第 86 回 MPS・第 27 回 BIO 合同研究発表会
December 1–2, 2011

Semi-Supervised Ligand Finding Using Formal Concept Analysis

Mahito SUGIYAMA[†,‡], Kentaro Imajo[†]
Keisuke Otaki[†], Akihiro YAMAMOTO[†]

[†]Kyoto University, [‡]JSPS Research Fellow

Summary

- ***Background:*** In organisms, receptors have crucial roles for signal processing
  - Ligands are key tools to investigate them in biochemical experiments
- ***Problem:*** Finding ligands is difficult
  - *In silico* approach is required for helping biologists
- ***Solution:*** A machine learning approach by formalizing the problem as multi-label classification
  - We develop a new algorithm LIFT (Ligand FInding via Formal ConcepT Analysis) for multi-label classification
    - Ligand data are treated in the semi-supervised manner
    - Clustering is realized by Formal Concept Analysis (FCA)

Contents

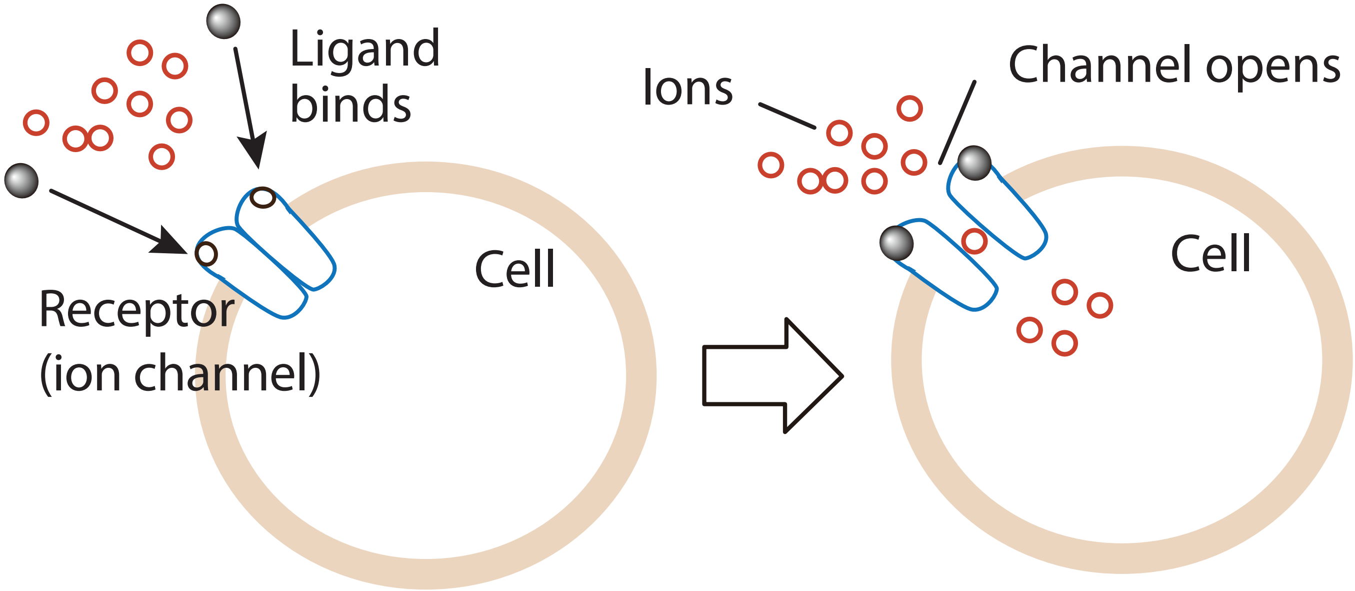
1. Motivation
2. Classification by LIFT
   - i) Data preprocessing
   - ii) Making concepts by LIFT
   - iii) Classifying unlabeled data
3. Experiments
4. Conclusion

Contents

1. Motivation
2. Classification by LIFT
   - i) Data preprocessing
   - ii) Making concepts by LIFT
   - iii) Classifying unlabeled data
3. Experiments
4. Conclusion

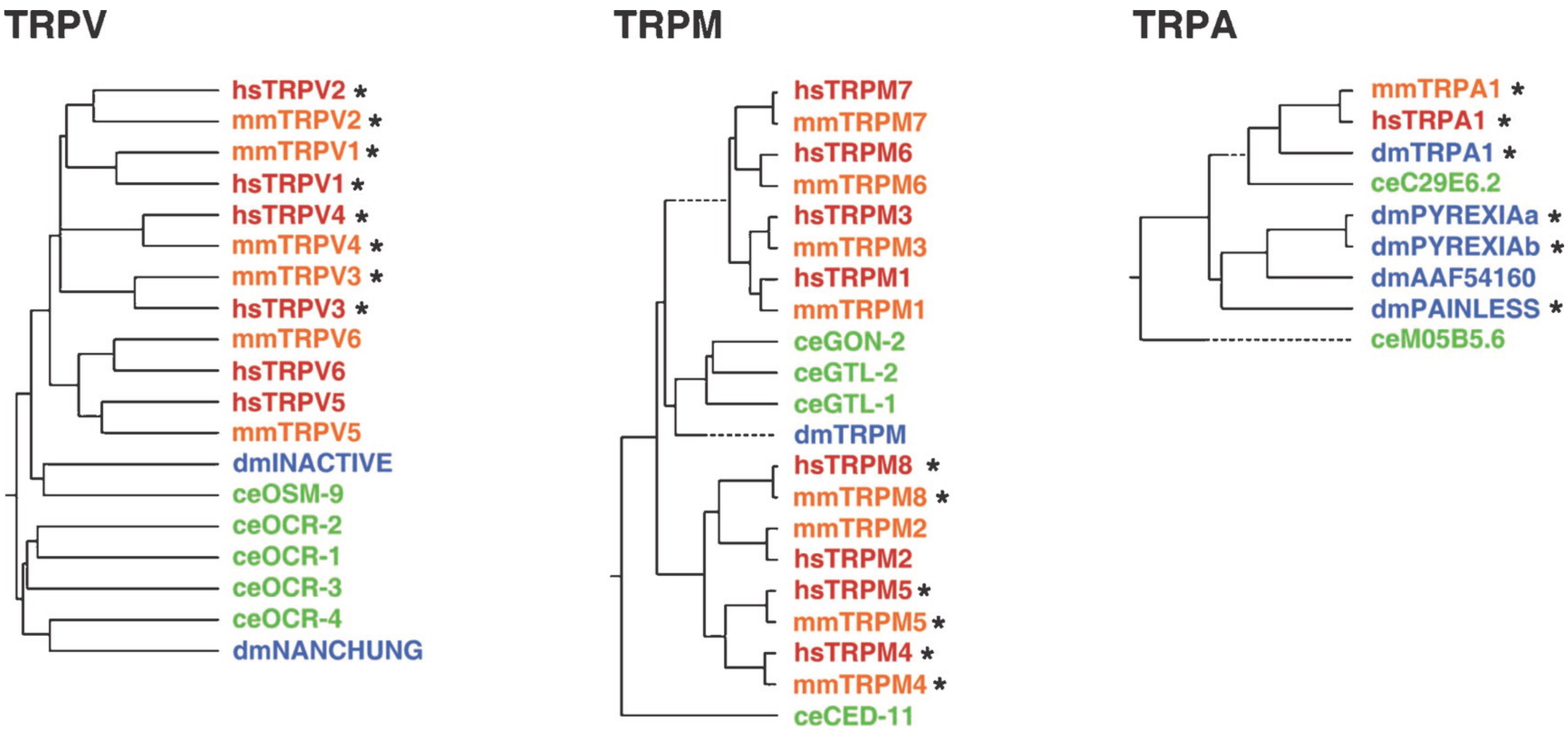
Background

- Receptors are "gates" of various responses of living things
  - A ligand is a chemical compound
  - It activates (agonist / activator) or inhibits (antagonist / inhibitor) receptors



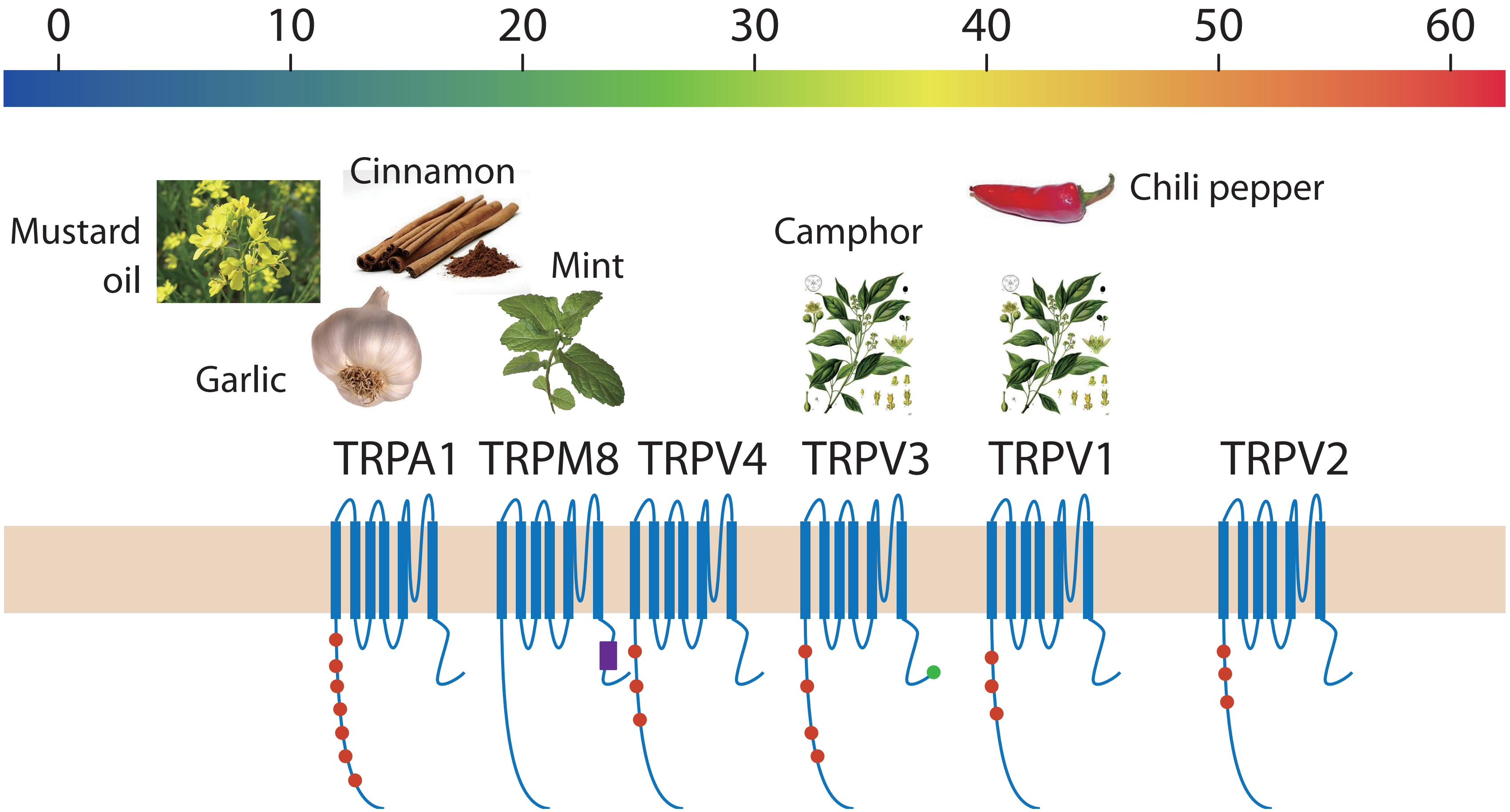
Example: TRP Ion Channels

- TRP ion channels form a family of ligand-gated channels

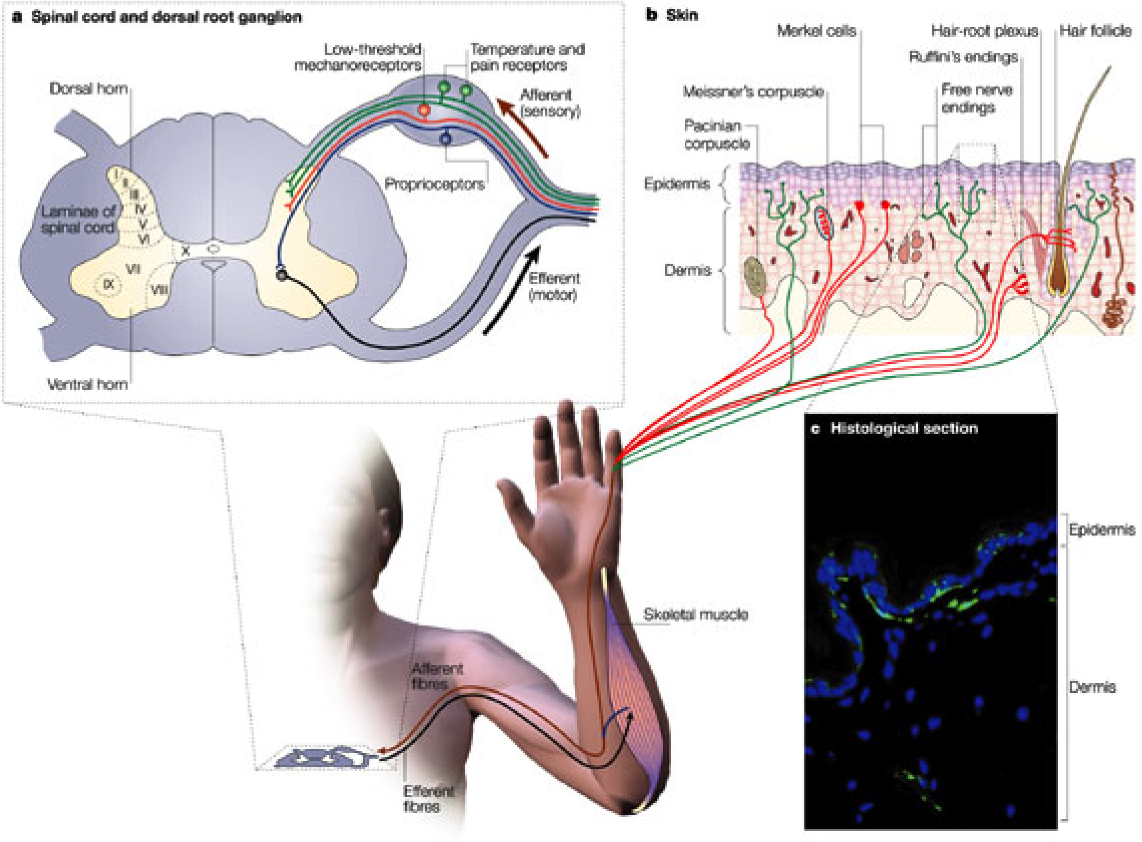


Dhaka A, et al. 2006.
Annu. Rev. Neurosci. 29:135–61

ThermoTRPs


0
10
20
30
40
50
60
Mustard oil
Cinnamon
Garlic
Mint
Camphor
Chili pepper
TRPA1
TRPM8
TRPV4
TRPV3
TRPV1
TRPV2

Signal Path of ThermoTRPs



Nature Reviews | Neuroscience

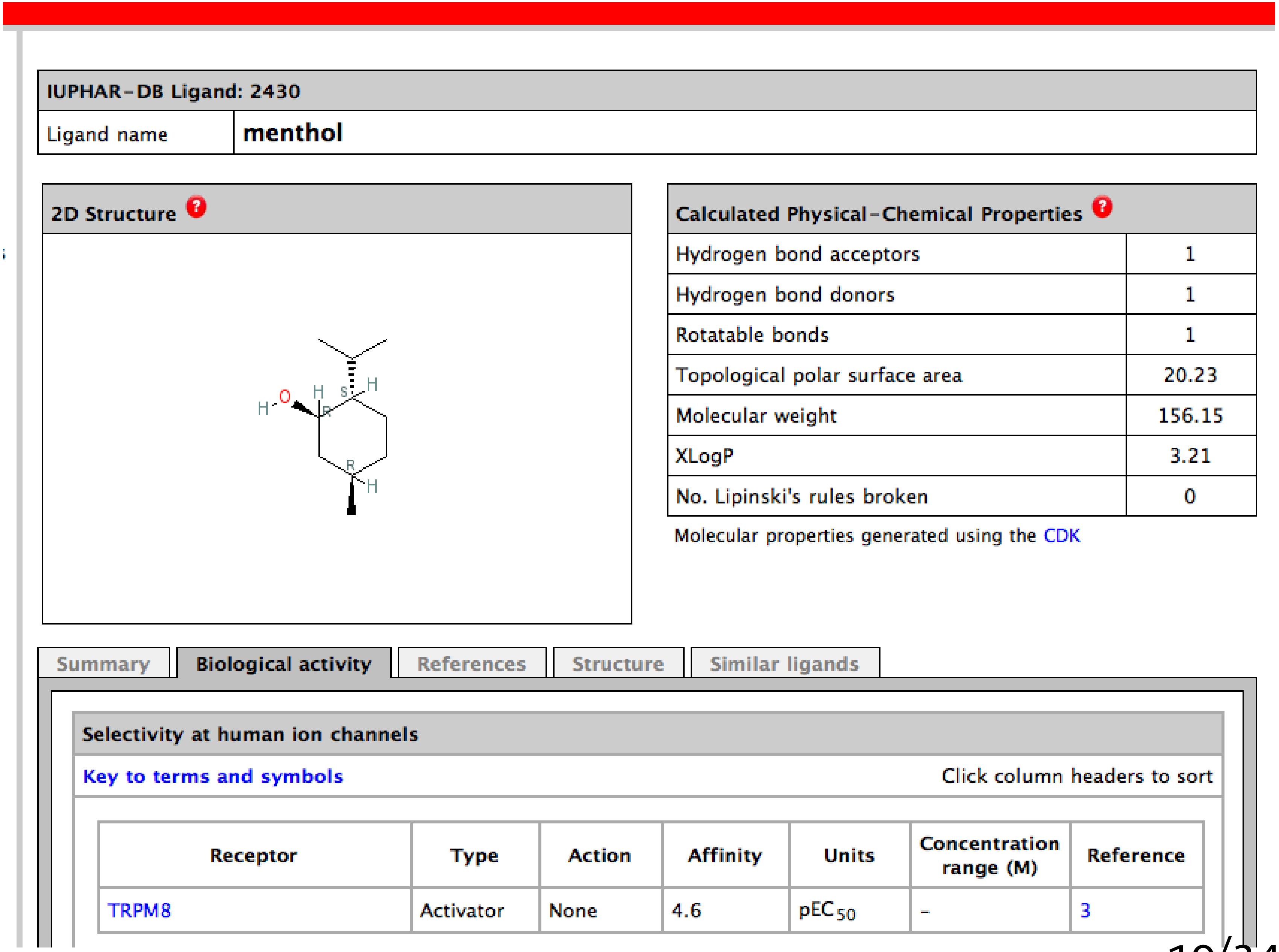
Patapoutian, A *et al.*, Nat. Rev. Neurosci. 4, 529–539

Problems

- Finding new convenient ligands is difficult
  - Choosing ligand candidates relies on expert knowledge of biologists
  - Conducting experiments to test whether or not candidates work *in vivo* or *in vitro* has a high cost in terms of time and money
- An *in silico* approach is required for helping biologists

Strategy

- We find candidates of ligands from databases in a machine learning approach
- We use IUPHAR database [Sharman *et al.*, 2011]
  - 1,782 ligands in total
  - Each ligand is characterized by 7 features
    - 3 continuous and 4 discrete variables
  - For each ligand, we can know ligands to which it binds, corresponding to class labels



IUPHAR-DB Ligand: 2430	
Ligand name	menthol

2D Structure ?

Calculated Physical-Chemical Properties ?

Hydrogen bond acceptors	1
Hydrogen bond donors	1
Rotatable bonds	1
Topological polar surface area	20.23
Molecular weight	156.15
XLogP	3.21
No. Lipinski's rules broken	0

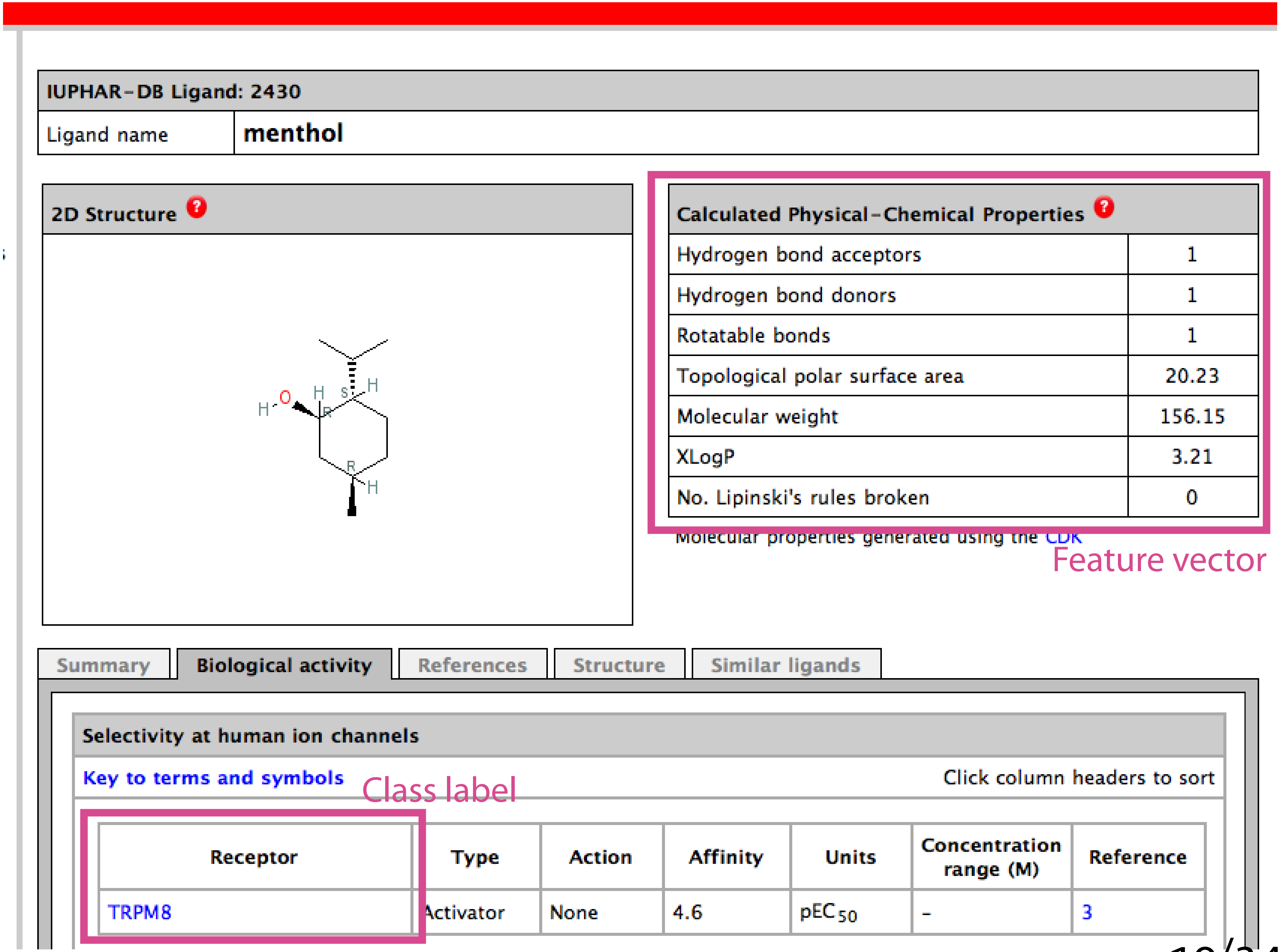
Molecular properties generated using the CDK

Summary | Biological activity | References | Structure | Similar ligands

Selectivity at human ion channels

Key to terms and symbols — Click column headers to sort

Receptor	Type	Action	Affinity	Units	Concentration range (M)	Reference
TRPM8	Activator	None	4.6	pEC_{50}	-	3



IUPHAR-DB Ligand: 2430	
Ligand name	**menthol**

2D Structure

Calculated Physical-Chemical Properties	
Hydrogen bond acceptors	1
Hydrogen bond donors	1
Rotatable bonds	1
Topological polar surface area	20.23
Molecular weight	156.15
XLogP	3.21
No. Lipinski's rules broken	0

Molecular properties generated using the CDK

Feature vector

Selectivity at human ion channels

Key to terms and symbols

Click column headers to sort

Class label

Receptor	Type	Action	Affinity	Units	Concentration range (M)	Reference
TRPM8	Activator	None	4.6	pEC_{50}	-	3

Mathematical Modeling

- We model ligand finding as multi-label classification
  - Ligand: Data point
  - Receptor: Class label
    - Each object has not a single label but a set of possible labels
    - Recently discussed in preference learning
- In the database, there are lots of ligands associated with non-related receptors
  - We adopt semi-supervised learning by using them as unlabeled data

Key Approach

- We use Formal Concept Analysis (FCA)
  - Used for frequent pattern mining [Pasquier *et al.*, 1999]
    - Closed patterns obtained by FCA is used as condensed "lossless" representations of original patterns
  - Databases are treated in a discrete manner
  - Fast algorithm LCM has been already proposed by Uno [Uno *et al.*, 2005]
- We discretize continuous values based on the binary encoding scheme to exploit FCA

Related Work (ML)

- No study treats machine learning for ligand finding in classification point of view
- There exists only one related study [Ballester and Mitchell, 2010]
  - A machine learning approach to predict the affinity of ligands
- Another approach was performed by King *et al.* [King *et al.*, 1996] for modeling structure-activity relationships (SAR)
  - However, their goal is to understand the chemical model by describing relations using inductive logic programming (ILP)

Related Work (*in silico* modeling)

- Most *in silico* studies about receptors and ligands [Huang *et al.*, 2006; Moitessier *et al.*, 2008] tried to construct a predictive model using domain-specific knowledge
  - The potential energy of a complex
  - The two-dimensional co-ordinates
  - The free energy of binding
- Lots of scoring methods were proposed; *e.g.*, AMBER [Cornell *et al.*, 1995], AutoDock [Huey *et al.*, 2007], and DrugScore [Gohlke *et al.*, 2000]
- However, some domain-specific background knowledge is required and results depend on them

Contents

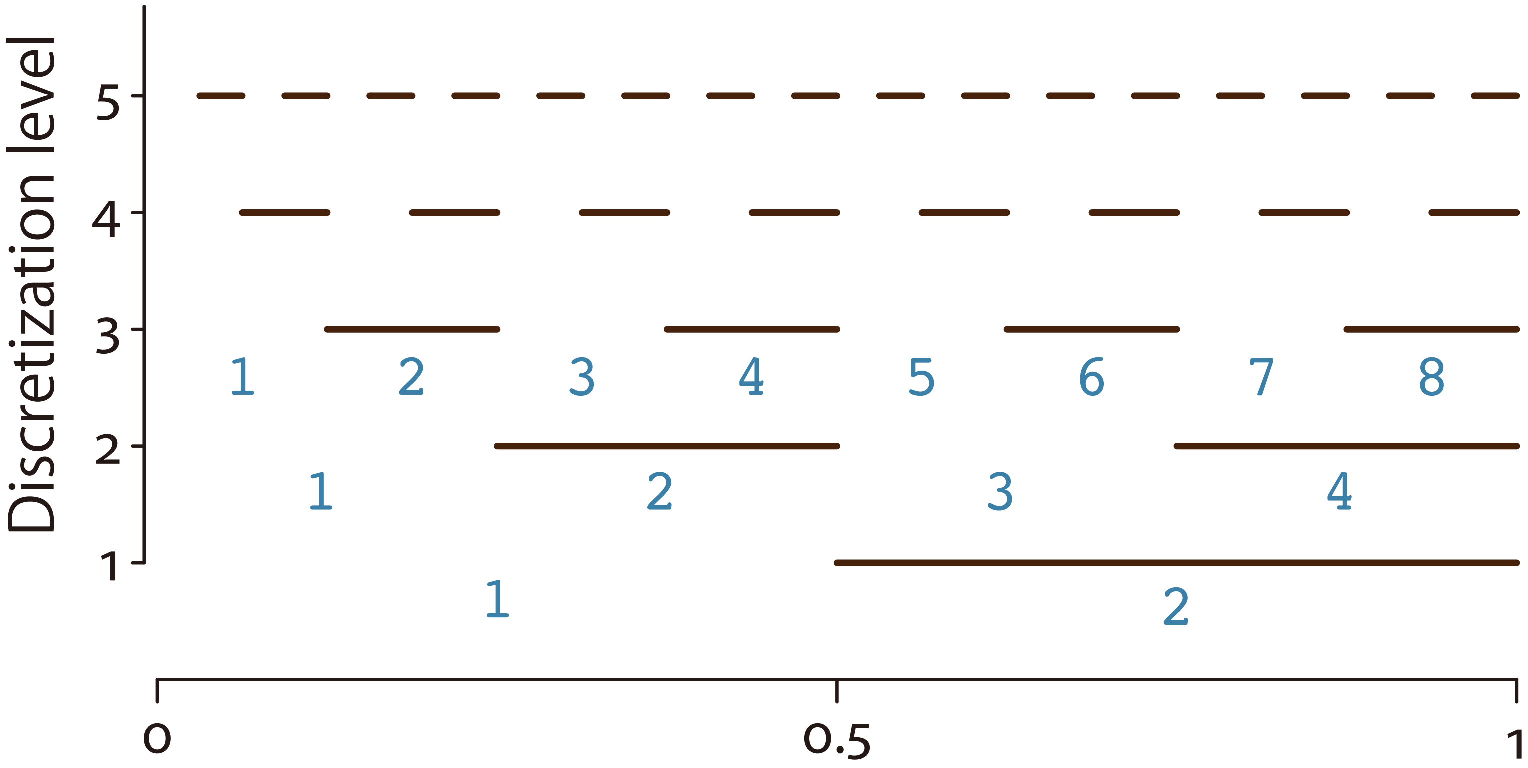
1. Motivation
2. Classification by LIFT
   - i) Data preprocessing
   - ii) Making concepts by LIFT
   - iii) Classifying unlabeled data
3. Experiments
4. Conclusion

Example of Classification

- Classify the test object y

	HBD	TPS	MW	Labels	
x_1	0	0.98	0.88	A	
x_2	1	0.41	0.48	B	C
x_3	2	0.12	0.71	A	C
y	0	0.77	0.79		

Binary Encoding

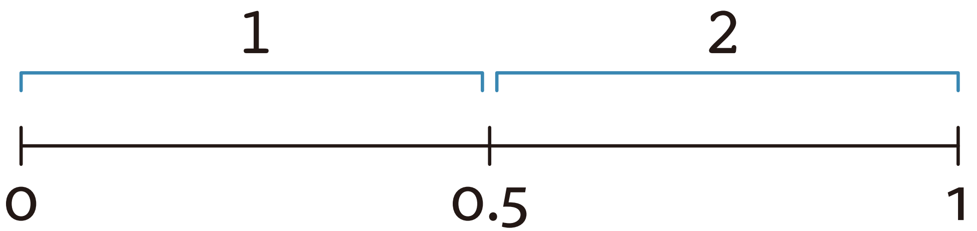


Discretization at level 1

- Classify the test object *y*

	HBD	TPS	MW	Labels
x_1	0	0.98	0.88	A
x_2	1	0.41	0.48	B C
x_3	2	0.12	0.71	A C
y	0	0.77	0.79	

  - At discretization level 1



Data preprocessing

- Classify the test object *y*

	HBD	TPS	MW	Labels	
x_1	0	0.98	0.88	A	
x_2	1	0.41	0.48	B	C
x_3	2	0.12	0.71	A	C
y	0	0.77	0.79		

  - It is converted as follows:

	H.0	H.1	H.2	T.1	T.2	M.1	M.2
x_1	×				×		×
x_2		×		×		×	
x_3			×	×			×
y	×				×		×

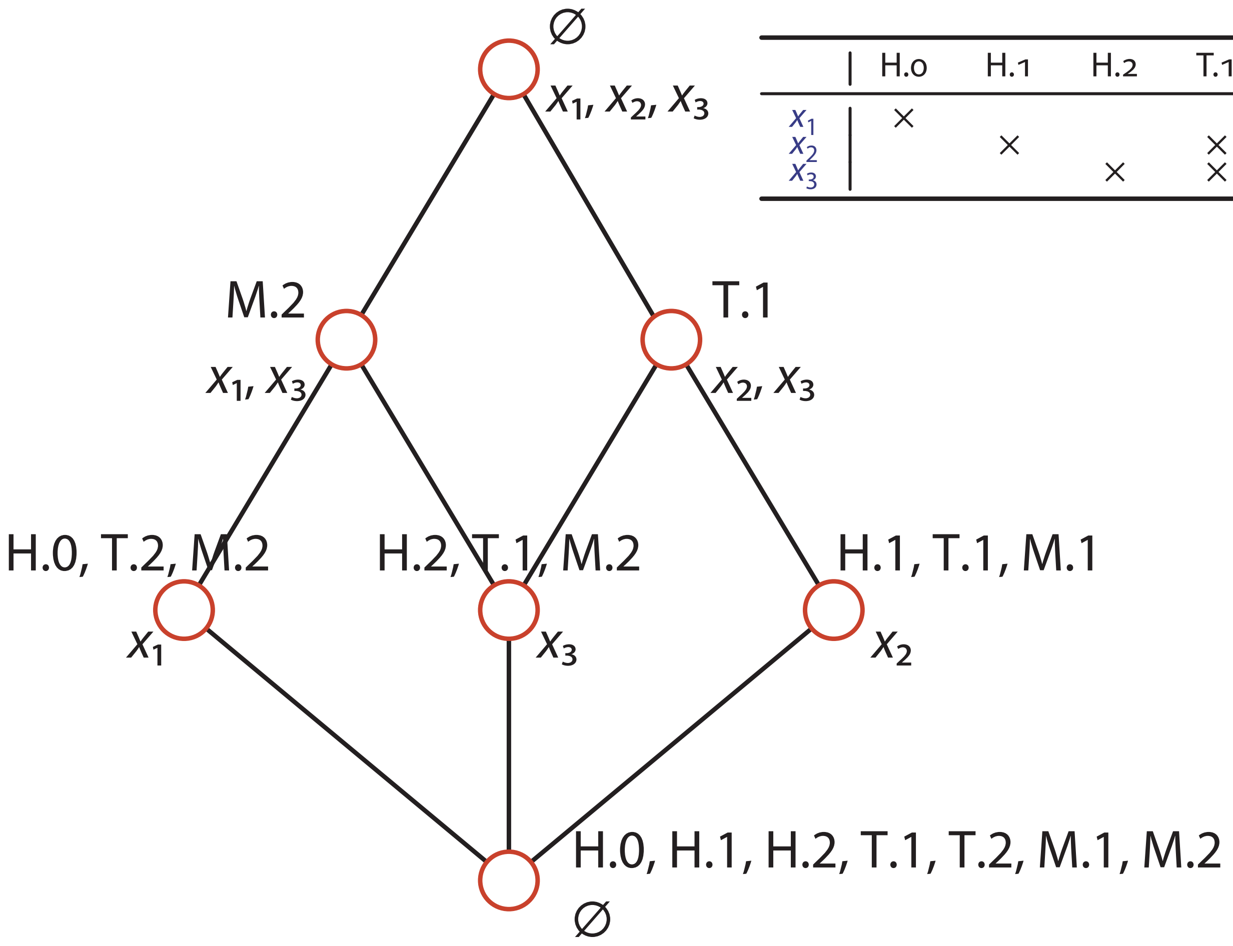
Method to Calculate Preference

- Classify the test object y

	H.0	H.1	H.2	T.1	T.2	M.1	M.2
x_1	×				×		×
x_2		×		×		×	
x_3			×	×			×
y	×				×		×

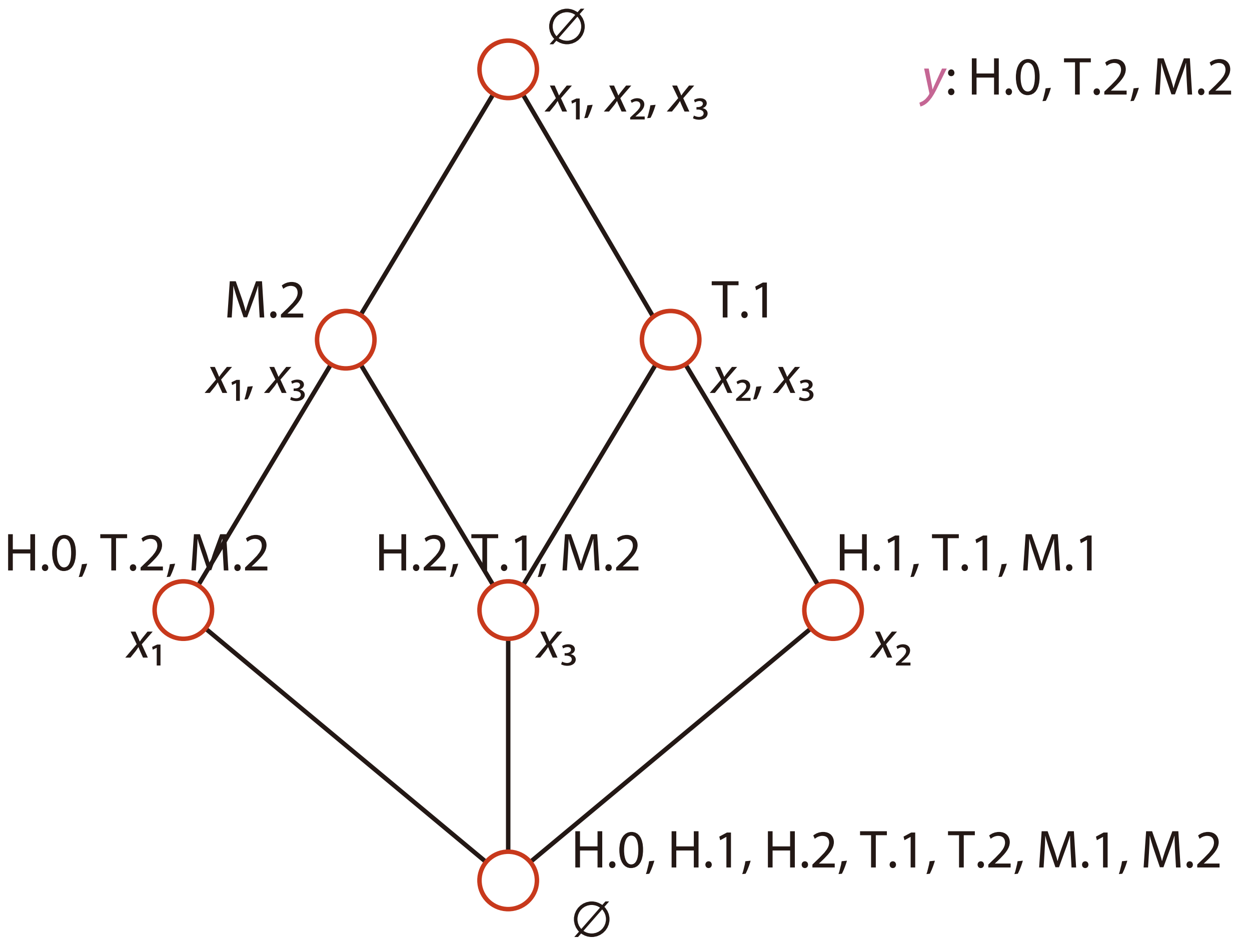
1. Make a concept lattice from training data x_1, x_2, x_3 by FCA
2. Find consistent concepts
3. Calculate preference for each label

Concept Lattice by FCA

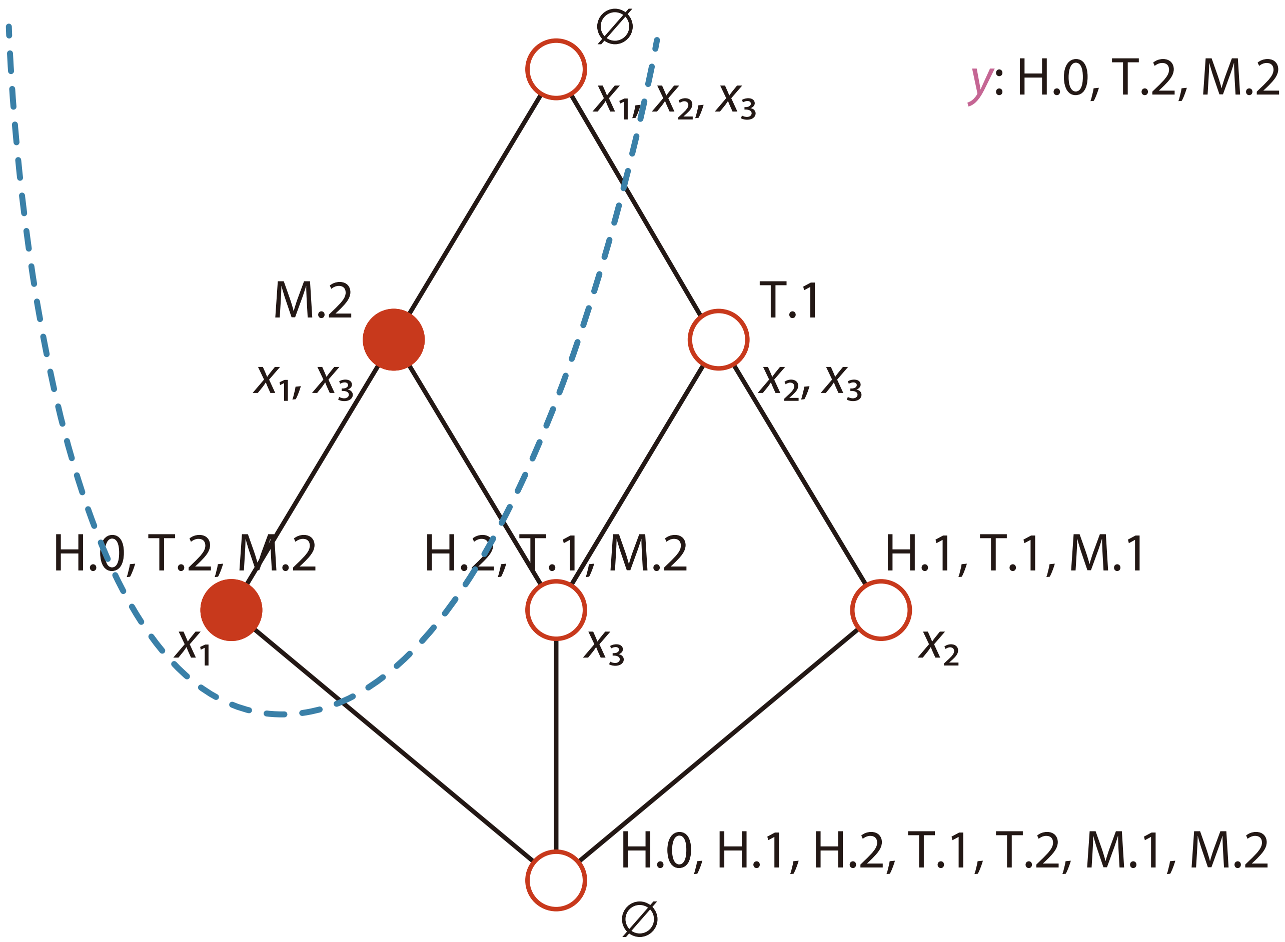


	H.0	H.1	H.2	T.1	T.2	M.1	M.2
x_1	×				×		×
x_2		×		×		×	
x_3			×	×			×

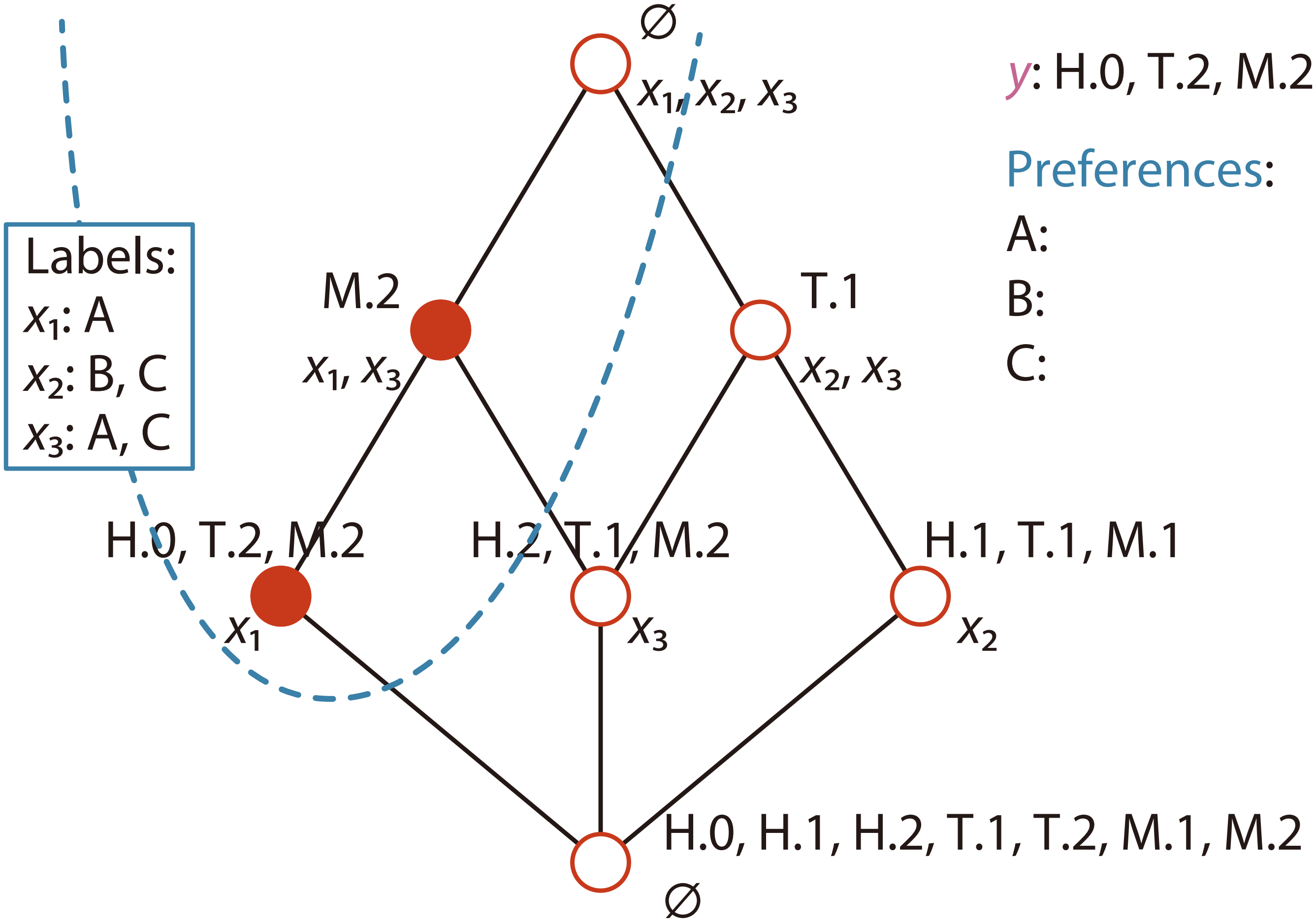
Learning on the Lattice



Learning on the Lattice



Learning on the Lattice

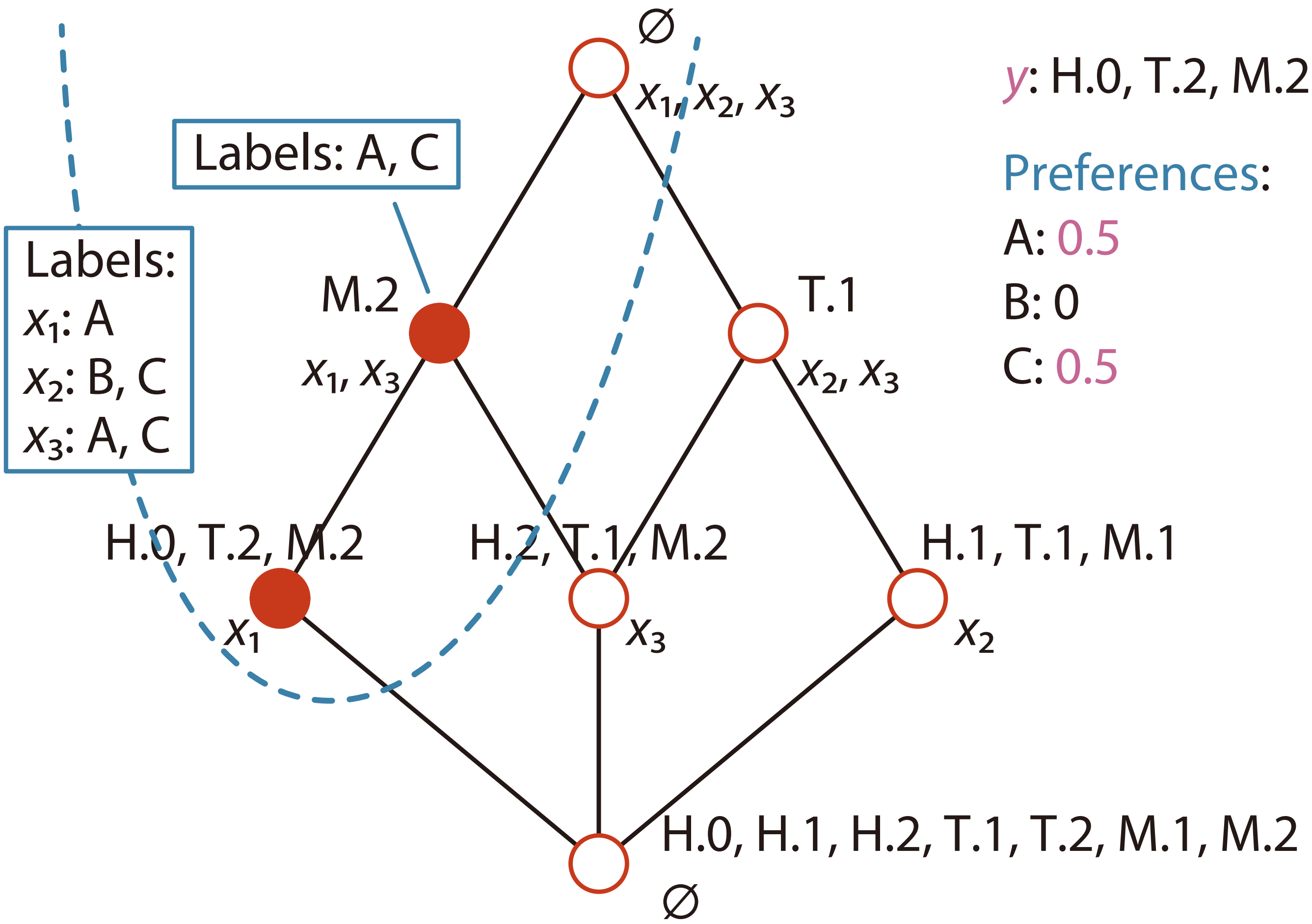


Labels:
x_1: A
x_2: B, C
x_3: A, C

y: H.0, T.2, M.2

Preferences:
A:
B:
C:

Learning on the Lattice



y: H.0, T.2, M.2

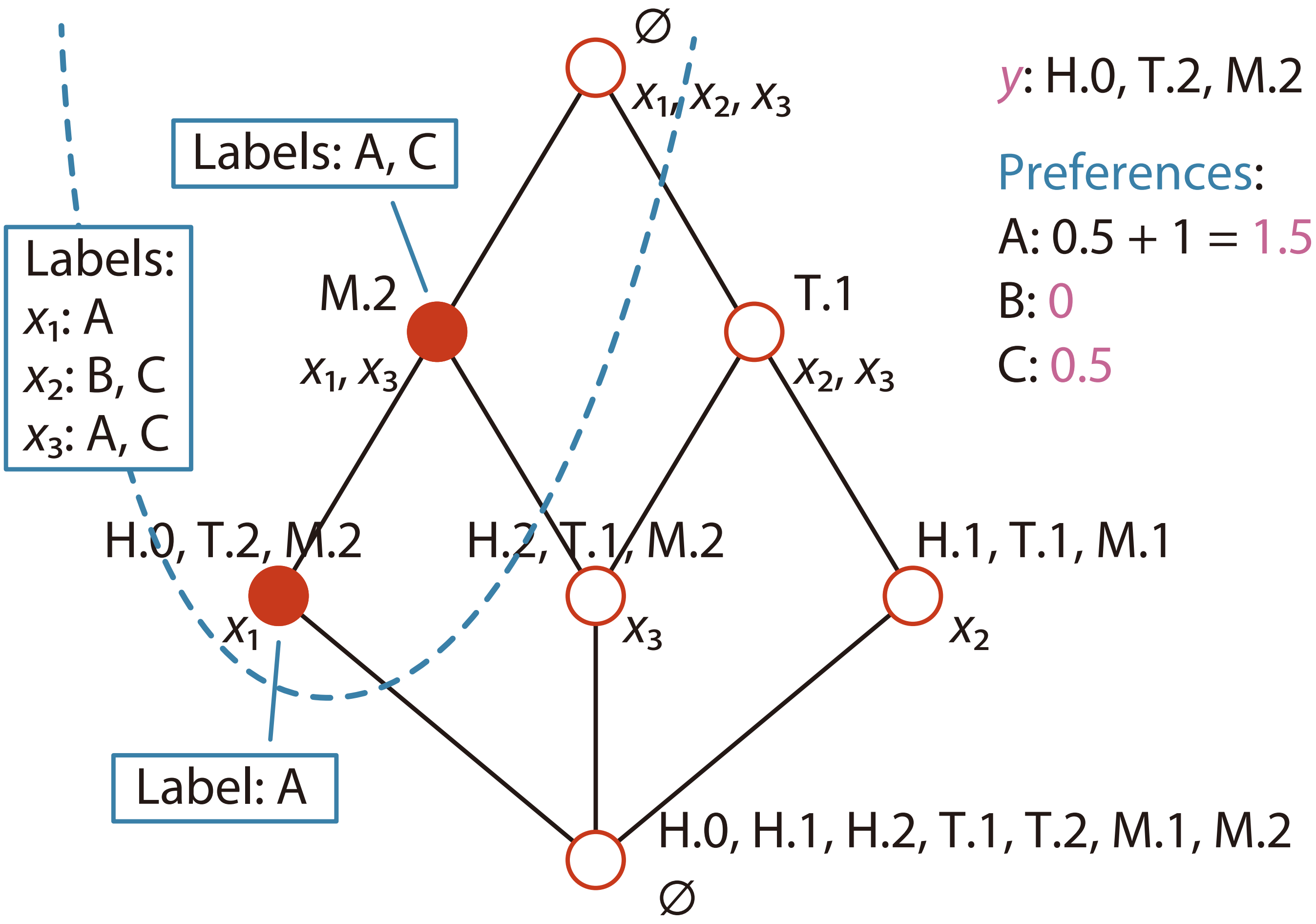
Preferences:

A: 0.5

B: 0

C: 0.5

Learning on the Lattice



y: H.0, T.2, M.2

Preferences:

A: 0.5 + 1 = 1.5

B: 0

C: 0.5

Preferences

- At discretization level 1, preferences are

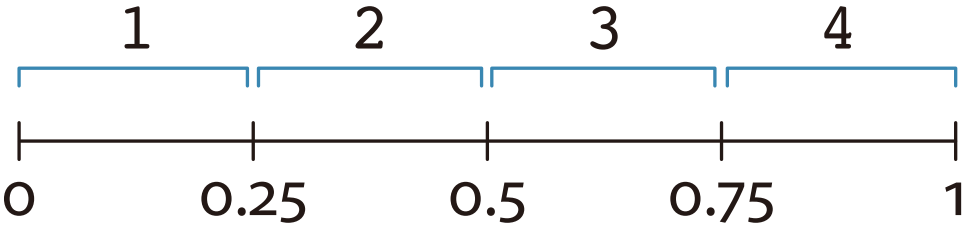
  $\psi_y^1(\mathrm{A}) = 1.5$, $\psi_y^1(\mathrm{B}) = 0$, and $\psi_y^1(\mathrm{C}) = 0.5$

Discretization at level 2

- Classify the test object y

	HBD	TPS	MW	Labels	
x_1	0	0.98	0.88	A	
x_2	1	0.41	0.48	B	C
x_3	2	0.12	0.71	A	C
y	0	0.77	0.79		

  - At discretization level 2



Data preprocessing

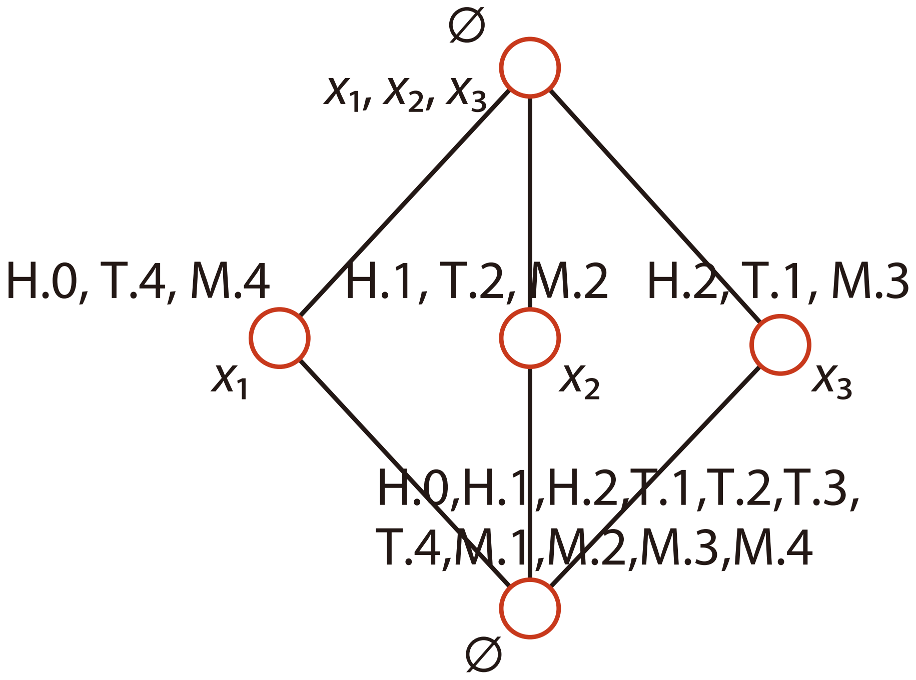
- Classify the test object y

	HBD	TPS	MW	Labels	
x_1	0	0.98	0.88	A	
x_2	1	0.41	0.48	B	C
x_3	2	0.12	0.71	A	C
y	0	0.77	0.79		

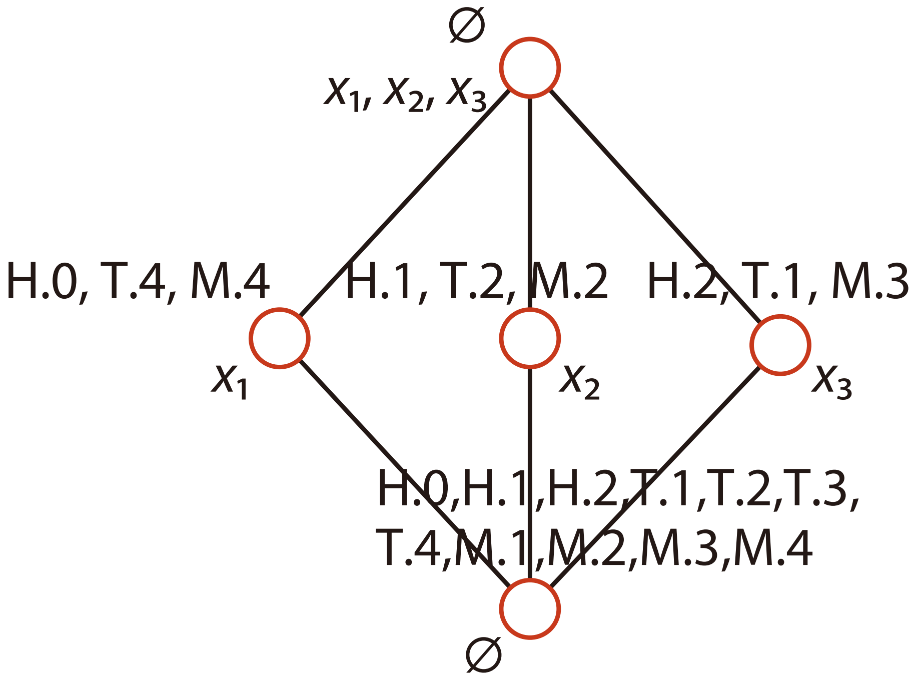
  - It is converted as follows:

	H.0	H.1	H.2	T.1	T.2	T.3	T.4	M.1	M.2	M.3	M.4
x_1	×						×				×
x_2		×			×				×		
x_3			×	×						×	
y	×						×				×

Concept Lattice by FCA

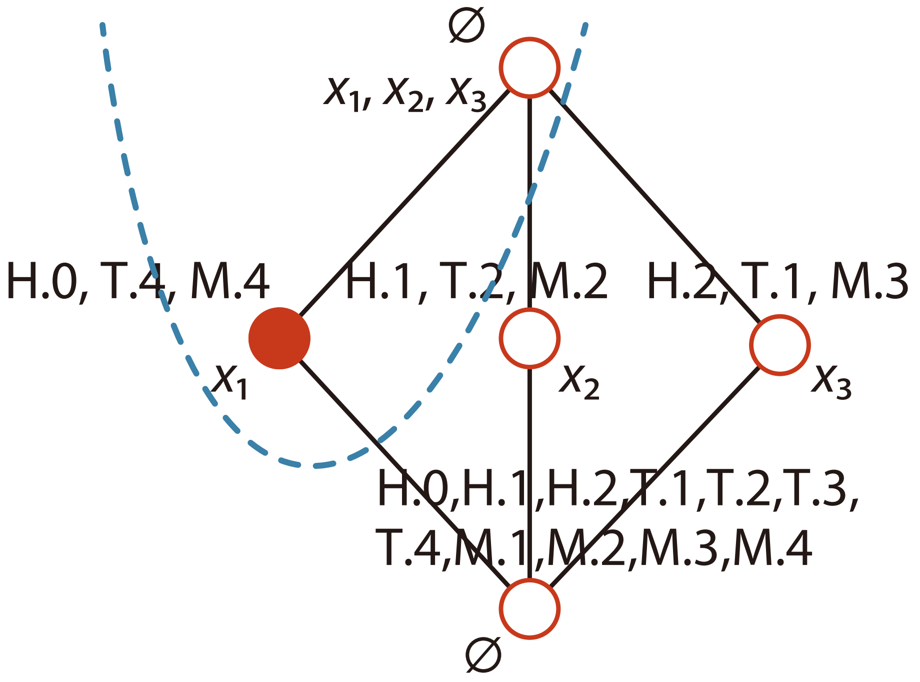


Learning on the Lattice



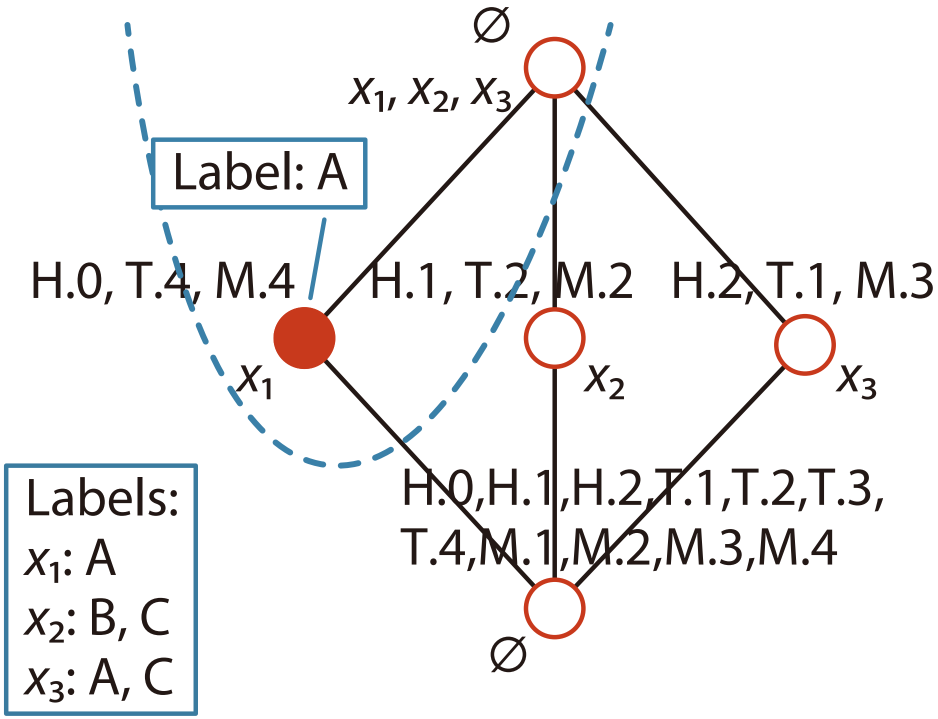
y: H.0, T.4, M.4

Learning on the Lattice



y: H.0, T.4, M.4

Learning on the Lattice



Labels:
x_1: A
x_2: B, C
x_3: A, C

y: H.0, T.4, M.4

Preferences:

A: 1
B: 0
C: 0

Preferences

- At discretization level 1, preferences are

  $\psi_y^1(\mathrm{A}) = 1.5$, $\psi_y^1(\mathrm{B}) = 0$, and $\psi_y^1(\mathrm{C}) = 0.5$
- At discretization level 2, preferences are

  $\psi_y^2(\mathrm{A}) = 1$, $\psi_y^2(\mathrm{B}) = 0$, and $\psi_y^2(\mathrm{C}) = 0$

Preferences

- At discretization level 1, preferences are

  $\psi_y^1(A) = 1.5$, $\psi_y^1(B) = 0$, and $\psi_y^1(C) = 0.5$
- At discretization level 2, preferences are

  $\psi_y^2(A) = 1$, $\psi_y^2(B) = 0$, and $\psi_y^2(C) = 0$
- Preference for each label is

  $\psi_y(A) = 1.5 + 1 = 2.5$,

  $\psi_y(B) = 0 + 0 = 0$,

  $\psi_y(C) = 0.5 + 0 = 0.5$
  - y should be associated with the label A, B
  - y's label ranking is A > C > B
    - The maximum level $k_{\max} = 2$ is given by the user

Definition for Preference

- For a context $(\{y\}, M, I)$ and a concept (A, B), y is consistent with $(A, B) \iff B \subseteq \{m \in M \mid (y, m) \in I\}$ and $B \neq \emptyset$
- Given tables $\tau = (H, X)$ and $\upsilon = (H, y)$ with $|\upsilon| = 1$.
  For each discretization level k and each label $\lambda \in \mathscr{L}$, define the preference of λ at discretization level k w.r.t. y by

  $$\psi_y^k(\lambda|\tau) := \textstyle\sum_{A \in \mathbf{A}} \#\Lambda(A)^{-1}, \text{ where}$$

  $$\mathbf{A} := \{A \mid y \text{ is consistent with } (A, B) \in \mathscr{B}^k(\tau) \text{ with } \lambda \in \Lambda(A)\}$$

  – Assume $\#\Lambda(A)^{-1} = 0 \iff \#\Lambda(A) = 0$ for simplicity
- Given a natural number $k_{\max}$ (input parameter).
  For each label $\lambda \in \mathscr{L}$ and y, define the preference of λ by

  $$\psi_y(\lambda|\tau) := \sum_{k=1}^{k_{\max}} \psi_y^k(\lambda|\tau)$$

Algorithm for Data Preprocessing

Input: Table $\tau = (H, X)$ and discretization level k
Output: Context (G, M^k, I^k)

```
function Context($\tau$, $k$)
    $G \leftarrow \mathrm{set}(X)$
    for each feature $h \in H$
      if Drom($h$) = $\mathbb{N}$ then $(M_h, I_h) \leftarrow$ ContextD($X$, $h$)
      else if Drom($h$) = $\mathbb{R}$ then $(M_h, I_h) \leftarrow$ ContextC($X$, $h$, $k$)
      end if
    combine $(M_{\mathrm{HBA}}, I_{\mathrm{HBA}}), (M_{\mathrm{HBD}}, I_{\mathrm{HBD}}), \dots, (M_{\mathrm{NLR}}, I_{\mathrm{NLR}})$ into $(M^k, I^k)$
    return $(G, M^k, I^k)$
```

Algorithm for Data Preprocessing

```
function ContextD($X, h$)
    $M \leftarrow \{h.m \mid m \in x(h)$ such that $x \in \text{set}(X)\}$
    $I \leftarrow \{(x, h.m) \mid x \in \text{set}(X)$ and $x(h) = m\}$
    return $(M, I)$

function ContextC($X, h, k$)
    $M \leftarrow \{1, 2, \ldots, 2^k\}, I \leftarrow \emptyset$
    Normalize the set $\{x(h) \mid x \in \text{set}(X)\}$
    for each $x \in \text{set}(X)$
      if $x(h) = 0$ then $I \leftarrow I \cup \{(x, h.1)\}$
      else if $x(h) \neq 0$ then
        $I \leftarrow I \cup \{(x, h.a)\}$, where $(a - 1) \cdot 2^{-k} < x(h) \leq a \cdot 2^{-k}$
      end if
    end for
    return $(M, I)$
```

The LIFT Algorithm

Input: Tables $\tau = (H, X)$ and $\upsilon = (H, y)$, and maximum level k_{max}
Output: Preference ψ_y for each label $\lambda \in \mathscr{L}$

```
function LIFT($\tau$, $\upsilon$, $k_{\mathrm{max}}$)
    $k \leftarrow 1$    // $k$ is discretization level
    for each label $\lambda \in \mathscr{L}$
      $\psi_y(\lambda|\tau) \leftarrow 0$    // initialization
    end for
    return Learning($\tau, \upsilon, k, k_{\mathrm{max}}$)
```

The LIFT Algorithm

```
function Learning($\tau, \upsilon, k, k_{\max}$)
    $(G(\tau), M^k(\tau), I^k(\tau)) \leftarrow \mathrm{Context}(\tau, k)$    // make a context from $\tau$
    $(G(\upsilon), M^k(\upsilon), I^k(\upsilon)) \leftarrow \mathrm{Context}(\upsilon, k)$    // make a context from $\upsilon$
    make a concept lattice $k(\tau)$ from $(G(\tau), M^k(\tau), I^k(\tau))$ by FCA
    for each label $\lambda \in \mathscr{L}$
      compute the preference $\psi_y^k(\lambda|X)$ at discretization level $k$
      $\psi_y(\lambda|X) \leftarrow \psi_y(\lambda|X) + \psi_y^k(\lambda|X)$
    end for
    if $k = k_{\max}$ then
      return $(\psi_y(\lambda|\tau))_{\lambda \in \mathscr{L}}$
    else
      return Learning($\tau, \upsilon, k + 1, k_{\max}$)
    end if
```

Contents

1. Motivation
2. Classification by LIFT
   i) Data preprocessing
   ii) Making concepts by LIFT
   iii) Classifying unlabeled data
3. Experiments
4. Conclusion

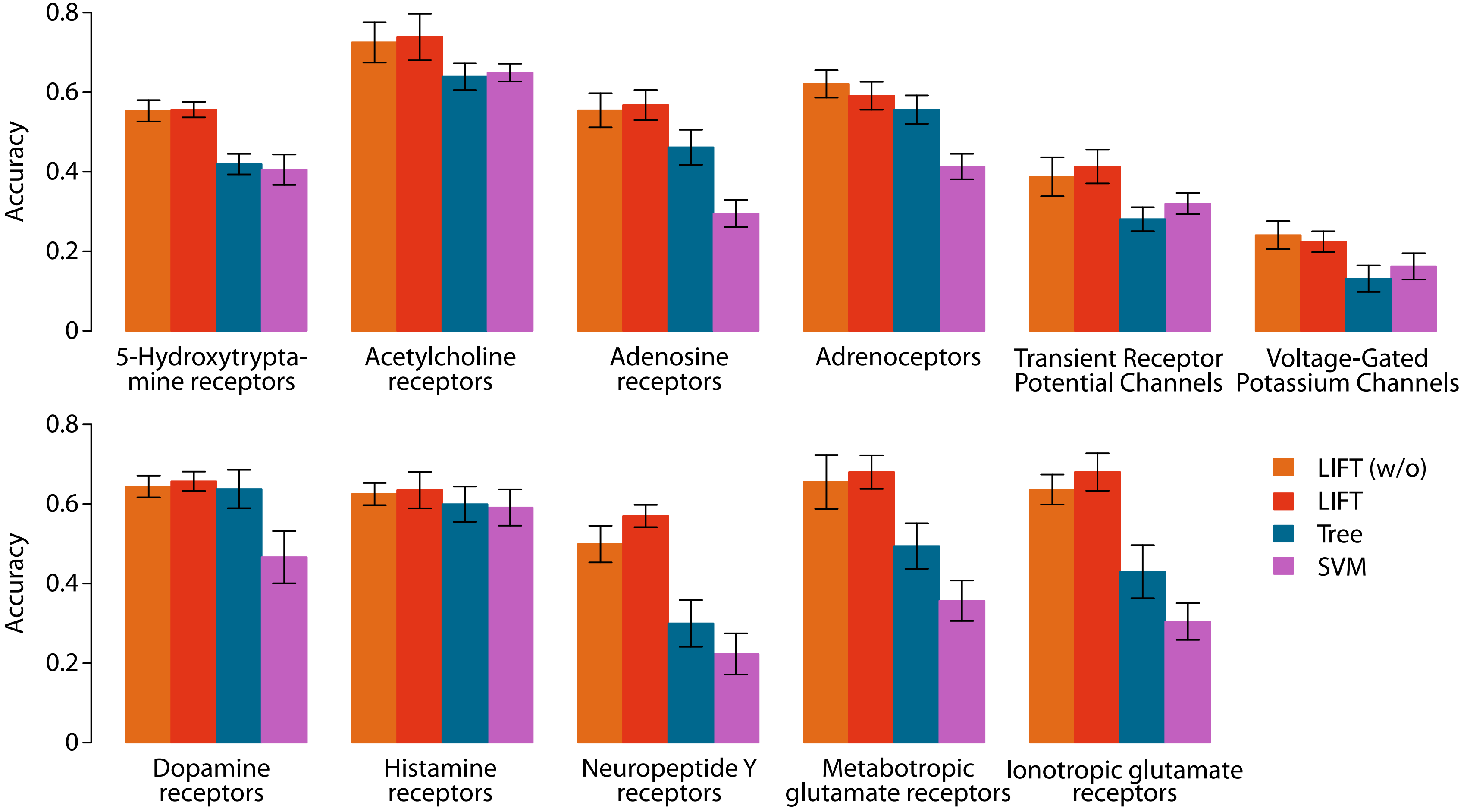
Experimental Methods

- LIFT was implemented in R version 2.12.2
  - LIFT uses LCM [Uno *et al.*, 2005] to construct a concept lattice
- We collected the entire 1,782 ligands in the IUPHAR database
- To measure the effectiveness of unlabeled ligand data, we used LIFT in two cases:
  1. Only labeled data were used in training
  2. All ligands except test data were used as unlabeled training data
- The maximum level k_{max} was set at 5
- 10-fold cross validation
- Control methods: SVM (RBF kernel) and the decision tree-based method implemented in R

Families of receptors

Family name	# Ligands (Data size)	# Receptors (Class size)
5-Hydroxytryptamine receptors	286	53
Acetylcholine receptors	100	68
Adenosine receptors	162	40
Adrenoceptors	111	35
Dopamine receptors	69	40
Histamine receptors	120	37
Neuropeptide Y receptors	76	34
Metabotropic glutamate receptors	73	9
Transient receptor potential channels	78	58
Voltage-gated potassium channels	61	71
Ionotropic glutamate receptors	81	14

Experimental Results



Contents

1. Motivation
2. Classification by LIFT
   - i) Data preprocessing
   - ii) Making concepts by LIFT
   - iii) Classifying unlabeled data
3. Experiments
4. Conclusion

Conclusion

- We have mathematically modeled the ligand finding problem as multi-label classification
- We have proposed the semi-supervised learning algorithm LIFT
  - Every dataset is translated into a context, followed by clustering of it by FCA by putting on a concept lattice
    - Each continuous value is discretized based on the binary encoding scheme
- LIFT should be valuable for finding new ligands compared to other machine learning methods
  - It may contribute to biology and biochemistry

Appendix

Related Work (FCA)

- Many studies used FCA for machine learning and knowledge discovery
  - Classification [Ganter and Kuznetsov, 2003]
  - Clustering [Zhang *et al.*, 2008]
  - Association rule mining [Pasquier *et al.*, 1999]
  - Bioinformatics [Blinova *et al.*, 2003; Kaytoue *et al.*, 2011; Kuznetsov and Samokhin, 2005]
- Ganter and Kuznetsov treated 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*

Notation

- A set of ligands is treated as a table $\tau = (H, X)$ (H: header, X: body)
  - The domain of h is denoted by $\mathrm{Drom}(h)$
  - A body X is a sequence of tuples $x_1, x_2, \ldots, x_n$
    - Each tuple x_i is a total function from H to $\mathrm{Drom}(H)$ such that $x_i(h) \in \mathrm{Drom}(h)$ for all $h \in H$.
  - $|\tau|$: the number of tuples (the table size) n
  - $\mathrm{set}(X)$: the body X treated as a set
- In the IUPHAR database, the header H is always the set {HBA, HBD, RB, TPS, MW, XLogP, NLR}, and

$$\mathrm{Drom}(\mathrm{HBA}) = \mathrm{Drom}(\mathrm{HBD}) = \mathrm{Drom}(\mathrm{RB}) = \mathrm{Drom}(\mathrm{NLR}) = \mathbb{N},$$

$$\mathrm{Drom}(\mathrm{TPS}) = \mathrm{Drom}(\mathrm{MW}) = \mathrm{Drom}(\mathrm{XLogP}) = \mathbb{R}$$

- The projection of x on $J \subset H$, denoted by $x|_J$, is exactly the same as the restriction of x to J

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 $O(N)$
  - N is the maximum number of concepts in concept lattices constructed in the learning process of LIFT
- The time complexity of SELF is $O(nd) + O(\Delta^3) + O(N)$

Data Preprocessing (mixed-type)

- Objective: Convert a given dataset to a context for FCA
  - Example:

	HBD	TPS	MW
x_1	0	0.61	0.98
x_2	0	0.44	0.74
x_3	1	0.72	0.34

Data Preprocessing (mixed-type)

- Objective: Convert a given dataset to a context for FCA
  - Example:

	HBD	TPS	MW
x_1	0	0.61	0.98
x_2	0	0.44	0.74
x_3	1	0.72	0.34

  - First, HBD is converted as follows:

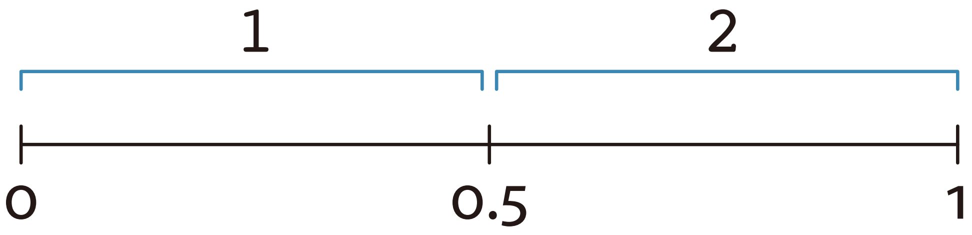
	HBD.0	HBD.1
x_1	×	
x_2	×	
x_3		×

Data Preprocessing (mixed-type)

- Objective: Convert a given dataset to a context for FCA
  - Example:

	HBD	TPS	MW
x_1	0	0.61	0.98
x_2	0	0.44	0.74
x_3	1	0.72	0.34

  - Second, discretize continuous values using binary encoding at discretization level 1



Data Preprocessing (mixed-type)

- Objective: Convert a given dataset to a context for FCA
  - Example:

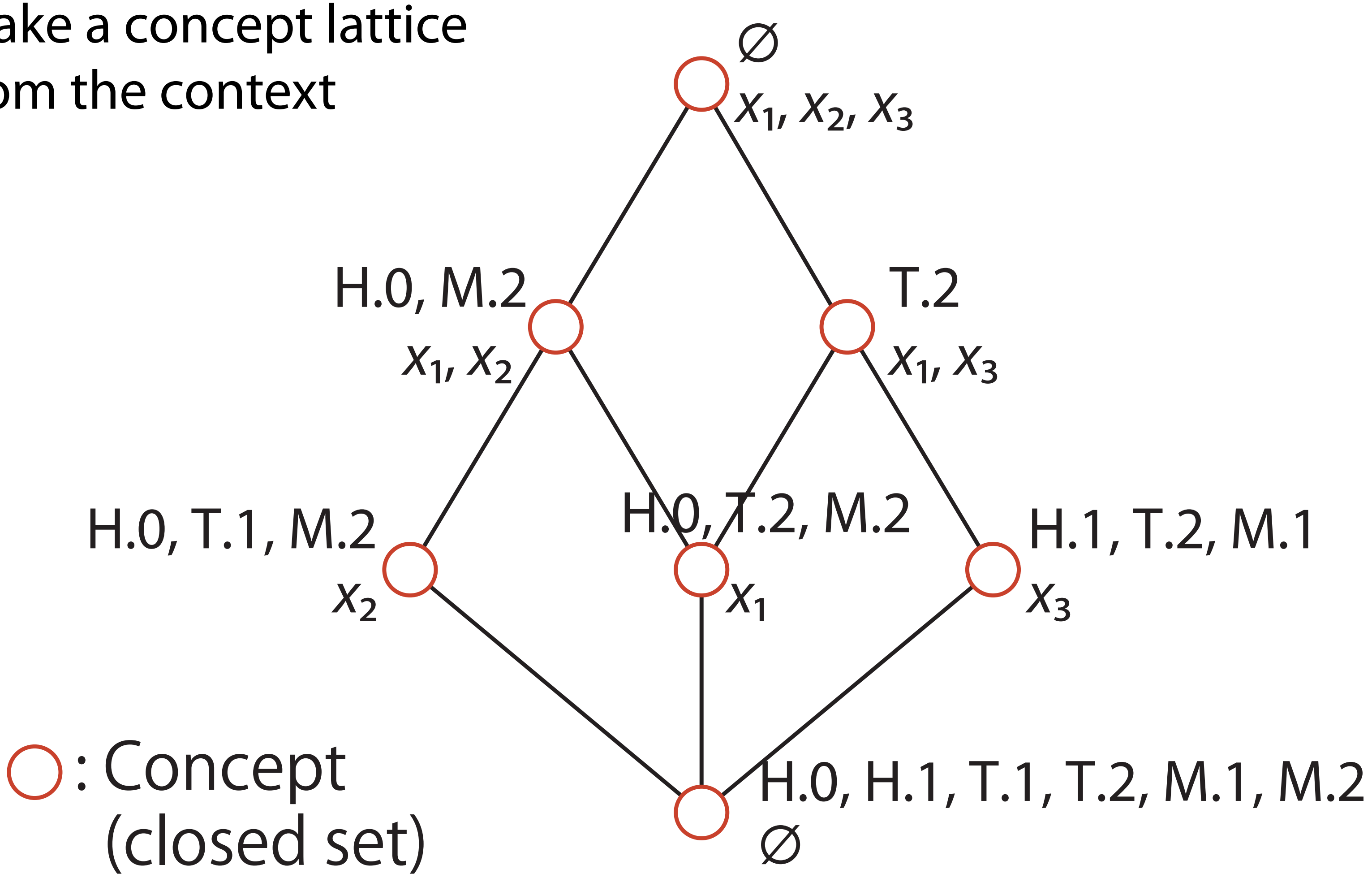
	HBD	TPS	MW
x_1	0	0.61	0.98
x_2	0	0.44	0.74
x_3	1	0.72	0.34

  - Finally, it is converted as follows:

	HBD.0	HBD.1	TPS.1	TPS.2	MW.1	MW.2
x_1	×			×		×
x_2	×		×			×
x_3		×		×	×	

Make a Concept Lattice by FCA

- Make a concept lattice from the context



○ : Concept (closed set)

Data Preprocessing (continuous)

- Objective: Convert a given dataset to a context for FCA
  - Example:

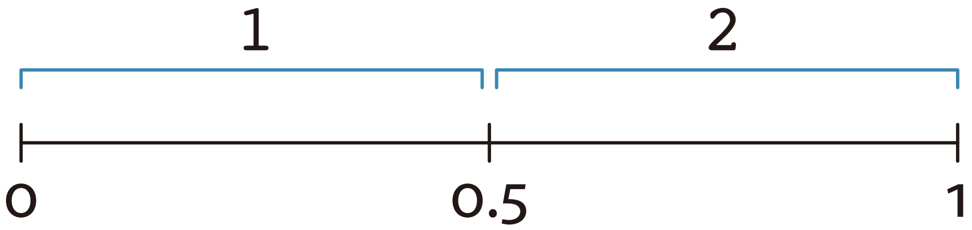
	TPS	MW	XLogP
x_1	0.23	0.12	0.18
x_2	0.35	0.03	0.74
x_3	0.41	0.79	0.91

Data Preprocessing (continuous)

- Objective: Convert a given dataset to a context for FCA
  - Example:

	TPS	MW	XLogP
x_1	0.23	0.12	0.18
x_2	0.35	0.03	0.74
x_3	0.41	0.79	0.91

  - Discretize continuous values using binary encoding



Data Preprocessing (continuous)

- Objective: Convert a given dataset to a context for FCA
  - Example:

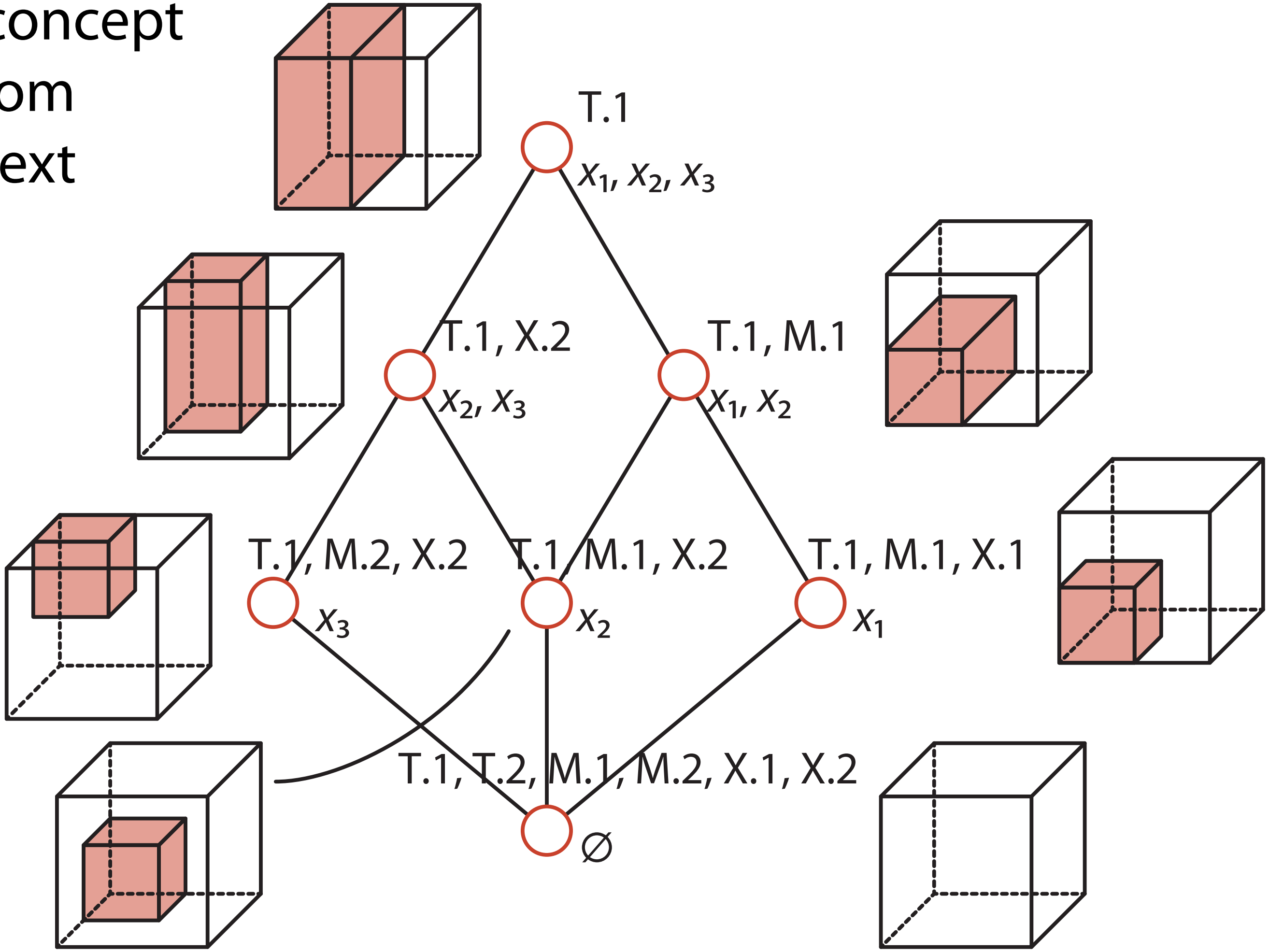
	TPS	MW	XLogP
x_1	0.23	0.12	0.18
x_2	0.35	0.03	0.74
x_3	0.41	0.79	0.91

  - It is converted as follows (discretization level 1):

	TPS.1	TPS.2	MW.1	MW.2	XLogP.1	XLogP.2
x_1	×		×		×	
x_2	×		×			×
x_3	×			×		×

Make a Concept Lattice by FCA

- Make a concept lattice from the context

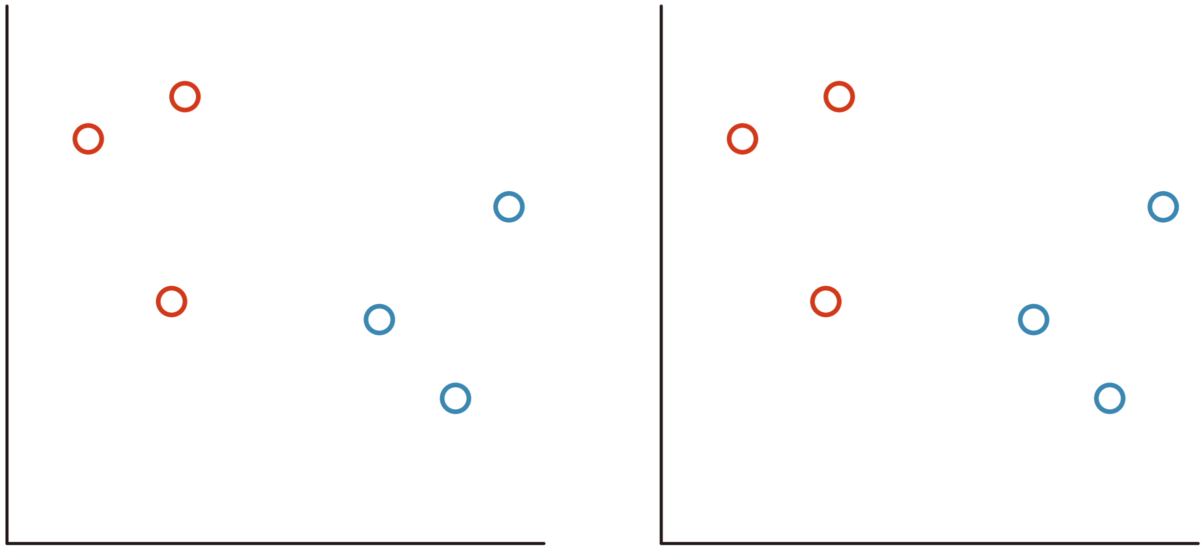


What is SSL?

- 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 [?]
  - 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

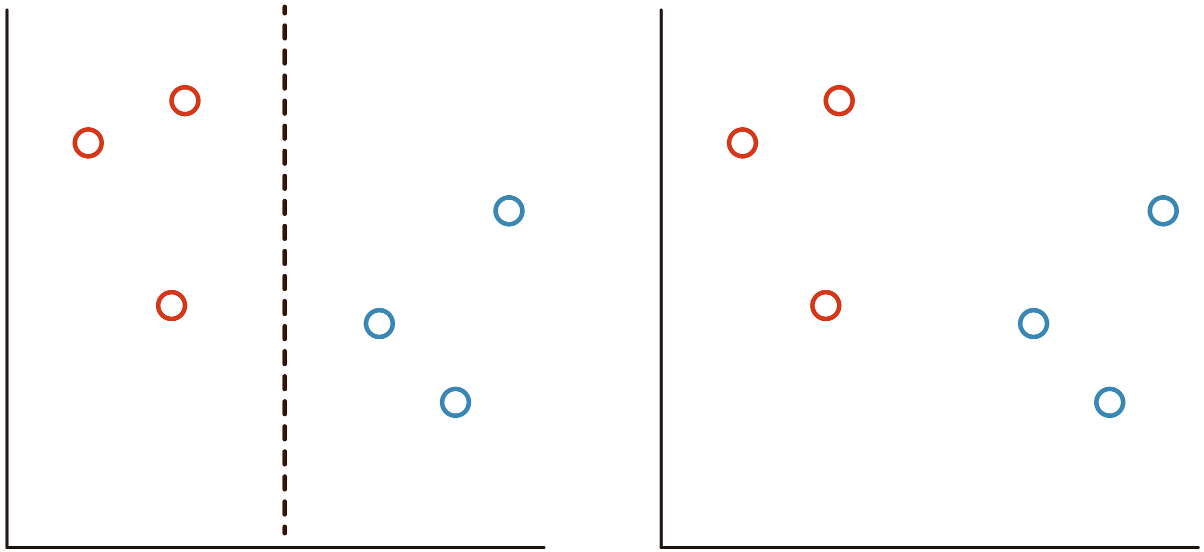
Semi-Supervised Learning

- 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]



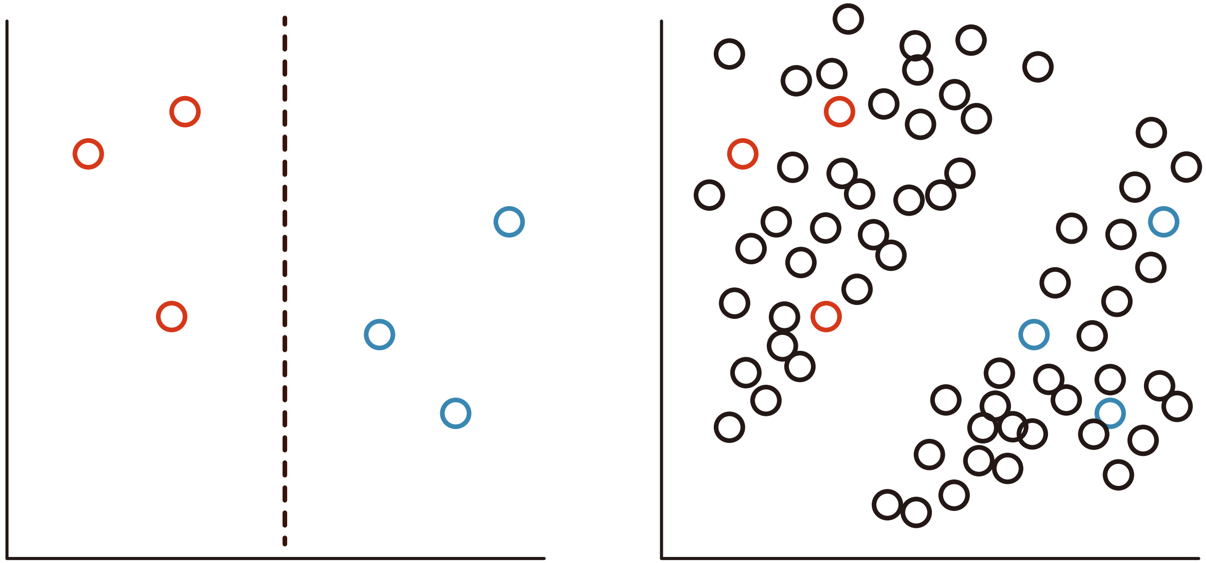
Semi-Supervised Learning

- 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]



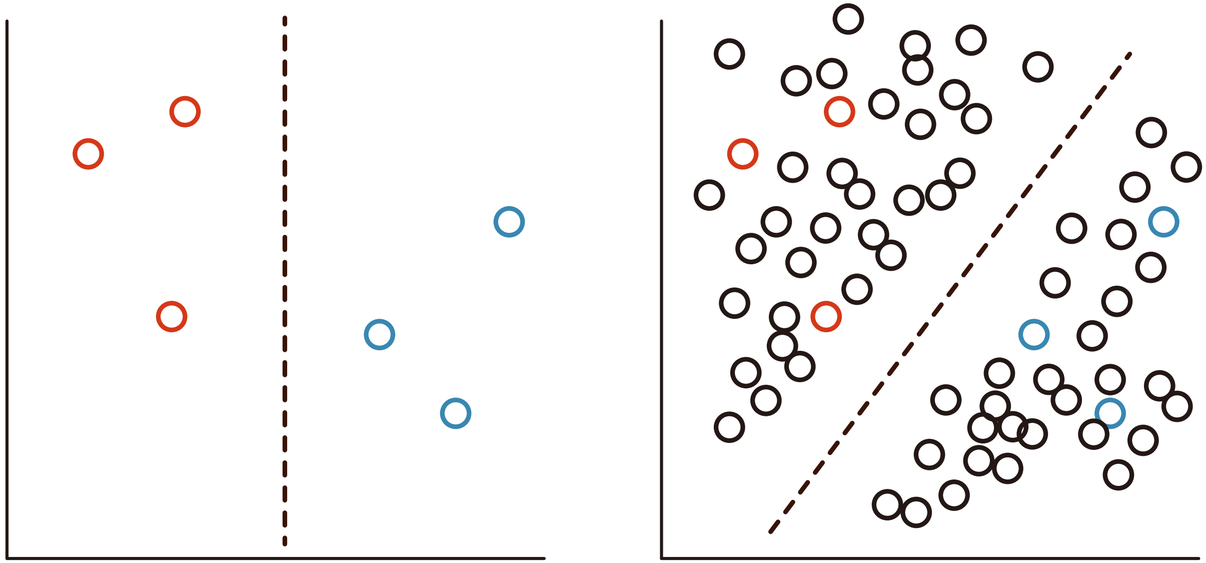
Semi-Supervised Learning

- 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]



References

[Ballester and Mitchell, 2010] P. J. Ballester and J. B. O. Mitchell. A machine learning approach to predicting protein–ligand binding affinity with applications to molecular docking. *Bioinformatics*, 26(9):1169–1175, 2010.

[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.

[Cornell *et al.*, 1995] W. D. Cornell, P. Cieplak, C. I. Bayly, I. R. Gould, K. M. Merz, D. M. Ferguson, D. C. Spellmeyer, T. Fox, J. W. Caldwell, and P. A. Kollman. A second generation force field for the simulation of proteins, nucleic acids, and organic molecules. *Journal of the American Chemical Society*, 117(19):5179–5197, 1995.

[Dara *et al.*, 2002] R. Dara, S. C. Kremer, and D. A. Stacey. Clustering unlabeled data with SOMs improves classification of labeled real-world data. In *Proceedings of the 2002 International Joint Conference on Neural Networks*, volume 3, pages 2237–2242, 2002.

[Date, 2003] C. J. Date. *An Introduction to Database Systems*. Addison Wesley, 8 edition, 2003.

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

[Demiriz *et al.*, 1999] A. Demiriz, K. P. Bennett, and M. J. Embrechts. Semi-supervised clustering using genetic algorithms. In *Proceedings of Artificial Neural Networks in Engineering*, pages 809–814, 1999.

[Fürnkranz and Hüllermeier, 2010] J. Fürnkranz and E. Hüllermeier, editors. *Preference learning*. Springer, 2010.

[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.

[Gohlke *et al.*, 2000] H. Gohlke, M. Hendlich, and G. Klebe. Knowledge-based scoring function to predict protein-ligand interactions1. *Journal of molecular biology*, 295(2):337–356, 2000.

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

[Huang *et al.*, 2006] N. Huang, C. Kalyanaraman, K. Bernacki, and M. P. Jacobson. Molecular mechanics methods for predicting protein–ligand binding. *Physical Chemistry Chemical Physics*, 8(44):5166–5177, 2006.

[Huey *et al.*, 2007] R. Huey, G. M. Morris, A. J. Olson, and D. S. Goodsell. A semiempirical free energy force field with charge-based desolvation. *Journal of computational chemistry*, 28(6):1145–1152, 2007.

[Karatzoglou *et al.*, 2004] A. Karatzoglou, A. Smola, K. Hornik, and A. Zeileis. kernlab–an S4 package for kernel methods in R. *Journal of Statistical Software*, 11(9):1–20, 2004.

[Kaytoue *et al.*, 2011] M. Kaytoue, S. O. Kuznetsov, A. Napoli, and S. Duplessis. Mining gene expression data with pattern structures in formal concept analysis. *Information Sciences*, 181:1989–2001, 2011.

[King *et al.*, 1996] R. D. King, S. H. Muggleton, A. Srinivasan, and M. J. E.

Sternberg. Structure-activity relationships derived by machine learning: The use of atoms and their bond connectivities to predict mutagenicity by inductive logic programming. *Proceedings of the National Academy of Sciences*, 93(1):438–442, 1996.

[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.

[Makino and Uno, 2004] K. Makino and T. Uno. New algorithms for enumerating all maximal cliques. In *SWAT 2004*, volume 3111 of *Lecture Notes in Computer Science*, pages 260–272. Springer, 2004.

[Moitessier *et al.*, 2008] N. Moitessier, P. Englebienne, D. Lee, J. Lawandi, and C. R. Corbeil. Towards the development of universal, fast and highly accurate docking/scoring methods: a long way to go. *British journal of pharmacology*, 153(S1):S7–S26, 2008.

[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.

[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.

[Sharman *et al.*, 2011] J. L. Sharman, C. P. Mpamhanga, M. Spedding, P. Germain, B. Staels, C. Dacquet, V. Laudet, A. J. Harmar, and NC-IUPHAR. IUPHAR-DB: New receptors and tools for easy searching and visualization of pharmacological data. *Nucleic Acids Research*, 39:D534–D538, 2011. Database Issue.

[Simovici and Djeraba, 2008] D. A. Simovici and C. Djeraba. *Mathematical Tools for Data Mining: Set Theory, Partial Orders, Combinatorics*. Springer, 2008.

[Sugiyama and Yamamoto, 2011] M. Sugiyama and A. Yamamoto. Semi-supervised learning for mixed-type data via formal concept analysis. In S. Andrews, S. Polovina, R. Hill, and B. Akhgar, editors, *Conceptual Structures for Discovering Knowledge*, volume 6828 of *Lecture Notes in Computer Science*, pages 284–297, 2011.

[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.

[Wille, 1982] R. Wille. Restructuring lattice theory: An approach based on hierarchies of concepts. In *Ordered Sets*, pages 445–470. D. Reidel Publishing Company, 1982. This article is included in Formal Concept Analysis, LNCS 5548, 314–339, Springer (2009).

[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.