Multi-Dimensional Classification via Sparse Label Encoding

ICML
International Conference On Machine Learning

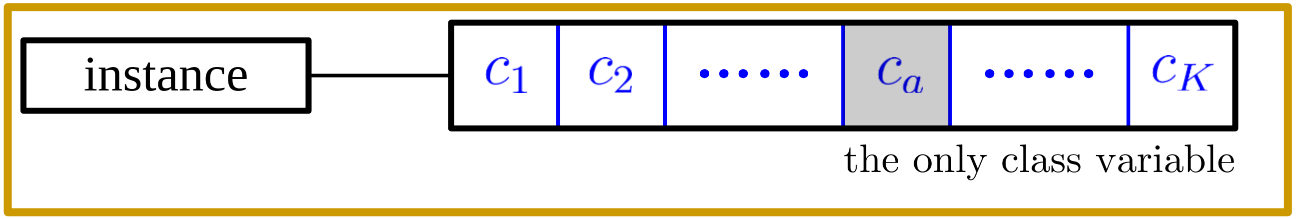
Bin-Bin Jia **Min-Ling Zhang**

School of Computer Science and Engineering,
MOE Key Laboratory of Computer Network & Information Integration
Southeast University, China

Virtual Event

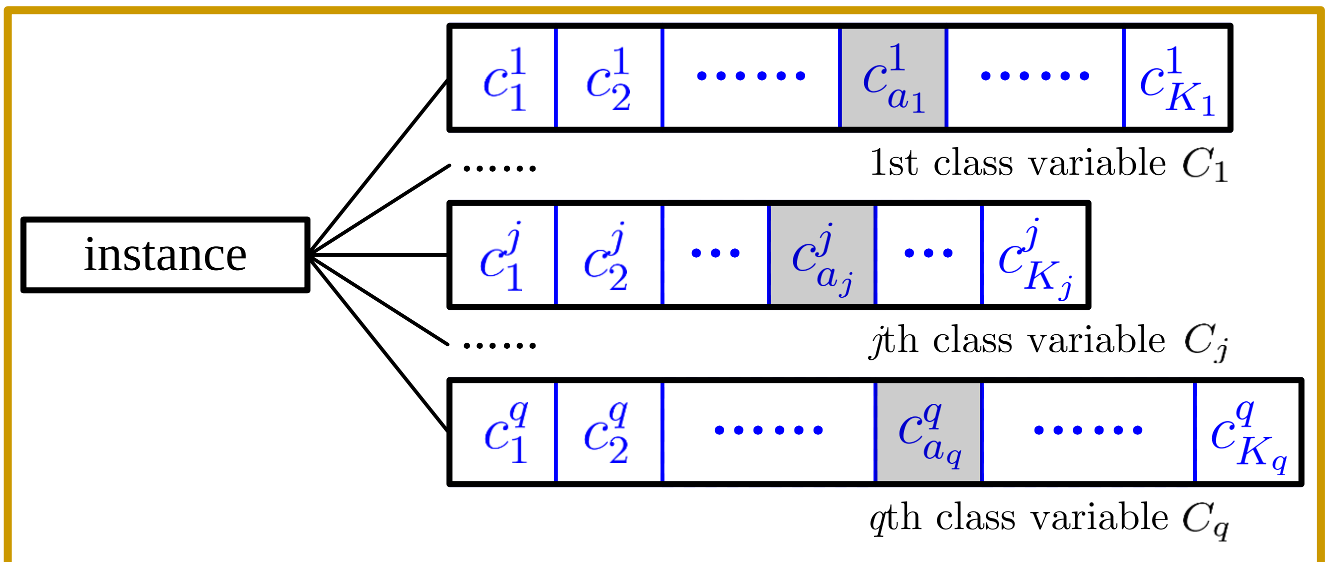
Multi-Dimensional Classification

Traditional Multi-class classification



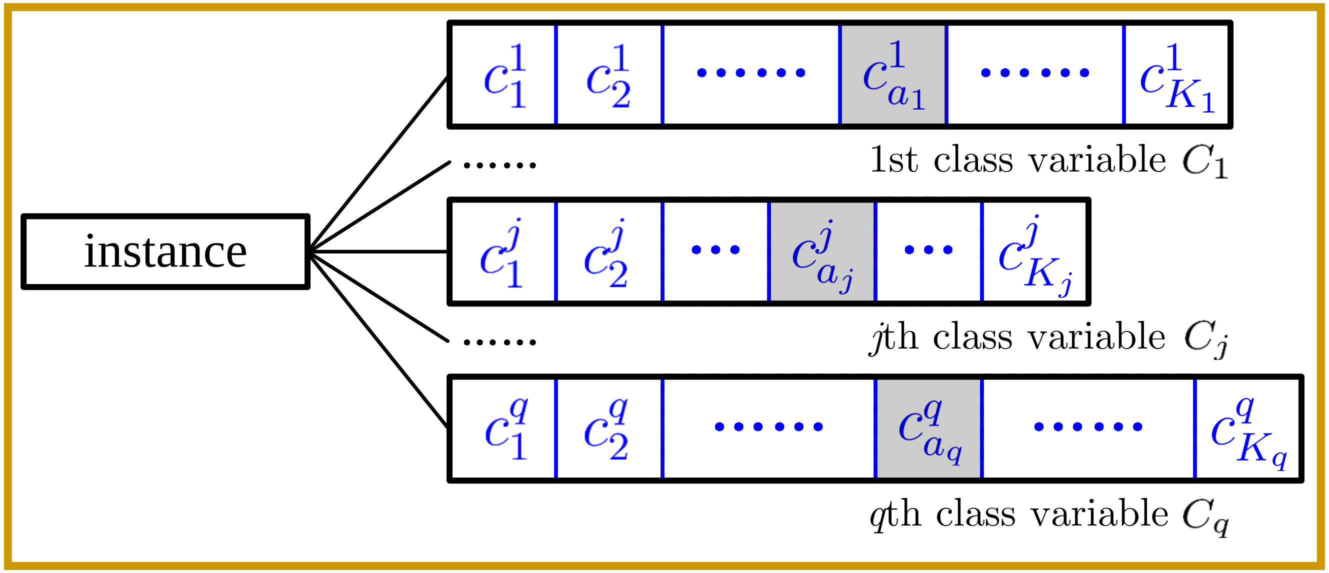
- Only one class variable

Multi-Dimensional Classification (MDC)



- Multiple class variables

The Problem



- Multiple class variables

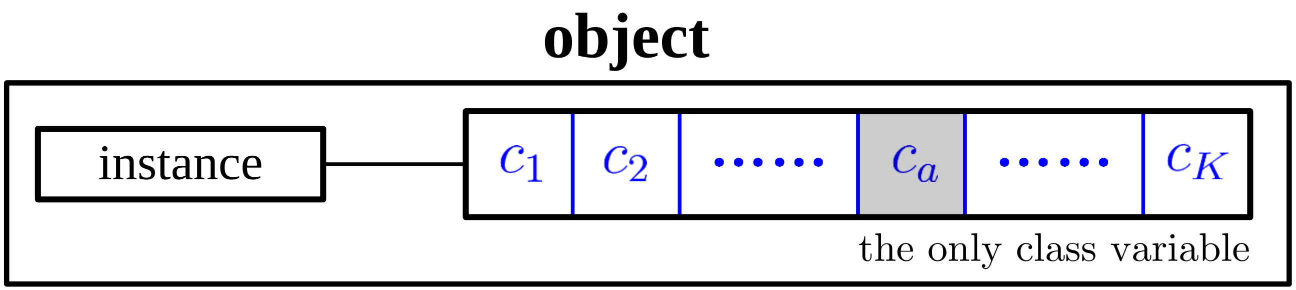
Existing works: learn the predictive model **in the original output space** for MDC where the dependencies among class variables are considered.

Our work: make a first attempt to learn the predictive model **in its transformed label space** for MDC.

Outline

- **Introduction**
- The proposed SLEM Approach
- Experiments
  - Experimental setup
  - Experimental results
- Conclusion

Multi-Class Classification (MCC)



Input Space

represented by a single instance characterizing its properties

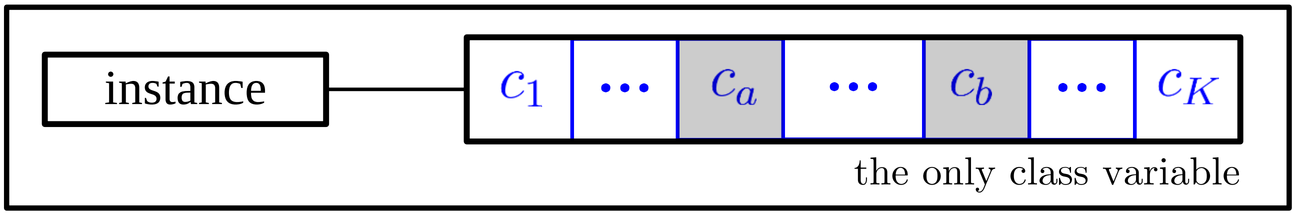
Output Space

associated with a single class variable characterizing its semantics

Only one label in the single class space is relevant.

Multi-Label Classification (MLC)

object



Input Space

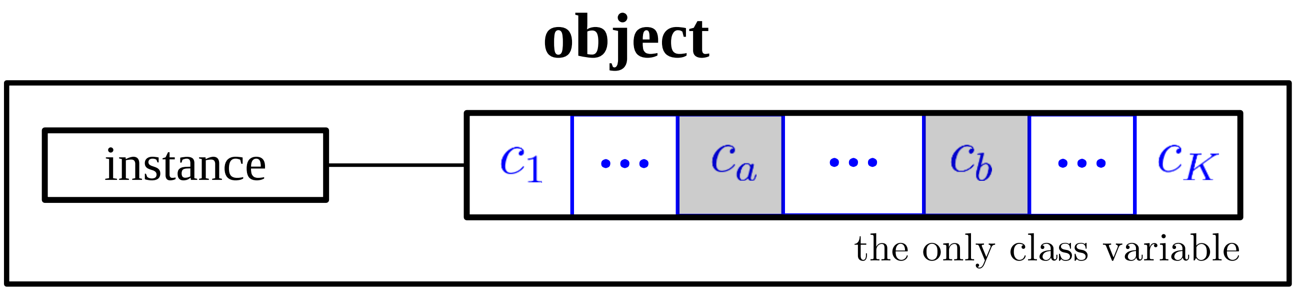
represented by a single instance characterizing its properties

Output Space

associated with a single class variable characterizing its semantics

Multiple labels in the single class space are relevant.

Multi-Label Classification (MLC)



Input Space

represented b… …res

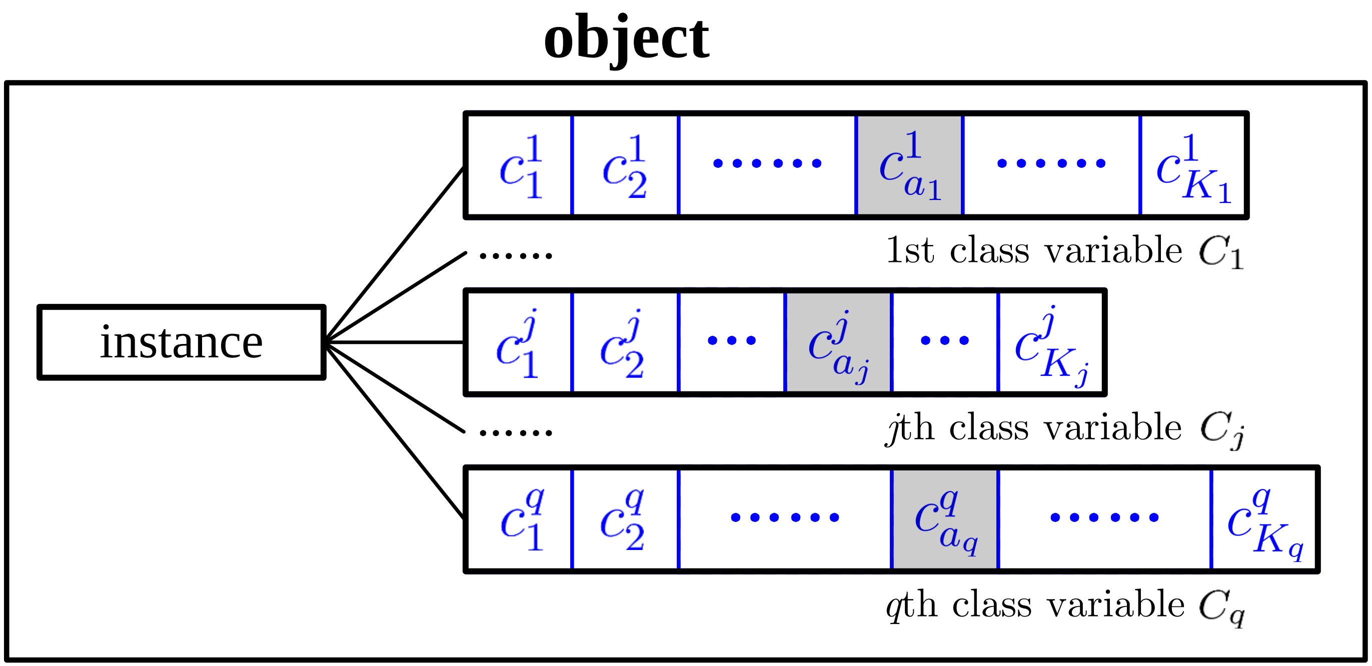
Outpu…

associated with a single class variable characterizing its semantics

Both MCC and MLC assume one single homogeneous class space.

Multiple labels in the single class space are relevant.

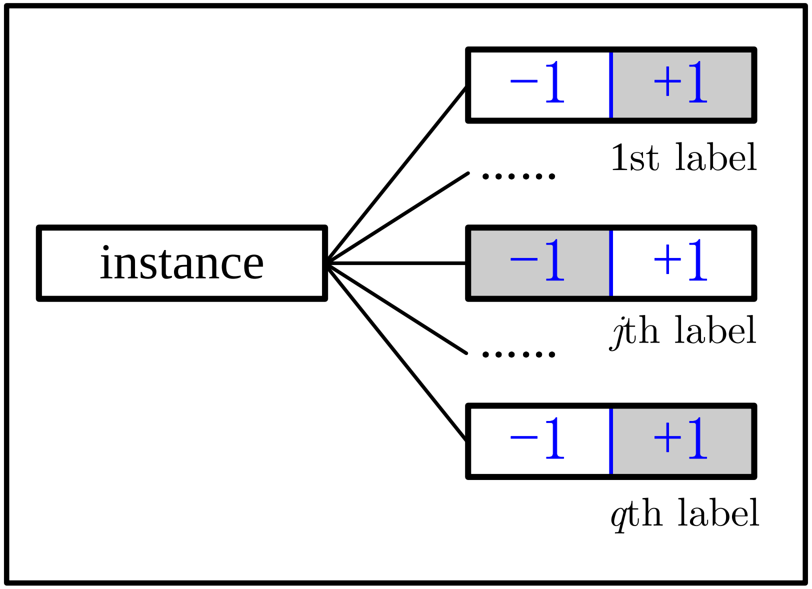
Multi-Dimensional Classification



Multiple labels are relevant with each of them from one heterogeneous class space.

What's More…

Mathematical view of multi-label classification:



MLC's output space can be regarded as multiple **binary** class variables similar to MDC. But conceptually, they are all in the same class space.

Key difference between MDC and MLC

MDC usually assumes ***heterogeneous*** semantic spaces

MLC usually assumes ***homogeneous*** semantic space

MDC Examples

Dim. 1: Genre ---> rock, popular, classical, etc.

Dim. 2: Instrument ---> piano, violin, guitar, etc.

Dim. 3: Language ---> English, Chinese, Spanish, etc.

A piece of music

Dim. 1: Topic ---> sports, politics, social, Sci&Tech, etc.

Dim. 2: Mood ---> good news, neutral news, bad news

Dim. 3: Zone ---> domestic, intra-/inter-continental

A news document

MDC Examples

Dim. 1: Genre

rock, popular,

Widely exist in real-world applications

- Text classification [Ortigosa-Hernandez et al., Neurocomputing12] [Serafino et al., LNAI15] [Tu et al., ACM TIST17]
- Bioinformatics [Read et al., TKDE14] [Fernandez-Gonzalez et al., ICML Workshop'15] [Bolt et al., IJAR17][Benjumeda et al., IJAR18]
- Computer vision [Ma et al., Neurocomputing18]
- Resource allocation [Muktadir et al., IEICE TIS19]
- Other areas [Tekinerdogan, SoSE'19] [Verma et al. Sci Total Environ21]

Dim. 3: Zone

/inter-continental

A news document

Existing Approaches

Intuitive strategies:

- Binary Relevance (BR): training an independent multi-class classifier w.r.t. each class space
- Class Powerset (CP): training a single multi-class classifier by conducting powset transformation

Other specifically designed approaches:

- Specifying chaining order over class variables [Zaragoza et al., IJCAI'11; Read et al., Pattern Recognition14]
- Partitioning class variables into groups [Read et al., TKDE14]
- Assuming DAG structure over class variables [Bolt & van der Gaag, IJAR17; Benjumeda et al., IJAR18; Gil-Begue et al., Artif. Intell. Rev.21]

Existing Approaches

Intuitive strategies:

- Binary Relevance (BR): training an independent multi-class classifier w.r.t. each class space
- C… …ulti-class classifier

in the original output space !!!

Other specifically designed approaches:

- Specifying chaining order ov… …al., IJCAI'11; Read et al., Pattern Reco…
- Partitioning class variables i… …al., TKDE14]
- Assuming DAG structure over class variables [Bolt & van der Gaag, IJAR17; Benjumeda et al., IJAR18; Gil-Begue et al., Artif. Intell. Rev.21]

Why not in the trans-formed output space?

Outline

- Introduction
- **The proposed SLEM Approach**
- Experiments
  - Experimental setup
  - Experimental results
- Conclusion

Formal Definition of MDC

Settings

$\mathcal{X} = \mathbb{R}^d$: d-dimensional input (feature) space

$\mathcal{Y} = C_1 \times C_2 \times \cdots \times C_q$, where $C_j = \{c_1^j, c_2^j, \ldots, c_{K_j}^j\}$

: output space which corresponds to the Cartesian product of q class spaces (dim.)

Inputs

$\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i) \mid 1 \leq i \leq m\}$: training data set, where

$$\boldsymbol{x}_i = [x_{i1}, x_{i2}, \ldots, x_{id}]^\top \in \mathcal{X}$$

$$\boldsymbol{y}_i = [y_{i1}, y_{i2}, \ldots, y_{iq}]^\top \in \mathcal{Y}$$

Outputs

f : multi-dimensional classifier $\mathcal{X} \to \mathcal{Y}$

Formal Definition of MDC

Settings

$\mathcal{X} = \mathbb{R}^d$: d-dimensional input (feature) space

$\mathcal{Y} = C_1 \times C_2 \times \cdots \times C$ where $C_j = \{c^j, c^j, \ldots, c^j\}$

General MDC approaches

Induce the MDC model ***in the original output space***

Our SLEM approach

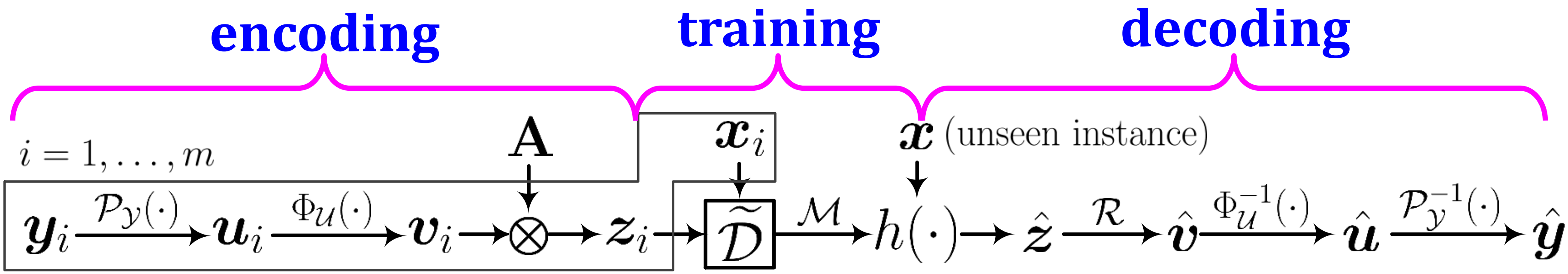
Induce the MDC model ***in the transformed output space***

$\boldsymbol{y}_i = [y_{i1}, y_{i2}, \ldots, y_{iq}]^\top \in \mathcal{Y}$

Outputs

f : multi-dimensional classifier $\mathcal{X} \to \mathcal{Y}$

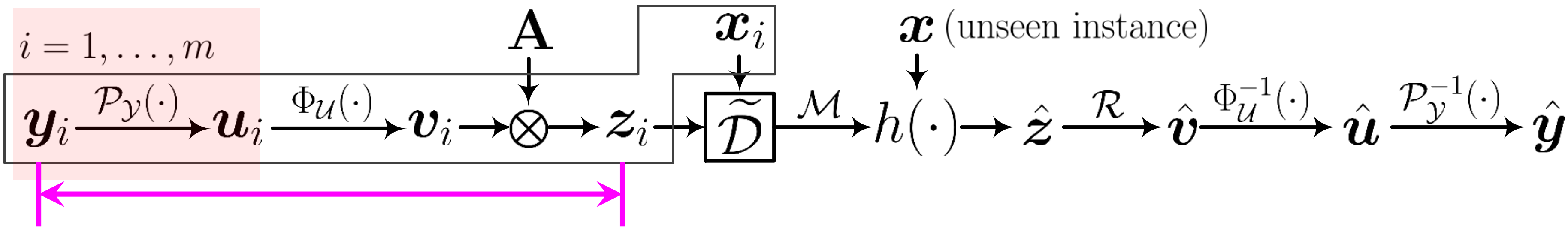
The SLEM Approach (1/8)



SLEM works in an *encoding-training-decoding* framework:

- In encoding phase, the categorical class vector $\boldsymbol{y}_i$ is transformed into a real-valued label vector $\boldsymbol{z}_i$.
- In training phase, a multi-output regression model $h(\cdot)$ is induced in the encoded label space.
- In decoding phase, the predicted class vector $\hat{\boldsymbol{y}}$ for unseen instance $\boldsymbol{x}$ is determined by conducting inverse operations in encoding phase based on $h(\boldsymbol{x})$.

The SLEM Approach (2/8)



- **Pairwise grouping:** $\mathcal{Y} = C_1 \times \cdots \times C_q \Rightarrow \mathcal{U} = C'_1 \times \cdots \times C'_{\lceil \frac{q}{2} \rceil}$

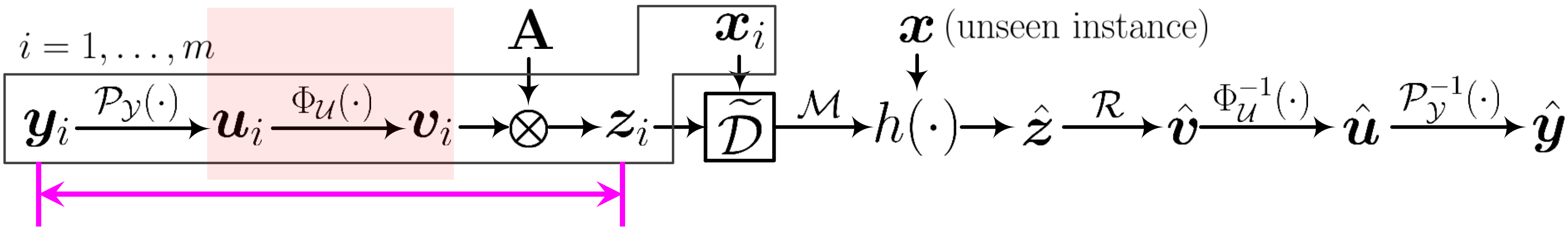
- Make the results of one-hot conversion in next step sparser.
- Group q class spaces into $\lfloor \frac{q}{2} \rfloor$ pairs (plus a singleton one if q is odd) according to the number of class labels in each class space.

$$C_1 = \{c_1^1, c_2^1, c_3^1, c_4^1\}$$
$$C_2 = \{c_1^2, c_2^2, c_3^2\}$$
$$C_3 = \{c_1^3, c_2^3, c_3^3, c_4^3, c_5^3\}$$
$$C_4 = \{c_1^4, c_2^4, c_3^4, c_4^4, c_5^4, c_6^4\}$$
$$C_5 = \{c_1^5, c_2^5\}$$

$$C'_1 \leftarrow C_5 \times C_3$$
$$C'_2 \leftarrow C_2 \times C_1$$
$$C'_3 \leftarrow C_4$$

q is odd

The SLEM Approach (3/8)



□ One-hot conversion: Categorical space $\mathcal{U} \Rightarrow$ Binary {0,1} space

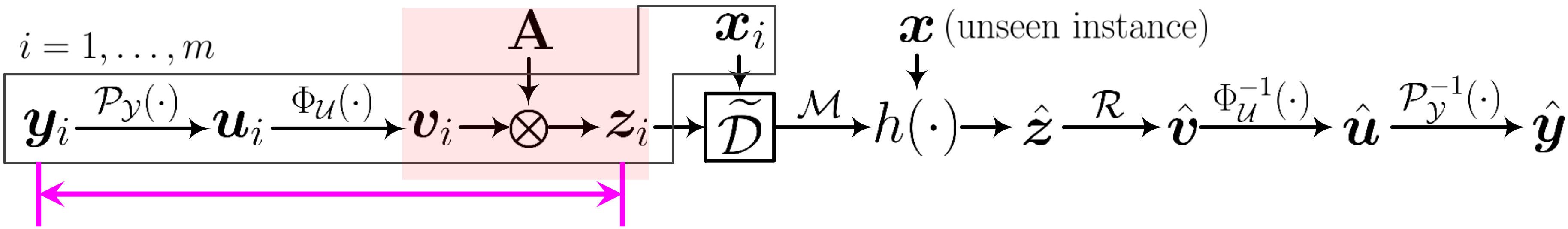
- Facilitate the following numeric computations.
- Convert each class label in categorical class vector $\boldsymbol{u}_i$ into its one-hot form and then concatenate them together.

$C'_1 = \{a, b, c, d\}$, $C'_2 =$ {I, II, III, IV, V, VI}, $C'_3 = \{\alpha, \beta, \gamma\}$ $(k = 3)$

$$\boldsymbol{u}_i = [b, \mathrm{III}, \gamma]^\top \quad \Rightarrow \boldsymbol{v}_i = [\underbrace{0, 1, 0, 0}, \underbrace{0, 0, 1, 0, 0, 0}, \underbrace{0, 0, 1}]^\top \qquad (k\text{-sparse})$$

Local sparsity: there is one and only one '1' among each local group

The SLEM Approach (4/8)



- **Sparse linear encoding:** Binary {0,1} space ⇒ Real-valued space

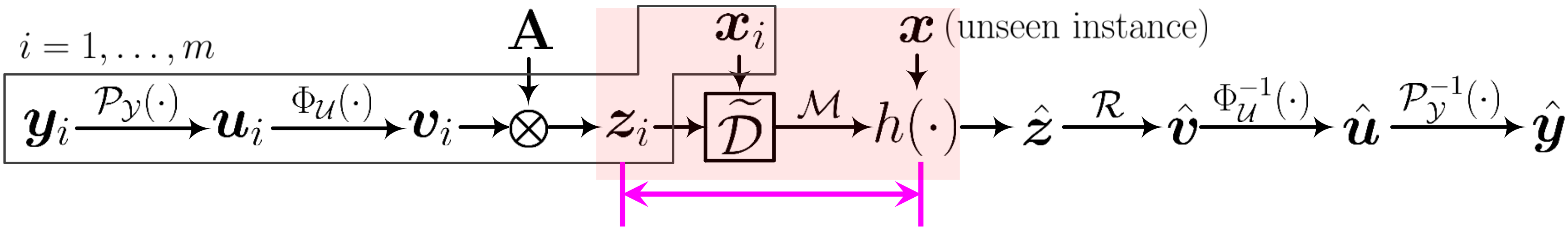
- Blend the heterogeneous class spaces into an integrated one.
- Encode binary label vector $\boldsymbol{v}_i$ into real-valued vector $\boldsymbol{z}_i$ with matrix $\mathbf{A}$ which satisfies k-RIP (i.e., $\boldsymbol{z}_i = \mathbf{A}\boldsymbol{v}_i$):

Definition 1. For matrix $\mathbf{A}$, if there is a constant $\delta_k \in [0, 1)$ which satisfies

$$(1 - \delta_k)\,\|\boldsymbol{v}\|_2^2 \leq \|\mathbf{A}\boldsymbol{v}\|_2^2 \leq (1 + \delta_k)\,\|\boldsymbol{v}\|_2^2$$

where $\boldsymbol{v}$ is any k-sparse vector, then $\mathbf{A}$ is known as satisfying k-order Restricted Isometry Property (k-RIP) (e.g., Gaussian matrix and Bernoulli matrix).

The SLEM Approach (5/8)



- **Training:** Learn multi-output regression model $h(\cdot)$ with algorithm $\mathcal{M}$

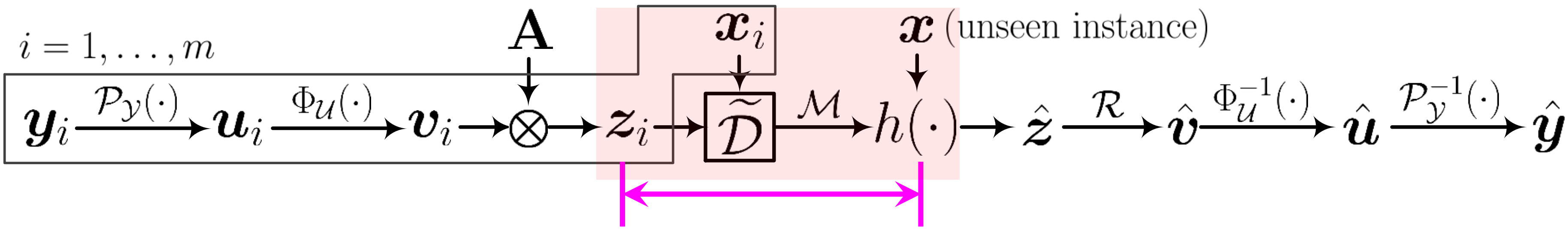
• Train predictive model over $\widetilde{\mathcal{D}} = \{(\boldsymbol{x}_i, \boldsymbol{z}_i) \mid 1 \leq i \leq m\}$.

• Learn multi-output regression model $h(\boldsymbol{x}) = \mathbf{W}^\top \boldsymbol{x} + \boldsymbol{b}$ via optimizing the following formulation:

$$\min_{\mathbf{W}, \boldsymbol{b}, \hat{\mathbf{V}}} \ \frac{1}{2} \|\mathbf{W}\|_F^2 + \lambda \sum_{i=1}^{m} \left[\|h(\boldsymbol{x}_i) - \boldsymbol{z}_i\|_2^2 + \gamma_1 \left(\|h(\boldsymbol{x}_i) - \mathbf{A}\hat{\boldsymbol{v}}_i\|_2^2 + \gamma_2 \|\hat{\boldsymbol{v}}_i - \boldsymbol{v}_i\|_1 \right) \right]$$

Here, $\mathbf{W} = [\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{s'}] \in \mathbb{R}^{d \times s'}$ and $\boldsymbol{b} = [b_1, b_2, \ldots, b_{s'}]^\top$ are the model parameters of h to be determined, $\hat{\mathbf{V}} = [\hat{\boldsymbol{v}}_1, \ldots, \hat{\boldsymbol{v}}_m]^\top \in \mathbb{R}^{m \times s}$ with $\hat{\boldsymbol{v}}_i$ corresponding to the recovered sparse vector for $\boldsymbol{v}_i$ based on its prediction $h(\boldsymbol{x}_i)$, and λ, γ_1 and γ_2 are three trade-off parameters.

The SLEM Approach (5/8)



- **Training:** Learn multi-output regression model $h(\cdot)$ with algorithm $\mathcal{M}$

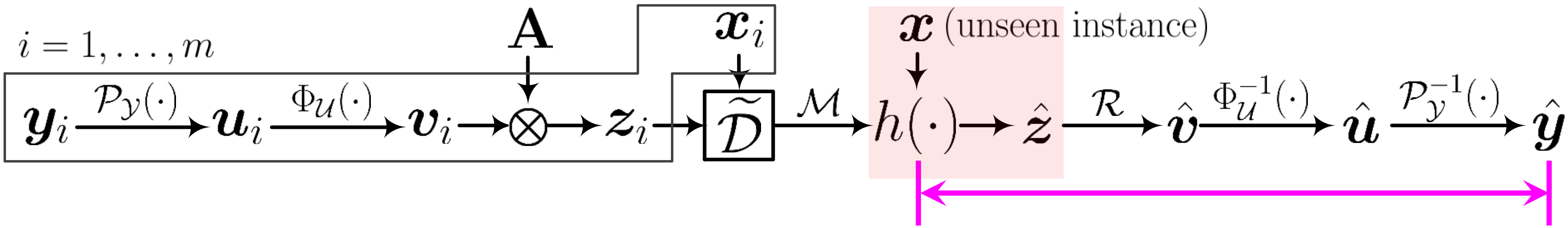
- Train predictive model over $\widetilde{\mathcal{D}} = \{(\boldsymbol{x}_i, \boldsymbol{z}_i) \mid 1 \leq i \leq m\}$.
- ...ou...on ... mizi... the following form...ation:

regularizer **Square loss** **Facilitate sparse reconstruction**

$$\min_{\mathbf{W}, \boldsymbol{b}, \hat{\mathbf{V}}} \frac{1}{2}\|\mathbf{W}\|_F^2 + \lambda \sum_{i=1}^{m} \left[\|h(\boldsymbol{x}_i) - \boldsymbol{z}_i\|_2^2 + \gamma_1 \left(\|h(\boldsymbol{x}_i) - \mathbf{A}\hat{\boldsymbol{v}}_i\|_2^2 + \gamma_2 \|\hat{\boldsymbol{v}}_i - \boldsymbol{v}_i\|_1 \right) \right]$$

Here, $\mathbf{W} = [\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{s'}] \in \mathbb{R}^{d \times s'}$ and $\boldsymbol{b} = [b_1, b_2, \ldots, b_{s'}]^\top$ are the model parameters of h to be determined, $\hat{\mathbf{V}} = [\hat{\boldsymbol{v}}_1, \ldots, \hat{\boldsymbol{v}}_m]^\top \in \mathbb{R}^{m \times s}$ with $\hat{\boldsymbol{v}}_i$ corresponding to the recovered sparse vector for $\boldsymbol{v}_i$ based on its prediction $h(\boldsymbol{x}_i)$, and λ, γ_1 and γ_2 are three trade-off parameters.

The SLEM Approach (6/8)



- **Testing:** Obtain the real-valued prediction $\hat{\boldsymbol{z}} = h(\boldsymbol{x})$ for unseen instance $\boldsymbol{x}$

Suppose that $\boldsymbol{y}$ is the ground-truth class vector of $\boldsymbol{x}$, then $\hat{\boldsymbol{z}} = h(\boldsymbol{x})$ should correspond to the prediction of $\boldsymbol{z}$:

$$\boldsymbol{y} \xrightarrow{\mathcal{P}_{\mathcal{Y}}(\cdot)} \boldsymbol{u} \xrightarrow{\Phi_{\mathcal{U}}(\cdot)} \boldsymbol{v} \to \underset{\uparrow\,\mathbf{A}}{\otimes} \to \boldsymbol{z}$$

The decoding phase corresponds to the inverse operations of encoding phase to obtain $\boldsymbol{x}$'s predicted class vector $\hat{\boldsymbol{y}}$ based on $\hat{\boldsymbol{z}}$.

Note that $\boldsymbol{v}$ is k-sparse, and there is one and only one '1' in each local group (i.e., local sparsity).

The SLEM Approach (7/8)

$i = 1, \ldots, m$

$\mathbf{A}$, $\boldsymbol{x}_i$, $\boldsymbol{x}$ (unseen instance)

$$\boldsymbol{y}_i \xrightarrow{\mathcal{P}_{\mathcal{Y}}(\cdot)} \boldsymbol{u}_i \xrightarrow{\Phi_{\mathcal{U}}(\cdot)} \boldsymbol{v}_i \rightarrow \otimes \rightarrow \boldsymbol{z}_i \rightarrow \boxed{\widetilde{\mathcal{D}}} \xrightarrow{\mathcal{M}} h(\cdot) \rightarrow \hat{\boldsymbol{z}} \xrightarrow{\mathcal{R}} \hat{\boldsymbol{v}} \xrightarrow{\Phi_{\mathcal{U}}^{-1}(\cdot)} \hat{\boldsymbol{u}} \xrightarrow{\mathcal{P}_{\mathcal{Y}}^{-1}(\cdot)} \hat{\boldsymbol{y}}$$

- Inverse of sparse label encoding

• Complete sparse reconstruction and keep local sparsity.

• Adapt the well-known orthogonal matching pursuit (OMP) algorithm to consider the local sparsity property.

The OMP algorithm recovers a k-sparse vector by choosing the column of $\mathbf{A}$ that is most strongly correlated with the residual at each iteration and repeating this procedure k times.

Our key adaption is to set the related columns to zero which belong to the same local group with the current selected column of $\mathbf{A}$.

The S

$i = 1, \dots, m$

$\boldsymbol{y}_i \xrightarrow{\mathcal{P}_{\mathcal{Y}}(\cdot)} \boldsymbol{u}_i$ … $\rightarrow \hat{\boldsymbol{u}} \xrightarrow{\mathcal{P}_{\mathcal{Y}}^{-1}(\cdot)} \hat{\boldsymbol{y}}$

Algorithm 2 LOMP: $\boldsymbol{v} = \mathcal{R}(\boldsymbol{z}, \mathbf{A}, k, \mathcal{I})$

Input: The encoding matrix $\mathbf{A} \in \mathbb{R}^{s' \times s}$, the real-valued vector $\boldsymbol{z} \in \mathbb{R}^{s'}$, sparsity level k, local sparsity information $\mathcal{I}: s_1, s_2, \dots, s_k$;

Output: The recovered k-sparse vector $\boldsymbol{v}$;

Initialize $\boldsymbol{v}$ as zero vector with length $s = \sum_{j=1}^{k} s_j$;
Initialize $\boldsymbol{r}_0 = \boldsymbol{z}$, $J = \varnothing$, $\mathbf{B} = \mathbf{A}$;
for $i = 1$ to k **do**
  $j_* = \arg\max_j |\langle \boldsymbol{r}_{i-1}, \mathbf{B}_{:j} \rangle|$;
  $\boldsymbol{v}(j_*) = 1$;
  $J = J \cup \{j_*\}$;
  $\boldsymbol{r}_i = \boldsymbol{z} - \mathbf{A}_{:J}(\mathbf{A}_{:J}^{\top}\mathbf{A}_{:J})^{-1}\mathbf{A}_{:J}^{\top}\boldsymbol{z}$;
  for $\kappa = 1$ to k **do**
    $t_f = \sum_{t=1}^{\kappa} s_t$;
    if $j_* \le t_f$ **then**
      $t_b = t_f - s_\kappa$;
      $T = \{t_b + 1, t_b + 2, \dots, t_f\}$;
      $\mathbf{B}_{:T} = 0$;
      break;
    end if
  end for
end for
Return $\boldsymbol{v}$.

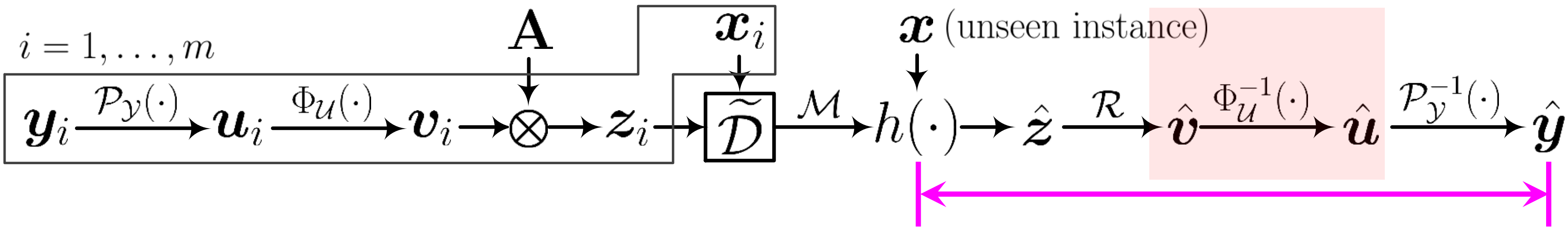
☐ Inverse

- Complet
- Adapt th … P) algorithm to c

The OMP … sing the column of … sidual at each iterat

Our key ac … h belong to the sam … of $\mathbf{A}$.

The SLEM Approach (8/8)

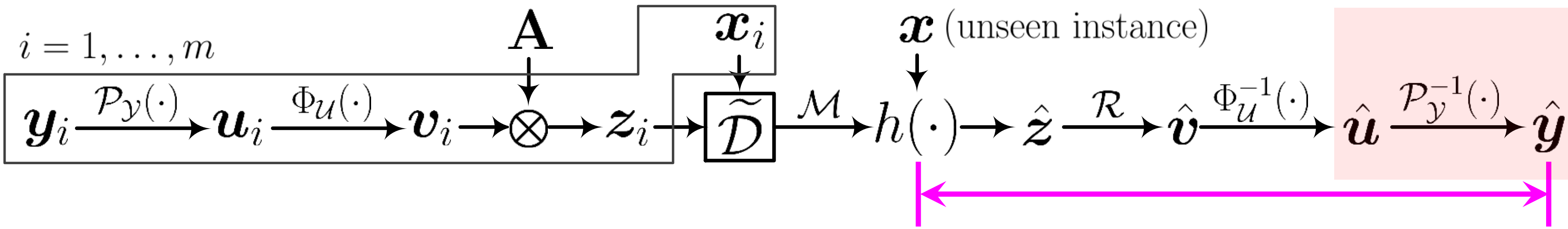


- Inverse of one-hot conversion

$$C'_1 = \{a, b, c, d\},\ C'_2 = \{\text{I, II, III, IV, V, VI}\},\ C'_3 = \{\alpha, \beta, \gamma\} \qquad (k = 3)$$

$$\boldsymbol{v}_i = [0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1]^\top \Rightarrow \boldsymbol{u}_i = [b, \text{III}, \gamma]^\top \qquad (k\text{-sparse})$$

The SLEM Approach (8/8)



- Inverse of one-hot conversion

$$C_1' = \{a, b, c, d\},\ C_2' = \{\text{I, II, III, IV, V, VI}\},\ C_3' = \{\alpha, \beta, \gamma\} \qquad (k = 3)$$

$$\boldsymbol{v}_i = [0,1,0,0 \,|\, 0,0,1,0,0,0 \,|\, 0,0,1]^\top \Rightarrow \boldsymbol{u}_i = [b, \text{III}, \gamma]^\top \qquad (k\text{-sparse})$$

- Inverse of pairwise grouping

$$C_1' \leftarrow C_5 \times C_3$$
$$C_2' \leftarrow C_2 \times C_1$$
$$C_3' \leftarrow C_4$$

$$C_1 = \{c_1^1, c_2^1, c_3^1, c_4^1\}$$
$$C_2 = \{c_1^2, c_2^2, c_3^2\}$$
$$C_3 = \{c_1^3, c_2^3, c_3^3, c_4^3, c_5^3\}$$
$$C_4 = \{c_1^4, c_2^4, c_3^4, c_4^4, c_5^4, c_6^4\}$$
$$C_5 = \{c_1^5, c_2^5\}$$

Outline

- Introduction
- The proposed SLEM Approach
- **Experiments**
  - **Experimental setup**
  - **Experimental results**
- Conclusion

Experimental Setup

Experimental data sets

Characteristics of the experimental MDC data sets.

Data Set	#Exam.	#Dim.	#Labels/Dim.	#Features†
Jura	359	2	4,5	$9n$
Oes10	403	16	3	$298n$
Voice	3136	2	4,2	$19n$
Scm20d	8966	16	4	$61n$
Rf1	8987	8	4,4,3,4,4,3,4,3	$64n$
Scm1d	9803	16	4	$280n$
CoIL2000	9822	5	6,10,10,4,2	$81x$
Flickr	12198	5	3,4,3,4,4	$1536n$
Disfa	13095	12	5,5,6,3,4,4,5,4,4,4,6,4	$136n$
Fera	14052	5	6	$136n$
Adult	18419	4	7,7,5,2	$5n$,$5x$

† n, x denote numeric and nominal features respectively.

Experimental Setup

Evaluation Metrics

Testing set: $\mathcal{S} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i) \mid 1 \leq i \leq p\}$, where $\boldsymbol{y}_i = [y_{i1}, y_{i2}, \ldots, y_{iq}]^\top$

Predicted class vector: $\hat{\boldsymbol{y}}_i = f(\boldsymbol{x}_i) = [\hat{y}_{i1}, \hat{y}_{i2}, \ldots, \hat{y}_{iq}]^\top$

For each MDC test example $(\boldsymbol{x}_i, \boldsymbol{y}_i)$: $r^{(i)} = \sum_{j=1}^{q} [\![y_{ij} = \hat{y}_{ij}]\!]$

Hamming Score: $$\mathrm{HS}_{\mathcal{S}}(f) = \frac{1}{p}\sum_{i=1}^{p}\frac{1}{q}\cdot r^{(i)}$$

Exact Match: $$\mathrm{EM}_{\mathcal{S}}(f) = \frac{1}{p}\sum_{i=1}^{p}[\![r^{(i)} = q]\!]$$

Sub-Exact Match: $$\mathrm{SEM}_{\mathcal{S}}(f) = \frac{1}{p}\sum_{i=1}^{p}[\![r^{(i)} \geq q-1]\!]$$

Experimental Setup

Compared Algorithms

BR: Learn q independent multi-class classifier, one per dimension

CP: Learn a single multi-class classifier via powerset transformation

BCC: Learn q chain-structured multi-class classifiers, one per dimension

ESC: Group the class variables into groups

gMML: Learn a regressor for each class label as well as a Mahalanobis distance metric to train all regressor in a joint manner

Experimental Protocol

Ten-fold cross-validation + Pairwise t-test

Experimental Results

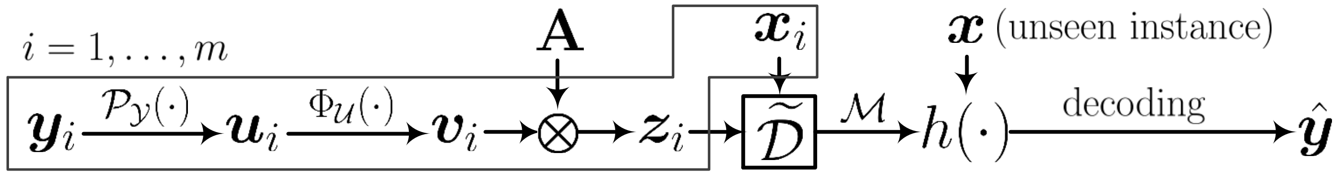
Win/tie/loss counts of pairwise t-test (at 0.05 significance level) between SLEM and each MDC approach.

Evaluation metric	SLEM against BR	CP	BCC	ESC	gMML
HS	9/1/1	7/0/0	10/1/0	8/0/0	8/0/3
EM	10/1/0	5/2/0	10/1/0	7/1/0	9/1/1
SEM	7/2/2	5/1/1	8/2/1	6/1/1	5/4/2
In Total	**26/4/3**	**17/3/1**	**28/4/1**	**21/2/1**	**22/5/6**

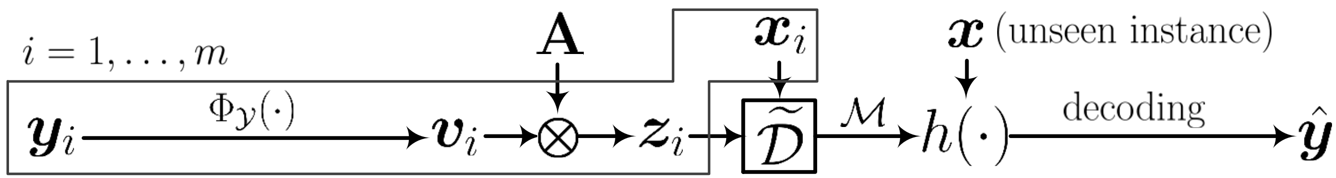
Among all the 144 configurations, SLEM achieves superior or at least comparable performance against the five compared approaches in 132 cases.

Further Analysis (1/2)

SLEM:



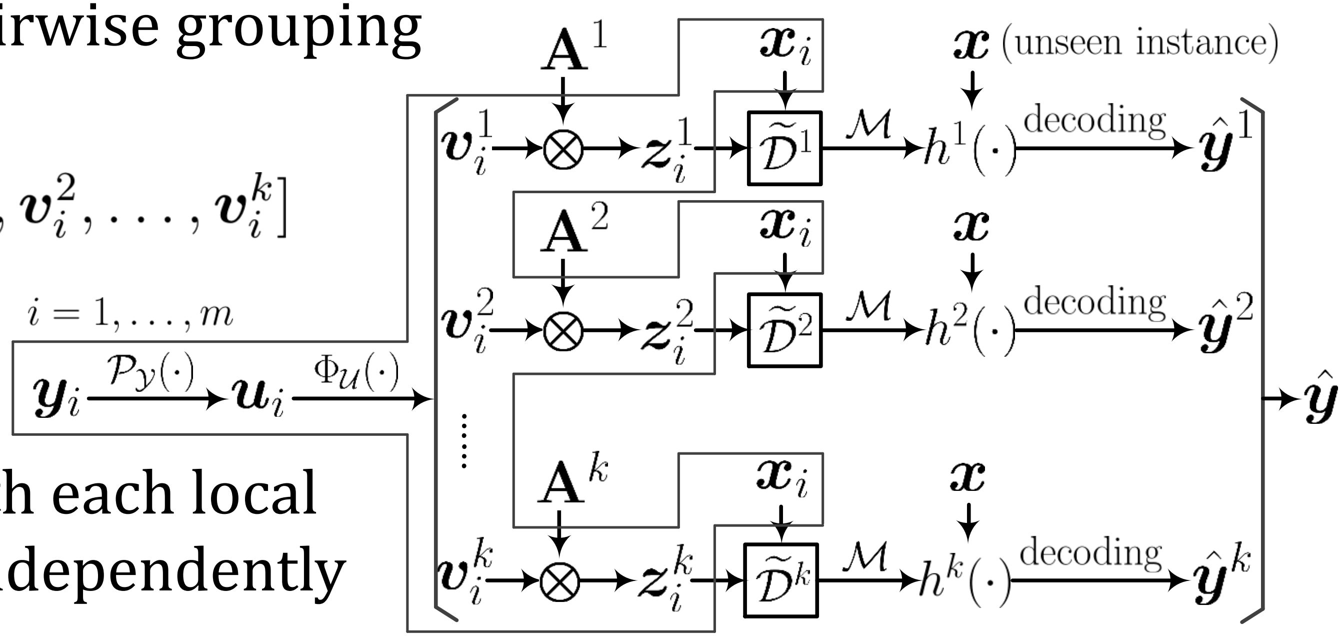
DeV1:



Omit pairwise grouping

$$\boldsymbol{v}_i = [\boldsymbol{v}_i^1, \boldsymbol{v}_i^2, \ldots, \boldsymbol{v}_i^k]$$
$$k = \lceil \tfrac{q}{2} \rceil$$

DeV2:



Deal with each local group independently

Further Analysis (2/2)

Wilcoxon signed-ranks test for SLEM against its two degenerated versions in terms of each evaluation metric (significance level $\alpha = 0.05$; p-values shown in the brackets).

SLEM	Evaluation metric		
versus	HS	EM	SEM
DeV1	**win**[9.77e-04]	**win**[9.77e-04]	**win**[9.77e-04]
DeV2	**win**[3.91e-03]	**win**[3.91e-03]	**win**[3.91e-03]

- The superiority of SLEM against DeV1 shows the benefits of the pairwise grouping operation.
- The superiority of SLEM against DeV2 shows the benefits of encoding the heterogeneous class spaces into an integrated one.

Outline

- Introduction
- The proposed SLEM Approach
- Experiments
  - Experimental setup
  - Experimental results
- **Conclusion**

Conclusion

- Different from most existing MDC approaches, we propose a first attempt towards learning predictive models in the transformed label space instead of the original one.

- We design a novel MDC approach named SLEM which works in an *encoding-training-decoding* framework by utilizing the sparse property of the transformed label space.

- Experimental results clearly validate the superiority of SLEM against state-of-the-art MDC approaches.

Thanks！

http://palm.seu.edu.cn/zhangml/files/SLEM.rar

Email: jiabb@seu.edu.cn