.1397 ± 0.0076
mirflickr	**0.4440** ± 0.0039	0.4393 ± 0.0083	0.4156 ± 0.0166	0.3895 ± 0.0030
corel5k	**0.2371** ± 0.0526	0.1087 ± 0.0163	0.1746 ± 0.0080	0.0263 ± 0.0166
espgame	**0.2143** ± 0.0256	0.1400 ± 0.0118	0.2047 ± 0.0079	0.1551 ± 0.0126
iaprtc12	**0.2258** ± 0.0382	0.0587 ± 0.0383	0.1770 ± 0.0059	0.1623 ± 0.0242

stone. In *Advances in Neural Information Processing Systems*, 757–765.

Guillaumin, M.; Mensink, T.; Verbeek, J.; and Schmid, C. 2009. Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *Proceedings of the International Conference on Computer Vision*, 309–316.

Gupta, A., and Nagar, D. 2000. *Matrix Variate Distributions*. Chapman & Hall/CRC.

Hariharan, B.; Zelnik-Manor, L.; Vishwanathan, S.; and Varma, M. 2010. Large scale max-margin multi-label classification with priors. In *Proceedings of the 27th International Conference on Machine Learning*, 423–430.

Hsu, D.; Kakade, S.; Langford, J.; and Zhang, T. 2009. Multi-label prediction via compressed sensing. In *Advances in Neural Information Processing Systems 22*, 772–780.

Kapoor, A.; Viswanathan, R.; and Jain, P. 2012. Multilabel classification using bayesian compressed sensing. In *Advances in Neural Information Processing Systems*, 2654–2662.

Koivisto, M., and Sood, K. 2004. Exact Bayesian structure discovery in Bayesian networks. *Journal of Machine Learning Research* 5:549–573.

Lauritzen, S. L. 1996. *Graphical Models*. Oxford University Press.

Petterson, J., and Caetano, T. 2011. Submodular multi-label learning. *Advances in Neural Information Processing Systems 24*.

Rai, P.; Kumar, A.; and Iii, H. D. 2012. Simultaneously leveraging output and task structures for multiple-output regression. In *Advances in Neural Information Processing Systems 25*, 3194–3202.

Read, J.; Pfahringer, B.; Holmes, G.; and Frank, E. 2009. Classifier chains for multi-label classification. In *Proceedings of the European Conference on Machine Learning*, 254–269.

Rothman, A. J.; Levina, E.; and Zhu, J. 2010. Sparse multivariate regression with covariance estimation. *Journal of Computational and Graphical Statistics* 19(4):947–962.

Tai, F., and Lin, H. 2012. Multilabel classification with principal label space transformation. *Neural Computation* 24(9):2508–2542.

Tsoumakas, G., and Katakis, I. 2007. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining* 3:1–13.

Tsoumakas, G.; Katakis, I.; and Vlahavas, I. 2010. Mining multi-label data. In Maimon, O., and Rokach, L., eds., *Data Mining and Knowledge Discovery Handbook*. Springer. 667–685.

Von Ahn, L., and Dabbish, L. 2004. Labeling images with a computer game. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 319–326.

Xu, M.; Jin, R.; and Zhou, Z.-H. 2013. Speedup matrix completion with side information: Application to multi-label learning. In *Advances in Neural Information Processing Systems*, 2301–2309.

Yu, H.-F.; Jain, P.; and Dhillon, I. S. 2014. Large-scale multi-label learning with missing labels. In *Proceedings of the 31th International Conference on Machine Learning*, 593–601.

Yuan, M., and Lin, Y. 2006. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society, Series B* 68(1):49–67.

Zhang, Y., and Yeung, D.-Y. 2010. A convex formulation for learning task relationships in multi-task learning. In *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence*, 733–742.

Zhang, M.-L., and Zhang, K. 2010. Multi-label learning by exploiting label dependency. In *Proceedings of the 16th International Conference on Knowledge Discovery and Data Mining*, 999–1008.

Zhang, M.-L., and Zhou, Z.-H. 2013. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering* 99(PrePrints):1.