Conditional Bernoulli Mixtures for Multi-label Classification

Cheng Li, Bingyu Wang, Virgil Pavlu, and Javed Aslam {chengli, rainicy, vip, jaa}@ ccs.neu.edu
College of Computer and Information Science, Northeastern University

1. Multi-label Classification

Assign a subset of candidate labels to an object (image, document, video)

`airport` ✗, `animal` ✗, `clouds` ✓, `book` ✗, `lake` ✓, `sunset` ✓, `sky` ✓, `cars` ✗, `water` ✓, `reflection` ✓

2. Existing Approaches

Binary Relevance: predict each binary label independently
☹ ignore label dependencies

Power-Set: treat each subset as a class + multi-class
☹ poor scalability; cannot predict unseen subsets

CRF: specify label dependencies with graphical models
☹ only model specified and limited (e.g., pair-wise) dependencies

PCC: predict next label based on previous labels
☹ hard to predict the jointly most probable subset

3. Proposed Model: Conditional Bernoulli Mixtures

Approximate the conditional joint by a Conditional Bernoulli Mixture (CBM) with fully factorized mixture components. $\mathbf{y}$=binary label vector of length L.

$$\textbf{CBM:} \quad p(\mathbf{y}|\mathbf{x}) = \sum_{k=1}^{K} \pi(z=k|\mathbf{x};\boldsymbol{\alpha}) \prod_{\ell=1}^{L} b(y_\ell|\mathbf{x};\beta_\ell^k)$$

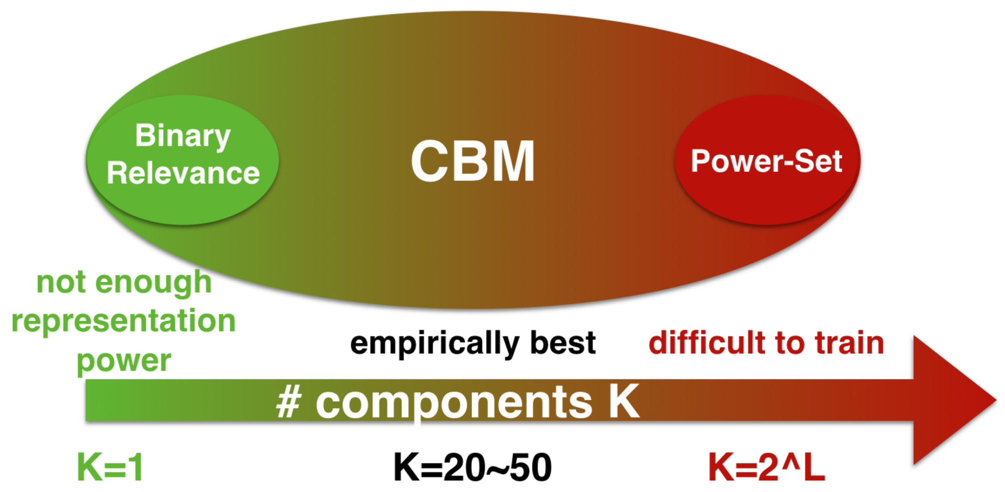
$\pi(z=k|\mathbf{x};\boldsymbol{\alpha})$: probability of belonging to component k;
instantiated with a multi-class classifier; e.g., multinomial LR

$b(y_\ell|\mathbf{x};\beta_\ell^k)$: probability of getting label y_ℓ in component k;
instantiated with a binary classifier; e.g., binary LR

☺ automatically capture label dependencies: $p(\mathbf{y}|\mathbf{x}) \neq \prod_{\ell=1}^{L} p(y_\ell|\mathbf{x})$

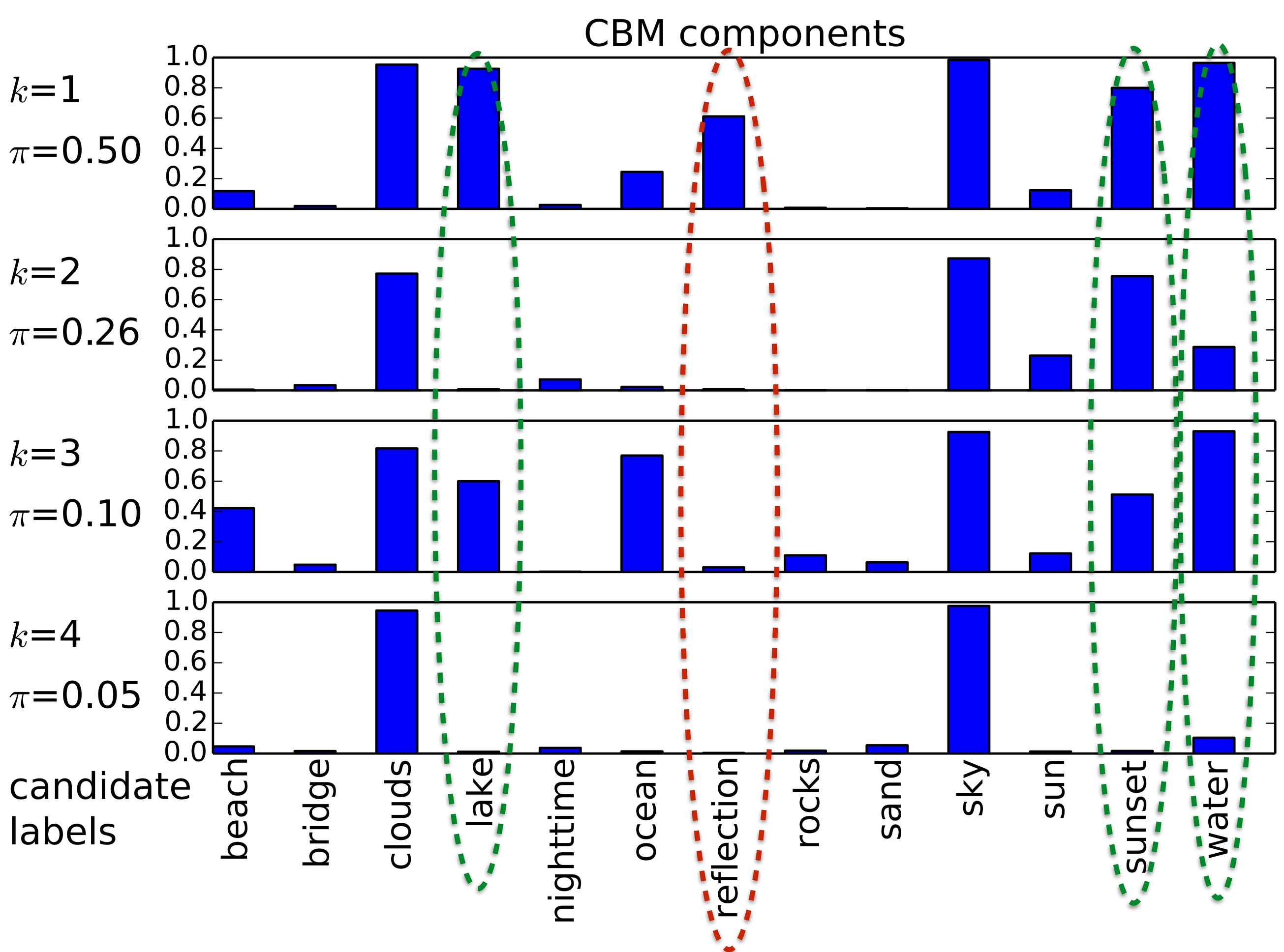
☺ a flexible reduction method: multi-label $\Rightarrow$ multi-class + binary

☺ subsume Binary Relevance and Power-Set as special cases



4. Capturing Label Dependencies: Illustration

top 4 most influential CBM components for the example image



- row = component; bar = individual label probability
- marginal probability = averaging bars weighted by π
- `water`, `lake`, `sunset` have high marginal probabilities; `reflection` has a low marginal; missed by independent prediction ☹
- `reflection` is positively correlated with `lake`, `water`, and `sunset` $\rho_{\texttt{reflection,lake}} = 0.5$, $\rho_{\texttt{reflection,water}} = 0.4$, $\rho_{\texttt{reflection,sunset}} = 0.17$
- predicting the most probable subset includes `reflection` ☺

5. Simple Training with EM

Given training dataset $\{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N$, use EM to minimize an upper bound of negative log likelihood:

$$\sum_{n=1}^{N} \mathbb{KL}(\Gamma(z_n)||\pi(z_n|\mathbf{x}_n;\boldsymbol{\alpha})) + \sum_{k=1}^{K}\sum_{\ell=1}^{L}\sum_{n=1}^{N} \gamma_n^k \mathbb{KL}(\text{Ber}(Y_{n\ell};y_{n\ell})||b(Y_{n\ell}|\mathbf{x}_n;\beta_\ell^k))$$

$\Gamma(z_n) = (\gamma_n^1, \gamma_n^2, ..., \gamma_n^K)$ is the posterior membership distribution $p(z_n|\mathbf{x}_n, \mathbf{y}_n)$.
$\text{Ber}(Y_{n\ell}; y_{n\ell})$ is the Bernoulli distribution with head probability $y_{n\ell}$.

E step: Re-estimate posterior membership probabilities:

$$\gamma_n^k = \frac{\pi(z_n = k|\mathbf{x}_n;\boldsymbol{\alpha}) \prod_{\ell=1}^{L} b(y_{n\ell}|\mathbf{x}_n;\beta_\ell^k)}{\sum_{k=1}^{K} \pi(z_n = k|\mathbf{x}_n;\boldsymbol{\alpha}) \prod_{\ell=1}^{L} b(y_{n\ell}|\mathbf{x}_n;\beta_\ell^k)}$$

M step: Update model parameters. Standard multi-class and binary classifier learning:

$$\boldsymbol{\alpha}_{new} = \underset{\boldsymbol{\alpha}}{\text{argmin}} \sum_{n=1}^{N} \mathbb{KL}(\Gamma(z_n)||\pi(z_n|\mathbf{x}_n;\boldsymbol{\alpha})) \quad \rightarrow \text{multi-class classification}$$

$$\beta^k_{\ell\, new} = \underset{\beta_\ell^k}{\text{argmin}} \sum_{n=1}^{N} \gamma_n^k \mathbb{KL}(\text{Ber}(Y_{n\ell};y_{n\ell})||b(Y_{n\ell}|\mathbf{x}_n;\beta_\ell^k)) \quad \rightarrow \text{binary classification}$$

Two concrete instantiations:

- with logistic regression (LR) learners: EM + gradient descent/LBFGS
- with gradient boosted trees (GB) learners: EM + gradient boosting

6. Fast Prediction by Dynamic Programming

Two common difficulties in prediction:

? given $p(\mathbf{y}|\mathbf{x})$ how to find $\text{argmax}_{\mathbf{y}}\, p(\mathbf{y}|\mathbf{x})$ without enumerating 2^L possibilities of $\mathbf{y}$?
? how to predict unseen subsets $\mathbf{y}$?

Find the exact $\text{argmax}_{\mathbf{y}}\, p(\mathbf{y}|\mathbf{x})$ efficiently by dynamic programming:

- to get a high overall probability, at least one component probability must be high
- in each component, list label subsets in a decreasing probability order with DP
- iterate round-robin across components and prune remaining suboptimal subsets

No problem if $\text{argmax}_{\mathbf{y}}\, p(\mathbf{y}|\mathbf{x})$ is an unseen subset

7. Results

We use the most stringent evaluation measure: subset accuracy $= \frac{1}{N}\sum_{n=1}^{N} \mathbb{1}[\hat{\mathbf{y}}_n = \mathbf{y}_n]$.
A predicted subset is considered correct only when it matches the true subset exactly.

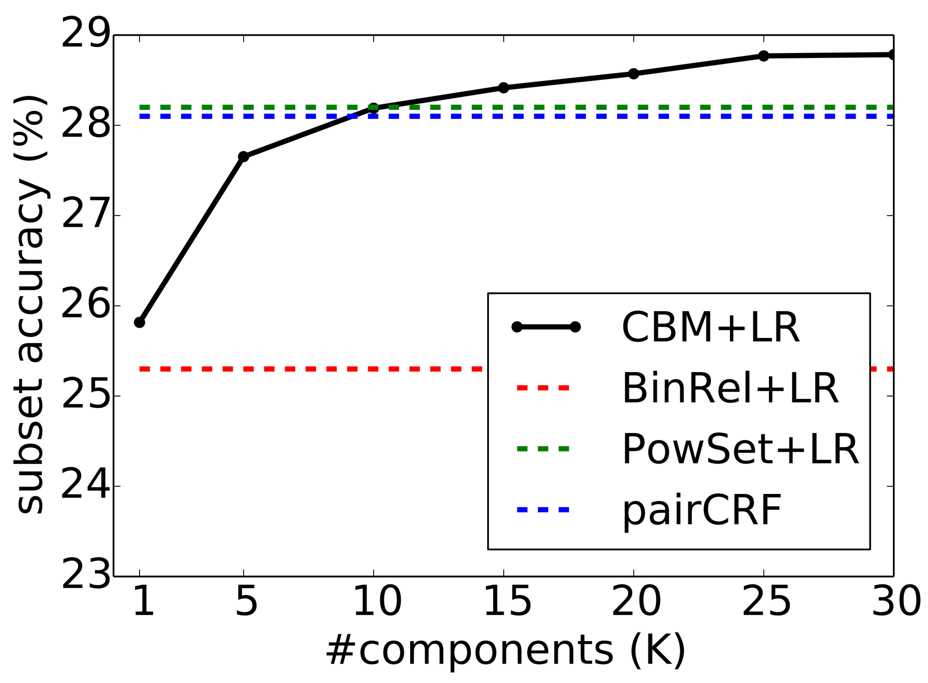
Test subset accuracy of different methods on five datasets. All numbers are in percentages.

	dataset	SCENE	RCV1	TMC2007	MEDIAMILL	NUS-WIDE
	domain	image	text	text	video	image
#labels /	#label subsets	6 / 15	103 / 799	22 / 1341	101 / 6555	81 / 18K
#features /	#datapoints	294 / 2407	47K / 6000	49K / 29K	120 / 44K	128 / 270K
Method	Learner					
BinRel	LR	51.5	40.4	25.3	9.6	24.7
PowSet	LR	68.1	50.2	28.2	9.0	26.6
CC	LR	62.9	48.2	26.2	10.9	26.0
PCC	LR	64.8	48.3	26.8	10.9	26.3
ECC-label	LR	60.6	46.5	26.0	11.3	26.0
ECC-subset	LR	63.1	49.2	25.9	11.5	26.0
CDN	LR	59.9	12.6	16.8	5.4	17.1
pairCRF	linear	68.8	46.4	28.1	10.3	26.4
CBM	LR	69.7	49.9	28.7	13.5	27.3
BinRel	GB	59.3	30.1	25.4	11.2	24.4
PowSet	GB	70.5	38.2	23.1	10.1	23.6
CBM	GB	70.5	43.0	27.5	14.1	26.5

- among all methods with LR learners, CBM is the best on 4 out of 5 datasets
- replace LR with GB $\Rightarrow$ further improvements on SCENE and MEDIAMILL

8. Analysis

Test subset accuracy on TMC dataset with varying number of components K for CBM+LR



- $K = 1$, CBM only estimates marginals and performs similarly to Binary Relevance
- $K > 1$, CBM becomes a better joint estimator and achieves better subset accuracy
- $K = 30$, performance asymptotes

➪ Our code is available at `https://github.com/cheng-li/pyramid`