Lect5: Tasks other than classification

1. Intro. to the frequent item set mining.
2. Intro. to the clustering.

1. Frequent item set mining
   (and association rule mining as its application).
2. Clustering.
3. On Exercise #5.

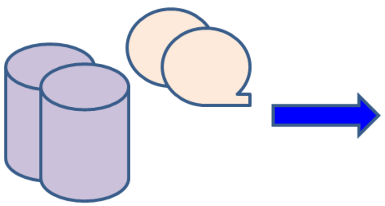
* Some of the slide materials (in particular, green ones) are from the slides of the authors of the textbook and their group at the University of Waikato.

1. Frequent item set mining

One of the earliest data mining examples investigated as a basic step for the association rule mining.

data mining of this type

supermarket costumer analysis



diapers + ?

milk
baby food, ...
standard staff, what else?

beer !

R. Agrawal, T. Imielinski, and A. N. Swami.
Mining association rules between sets of items in large databases.
in *Proc. SIGMOD Conference* 1993, pp. 207-216 (1993).

Famous database conference
started from 1975*!!*

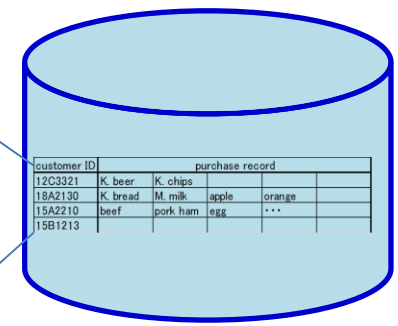
1. Frequent item set mining

glossaries

transaction database = a set of transactions.

transaction = a set of (usually, a sequence of) *item*s
recorded as one record in a database.

customer ID	purchase record				
12C3321	K. beer	K. chips			
18A2130	K. bread	M. milk	apple	orange	
15A2210	beef	pork ham	egg	・・・	
15B1213					



a transaction

an item

database

item set = a set of items in general.

frequency of an item set I = # of transactions that contain I.

↑ also called support

frequent item set = an item set whose frequency is greater than or equal to a specified min. freq. θ.

1. Frequent item set mining

=

a task of enumerating *all* frequent item sets
from a given database $\mathscr{D}$ and a min. freq. parameter θ.

Example: Weather database →
$\theta = 2$.

Outlook	Temp	Humidity	Windy	Play
Sunny	Hot	High	False	No
Rainy	Mild	Normal	False	Yes
			True	Yes
			True	Yes
			False	Yes
			True	No

One-item sets	Two-item sets	Three-item sets	Four-item sets
Outlook = Sunny (5)	Outlook = Sunny Temperature = Hot (2)	Outlook = Sunny Temperature = Hot Humidity = High (2)	Outlook = Sunny Temperature = Hot Humidity = High Play = No (2)
Temperature = Cool (4)	Outlook = Sunny Humidity = High (3)	Outlook = Sunny Humidity = High Windy = False (2)	Outlook = Rainy Temperature = Mild Windy = False Play = Yes (2)
…	…	…	…

❖ In total: 12 one-item sets, 47 two-item sets, 39 three-item sets, 6 four-item sets and 0 five-item sets (with minimum support of two)

1. Frequent item set mining
1.1. Algorithms

There are quite good number of algorithms for enumerating frequent item sets. Here are two well-known approaches.

Apriori methods:
The first apriori algorithm was proposed independently by Agrawal-Srikant and Mannila-Toivonen-Verkamo.

R. Agrawal and R. Srikant
Fast algorithms for mining association rules in large databases.
in Proc. VLDB 1994, pp. 487-499 (1994)
H. Mannila, H. Toivonen, and A. I. Verkamo.
Efficient algorithms for discovering association rules.
in *Proc. KDD Workshop* 1994: pp. 181-192 (1994).

Backtrack methods:
Algorithms known as *LCM* by Uno et al. are typical examples.

See http://research.nii.ac.jp/~uno/code/lcm.html

1. Frequent item set mining
1.2. Association rule mining: Application

Association rule mining is to derive a relation among items (in general, attribute values) with a certain "significance".

Outlook	Temp	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

```
If windy = false and play = no
   then outlook = sunny and humidity = high
```

customer ID	purchase record				
12C3321	K. beer	K. chips			
18A2130	K. bread	M. milk	apple	orange	
15A2210	beef	pork ham	egg	･･･	
15B1213					

If (buying) diapers & chips
then (buying) beer

1. Frequent item set mining
1.2. Association rule mining: Application

glossaries

- Support: number of instances p
correctly
- Confidence: number of correct
proportion of all instances the r
- Example: 4 cool days with norr

`If temperature = cool then humidit`

⇒ Support = 4, confidence = 100%

- Normally: minimum support an
pre-specified (e.g. 58 rules with
and confidence ≥ 95% for weat

Outlook	Temp	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

1. Frequent item set mining

1.2. Association rule mining: A

Outlook	Temp	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

- Once all item sets with minimum suppor have been generated, we can turn them into rules
- Example:

```
Humidity = Normal, Windy = False, Play = Yes (
```

- Seven (2^N-1) potential rules:

```
If Humidity = Normal and Windy = False then Play = Yes    4/4
If Humidity = Normal and Play = Yes then Windy = False    4/6
If Windy = False and Play = Yes then Humidity = Normal    4/6
If Humidity = Normal then Windy = False and Play = Yes    4/7
If Windy = False then Humidity = Normal and Play = Yes    4/8
If Play = Yes then Humidity = Normal and Windy = False    4/9
If True then Humidity = Normal and Windy = False
   and Play = Yes                                         4/12
```

2. Clustering

- Clustering techniques apply when there is no class to be predicted
- Aim: divide instances into “natural” groups

There are several ways to represent clusters, and several ways to measure the "appropriateness" of clusters.

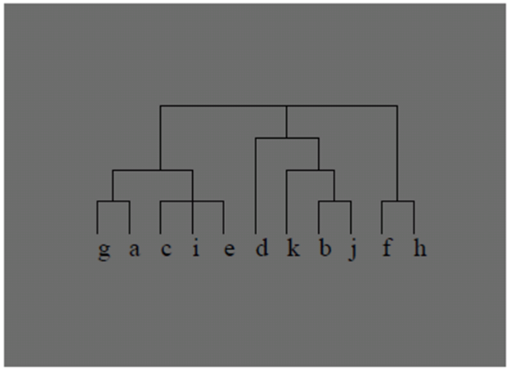
Representing clusters

1. Give a cluster label to each instance: disjoint sets.
2. By giving sets of instances: may have some overlaps.

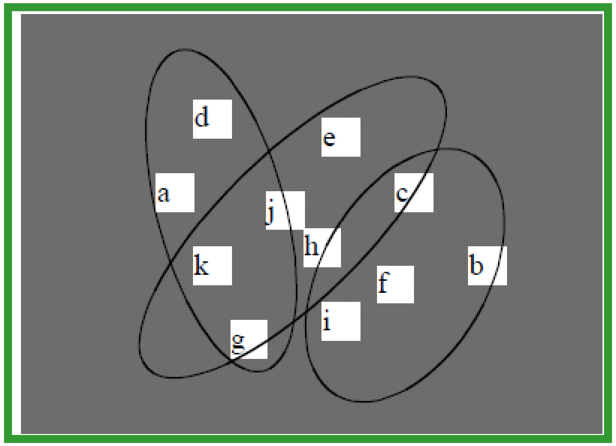
3. **Probabilistic assignment**

	1	2	3
a	0.4	0.1	0.5
b	0.1	0.8	0.1
c	0.3	0.3	0.4
d	0.1	0.1	0.8
e	0.4	0.2	0.4
f	0.1	0.4	0.5
g	0.7	0.2	0.1
h	0.5	0.4	0.1
…			

4. **Dendrogram**



NB: dendron is the Greek word for tree



2. Clustering
2.1. Two major algorithms

We explain two major algorithms for clustering. Both are designed w.r.t. a certain way to measure the "appropriateness" of clusters. Here we mainly discuss these measures.

k-means: An algorithm to separating instances to k disjoint sets (for a given k) so that the total distance from each center becomes small. ($\leftarrow$ usually, hard to get the smallest one)

- Simplest case: one numeric attribute
  - Distance is the difference between the two attribute values involved (or a function thereof)
- Several numeric attributes: normally, Euclidean distance is used and attributes are normalized
- Nominal attributes: distance is set to 1 if values are different, 0 if they are equal

2. Clustering
2.1. Two major algorithms

k-means: An ~~algorithm~~ heuristics to separating instances to k disjoint sets (for a given k) so that the total distance from each center becomes small. (← usually, hard to get the smallest one)

- To cluster data into k groups: (k is predefined)

1. Choose k cluster centers
   - e.g. at random
2. Assign instances to clusters
   - based on distance to cluster centers
3. Compute *centroids* of clusters
4. Go to step 1
   - until convergence

well...
simplified very much!

2. Clustering
2.1. Two major algorithms

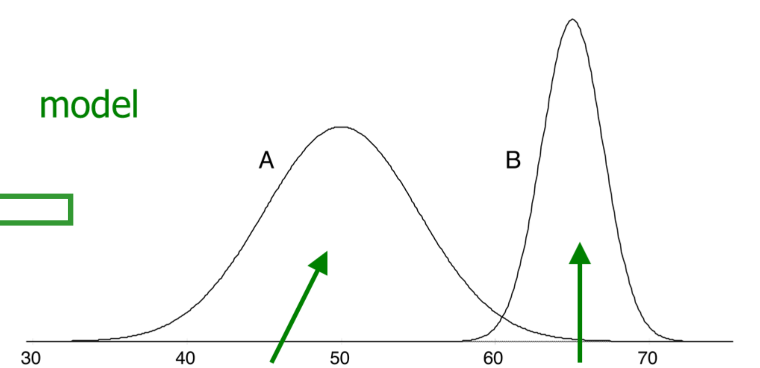
EM-algorithm: Probabilistic version of k-means
that tries to get "most likely" clusters for a given dataset.

under a certain probabilistic assumption

Most typically, we assume that instances are generated randomly under a mixture of (several) normal distributions.

data

A	51	B	62	B	64	A	48	A	39	A	51
A	43	A	47	A	51	B	64	B	62	A	48
B	62	A	52	A	52	A	51	B	64	B	64
B	64	B	64	B	62	B	63	A	52	A	42
A	45	A	51	A	49	A	43	B	63	A	48
A	42	B	65	A	48	B	65	B	64	A	41
A	46	A	48	B	62	B	66	A	48		
A	45	A	49	A	43	B	65	B	64		
A	45	A	46	A	40	A	46	A	48		

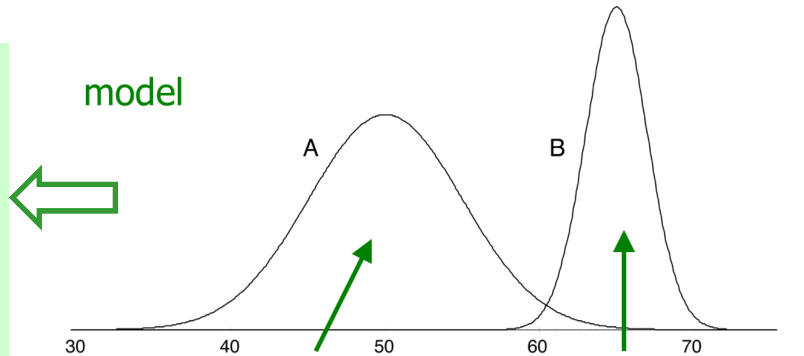


$\mu_A=50,\ \sigma_A=5,\ p_A=0.6$ $\mu_B=65,\ \sigma_B=2,\ p_B=0.4$

EM-algorithm: Probabilistic version of k-means that tries to get "most likely" clusters for a given dataset.

data

A	51	B	62	B	64	A	48	A	39	A	51
A	43	A	47	A	51	B	64	B	62	A	48
B	62	A	52	A	52	A	51	B	64	B	64
B	64	B	64	B	62	B	63	A	52	A	42
A	45	A	51	A	49	A	43	B	63	A	48
A	42	B	65	A	48	B	65	B	64	A	41
A	46	A	48	B	62	B	66	A	48		
A	45	A	49	A	43	B	65	B	64		
A	45	A	46	A	40	A	46	A	48		



$\mu_A=50$, $\sigma_A=5$, $p_A=0.6$ $\mu_B=65$, $\sigma_B=2$, $p_B=0.4$

- Probability that instance x belongs to cluster A:

$$\Pr[A \mid x] = \frac{\Pr[x \mid A]\Pr[A]}{\Pr[x]} = \frac{f(x;\mu_A,\sigma_A)\,p_A}{\Pr[x]} \quad \text{with} \quad f(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- *Likelihood* of an instance given the clusters:

$$\Pr[x] = \sum_j \Pr\left[x \mid \text{cluster}_j\right] \cdot \Pr\left[\text{cluster}_j\right]$$

$$= \sum_j \Pr\left[x \,\&\, \text{cluster}_j\right]$$

- Log-likelihood of n instances in the training set:

$$\log \prod_j \Pr\left[x_j\right] = \sum_j \log \Pr\left[x_j\right] \quad \leftarrow x_1, \ldots, x_n$$

2. Clustering
2.2. How to determine # of clusters?

A general question on clustering is to a way to determine the number k of clusters. In fact, this is the topic of our Ex.#5. Please do some literature study and try two or three (or even more) ways.

Note that the important point is a way to evaluate the "appropriateness" of k because if we have a good way for measuring the appropriateness of k, then we would be able to design a binary search type algorithm for determining k.