ICCS2011
July 25–29, 2011

Semi-Supervised Learning for Mixed-Type Data via Formal Concept Analysis

Mahito SUGIYAMA[†,‡], Akihiro YAMAMOTO[†]

[†]Kyoto University

[‡]JSPS Research Fellow

Summary

- We propose a semi-supervised learning (SSL) method, called **SELF** (SEmi-supervised Learning via FCA), using Formal Concept Analysis (FCA)
  - It can handle mixed-type data containing both discrete and continuous variables
    - Numerical data are discretized by binary encoding

Summary

- We propose a semi-supervised learning (SSL) method, called **SELF** (SEmi-supervised Learning via FCA), using Formal Concept Analysis (FCA)
  - It can handle mixed-type data containing both discrete and continuous variables
    - Numerical data are discretized by binary encoding
- ***Main contributions***
  1. The first direct SSL method for mixed-type data
     - FCA is shown to be a powerful tool for machine learning and knowledge discovery
  2. Can handle incomplete datasets
     - missing values and missing labels
  3. Achieve good accuracy of classification experimentally

Contents

Problems in Knowledge Discovery

- We need to treat more and more massive datasets in KDD
- ***Problems:***
  1. Many datasets are mixed-type: consist of both discrete and continuous variables
  2. Many datasets are incomplete: contain NULL values
     - A missing value: Some value of a datum is missing
     - A missing label: Class label of a datum is missing

Example: The Horse-Colic Dataset in UCI Repository

```
2 1 530101 38.50 66 28 3 3 ? 2 5 4 4 ? ? ? 3 5 45.00 8.40 ? ? 2 2 11300 00000 00000 2
1 1 534817 39.2 88 20 ? ? 4 1 3 4 2 ? ? ? 4 2 50 85 2 2 3 2 02208 00000 00000 2
2 1 530334 38.30 40 24 1 1 3 1 3 3 1 ? ? ? 1 1 33.00 6.70 ? ? 1 2 00000 00000 00000 1
1 9 5290409 39.10 164 84 4 1 6 2 2 4 4 1 2 5.00 3 ? 48.00 7.20 3 5.30 2 1 02208 00000 00000 1
2 1 530255 37.30 104 35 ? ? 6 2 ? ? ? ? ? ? ? ? 74.00 7.40 ? ? 2 2 04300 00000 00000 2
2 1 528355 ? ? ? 2 1 3 1 2 3 2 2 1 ? 3 3 ? ? ? ? 1 2 00000 00000 00000 2
1 1 526802 37.90 48 16 1 1 1 1 3 3 3 1 1 ? 3 5 37.00 7.00 ? ? 1 1 03124 00000 00000 2
1 1 529607 ? 60 ? 3 ? ? 1 ? 4 2 2 1 ? 3 4 44.00 8.30 ? ? 2 1 02208 00000 00000 2
2 1 530051 ? 80 36 3 4 3 1 4 4 4 2 1 ? 3 5 38.00 6.20 ? ? 3 1 03205 00000 00000 2
2 9 5299629 38.30 90 ? 1 ? 1 1 5 3 1 2 1 ? 3 ? 40.00 6.20 1 2.20 1 2 00000 00000 00000 1
1 1 528548 38.10 66 12 3 3 5 1 3 3 1 2 1 3.00 2 5 44.00 6.00 2 3.60 1 1 02124 00000 00000 1
2 1 527927 39.10 72 52 2 ? 2 1 2 1 2 1 1 ? 4 4 50.00 7.80 ? ? 1 1 02111 00000 00000 2
1 1 528031 37.20 42 12 2 1 1 1 3 3 3 3 1 ? 4 5 ? 7.00 ? ? 1 2 04124 00000 00000 2
2 9 5291329 38.00 92 28 1 1 2 1 1 3 2 3 ? 7.20 1 1 37.00 6.10 1 ? 2 2 00000 00000 00000 1
1 1 534917 38.2 76 28 3 1 1 1 3 4 1 2 2 ? 4 4 46 81 1 2 1 1 02112 00000 00000 2
1 1 530233 37.60 96 48 3 1 4 1 5 3 3 2 3 4.50 4 ? 45.00 6.80 ? ? 2 1 03207 00000 00000 2
1 9 5301219 ? 128 36 3 3 4 2 4 4 3 3 ? ? 4 5 53.00 7.80 3 4.70 2 2 01400 00000 00000 1
2 1 526639 37.50 48 24 ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? 1 2 00000 00000 00000 2
1 1 5290481 37.60 64 21 1 1 2 1 2 3 1 1 1 ? 2 5 40.00 7.00 1 ? 1 1 04205 00000 00000 1
2 1 532110 39.4 110 35 4 3 6 ? ? 3 3 ? ? ? ? ? 55 8.7 ? ? 1 2 00000 00000 00000 2
1 1 530157 39.90 72 60 1 1 5 2 5 4 4 3 1 ? 4 4 46.00 6.10 2 ? 1 1 02111 00000 00000 2
2 1 529340 38.40 48 16 1 ? 1 1 1 3 1 2 3 5.50 4 3 49.00 6.80 ? ? 1 2 00000 00000 00000 2
1 1 521681 38.60 42 34 2 1 4 ? 2 3 1 ? ? ? 1 ? 48.00 7.20 ? ? 1 1 03111 00000 00000 2
1 9 534998 38.3 130 60 ? 3 ? 1 2 4 ? ? ? ? ? ? 50 70 ? ? 1 1 03111 00000 00000 2
1 1 533692 38.1 60 12 3 3 3 1 ? 4 3 3 2 2 ? ? 51 65 ? ? 1 1 03111 00000 00000 2
2 1 529518 37.80 60 42 ? ? ? 1 ? ? ? ? ? ? ? ? ? ? ? ? 1 2 00000 00000 00000 2
```

Mixed-Type Datasets

- Lots of mixed-type data with discrete and continuous variables are available for KDD
  - *e.g.* traffic data for intrusion detection, biochemical data, ...
  - Discrete variable: binary (**T**, **F**), nominal (A, B, ..., v)
    - *e.g.*, sex, buying history, questionnaire results, ...
  - Continuous variable: real-valued ($\mathbb{R}$)
    - Mainly obtained by measurement or observation
  - cf. scales of measure (nominal, ordinal, interval, ratio)
- Only few machine learning and KDD methods can directly handle mixed-type data
  - *e.g.* Decision tree-based methods such as C4.5
- Modern efficient and effective KDD method is needed

Incomplete Datasets

- How to treat incomplete datasets with NULL ($\perp$) values is important problem in KDD
  - There are various types of NULL
- We consider the following two types of NULL
  - A missing value: Some value of a datum is missing
  - A missing label: Class label of a datum is missing
- Treat errors of discretized (quantized) real-valued data
  - *e.g.* If a real number $\pi = 3.1415\ldots$ is discretized to 3.1, the subsequent bits become unknown
  - This error is not treated in most methods in (statistical) machine learning and KDD, but there always exist

Problems in Knowledge Discovery

- We need to treat more and more massive datasets in KDD
- ***Problems:***
  1. Many datasets are mixed-type: consist of both discrete and continuous variables
  2. Many datasets are incomplete: contain NULL values
     - A missing value: Some value of a datum is missing
     - A missing label: Class label of a datum is missing

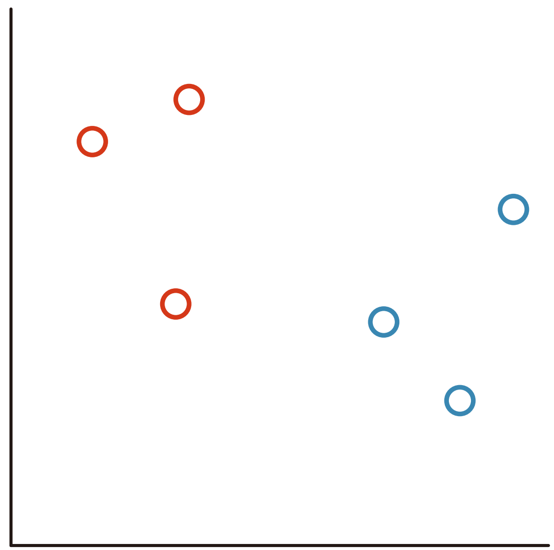
Problems in Knowledge Discovery

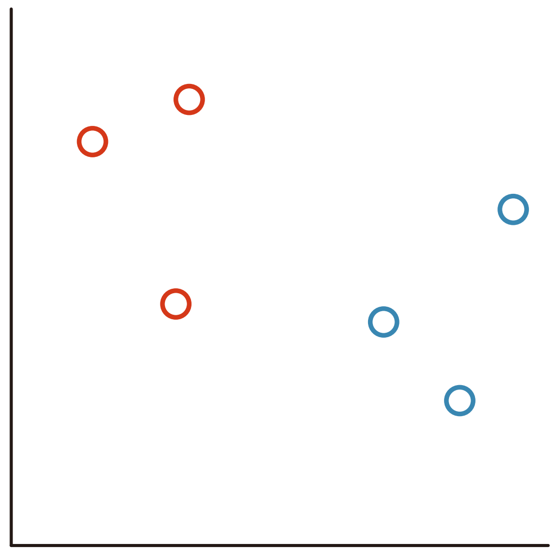
- We need to treat more and more massive datasets in KDD
- ***Problems:***

1. Many datasets are mixed-type: consist of both discrete and continuous variables
2. Many datasets are incomplete: contain NULL values
   - A missing value: Some value of a datum is missing
   - A missing label: Class label of a datum is missing
3. Labeling task costs high (money, time, …)
   - We need class labels to obtain classification rules
   - Yet we have lots of unlabeled data to be analyzed
   - The concept of semi-supervised learning arose in machine learning and KDD

Semi-Supervised Learning

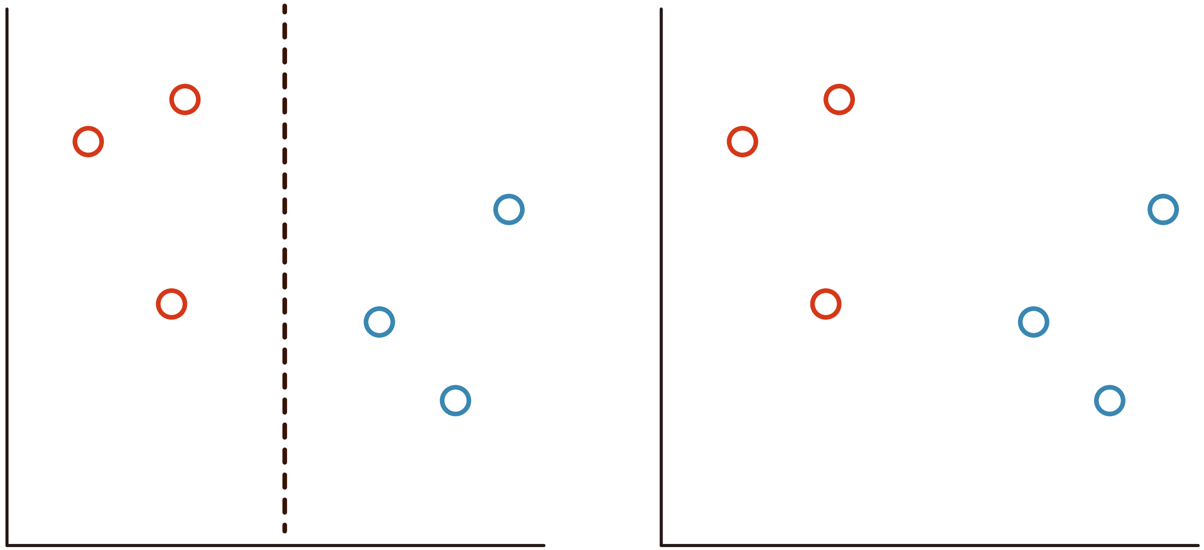
- It is a special form of classification
- Goal: Using (large amount of) unlabeled data effectively, together with (only few) labeled data, build better classifiers [Zhu and Goldberg, 2009]





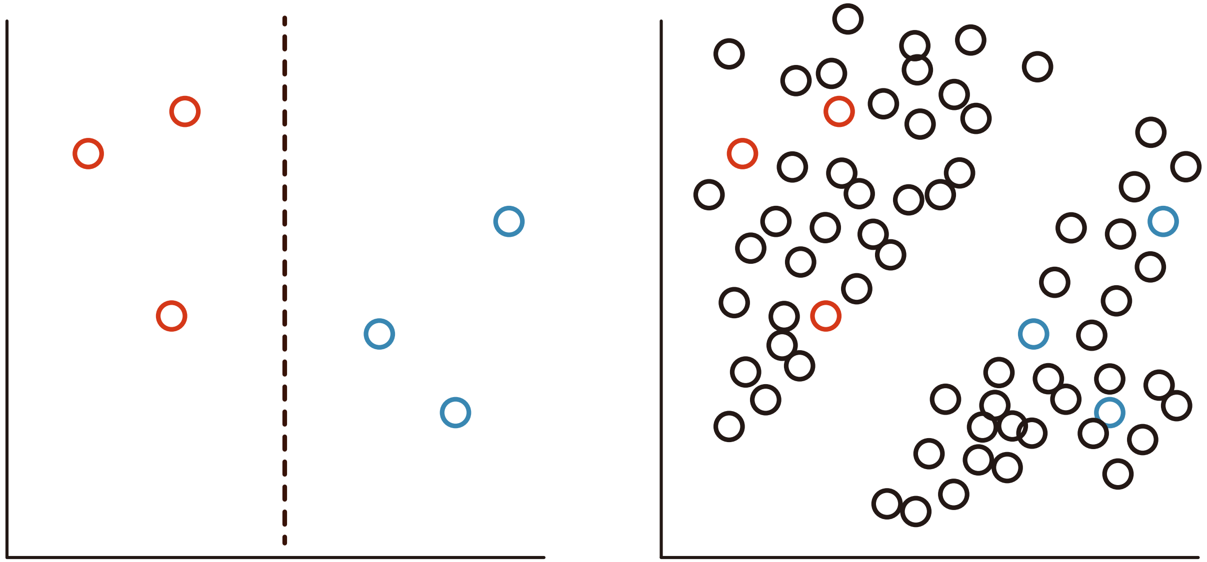
Semi-Supervised Learning

- It is a special form of classification
- Goal: Using (large amount of) unlabeled data effectively, together with (only few) labeled data, build better classifiers [Zhu and Goldberg, 2009]



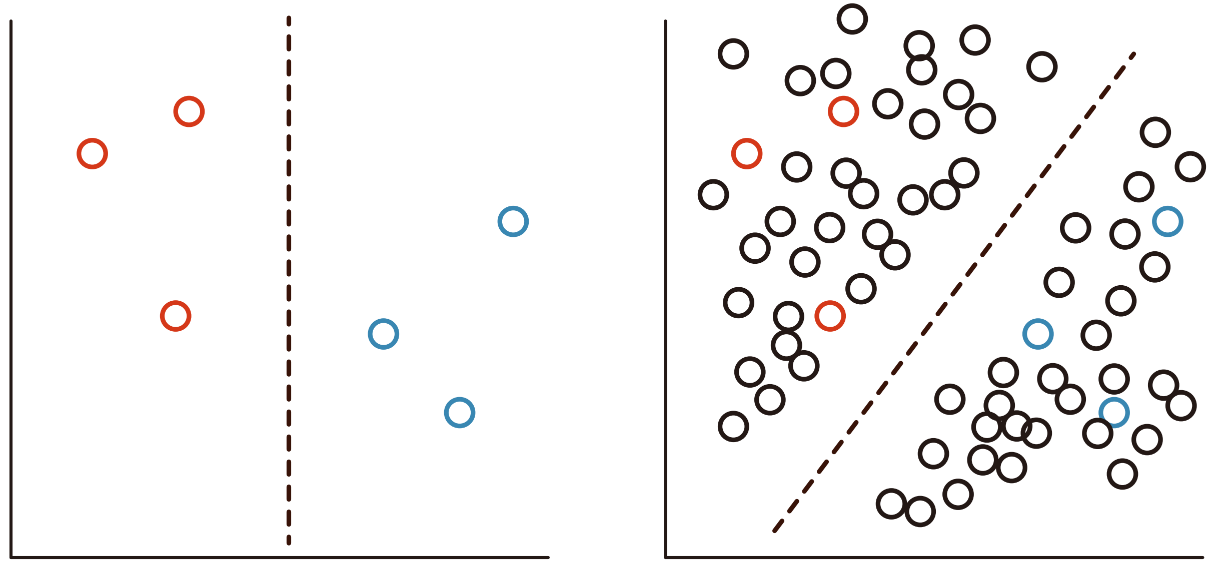
Semi-Supervised Learning

- It is a special form of classification
- Goal: Using (large amount of) unlabeled data effectively, together with (only few) labeled data, build better classifiers [Zhu and Goldberg, 2009]



Semi-Supervised Learning

- It is a special form of classification
- Goal: Using (large amount of) unlabeled data effectively, together with (only few) labeled data, build better classifiers [Zhu and Goldberg, 2009]



Problems in SSL

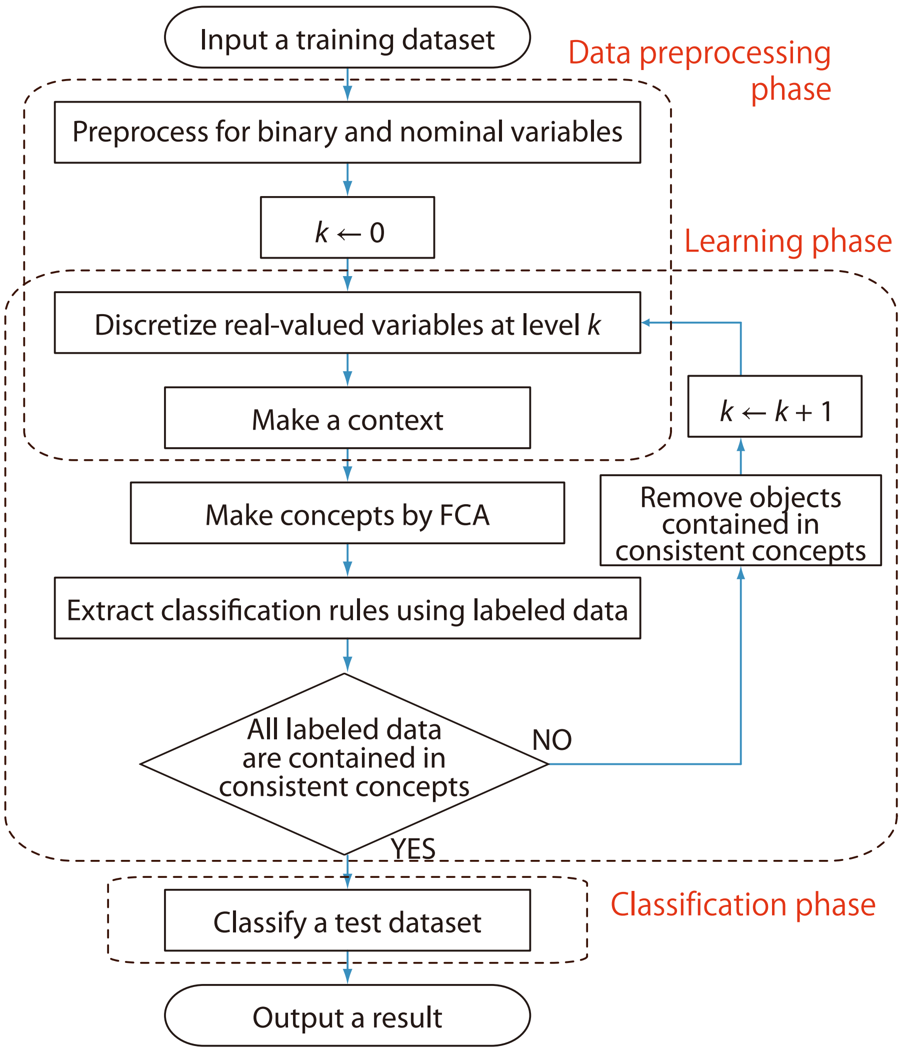
- It is a special form of classification
- Goal: Using (large amount of) unlabeled data effectively, together with (only few) labeled data, build better classifiers [Zhu and Goldberg, 2009]
- However, to date, no SSL method can treat mixed-type datasets directly

Problems in SSL and Our Solution

- It is a special form of classification
- Goal: Using (large amount of) unlabeled data effectively, together with (only few) labeled data, build better classifiers [Zhu and Goldberg, 2009]
- However, to date, no SSL method can treat mixed-type datasets directly
- We solve this problem by using Formal Concept Analysis
  - We present a new SSL method, called SELF, for incomplete mixed-type datasets

Contents

Flowchart of SELF



Contents

Learning by SELF

- Learning from the following dataset ($\perp$ is the missing value)

	Feature 1	Feature 2	Feature 3	Label
1	**T**	C	0.28	1
2	**F**	A	0.54	1
3	**T**	B	$\perp$	$\perp$
4	**F**	A	0.79	2
5	**T**	C	0.81	$\perp$

Data Preprocessing (1/3)

- Learning from the following dataset (⊥ is the missing value)

	Feature 1	Feature 2	Feature 3	Label
1	**T**	C	0.28	1
2	**F**	A	0.54	1
3	**T**	B	⊥	⊥
4	**F**	A	0.79	2
5	**T**	C	0.81	⊥

- Make a context in the data preprocessing phase

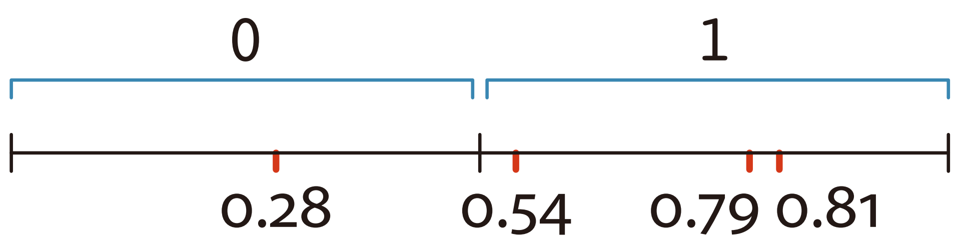
	1.**T**	2.A	2.B	2.C
1	×			×
2		×		
3	×		×	
4		×		
5	×			×

Data Preprocessing (2/3)

- Learning from the following dataset (⊥ is the missing value)

	Feature 1	Feature 2	Feature 3	Label
1	**T**	C	0.28	1
2	**F**	A	0.54	1
3	**T**	B	⊥	⊥
4	**F**	A	0.79	2
5	**T**	C	0.81	⊥

- Discretize the continuous feature (F3) using binary encoding



Data Preprocessing (3/3)

- Learning from the following dataset ($\perp$ is the missing value)

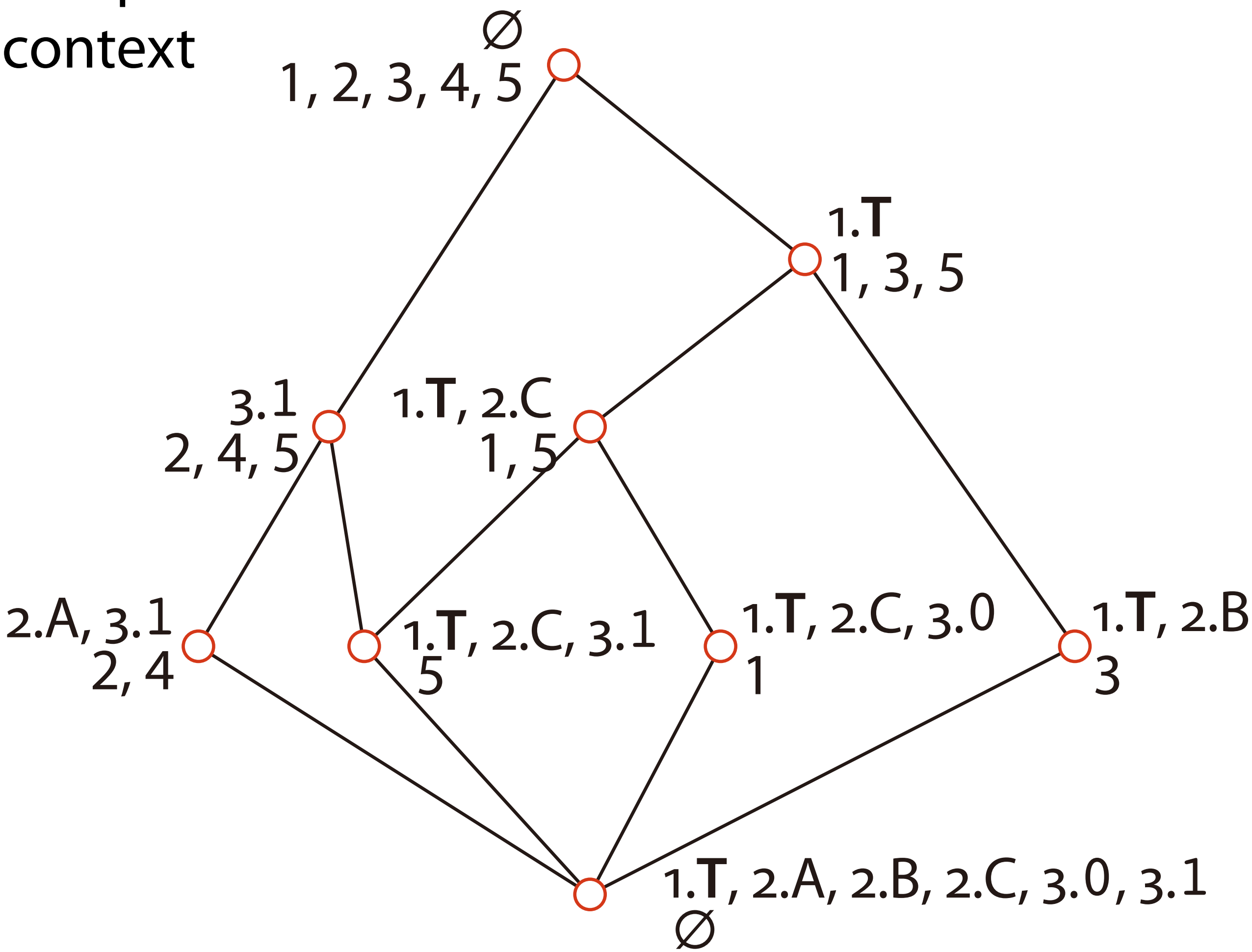
	Feature 1	Feature 2	Feature 3	Label
1	**T**	C	0.28	1
2	**F**	A	0.54	1
3	**T**	B	$\perp$	$\perp$
4	**F**	A	0.79	2
5	**T**	C	0.81	$\perp$

- Make a context in the data preprocessing phase

	1.**T**	2.A	2.B	2.C	3.1	3.2
1	×			×	×	
2		×				×
3	×		×			
4		×				×
5	×			×		×

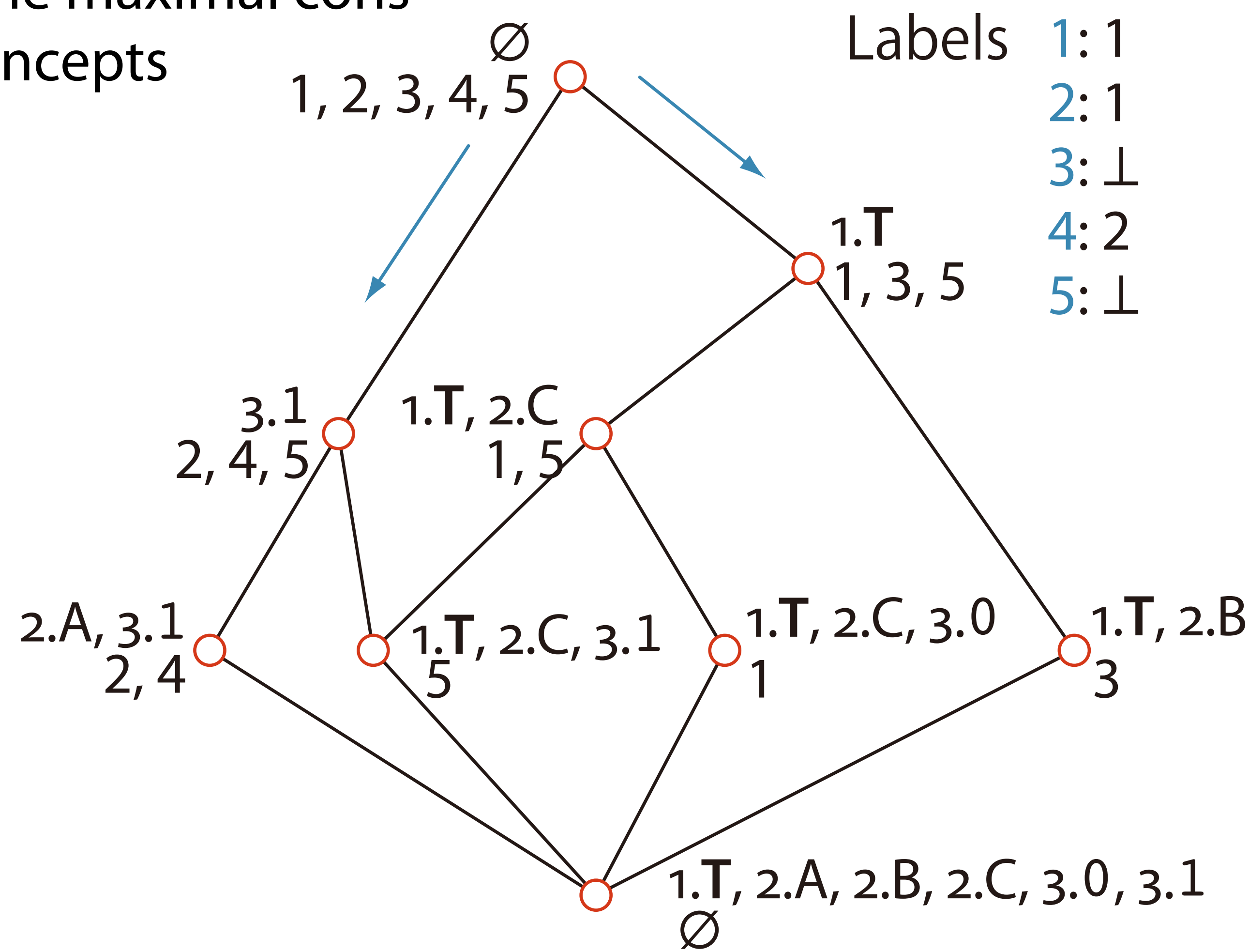
Make a Concept Lattice by FCA

- Make a concept lattice from the context



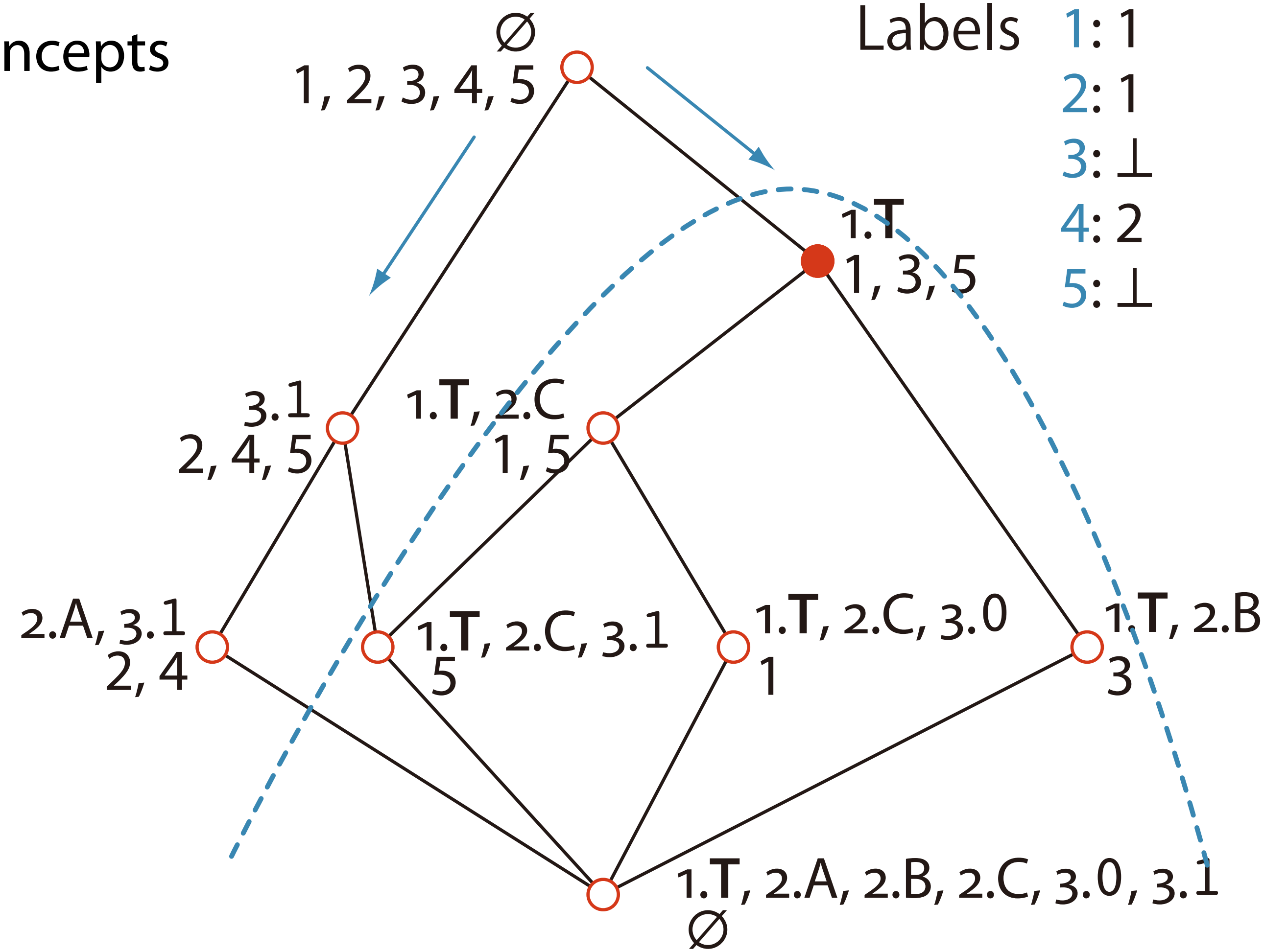
Learn Classification Rules

- Search the maximal consistent concepts



Learn Classification Rules

- Search the maximal cons-
  istent concepts



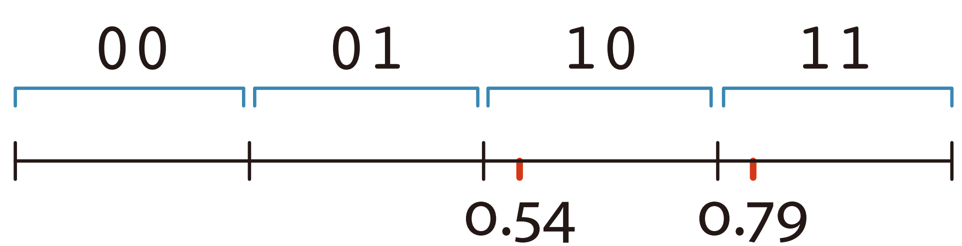
Data Preprocessing

- Learning from the remaining data

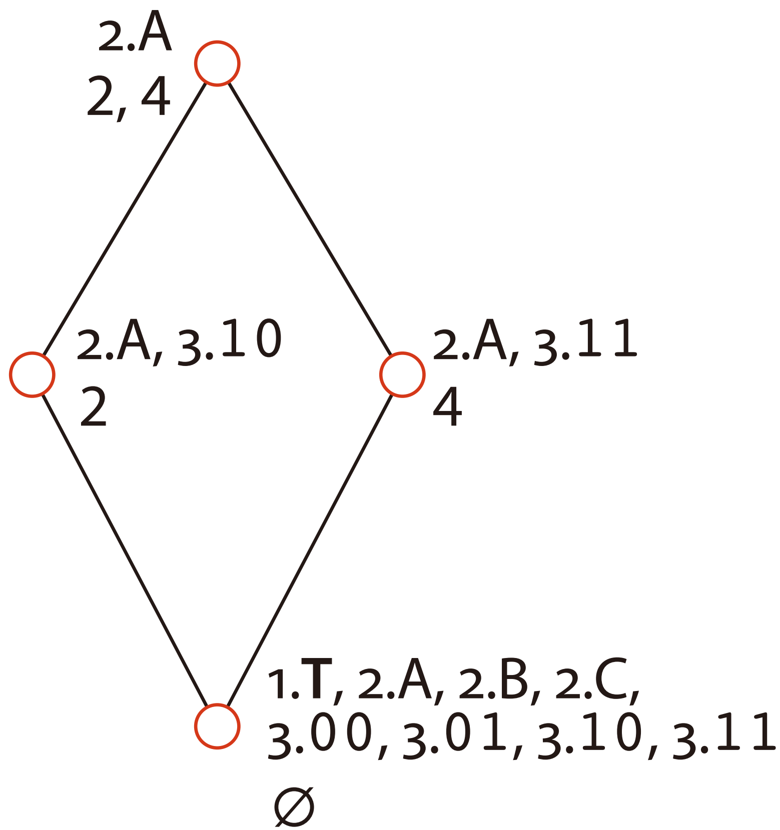
	Feature 1	Feature 2	Feature 3	Label
2	**F**	A	0.54	1
4	**F**	A	0.79	2

- Refine discretization, and obtain the following context

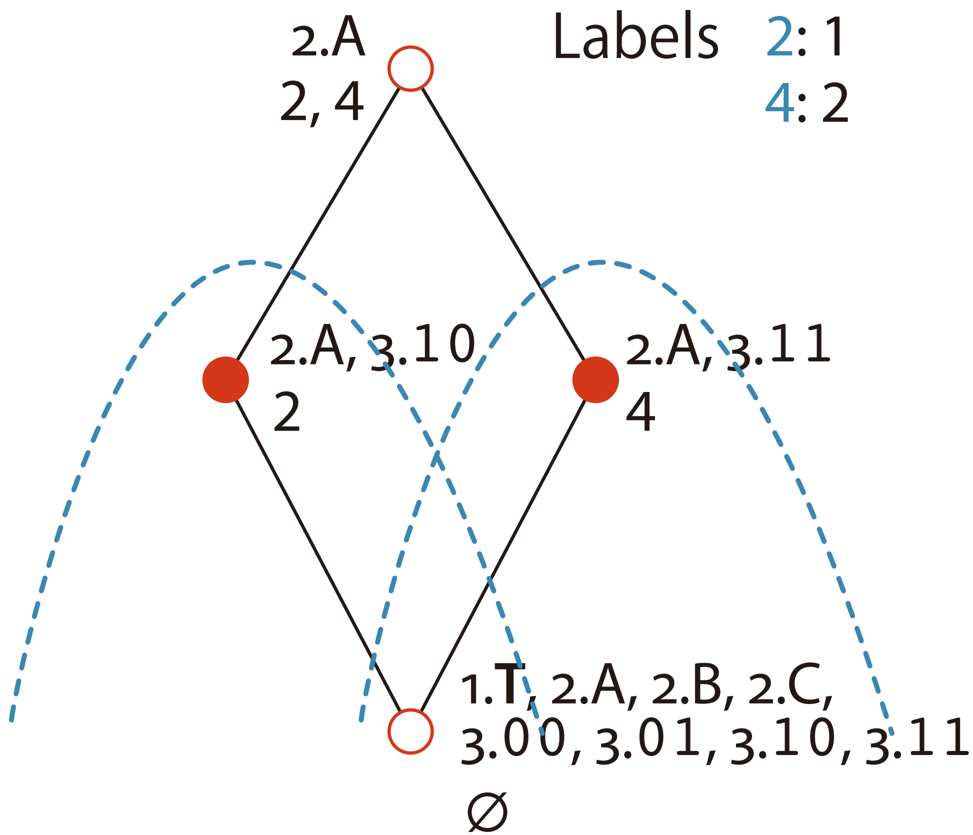
	1.T	2.A	2.B	2.C	3.00	3.01	3.10	3.11
2		×					×	
4		×						×



Make a Concept Lattice by FCA



Learn Classification Rules



Classify Unlabeled Data

- We obtain classification rules for each discretization level:

$$\begin{aligned} \mathscr{R} &= \{\mathscr{R}_1, \mathscr{R}_2\}, \\ \mathscr{R}_1 &= \{(\{1.\mathrm{T}\}, 1)\}, \\ \mathscr{R}_2 &= \{(\{2.\mathrm{A}, 3.10\}, 1), (\{2.\mathrm{A}, 3.11\}, 2)\} \end{aligned}$$

  – Each $\mathscr{R}_i$ is a set of pairs (attributes of a maximal concept, a class label)

- Classification of unlabeled (test) data
  – Datum (**T**, B, 0.45) belongs to class 1 since the 1st variable is **T**
  – Datum (**F**, A, 0.84) belongs to class 2 since the 2nd variable is A and the 3rd variable is in [0.75, 1]

Contents

Methods

1. SELF v.s. the decision tree-based method using mixed-type datasets
2. SELF v.s. other SSL methods using continuous datasets
   - There is no SSL method for mixed-type datasets

- SELF is implemented in R 2.10.1
- To enumerate all concepts, we use LCM [Uno *et al.*, 2005] presented by Uno
  - One of the fastest algorithm
- If # label candidates is more than two, we adopt the mode
  - If there is no label candidate, we adopt the mode of labels of a training dataset

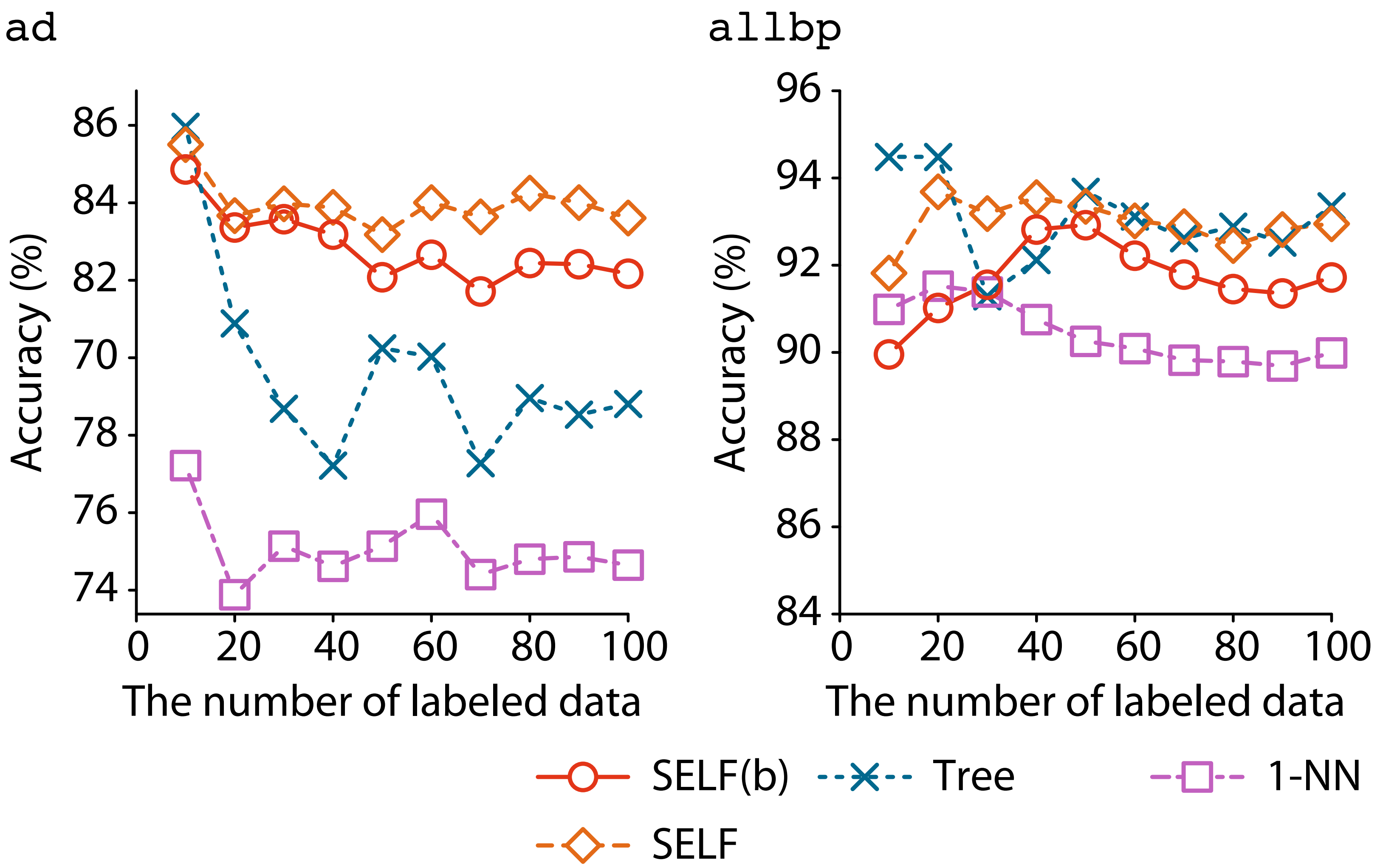
Methods of Exp. 1

- We used 10 mixed-type datasets from UCI repository
- To check effectivity of unlabeled data, we tested the following two cases
  1. Use labeled and unlabeled data for training
  2. Use only labeled data for training
- We used the decision tree-based method in R for control
- For reference, we used 1-NN using only continuous features
- We used 10-fold cross validation
  - One fold is labeled training data, another one fold is test data, and the other 8 folds are unlabeled training data
  - We fixed the number of labeled training data as 10∼100
- We compared the accuracy

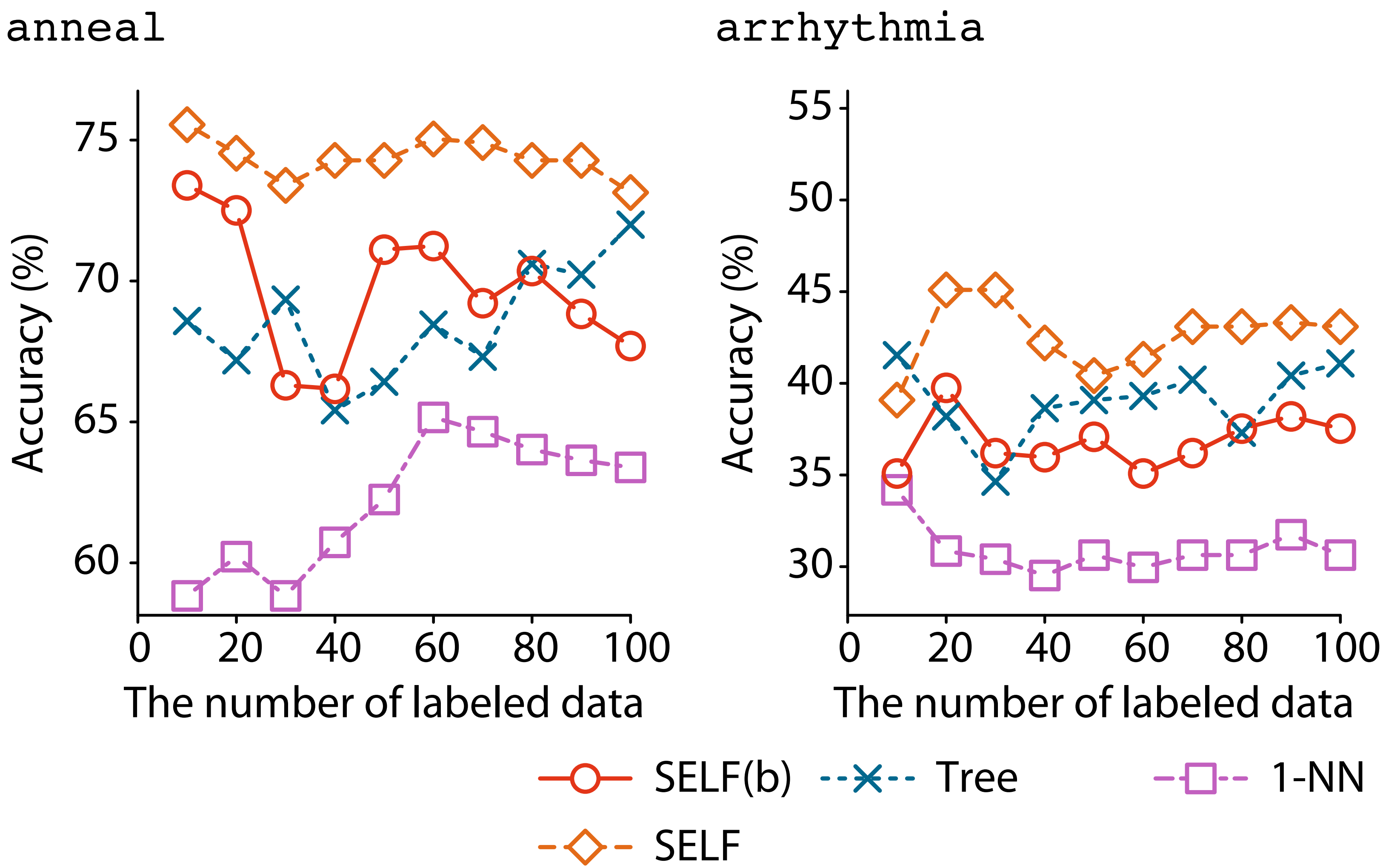
Datasets for Exp. 1

Name	# Data	# Classes	Bin.	Nom.	Real.
ad	3729	2	7	0	3
allbp	2800	3	2	0	3
anneal	798	5	0	26	6
arrhythmia	452	16	5	0	5
australian	690	2	4	4	6
crx	690	2	4	5	6
echoc	131	2	1	0	8
heart	270	2	3	4	6
hepatitis	155	2	13	0	6
horse	200	2	2	5	3

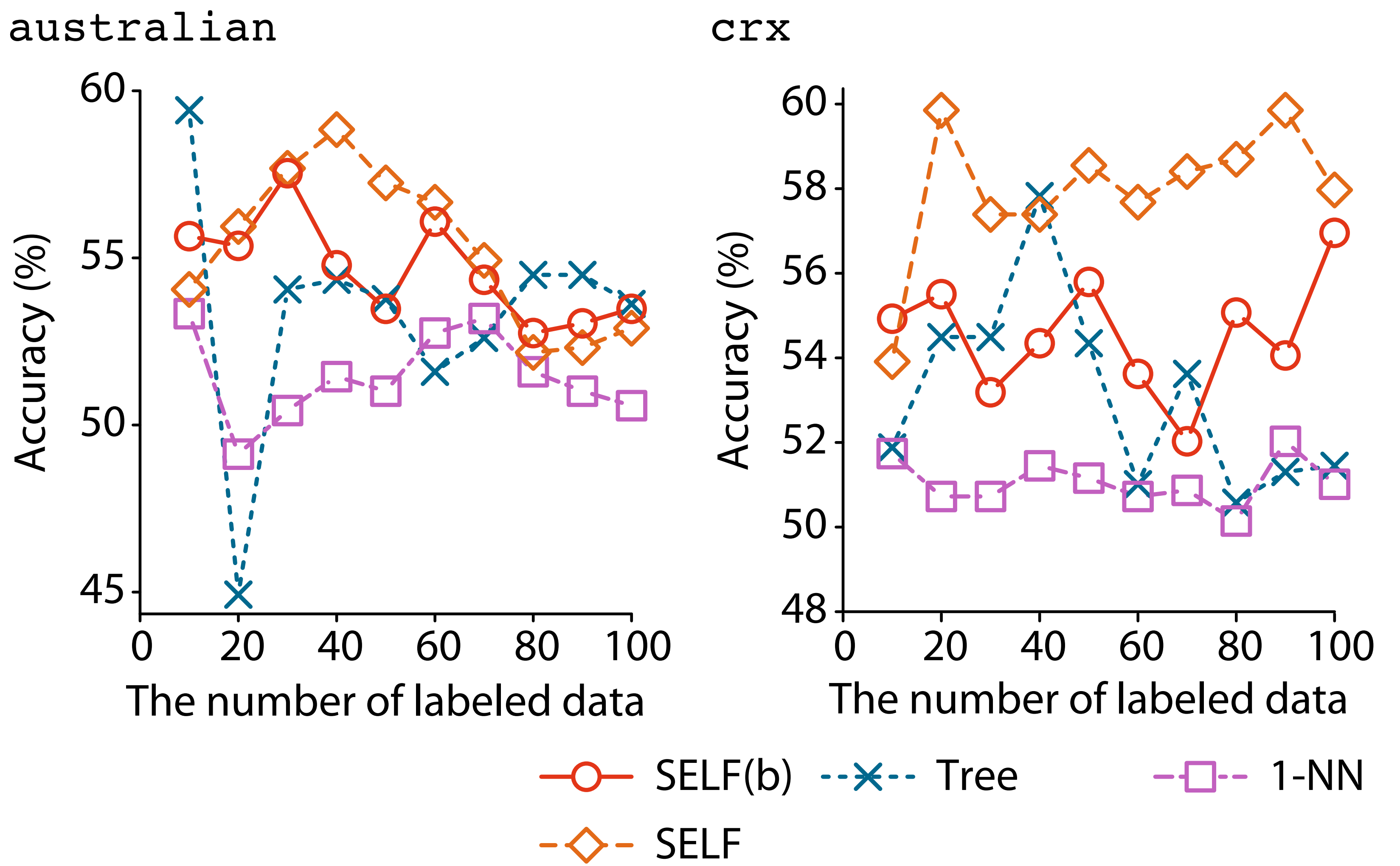
Results for Exp. 1



Results for Exp. 1

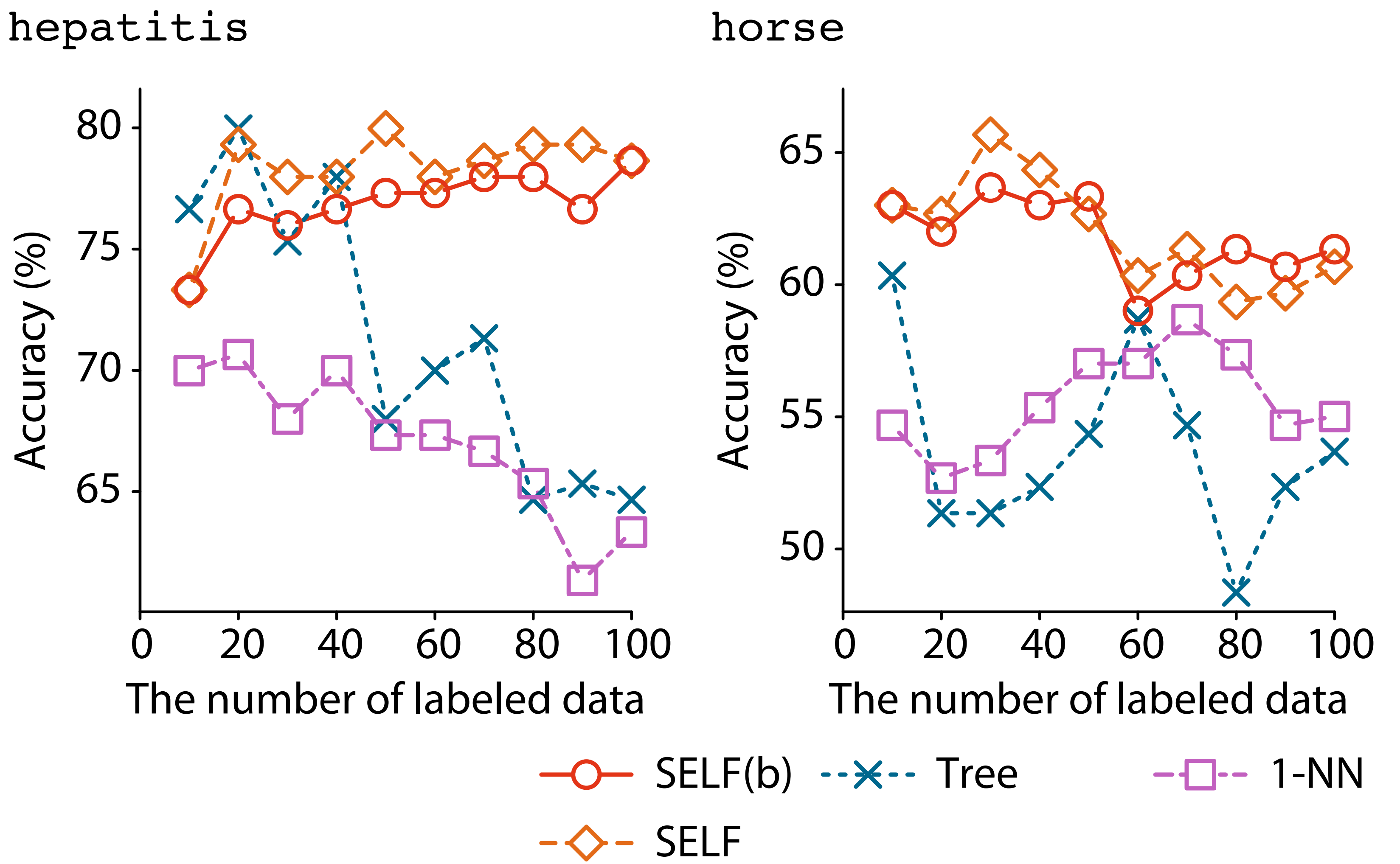


Results for Exp. 1



Results for Exp. 1

Results for Exp. 1



Methods for Exp. 2

- Use the benchmark datasets in [Chapelle *et al.*, 2006]
- Check two cases like exp. 1
  1. Use labeled and unlabeled data for training
  2. Use only labeled data for training
- Compared our results to those in [Chapelle *et al.*, 2006]

Datasets for Exp. 2

Name	# Data	# Classes	# features
`g241c` (D1)	1500	2	241
`g241d` (D2)	1500	2	241
`Digit1` (D3)	1500	2	241
`USPS` (D4)	1500	2	241
`COIL` (D5)	1500	6	241
`BCI` (D6)	400	2	117

Results for Exp. 2 (with 10 labeled data)

	D1	D2	D3	D4	D5	D6
SELF(b)	52.33	52.41	58.36	75.12	39.91	58.67
SELF	50.55	51.27	53.03	75.04	23.42	50.44
1-NN	55.95	56.78	76.53	80.18	34.09	51.26
SVM	52.68	53.34	69.40	79.97	31.64	50.15
MVU	51.32	52.72	88.08	85.12	34.28	49.76
LEM	52.53	54.66	87.96	80.86	32.04	50.06
QC	60.04	53.45	90.20	86.39	40.37	49.64
Disc.Reg.	50.41	50.95	87.36	83.93	36.62	50.49
TSVM	75.29	49.92	82.23	74.80	32.50	50.85
SGT	77.24	81.36	91.08	74.64	NA	50.41
C.Kernel	51.72	57.95	81.27	80.59	32.68	51.69
D.Reg.	58.75	54.11	87.51	82.04	36.35	49.79
LDS	71.15	49.37	84.37	82.43	38.10	50.73
RLS	56.05	54.32	94.56	81.01	45.46	51.03
CHM	60.97	56.99	85.14	79.47	NA	53.10

Results for Exp. 2 (with 100 labeled data)

	D1	D2	D3	D4	D5	D6
SELF(b)	67.01	67.03	72.62	83.19	70.18	88.08
SELF	54.37	53.87	59.98	77.44	46.09	64.56
1-NN	59.72	62.51	93.88	92.36	76.73	55.17
SVM	76.89	75.36	94.47	90.25	77.07	65.69
MVU	55.95	56.79	96.01	93.91	67.73	52.58
LEM	57.86	60.57	97.48	93.91	63.51	51.36
QC	77.95	71.80	96.85	93.64	89.97	53.78
Disc.Reg.	56.35	58.35	97.23	95.32	90.39	52.33
TSVM	81.54	77.58	93.85	90.23	74.20	66.75
SGT	82.59	90.89	97.39	93.20	NA	54.97
C.Kernel	86.51	95.05	96.21	90.32	78.01	64.83
D.Reg.	79.69	67.18	97.56	94.90	88.54	52.53
LDS	81.96	76.26	96.54	95.04	86.28	56.03
RLS	75.64	73.54	97.08	95.32	88.08	68.64
CHM	75.18	74.33	96.21	92.35	NA	63.97

Contents

Conclusion

- We have presented a semi-supervised learning (SSL) method, SELF, for mixed-type data
- ***Contribution to the FCA:***
  - A novel application of FCA
- ***Contribution to the KDD:***
  - The first direct SSL method for mixed-type datasets
    - An original dataset is lifted to the concept lattice using FCA (Formal Concept Analysis)
    - Classification rules are learned in the space
  - Moreover, SELF can apply to incomplete data with missing values

Appendix

Related Work (in FCA)

- Many studies used FCA for machine learning and knowledge discovery [Kuznetsov, 2004]
  - Classification [Ganter and Kuznetsov, 2000; Ganter and Kuznetsov, 2003]
  - Clustering [Zhang *et al.*, 2008]
  - Association rule mining [Jaschke *et al.*, 2006; Pasquier *et al.*, 1999; Valtchev *et al.*, 2004]
  - Bioinformatics [Blinova *et al.*, 2003; Kaytoue *et al.*, 2010; Kuznetsov and Samokhin, 2005]
- Ganter and Kuznetsov attacked to the problem of binary classification for real-valued data
  - Their method discretizes real-valued variables by conceptual scaling [Ganter and Wille, 1998], that are given *a priori*

Related Work (in ML)

- Decision tree-based methods, *e.g.*, C4.5 [Quinlan, 1993; Quinlan, 1996], can treat mixed-type data
- Discretization techniques [Fayyad and Irani, 1993; Liu *et al.*, 2002; Skubacz and Hollmén, 2000]
  - Our approach is different from them since we integrate discretization process into learning process and avoid overfitting
- Kok and Domingos [Kok and Domingos, 2009] have proposed a learning method via hypergraph lifting
  - Construct clusters by hypergraphs and learns on them
  - it is difficult to treat continuous variables in their approach

Time Complexity

- Data preprocessing takes $O(nd)$
  - n is the number of objects
  - d is the number of attributes
- Making concepts takes $O(\Delta^3)$
  - $\Delta = \max\{\#J \mid J \subseteq I,\ g = h$ for all $(g, m), (h, l) \in J$, or $m = l$ for all $(g, m), (h, l) \in J\}$
- Judging consistency of concepts takes less than $O(\Lambda)$
  - Λ is the number of concepts at discretization level 1
- The time complexity of SELF is $O(nd) + O(\Delta^3) + O(\Lambda)$

An Fatal Error Caused by Discretization

- Solve the system of linear equations [Schroder, 03]

$$40157959.0\,x + 67108865.0\,y = 1$$
$$67108864.5\,x + 112147127.0\,y = 0$$

  – Obtained by the well-known formula

$$x = \frac{b_1 a_{22} - b_2 a_{12}}{a_{11} a_{22} - a_{21} a_{12}}, \quad y = \frac{b_2 a_{11} - b_1 a_{21}}{a_{11} a_{22} - a_{21} a_{12}}$$

- By floating point arithmetic with double precision variables (IEEE 754) :

$$x = 112147127, \quad y = -67108864.5$$

- The correct solution:

$$x = 224294254, \quad y = -134217729$$

What is SSL?

- It is a special form of classification
- Goal: Using (large amount of) unlabeled data effectively, together with labeled data, build better classifiers [Zhu and Goldberg, 2009]
  - Transductive learning focuses on classification of unlabeled data in the training data [Vapnik and Sterin, 1977]
  - In contrast, in SSL we treat learning of classification rules and classification of unseen data
- Usual assumption: There are only few labeled data (10∼100) and lots of unlabeled data (∼1000)
  - Labeling costs high in real situation

Data Preprocessing for Binary and Nominal Variables

- Goal: Make a context from a given dataset to use FCA
  - A context is a triple (G, M, I), G and M are sets and $I \subseteq G \times M$
    - Elements in G and M are objects and attributes, resp.
    - gIm means an object g has an attribute m
    - Represented by a cross-table
- Strategy: Convert each feature in the given dataset into a context, and merge into one context
  - First we make a context from discrete variables
  - The process of making a context from continuous variables is embedded into the learning process
    - Increase discretization level along with the learning process
    - We can avoid overfitting

Data Preprocessing Algorithm for Binary and Nominal Variables

Input: Dataset $X = [x_{ij}]_{n \times q}$ whose variables are binary or nominal

Output: Context $(G, M_{\mathrm{BN}}, I_{\mathrm{BN}})$

function ContextBN(X)

$G \leftarrow \{1, 2, \dots, n\}$
for each $j \in \{1, 2, \dots, d\}$
  if the feature j of X is binary and has no missing value **then**
    $M_j \leftarrow \{\mathbf{T}\}, I_j \leftarrow \{(i, x_{ij}) \mid i \in G \text{ and } x_{ij} = \mathbf{T}\}$
  else if the feature j of X is binary and has missing value **then**
    $M_j \leftarrow \{\mathbf{T}, \mathbf{F}\}, I_j \leftarrow \{(i, x_{ij}) \mid i \in G \text{ and } x_{ij} \neq \perp\}$
  else // the feature j is nominal
    $M_j \leftarrow \{1, \dots, v_j\}, I_j \leftarrow \{(i, x_{ij}) \mid i \in G \text{ and } x_{ij} \neq \perp\}$
combine $(G, M_1, I_1), (G, M_2, I_2), \dots, (G, M_d, I_d)$ into $(G, M_{\mathrm{BN}}, I_{\mathrm{BN}})$
return $(G, M_{\mathrm{BN}}, I_{\mathrm{BN}})$

Example of Data Preprocessing for Discrete Variables

- Data preprocessing for the following dataset

	Feature 1	Feature 2	Feature 3
1	**T**	⊠	C
2	**F**	**F**	⊠

  – Features 1 and 2 are binary, and the feature 3 is nominal

Example of Data Preprocessing for Discrete Variables

- Data preprocessing for the following dataset

	Feature 1	Feature 2	Feature 3
1	**T**	⊠	C
2	**F**	**F**	⊠

  – Features 1 and 2 are binary, and the feature 3 is nominal
- We obtain the context as follows:

	1.T		
1	×		
2			

Example of Data Preprocessing for Discrete Variables

- Data preprocessing for the following dataset

	Feature 1	Feature 2	Feature 3
1	**T**	⊠	C
2	**F**	**F**	⊠

  – Features 1 and 2 are binary, and the feature 3 is nominal
- We obtain the context as follows:

	1.**T**	2.**T**	2.**F**	
1	×			
2			×	

Example of Data Preprocessing for Discrete Variables

- Data preprocessing for the following dataset

	Feature 1	Feature 2	Feature 3
1	**T**	⊠	C
2	**F**	**F**	⊠

  - Features 1 and 2 are binary, and the feature 3 is nominal
- We obtain the context as follows:

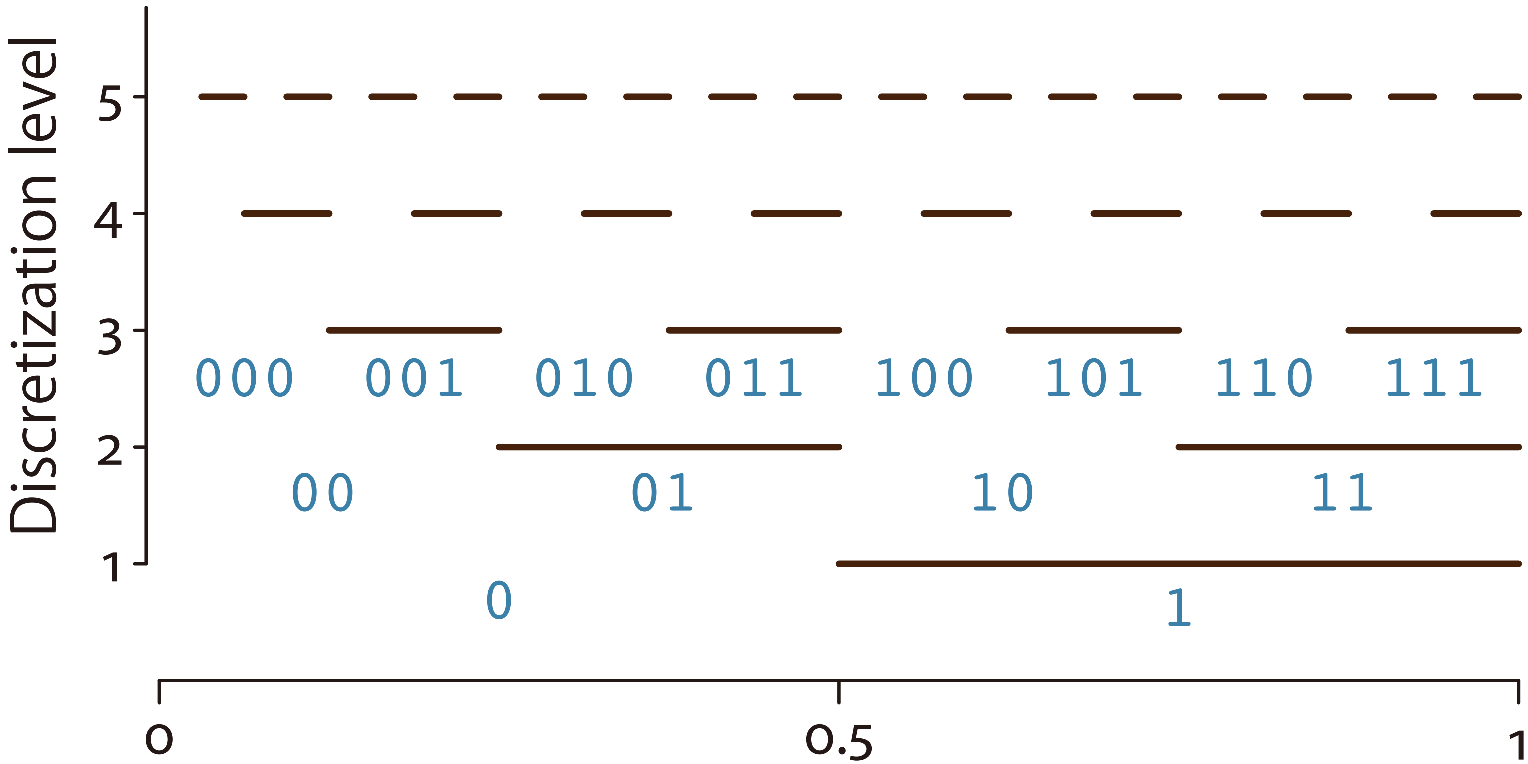
	1.**T**	2.**T**	2.**F**	3.A	3.B	3.C
1	×					×
2			×			

Data Preprocessing Algorithm for Real-Valued Variables

Input: Real-valued dataset X and discretization level k
Output: Context $(G, M_{\mathrm{R}}, I_{\mathrm{R}})$

function ContextR(X, k)
  $G \leftarrow \{1, 2, \dots, n\}$
  for each $j \in \{1, 2, \dots, d\}$
    $M_j \leftarrow \{1, 2, \dots, \beta^k\}$
    Normalize X_j by min-max normalization
    for each $i \in \{1, 2, \dots, n\}$
      if $x_{ij} = 0$ then $I_j \leftarrow I_j \cup \{(i, 1)\}$
      else if $x_{ij} \neq 0$ and $x_{ij} \neq \perp$ then
      $I_j \leftarrow I_j \cup \{(i, m)\}$, where $x_{ij} \in ((m-1)/\beta^k, m/\beta^k]$
  combine $(G, M_1, I_1), (G, M_2, I_2), \dots, (G, M_d, I_d)$ into $(G, M_{\mathrm{R}}, I_{\mathrm{R}})$
  return $(G, M_{\mathrm{R}}, I_{\mathrm{R}})$

Discretization by Binary Encoding



Example of Data Preprocessing for Real-Valued Variables

- Data preprocessing for the following dataset

	Feature 1	Feature 2	Feature 3
1	**T**	0.35	0.78
2	**F**	0.813	⊠

  – Features 1 is binary, and feature 2 and 3 are real-valued

Example of Data Preprocessing for Real-Valued Variables

- Data preprocessing for the following dataset

	Feature 1	Feature 2	Feature 3
1	**T**	0.35	0.78
2	**F**	0.813	⊠

  - Features 1 is binary, and feature 2 and 3 are real-valued
- At discretization level 1, we obtain the context as follows:

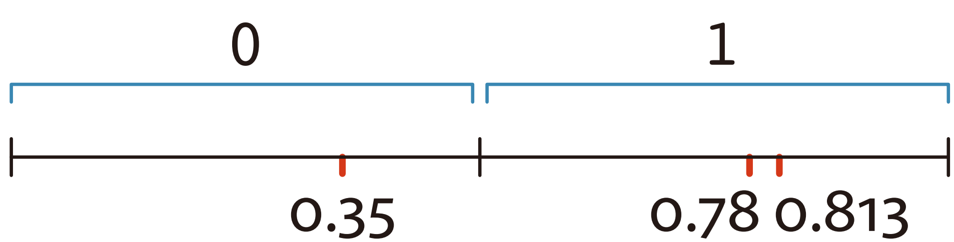
	1.T		
1	×		
2			

Example of Data Preprocessing for Real-Valued Variables

- Data preprocessing for the following dataset

	Feature 1	Feature 2	Feature 3
1	**T**	0.35	0.78
2	**F**	0.813	[illegible]

  – Features 1 is binary, and feature 2 and 3 are real-valued
- At discretization level 1, we obtain the context as follows:



Example of Data Preprocessing for Real-Valued Variables

- Data preprocessing for the following dataset

	Feature 1	Feature 2	Feature 3
1	**T**	0.35	0.78
2	**F**	0.813	⊠

  - Features 1 is binary, and feature 2 and 3 are real-valued
- At discretization level 1, we obtain the context as follows:

	1.T	2.0	2.1
1	×	×	
2			×

Example of Data Preprocessing for Real-Valued Variables

- Data preprocessing for the following dataset

	Feature 1	Feature 2	Feature 3
1	**T**	0.35	0.78
2	**F**	0.813	⊠

  - Features 1 is binary, and feature 2 and 3 are real-valued
- At discretization level 1, we obtain the context as follows:

	1.T	2.0	2.1	3.0	3.1
1	×	×			×
2			×		

Formal Concept Analysis (FCA)

- Generate a concept lattice using the algebraic “closed” property
- For $A \subseteq G$ and $B \subseteq M$ of a context (G, M, I),

  $$A' = \{m \in M \mid (\forall g \in A)\, gIm\},\ B' = \{g \in G \mid (\forall m \in B)\, gIm\}$$

- A concept of a (G, M, I) is a pair (A, B) $(A \subseteq G, B \subseteq M)$ such that $A' = B$ and $B' = A$
  - A is an extent and B is an interior, A is extent $\iff A'' = A'$
  - $\mathscr{B}(G, M, I)$ is the set of concept
- For concepts $(A_1, B_1), (A_2, B_2), A_1 \subseteq A_2 \Rightarrow (A_1, B_1) \leq (A_2, B_2)$

  $$(A_1, B_1) \leq (A_2, B_2) \iff A_1 \subseteq A_1 \iff B_1 \supseteq B_2$$

- $\leq$ is an order of $\mathscr{B}(G, M, I)$, $\langle \mathscr{B}(G, M, I), \leq \rangle$ is a complete lattice

– The concept lattice of a context (G, M, I)

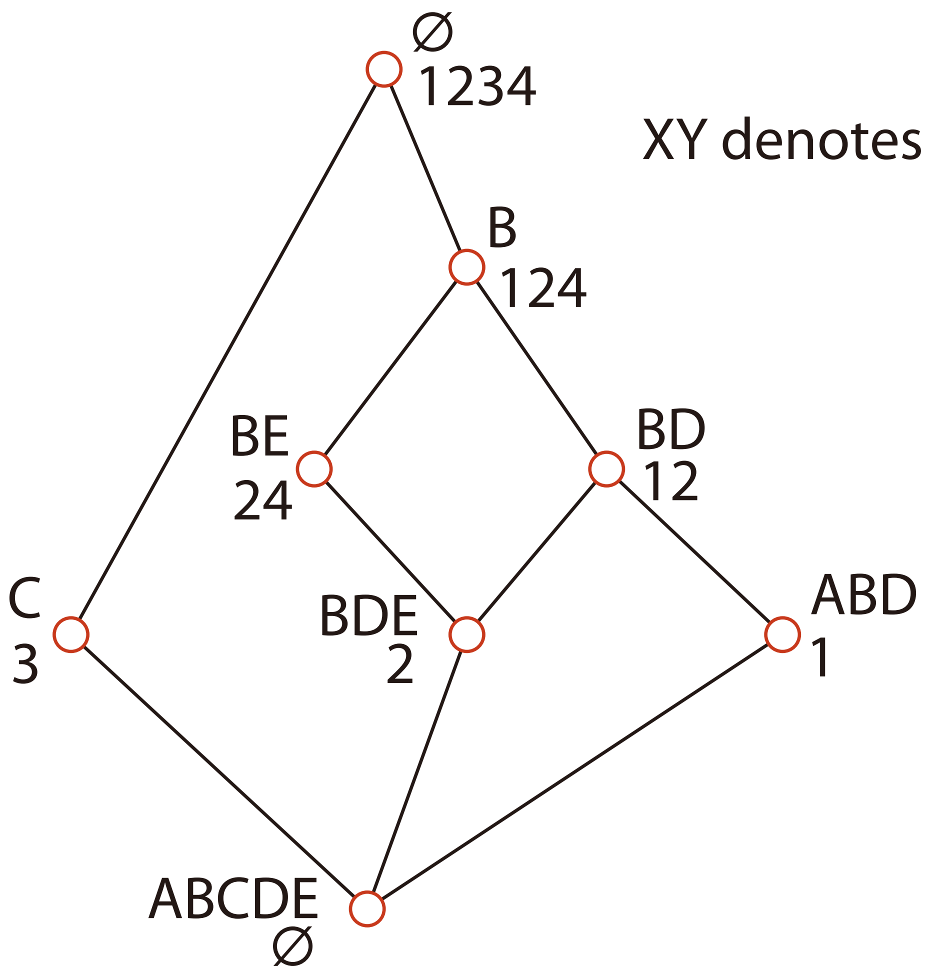
Example of FCA

- Given the following context

	A	B	C	D	E
1	×	×		×	
2		×		×	×
3			×		
4		×			×

- There are 8 concepts as follows:
  - $(\emptyset, \{A, B, C, D, E\})$, $(\{1\}, \{A, B, D\})$, $(\{2\}, \{B, D, E\})$, $(\{3\}, \{C\})$, $(\{1,2\}, \{B, D\})$, $(\{2,4\}, \{B, E\})$, $(\{1,2,4\}, \{B\})$, $(\{1,2,3,4\}, \emptyset)$.
- The concept lattice is constructed from these concepts

Example of FCA



XY denotes the set {X, Y}

Learning Classification Rules

- Idea: If the label of an object in a concept is c, those of the other objects in the same concept are also c
- Basic strategy: Search the concept lattice from the top concept, and find maximal and consistent concepts
  - The attribute B of such a concept (A, B) is a rule
    - A is a dataset classified to the class by the rule B
  - Each concept can be viewed as a base cluster
- For each object $g \in G$, g's label is denoted by $\gamma(g)$
  - $\Gamma(G) := \{g \in G \mid \gamma(g) \neq \bot\}$
    - $\gamma(g) = \bot \iff g$ is unlabeled data
- For a concept (A, B), if $\Gamma(A) \neq \emptyset$ and $\gamma(g) = \gamma(h)$ for all $g, h \in \Gamma(A)$, then (A, B) is consistent

The SELF Algorithm (1/2)

Input: Dataset X with n objects and d attributes
Output: A set of classification rules $\mathscr{R}$

function Main(X)

Divide X into two datasets X_{BN} and X_{R}, where X_{BN} contains all binary and nominal variables in X, and X_{R} contains all real-valued variables in X

$(G, M_{\mathrm{BN}}, I_{\mathrm{BN}}) \leftarrow$ ContextBN(X_{BN})
// make a context from binary and nominal variables of X

$k \leftarrow 1$ // k is level of discretization

$\mathscr{R} \leftarrow$ Learning($X_{\mathrm{R}}, G, M_{\mathrm{BN}}, I_{\mathrm{BN}}, k, \emptyset$)
// use this function recursively

return $\mathscr{R}$

The SELF Algorithm (2/2)

function Learning($X_{\mathrm{R}}, G, M_{\mathrm{BN}}, I_{\mathrm{BN}}, k, \mathscr{R}$)

$(G, M_{\mathrm{R}}, I_{\mathrm{R}}) \leftarrow \mathrm{ContextR}(X_{\mathrm{R}}, k)$
// make a context from real-valued variables of X at level k

make (G, M, I) from $(G, M_{\mathrm{BN}}, I_{\mathrm{BN}})$ and $(G, M_{\mathrm{R}}, I_{\mathrm{R}})$

build the concept lattice $\mathfrak{B}(G, M, I)$ from (G, M, I)

$\mathscr{C} \leftarrow \{(A, B) \in \mathfrak{B}(G, M, I) \mid (A, B) \text{ is consistent}\}$

$\mathscr{R}_k \leftarrow \{(B, \gamma(a)) \mid (A, B) \in \mathrm{Max}\,\mathscr{C} \text{ and } a \in \Gamma(A)\}$

$\mathscr{R} \leftarrow \mathscr{R} \cup (\mathscr{R}_k, k)$ // add the current result $\mathscr{R}_k$

$G \leftarrow G \setminus \{g \mid g \in A \text{ for some } (A, B) \in \mathscr{C}\}$

remove corresponding attributes and relations from M_{BN}
and I_{BN}, respectively

remove corresponding objects from X_{R}

if $\Gamma(G) = \emptyset$ **then return** $\mathscr{R}$

else return Learning($X_{\mathrm{R}}, G, M_{\mathrm{BN}}, I_{\mathrm{BN}}, k + 1, \mathscr{R}$)

end if

Classify unlabeled data

- Assume we obtain the rules $\mathscr{R} = \{\mathscr{R}_1, \dots, \mathscr{R}_k\}$ by SELF
- To classify a test datum, check rules from $\mathscr{R}_1$ to $\mathscr{R}_k$
- For each level k,
  1. Make a context (G, M, I) by data preprocessing
  2. Enumerate all l such that $B \subseteq M$ for $(B, l) \in \mathscr{R}_k$
- We obtain label candidates $L = \{l_1, \dots, l_c\}$ and sometimes cannot decide an unique label
  - All one-against-all classification methods have the same problem
    - One of future works

Classification Algorithm

Input: Classification rules $\mathscr{R} = \{\mathscr{R}_1, \mathscr{R}_2, \ldots, \mathscr{R}_K\}$ and $x = [x_{ij}]_{1\times d}$
Output: Label candidates $L = \{l_1, l_2, \ldots, l_C\}$

function Classify(R, x)

    $L \leftarrow \varnothing$
    divide x into two data x_{BN} and x_{R}, where x_{BN} contains all binary and nominal variables, x_{R} contains all real-valued variables
    $(G, M_{\mathrm{BN}}, I_{\mathrm{BN}}) \leftarrow$ ContextBN(x_{BN})
    // make a context from binary and nominal variables of x
    for each $k \in \{1, 2, \ldots, K\}$
      $(G, M_{\mathrm{R}}, I_{\mathrm{R}}) \leftarrow$ ContextR(x_{R}, k)
      // make a context from real-valued variables of x at level k
      make the context (G, M, I) from $(G, M_{\mathrm{BN}}, I_{\mathrm{BN}})$ and $(G, M_{\mathrm{R}}, I_{\mathrm{R}})$
      add l to L if $R \in M$ for some $(R, l) \in \mathscr{R}_k$
    end for; **return** L

References

[Abe, 2003] S. Abe. Analysis of multiclass support vector machines. In *Proceedings of International Conference on Computational Intelligence for Modeling Control and Automation*, pages 385–396, 2003.

[Blinova *et al.*, 2003] V. G. Blinova, D. A. Dobrynin, V. K. Finn, S. O. Kuznetsov, and E. S. Pankratova. Toxicology analysis by means of the JSM-method. *Bioinformatics*, 19(10):1201–1207, 2003.

[Chapelle *et al.*, 2006] O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-Supervised Learning*. MIT Press, 2006.

[Davey and Priestley, 2002] B. A. Davey and H. A. Priestley. *Introduction to lattices and order*. Cambridge University Press, 2 edition, 2002.

[de Brecht and Yamamoto, 2010] Matthew de Brecht and Akihiro Yamamoto. Topological properties of concept spaces (full version). *Information and Computation*, 208:327–340, 2010.

[Fayyad and Irani, 1993] U. M. Fayyad and K. B. Irani. Multi-interval discretization of continuous-valued attributes for classification learning.

In *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, pages 1022–1029, 1993.

[Frank and Asuncion, 2010] A. Frank and A. Asuncion. UCI machine learning repository, 2010.

[Ganter and Kuznetsov, 2000] B. Ganter and S. Kuznetsov. Formalizing hypotheses with concepts. In B. Ganter and G. W. Mineau, editors, *Conceptual Structures: Logical, Linguistic, and Computational Issues*, volume 1867 of *Lecture Notes in Computer Science*, pages 342–356. Springer, 2000.

[Ganter and Kuznetsov, 2003] B. Ganter and S. Kuznetsov. Hypotheses and version spaces. In A. de Moor, W. Lex, and B. Ganter, editors, *Conceptual Structures for Knowledge Creation and Communication*, volume 2746 of *Lecture Notes in Computer Science*, pages 83–95. Springer, 2003.

[Ganter and Wille, 1998] B. Ganter and R. Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer, 1998.

[Garcia-Molina *et al.*, 2008] H. Garcia-Molina, J. D. Ullman, and J. Widom. *Database systems: The complete book*. Prentice Hall Press, 2008.

[Han and Kamber, 2006] J. Han and M. Kamber. *Data Mining*. Morgan Kaufmann, 2 edition, 2006.

[Jaschke *et al.*, 2006] R. Jaschke, A. Hotho, C. Schmitz, B. Ganter, and G. Stumme. TRIAS–An algorithm for mining iceberg tri-lattices. In *Proceedings of the 6th International Conference on Data Mining (ICDM'06)*, pages 907–911. IEEE, 2006.

[Kaytoue *et al.*, 2010] M. Kaytoue, S. O. Kuznetsov, A. Napoli, and S. Duplessis. Mining gene expression data with pattern structures in formal concept analysis. *Information Sciences*, 2010.

[Kok and Domingos, 2009] S. Kok and P. Domingos. Learning Markov logic network structure via hypergraph lifting. In *Proceedings of the 26th International Conference on Machine Learning*, pages 505–512, 2009.

[Kuznetsov and Samokhin, 2005] S. O. Kuznetsov and M. V. Samokhin. Learning closed sets of labeled graphs for chemical applications. In S. Kramer and B. Pfahringer, editors, *Inductive Logic Programming*, volume 3625 of *Lecture Notes in Computer Science*, pages 190–208. Springer, 2005.

[Kuznetsov, 2004] S. O. Kuznetsov. Machine learning and formal concept analysis. In P. Eklund, editor, *Concept Lattices*, volume 2961 of *Lecture Notes in Computer Science*, pages 287–312. Springer, 2004.

[Liu *et al.*, 2002] H. Liu, F. Hussain, C. L. Tan, and M. Dash. Discretization: An enabling technique. *Data Mining and Knowledge Discovery*, 6(4):393–423, 2002.

[Makino and Uno, 2004] K. Makino and T. Uno. New algorithms for enumerating all maximal cliques. *Algorithm Theory-SWAT 2004*, pages 260–272, 2004.

[Murthy, 1998] S. K. Murthy. Automatic construction of decision trees from data: A multi-disciplinary survey. *Data Mining and Knowledge Discovery*, 2(4):345–389, 1998.

[Pasquier *et al.*, 1999] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Efficient mining of association rules using closed itemset lattices. *Information Systems*, 24(1):25–46, 1999.

[Quinlan, 1993] J. R. Quinlan. *C4.5: programs for machine learning*. Morgan Kaufmann, 1993.

[Quinlan, 1996] J. R. Quinlan. Improved use of continuous attributes in C4.5. *Journal of Artificial Intelligence Research*, 4:77–90, 1996.

[R Development Core Team, 2011] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, 2011.

[Ripley, 1996] B. D. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, 1996.

[Skubacz and Hollmén, 2000] M. Skubacz and J. Hollmén. Quantization of continuous input variables for binary classification. In *Intelligent Data Engineering and Automated Learning — IDEAL 2000. Data Mining, Financial Engineering, and Intelligent Agents*, volume 1983 of *Lecture Notes in Computer Science*, pages 42–47. Springer, 2000.

[Sugiyama and Yamamoto, 2010] M. Sugiyama and A. Yamamoto. The coding divergence for measuring the complexity of separating two sets. In *Proceedings of 2nd Asian Conference on Machine Learning*, volume 13 of *JMLR Workshop and Conference Proceedings*, pages 127–143, 2010.

[Uno *et al.*, 2005] T. Uno, M. Kiyomi, and H. Arimura. LCM ver. 3: Collaboration of array, bitmap and prefix tree for frequent itemset mining. In *Proceedings of the 1st International Workshop on Open Source Data Mining: Frequent Pattern Mining Implementations*, pages 77–86. ACM, 2005.

[Valtchev *et al.*, 2004] P. Valtchev, R. Missaoui, and R. Godin. Formal concept analysis for knowledge discovery and data mining: The new challenges. *Concept Lattices*, pages 3901–3901, 2004.

[Vapnik and Sterin, 1977] V. Vapnik and A. Sterin. On structural risk minimization or overall risk in a problem of pattern recognition. *Automation and Remote Control*, 10(3):1495–1503, 1977.

[Vapnik, 2000] V. N. Vapnik. *The nature of statistical learning theory*. Springer, 2000.

[Zhang *et al.*, 2008] Y. Zhang, B. Feng, and Y. Xue. A new search results clustering algorithm based on formal concept analysis. In *Proceedings of 5th International Conference on Fuzzy Systems and Knowledge Discovery*, pages 356–360. IEEE, 2008.

[Zhu and Goldberg, 2009] X. Zhu and A. B. Goldberg. *Introduction to semi-supervised learning*. Morgan and Claypool Publishers, 2009.