Conditional Bernoulli Mixtures for Multi-label Classification

Cheng Li, Bingyu Wang, Virgil Pavlu, and Javed Aslam

Northeastern University

Task: Multi-label Classification

- binary classification: 1 out of 2
- multi-class classification: 1 out of many
- **multi-label classification: many out of many**

Multi-label Classification: Example

News Article Categorization

Breakingviews

Twitter may score big with football digital rights

By Jennifer Saba | April 5, 2016

Tags: INTERNET | NFL | SPORTS | SPORTS BUSINESS | TWITTER

The author is a Reuters Breakingviews columnist. The opinions expressed are her own.

Twitter may finally be gaining some ground. Chief Executive Jack Dorsey's social-media company has won the rights to stream National Football League games on 10 Thursday nights for roughly $10 million, according to technology site Re/code. That's about the price of a one-minute Super Bowl commercial. After fumbling with stalled growth in the number of users, Twitter may have found a cheap way to stay on the field with rivals like Facebook.

Internet ✓, crime ✗, NFL ✓, government ✗, Asia ✗, sports ✓, politics ✗, sports business ✓, Twitter ✓

Multi-label Classification: Example

Image Tagging

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

Multi-label Classification: Mathematical Formulation

$$\mathbf{x} \xrightarrow{h} \mathbf{y} = [\overbrace{1, 0, 0, 1, 0, ..., 1}^{\text{length } L}]$$

L: # candidate labels

$\mathbf{x}$: instance

$\mathbf{y}$: label subset, written as binary vector of length L

$\mathbf{y}_\ell = 1$ if label ℓ occurs

Naive Approach: Predict Each Label Independently

Binary Relevance: not always effective

- `water`: easy to predict directly
- `reflection`: hard to predict directly (based on the given feature representation)

Better Solution: Exploit Label Dependencies

let easy labels help difficult labels

- `water`: easy to predict directly
- `reflection`: often co-occurs with `water`

How to Model Label Dependencies?

Existing approaches

- Power-Set: treat each subset as a class + multi-class
  ☹ $2^L \Rightarrow$ poor scalability; cannot predict unseen subsets

How to Model Label Dependencies?

Existing approaches

- Power-Set: treat each subset as a class + multi-class
  ☹ $2^L \Rightarrow$ poor scalability; cannot predict unseen subsets

- Conditional Random Field: manually specify label dependencies with a graphical model
  ☹ only model specified (e.g., all pair-wise) dependencies

How to Model Label Dependencies?

Existing approaches

- Power-Set: treat each subset as a class + multi-class
  ☹ $2^L \Rightarrow$ poor scalability; cannot predict unseen subsets

- Conditional Random Field: manually specify label dependencies with a graphical model
  ☹ only model specified (e.g., all pair-wise) dependencies

- Classifier Chain: $h(\mathbf{x}, y_1, y_2, ..., y_{\ell-1}) \rightarrow y_\ell$
  ☹ hard to predict the jointly most probable subset

Proposed Model: Conditional Bernoulli Mixtures (CBM)

Idea: approximate $p(\mathbf{y}|\mathbf{x})$ by a Conditional Bernoulli Mixture (CBM) with fully factorized mixture components

Proposed Model: Conditional Bernoulli Mixtures (CBM)

Idea: approximate $p(\mathbf{y}|\mathbf{x})$ by a Conditional Bernoulli Mixture (CBM) with fully factorized mixture components

- Step 1. write $p(\mathbf{y})$ as a mixture

$$\text{Mixture:} \quad p(\mathbf{y}) = \sum_{k=1}^{K} \pi^k p(\mathbf{y}; \boldsymbol{\beta}^k)$$

Proposed Model: Conditional Bernoulli Mixtures (CBM)

Idea: approximate $p(\mathbf{y}|\mathbf{x})$ by a Conditional Bernoulli Mixture (CBM) with fully factorized mixture components

- Step 1. write $p(\mathbf{y})$ as a mixture

$$\text{Mixture:} \quad p(\mathbf{y}) = \sum_{k=1}^{K} \pi^k p(\mathbf{y}; \boldsymbol{\beta}^k)$$

- Step 2: factorize component density

$$\text{Bernoulli Mixture:} \quad p(\mathbf{y}) = \sum_{k=1}^{K} \pi^k \prod_{\ell=1}^{L} b(y_\ell; \boldsymbol{\beta}_\ell^k)$$

Proposed Model: Conditional Bernoulli Mixtures (CBM)

Idea: approximate $p(\mathbf{y}|\mathbf{x})$ by a Conditional Bernoulli Mixture (CBM) with fully factorized mixture components

- Step 1. write $p(\mathbf{y})$ as a mixture

$$\text{Mixture:} \quad p(\mathbf{y}) = \sum_{k=1}^{K} \pi^k p(\mathbf{y}; \boldsymbol{\beta}^k)$$

- Step 2: factorize component density

$$\text{Bernoulli Mixture:} \quad p(\mathbf{y}) = \sum_{k=1}^{K} \pi^k \prod_{\ell=1}^{L} b(y_\ell; \boldsymbol{\beta}_\ell^k)$$

- Step 3: condition on $\mathbf{x}$

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

Proposed Model: Conditional Bernoulli Mixtures (CBM)

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

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

$b(y_\ell|\mathbf{x}; \boldsymbol{\beta}_\ell^k)$: probability of $\mathbf{x}$ having label y_ℓ in component k;
instantiated with a binary classifier
e.g., binary logistic regression with weight $\boldsymbol{\beta}_\ell^k$.

$$\text{Prediction:} \quad \operatorname*{argmax}_{\mathbf{y}} p(\mathbf{y}|\mathbf{x})$$

Proposed Model: Conditional Bernoulli Mixtures (CBM)

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

- ☺ Property 1: automatically capture label dependencies
- ☺ Property 2: a flexible reduction method
- ☺ Property 3: easily adjust the complexity by changing the number of components K
- ☺ Property 4: simple training with EM
- ☺ Property 5: fast prediction by dynamic programming

Proposed Model: Conditional Bernoulli Mixtures (CBM)

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

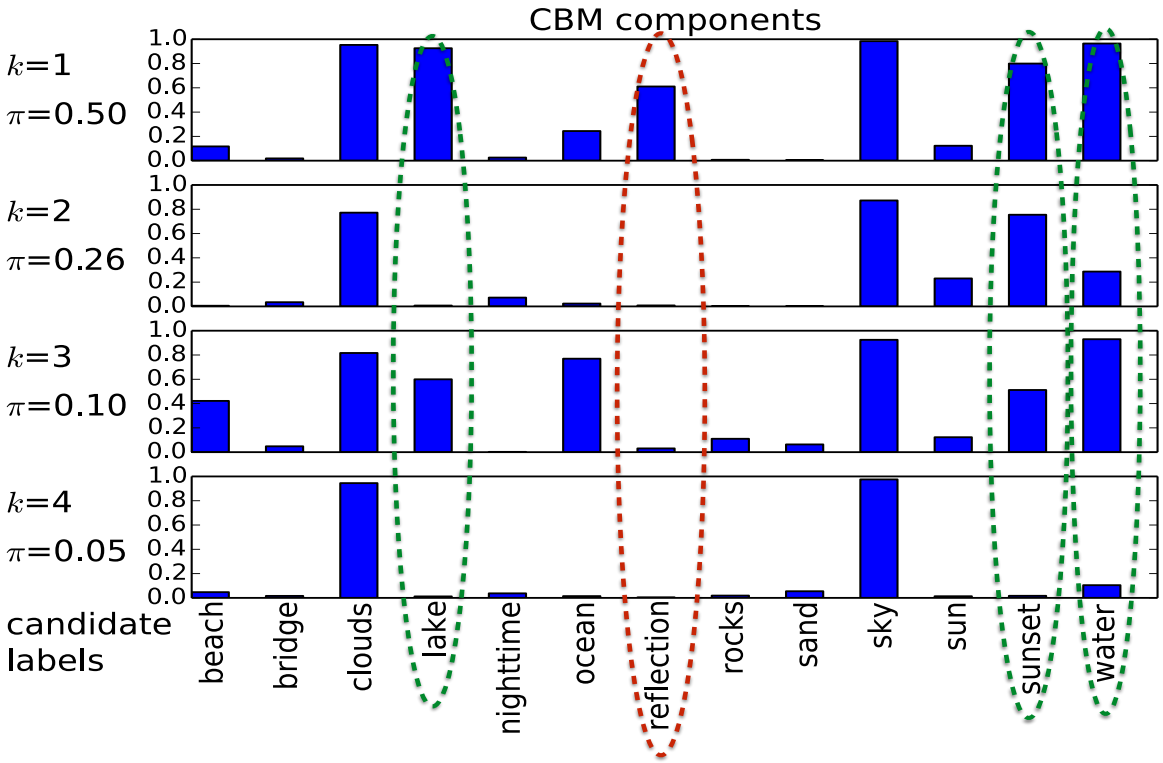
Property 1: automatically capture label dependencies

$$p(\mathbf{y}|\mathbf{x}) \neq \prod_{\ell=1}^{L} p(y_\ell|\mathbf{x})$$

analogy: Gaussian mixture with fully factorized components can represent a complex joint

Proposed Model: Conditional Bernoulli Mixtures (CBM)

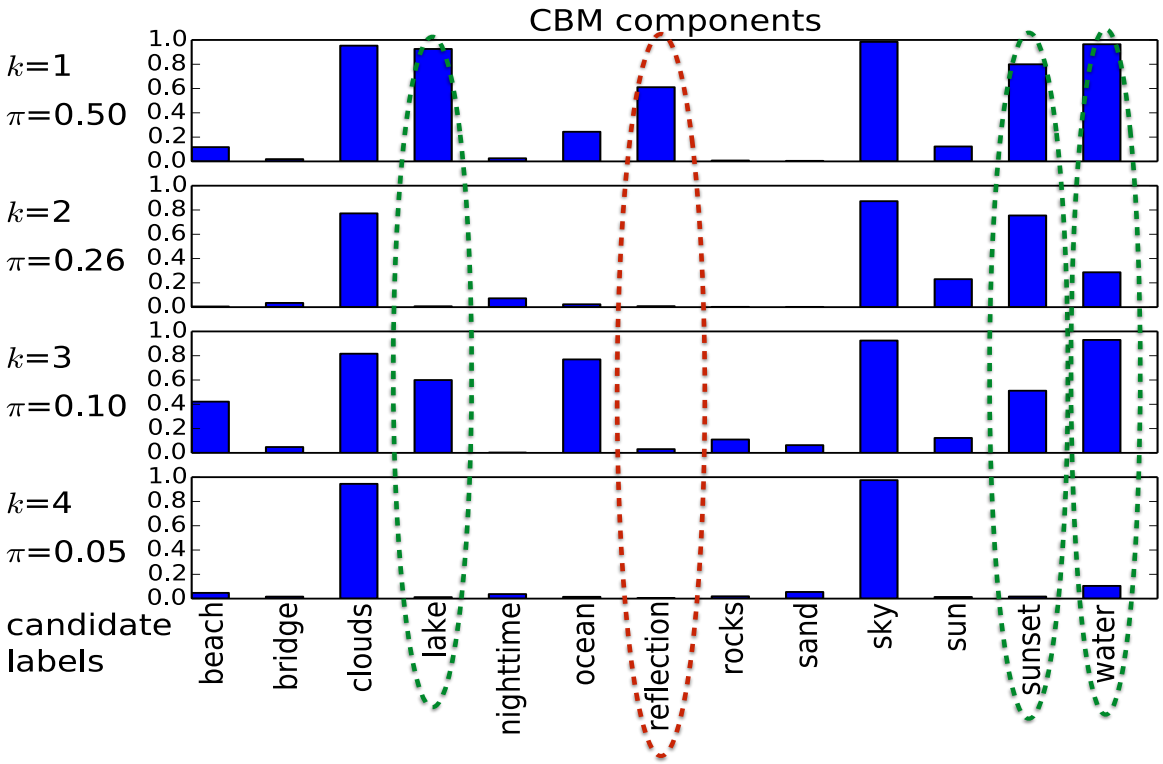
Property 1: capture label dependencies – illustration



- $p(\mathbf{y}|\mathbf{x})$ estimation provided by CBM
- showing only top 4 components; row = component; bar = individual label probability; π = mixture coefficient

Proposed Model: Conditional Bernoulli Mixtures (CBM)

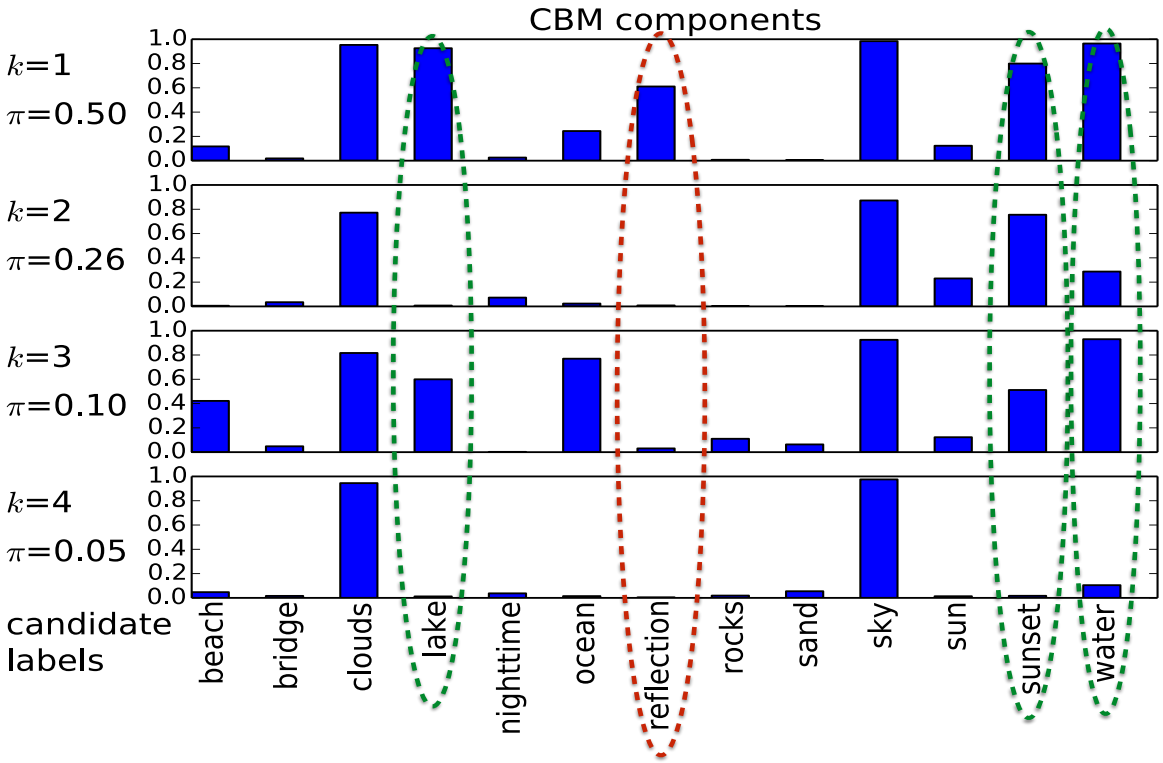
Property 1: capture label dependencies – illustration



- marginal probability = averaging bars weighted by π
- $p(\texttt{water}|\mathbf{x}) = 0.69$, $p(\texttt{lake}|\mathbf{x}) = 0.56$, $p(\texttt{sunset}|\mathbf{x}) = 0.66$
- $p(\texttt{reflection}|\mathbf{x}) = 0.32$
  $\Rightarrow$ missed by independent prediction ☹

Proposed Model: Conditional Bernoulli Mixtures (CBM)

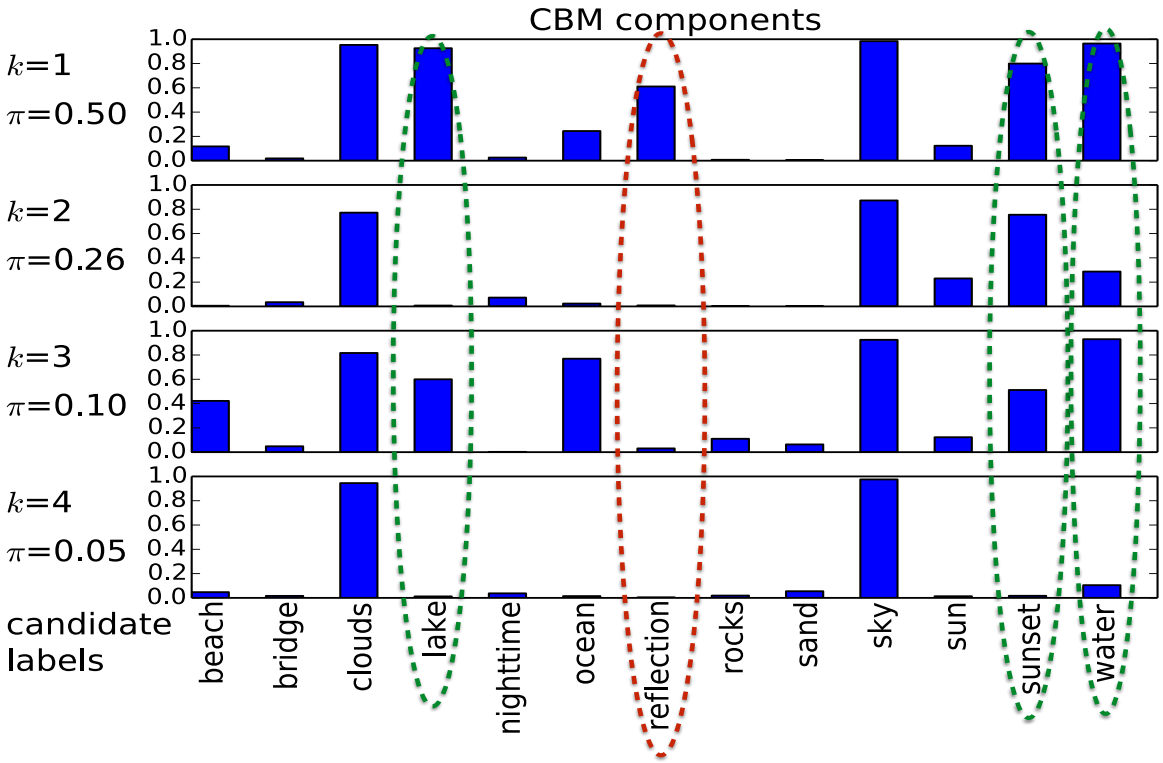
Property 1: capture label dependencies – illustration



- `reflection` is positively correlated with `lake`, `water`, and `sunset`;
  $p(\mathbf{y}|\mathbf{x}) \Rightarrow \rho_{\texttt{reflection,lake}} = 0.5$, $\rho_{\texttt{reflection,water}} = 0.4$, $\rho_{\texttt{reflection,sunset}} = 0.17$

Proposed Model: Conditional Bernoulli Mixtures (CBM)

Property 1: capture label dependencies – illustration



$p(\{\texttt{clouds, lake, sky, sunset, water, reflection}\}|\mathbf{x}) = 0.09$

$p(\{\texttt{clouds, lake, sky, sunset, water}\}|\mathbf{x}) = 0.06$

$\Rightarrow$ predicting the most probable subset includes `reflection` ☺

Proposed Model: Conditional Bernoulli Mixtures (CBM)

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

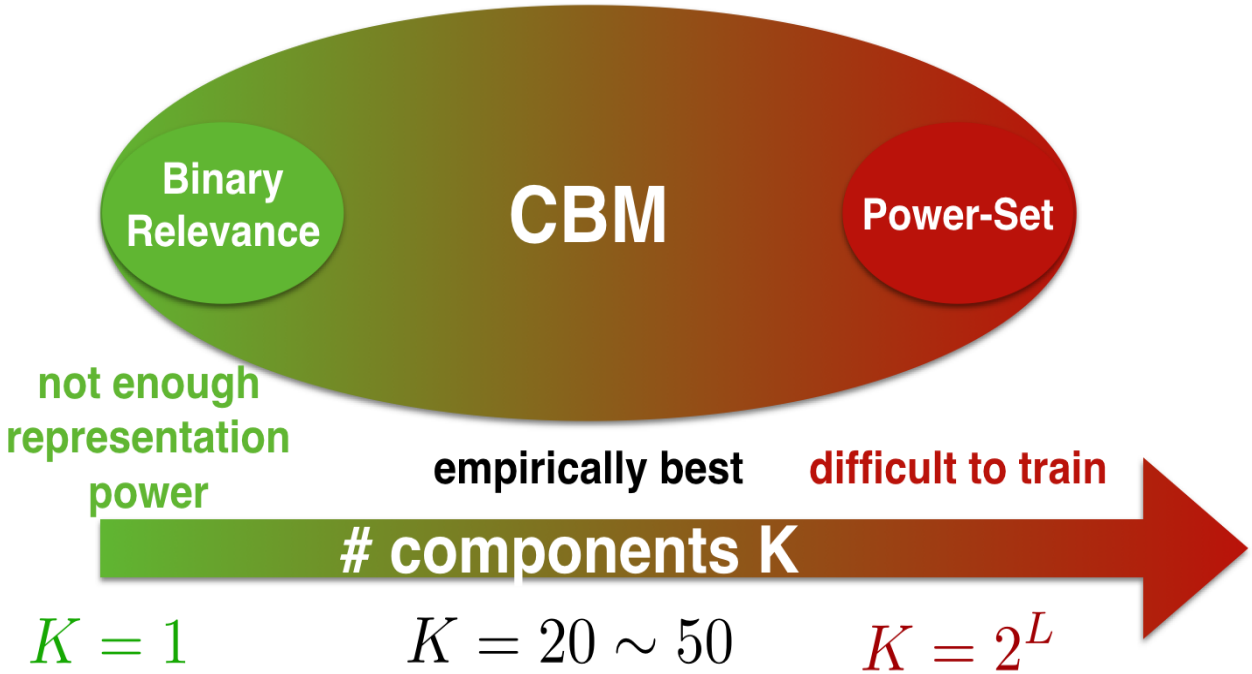
Property 2: a flexible reduction method

- multi-label $\Rightarrow$ multi-class + binary
- instantiated by many binary/multi-class classifiers
  e.g., logistic regressions, gradient boosted trees, neural networks

Proposed Model: Conditional Bernoulli Mixtures (CBM)

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

Property 3: easily adjust the complexity by changing the number of components K



Proposed Model: Conditional Bernoulli Mixtures (CBM)

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

Property 4: Simple Training with EM

Idea:

- maximum likelihood
- hidden variables $\Rightarrow$ EM
- update parameters $\Rightarrow$ binary and multi-class classifier learning

Proposed Model: Conditional Bernoulli Mixtures (CBM)

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

Property 4: Simple Training with EM

Denote the posterior membership distribution $p(z_n|\mathbf{x}_n, \mathbf{y}_n)$ as $\Gamma(z_n) = (\gamma_n^1, \gamma_n^2, ..., \gamma_n^K)$.

E step: Re-estimate posterior membership probabilities:

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

Proposed Model: Conditional Bernoulli Mixtures (CBM)

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

Property 4: Simple Training with EM

M step: Update model parameters. Decompose into simple classification problems:

$$\boldsymbol{\alpha}_{new} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \sum_{n=1}^{N} \mathbb{KL}(\Gamma(z_n)||\pi(z_n|\mathbf{x}_n; \boldsymbol{\alpha}))$$

(multi-class classification with soft target labels)

$$\boldsymbol{\beta}_{\ell\,new}^k = \underset{\boldsymbol{\beta}_\ell^k}{\operatorname{argmin}} \sum_{n=1}^{N} \gamma_n^k \mathbb{KL}(\text{Ber}(Y_{n\ell}; y_{n\ell})||b(Y_{n\ell}|\mathbf{x}_n; \boldsymbol{\beta}_\ell^k))$$

(weighted binary classification)

Proposed Model: Conditional Bernoulli Mixtures (CBM)

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

Property 5: Fast Prediction by Dynamic Programming

A common difficulty in prediction:

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

Proposed Model: Conditional Bernoulli Mixtures (CBM)

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

Property 5: Fast Prediction by Dynamic Programming

A common difficulty in prediction:

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

Existing solutions used in Power-Set, CRF, and Classifier Chain:

- ☹ restrict to $\mathbf{y}$ in training set $\Rightarrow$ will not predict unseen $\mathbf{y}$
- ☹ approximate inference $\Rightarrow$ suboptimal

Proposed Model: Conditional Bernoulli Mixtures (CBM)

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

Property 5: Fast Prediction by Dynamic Programming

A common difficulty in prediction:

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

Existing solutions used in Power-Set, CRF, and Classifier Chain:

- ☹ restrict to $\mathbf{y}$ in training set $\Rightarrow$ will not predict unseen $\mathbf{y}$
- ☹ approximate inference $\Rightarrow$ suboptimal

CBM:

- ☺ efficiently find the exact $\text{argmax}_{\mathbf{y}}\, p(\mathbf{y}|\mathbf{x})$ by DP

Proposed Model: Conditional Bernoulli Mixtures (CBM)

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

Summary

- ☺ Property 1: automatically capture label dependencies
- ☺ Property 2: a flexible reduction method
- ☺ Property 3: easily adjust the complexity by changing the number of components K
- ☺ Property 4: simple training with EM
- ☺ Property 5: fast prediction by dynamic programming

Experimental Results on Benchmark Datasets

- 5 Datasets of different types

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

Experimental Results on Benchmark Datasets

- 5 Datasets of different types

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

- 2 instantiations of CBM: LR and GB

Experimental Results on Benchmark Datasets

- 5 Datasets of different types

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

- 2 instantiations of CBM: LR and GB
- 8 baselines: BinRel, PowSet, CC, PCC, ECC-label, ECC-subset, CDN, pairCRF

Experimental Results on Benchmark Datasets

- 5 Datasets of different types

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

- 2 instantiations of CBM: LR and GB
- 8 baselines: BinRel, PowSet, CC, PCC, ECC-label, ECC-subset, CDN, pairCRF
- evaluation measure: subset accuracy

Experimental Results on Benchmark Datasets

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

- with LR learner, CBM is the best on 4 out of 5 datasets

Experimental Results on Benchmark Datasets

	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

- replace LR with GB $\Rightarrow$ further improvements on 2 datasets
  SCENE: 69.7$\rightarrow$70.5; MEDIAMILL: 13.5$\rightarrow$14.1
- use different learners for different applications

Conclusion

- proposed a new multi-label model CBM
- enjoys many nice properties
- performs well on real data
- code available at `https://github.com/cheng-li/pyramid`

Thank You!