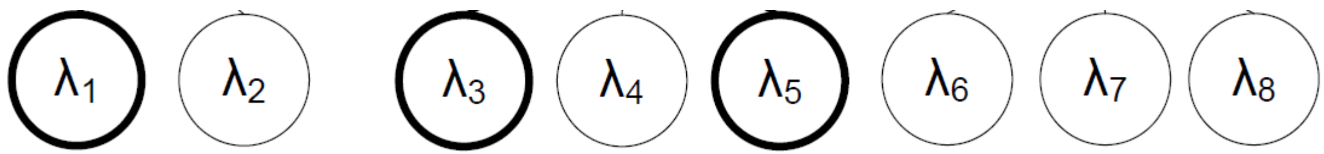
Decomposition and structuring of the output space in multi-label classification

Gjorgji Madjarov

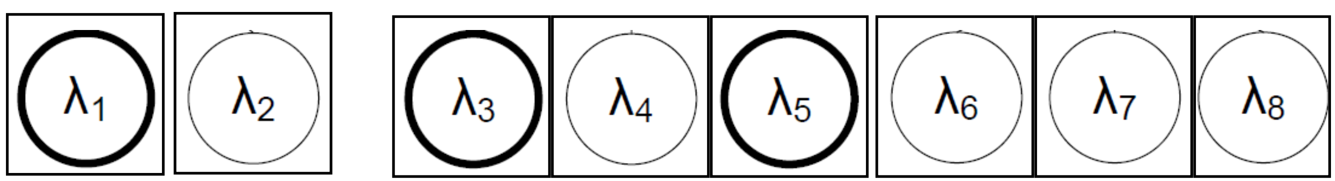
MAESTRA
LEARNING FROM MASSIVE, INCOMPLETELY ANNOTATED, AND STRUCTURED DATA

What is a decomposition of the output space?

- The output space in multi-label learning



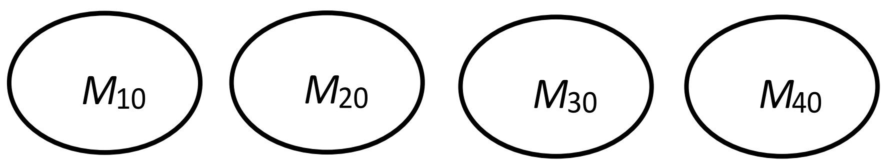
- A global model predict all label at once
- A decomposition of the above multi-label problem



- A set of local models predict one label each
  - multi-label problem decomposed into several single-label problems

Binary relevance methods

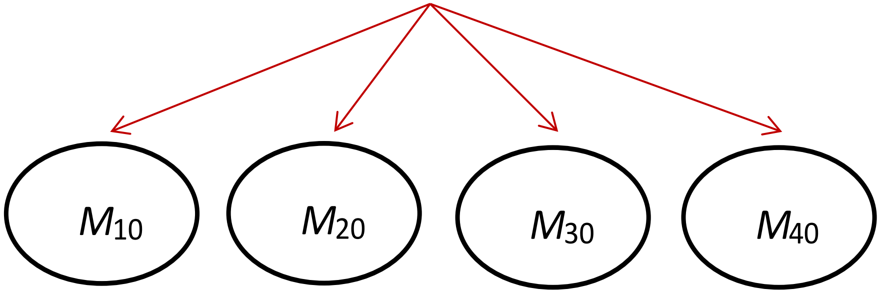
- Binary relevance (***BR***)



Binary relevance methods

- Binary relevance (***BR***)

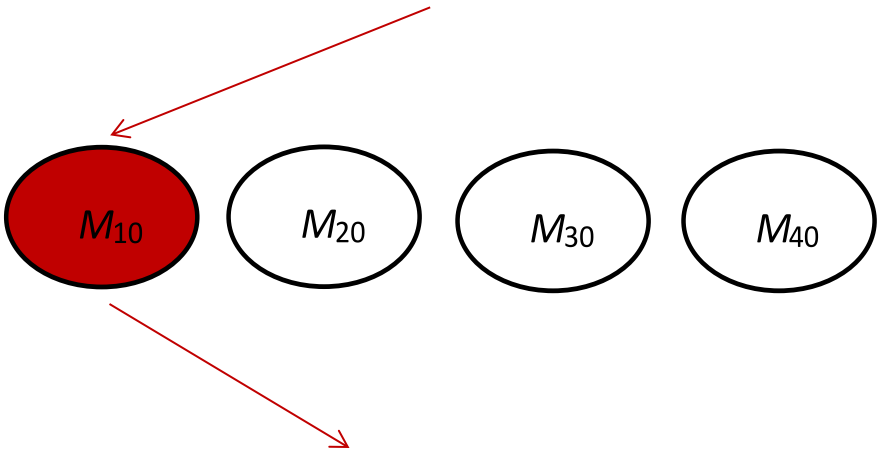
$$\mathrm{x}_{\mathrm{i}} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$



Binary relevance methods

- Classifier chains (***CC***)

$$\mathrm{x}_{\mathrm{i}} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$

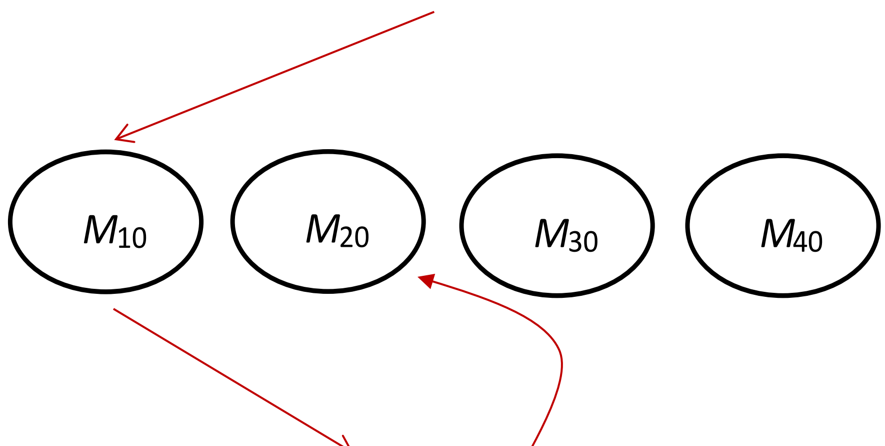


$$\mathrm{x}_{\mathrm{i}} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}, BP_{M_{10}}\}$$

Binary relevance methods

- Classifier chains (***CC***)

$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$

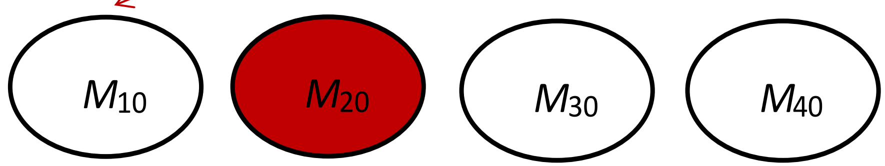


$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}, BP_{M_{10}}\}$$

Binary relevance methods

- Classifier chains (***CC***)

$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$

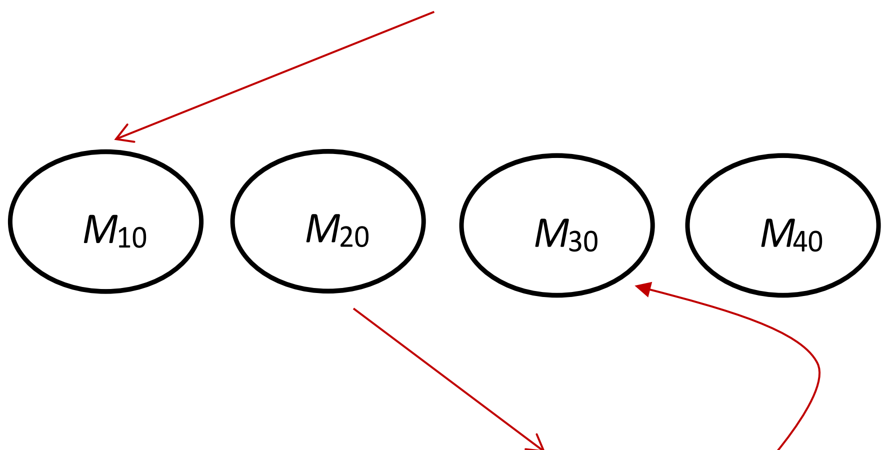


$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}, BP_{M_{10}}, BP_{M_{20}}\}$$

Binary relevance methods

- Classifier chains (***CC***)

$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$

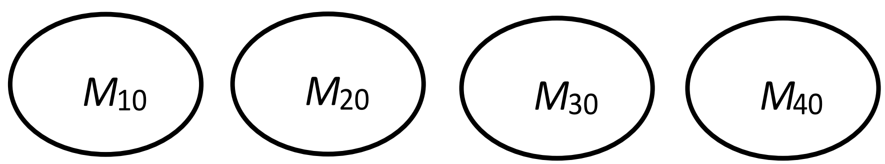


$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}, BP_{M_{10}}, BP_{M_{20}}\}$$

Binary relevance methods

- Classifier chains (***CC***)

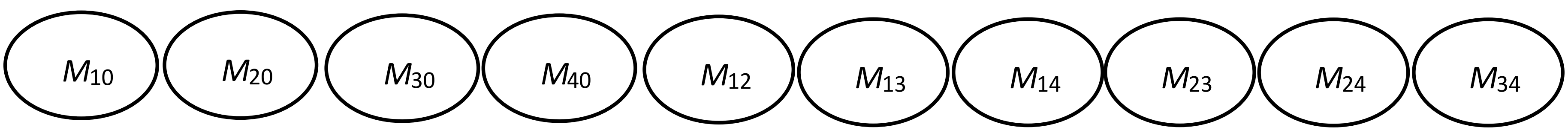
$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$



$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}, BP_{M_{10}}, BP_{M_{20}}, BP_{M_{30}}, BP_{M_{40}}\}$$

Pairwise methods

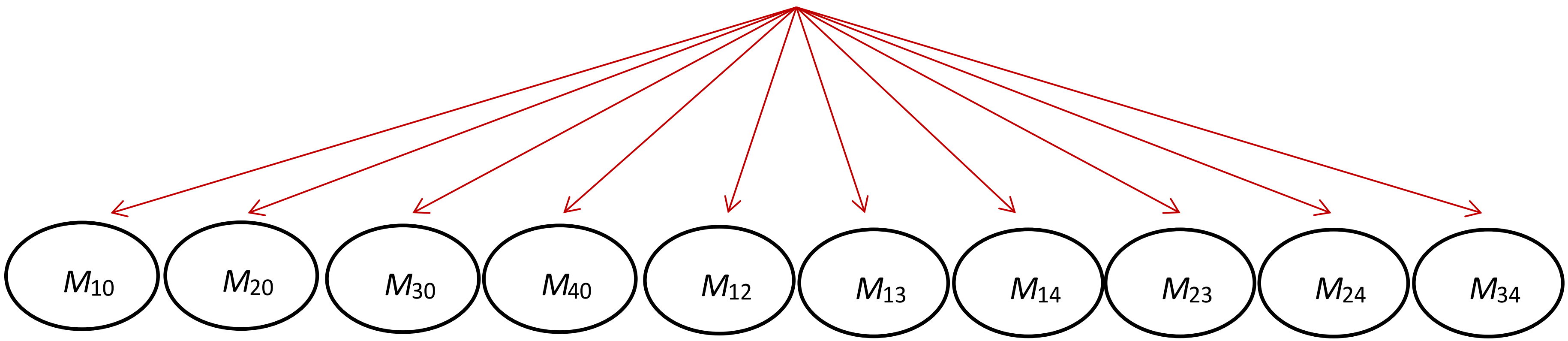
- Calibrated label ranking (***CLR***)



Pairwise methods

- Calibrated label ranking (***CLR***)

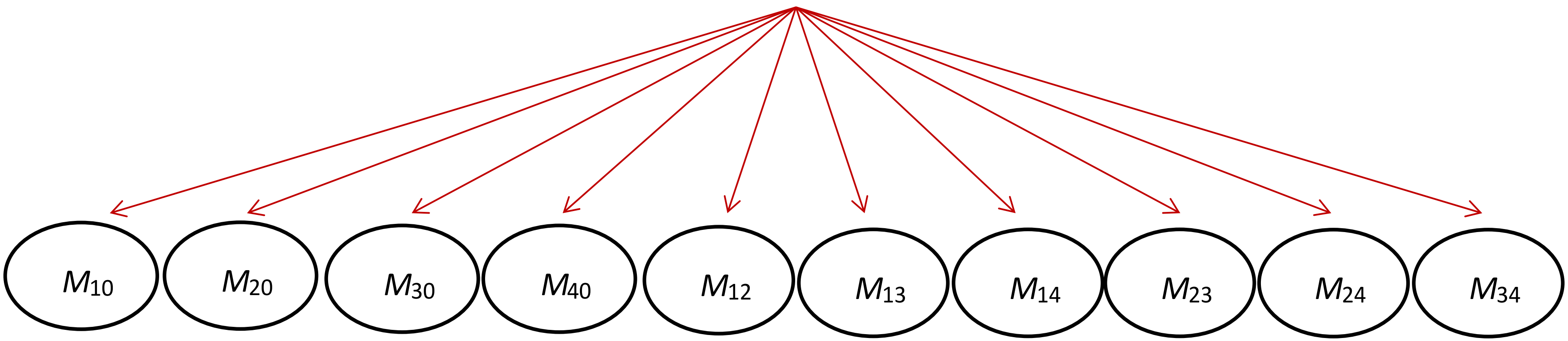
$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$



Pairwise methods

- Calibrated label ranking (***CLR***)

$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$



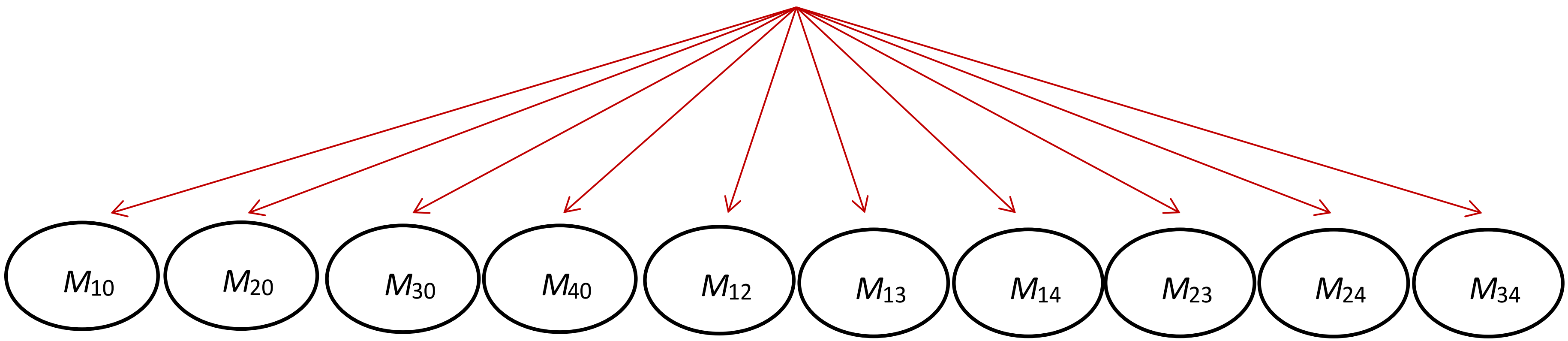
Labels

2 4 0 3 1

Pairwise methods

- Calibrated label ranking (***CLR***)

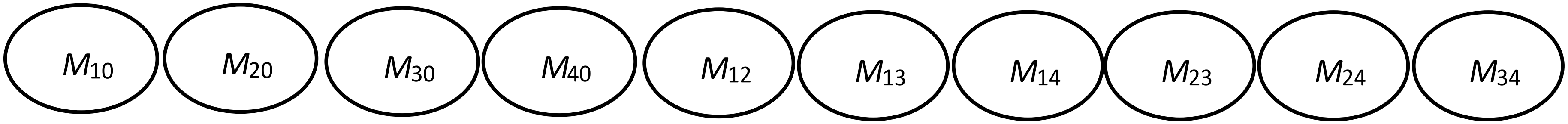
$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$



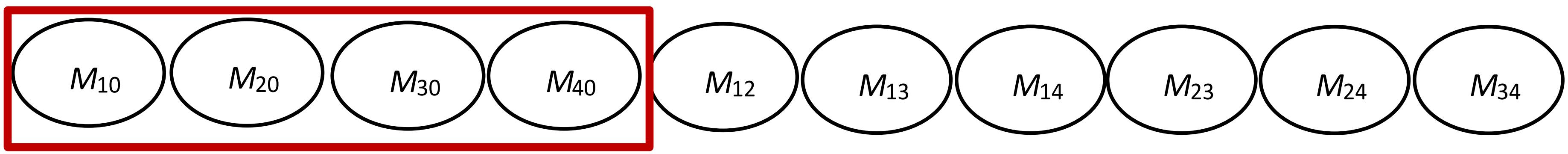
Labels

2 4 **0** 3 1

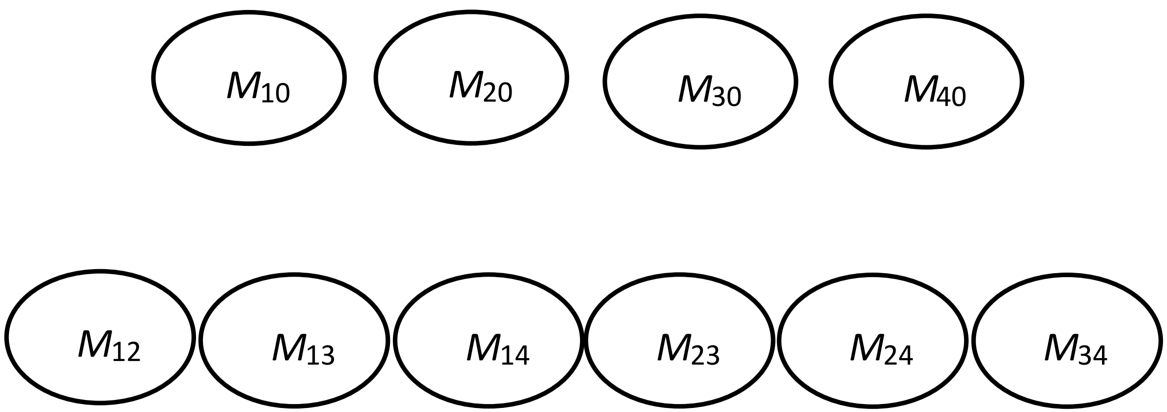
Two stage architecture



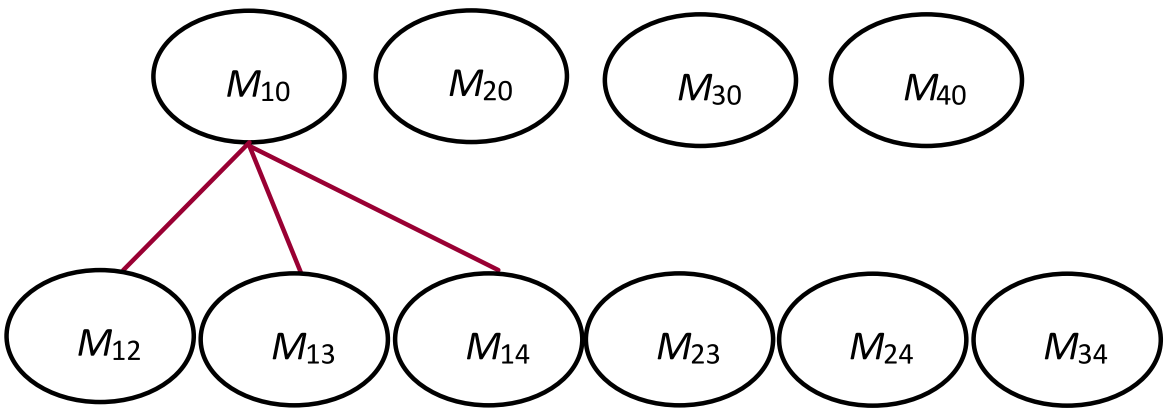
Two stage architecture



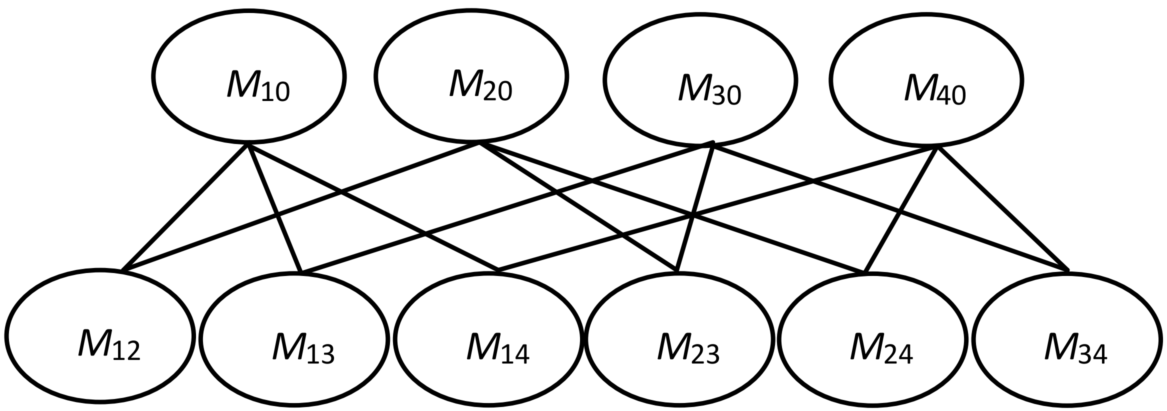
Two stage architecture



Two stage architecture

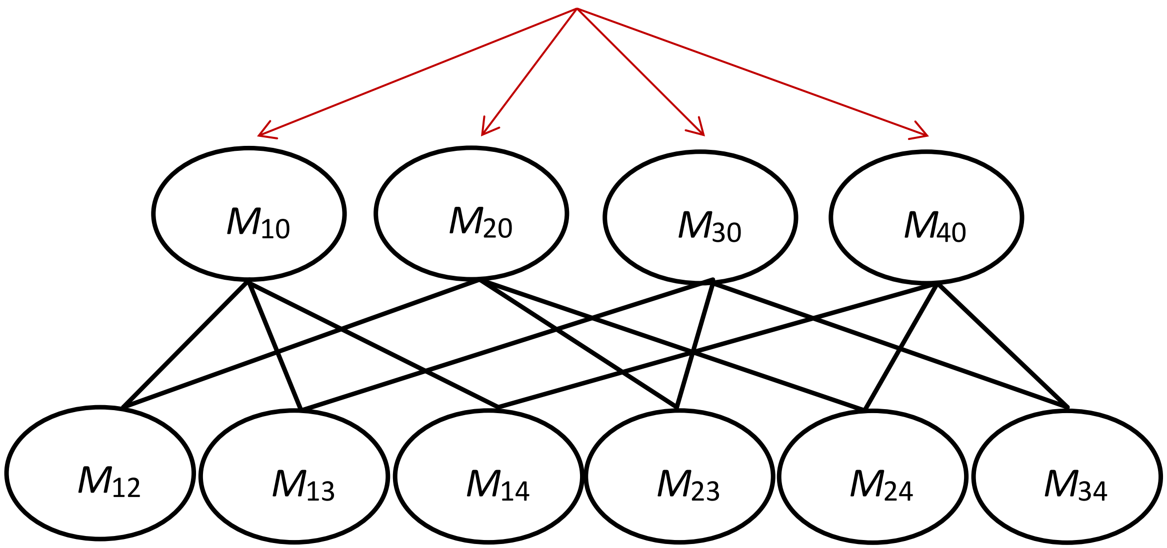


Two stage architecture



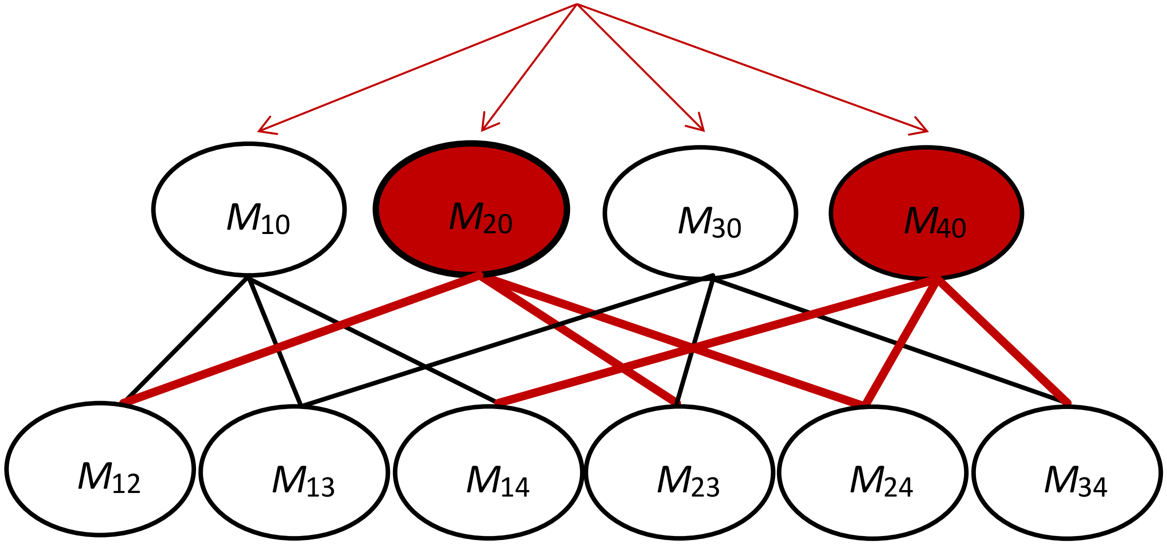
Two stage architecture

$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$



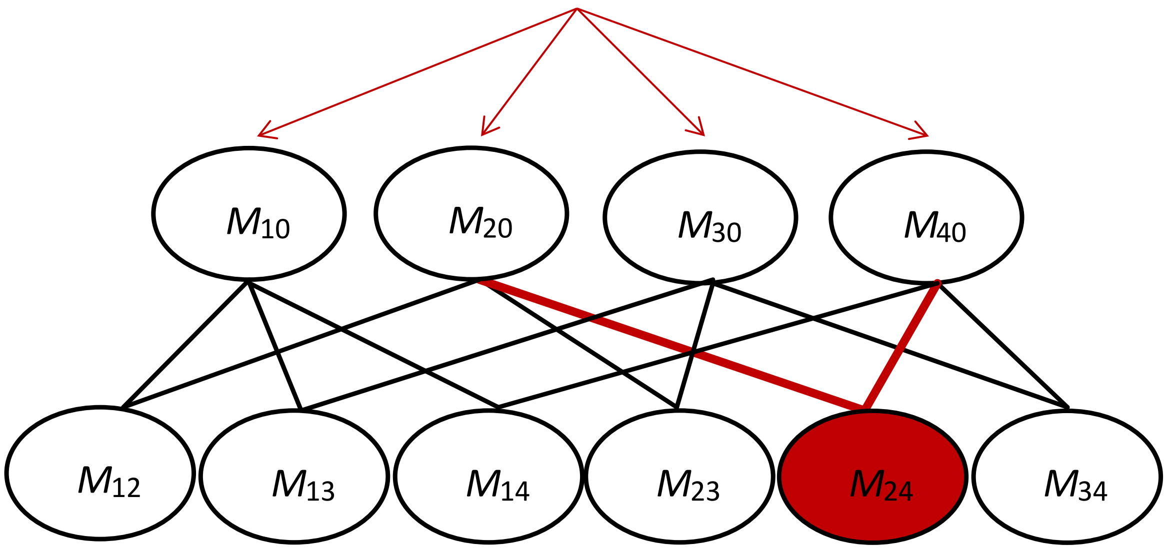
Two stage architecture

$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$



Two stage architecture

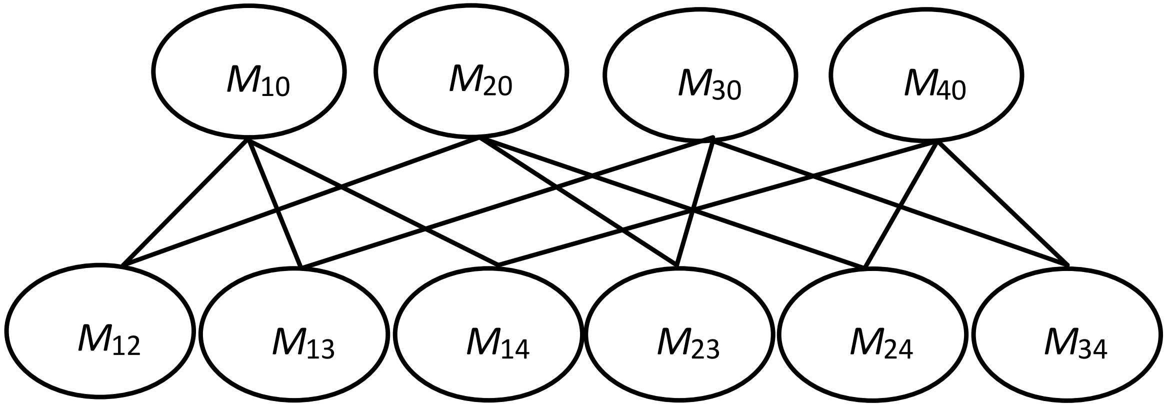
$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$



Labels

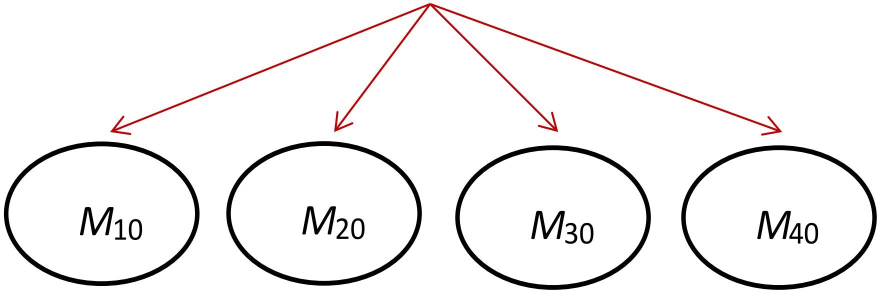
2 4 **0** 3 1

Two stage classifier chains architecture



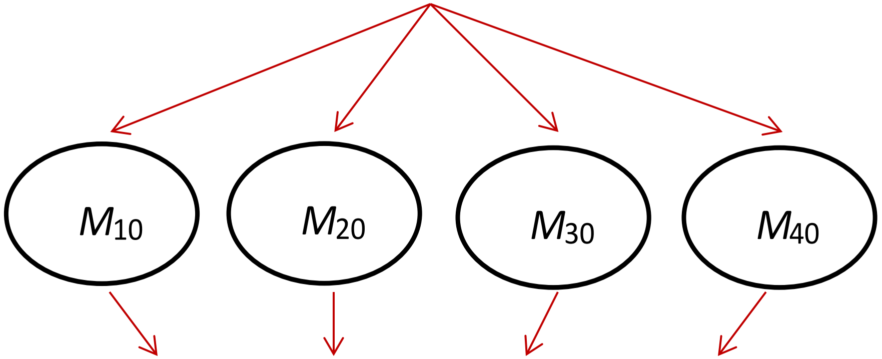
Two stage classifier chains architecture

$$\mathrm{x}_{\mathrm{i}} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$



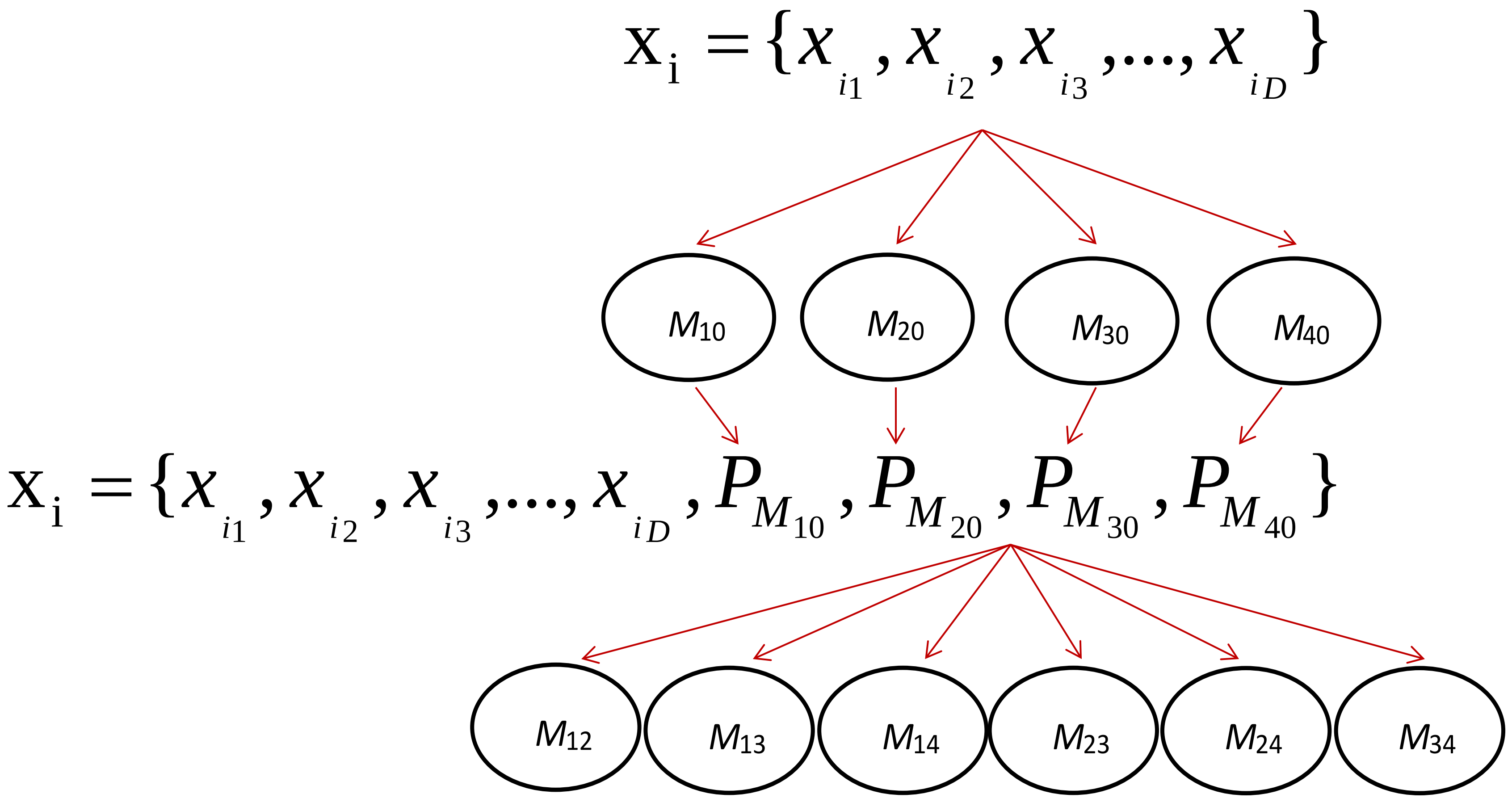
Two stage classifier chains architecture

$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$



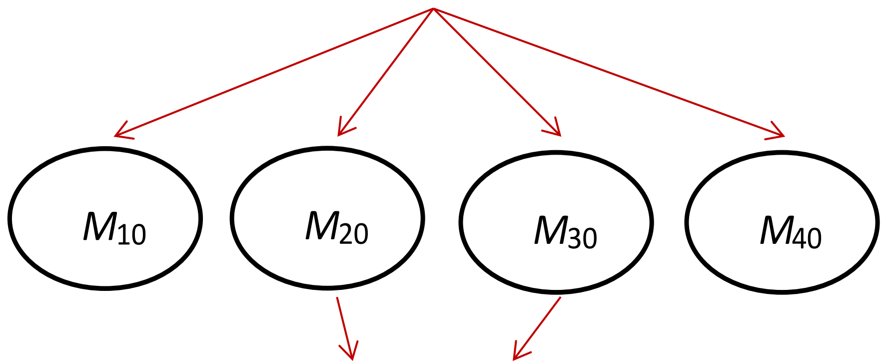
$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}, P_{M_{10}}, P_{M_{20}}, P_{M_{30}}, P_{M_{40}}\}$$

Two stage classifier chains architecture

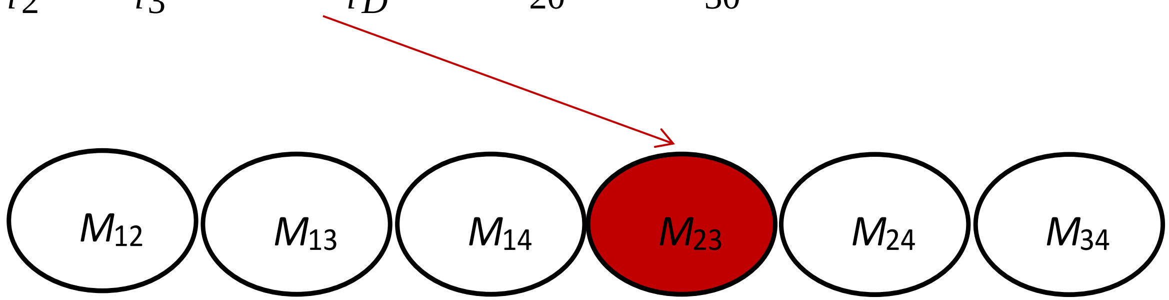


Two stage classifier chains architecture

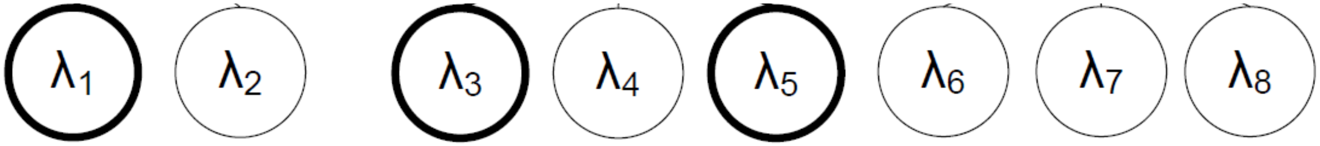
$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}\}$$



$$\mathrm{x_i} = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}, P_{M_{20}}, P_{M_{30}}\}$$



What is the structuring of the output space?



- Hierarchical multi-label classification
  - A hierarchical structure imposed on the label space

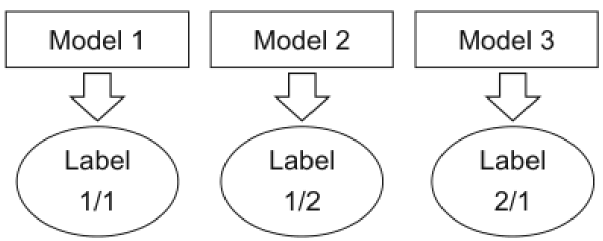
The importance of the label hierarchy in HMC

- The task of learning predictive models for hierarchical multi-label classification is addressed
- Investigation is made on
  - the differences in performance and interpretability of the local and global models
    - whether including information in the form of hierarchical relationships among the labels helps to improve the performance of the predictive models
  - inclusion of the information on the output structure also improves the performance of ensemble models.

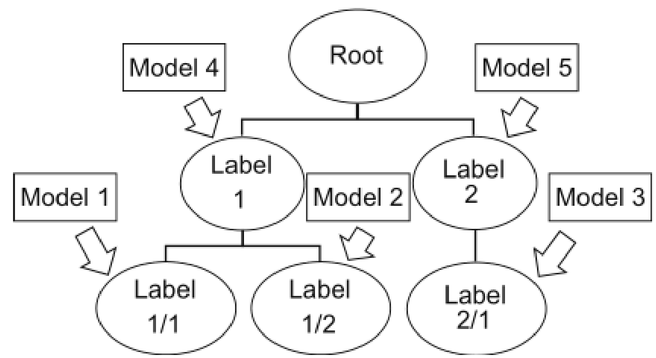
The importance of the label hierarchy in HMC

- Two local and two global modeling tasks that exploit different amounts of the information provided by the label hierarchy were considered

Local approaches

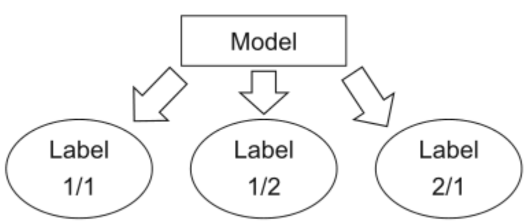


Single-label classification

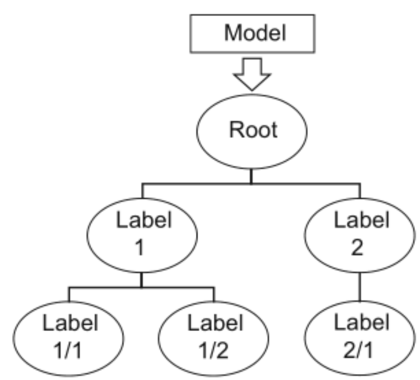


Hierarchical single-label classification (HSC)

Global approaches



Multi-label classification



Hierarchical multi label classification (HMC)

The importance of the label hierarchy in HMC - conclusions

Single tree

- **Label hierarchy improves the predictive performance of single trees**
- HMC trees should be used on domains with well populated label hierarchy
- HSC tree architecture should be used if the number of labels per example is closer to one

Random Forests

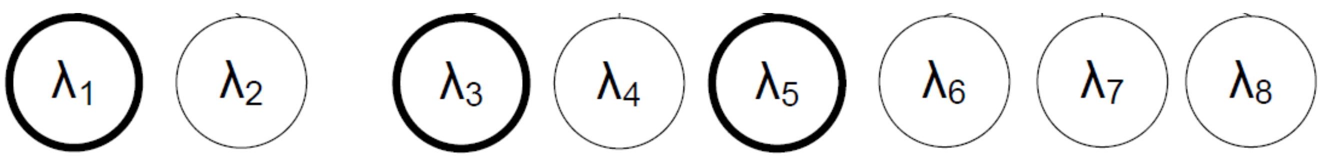
Bagging

- **Label hierarchy brings less (or no) advantage in terms of predictive performance to ensembles**
- However, there are considerable differences in the learning time between global and local ensemble methods
- HMC ensembles are much more efficient in terms of learning time than the single-label ensembles and should be used if time is an issue (especially random forests)

But what if we don't have a structure?

- Derive a structure from the data
  - Input space
  - Output space
  - Combination of the input and output space (no experimental results)

An example of a ML dataset and its transformed HMC dataset.



example	features	original labels
$\mathbf{x_1}$	$x_{1\,1}, x_{1\,2}, \dots, x_{1\,n}$	$\{\lambda_1\}$
$\mathbf{x_2}$	$x_{2\,1}, x_{2\,2}, \dots, x_{2\,n}$	$\{\lambda_3, \lambda_5\}$
$\mathbf{x_3}$	$x_{3\,1}, x_{3\,2}, \dots, x_{3\,n}$	$\{\lambda_6\}$
$\mathbf{x_4}$	$x_{4\,1}, x_{4\,2}, \dots, x_{4\,n}$	$\{\lambda_1, \lambda_6\}$
$\mathbf{x_5}$	$x_{5\,1}, x_{5\,2}, \dots, x_{5\,n}$	$\{\lambda_1, \lambda_2, \lambda_6\}$

Structuring of the output space – output data

- Label hierarchy based on the clustering of occurrence profiles of labels across instances
  - Identifying the relationships between labels by using expert provided information (maybe some features are not relevant for particular problem)
  - Not very relevant if the output space is sparse

Structuring of the output space – output data conclusions

- We have compared four different approaches to deriving label hierarchies
  - balanced k-means
  - hierarchical agglomerative clustering (single and complete linkage)
  - PCTs
- The hierarchies derived by using balanced k-means are clearly better to the ones derived by using the other approaches, yielding the highest improvements in predictive performance

Structuring of the output space – output space conclusions

- We have also compared data-derived hierarchies to expert-provided ones (where such hierarchies are available)

- The results reveal that they have approximately the same utility, i.e., both yield similar improvements in predictive performance

Structuring of the output space – input data

- We construct label hierarchies from the relevance scores of the features for every label
  - Each label from the output space is described (represented) by the relevance scores of the descriptive features for that particular label computed by using Relief
  - Balanced K-means (k=2,3,4,5)

Structuring of the output space – input data conclusions

- Great improvements as compared to the approach that does not use the structured output
- More general approach for structuring the output space (applicable even for multi-class classification problems)
- One extra step
  - Compute the relevance scores of the features for each label in the classification problem

Further work

- Combining the descriptions of the labels that come from (both) the input (relevance score) and the output (co-occurrence relationships) space

- Decomposing the data-derived output space

- Structuring with constraints