Announcements

- Survey & quiz scores posted
- Office Hours: after class in E2 559, none Monday
- Assignment progress & feedback, extension
- Questions and communications
  - Forum: assignment/reading/quiz questions
  - E-mail: individual-specific questions

SIMILARITY METRICS (CONTINUED)

Similarity Between Sets

- Comparing sets of items:
  - Set 1: A, B, D, E, F, J
  - Set 2: A, C, D, H
- Simple Matching:
  - Find similarity of absences/presences
  - Both have: A and D
  - Neither have: G and I
- (A+D+G+I)/(A+B+C+D+E+F+G+H+I+J) = 0.4

Similarity Between Sets

- Comparing sets of items:
  - Set 1: A, B, D, E, F, J
  - Set 2: A, C, D, H
- Jaccard:
  - Find the ratio of intersection and union
  - Both have: A and D
  - Total of 8 items (G/I unobserved)
- (A+D)/(A+B+C+D+E+F+H+J) = 2/8 = 0.25

Similarity Between Binary Vectors

- Common situation is that objects, *p* and *q*, have only binary attributes

- Compute similarities using the following quantities

M_{01} = the number of attributes where p was 0 and q was 1

M_{10} = the number of attributes where p was 1 and q was 0

M_{00} = the number of attributes where p was 0 and q was 0

M_{11} = the number of attributes where p was 1 and q was 1

- Simple Matching and Jaccard Coefficients

SMC = number of matches / number of attributes

$= (M_{11} + M_{00}) / (M_{01} + M_{10} + M_{11} + M_{00})$

J = number of 11 matches / number of not-both-zero attributes values

$= (M_{11}) / (M_{01} + M_{10} + M_{11})$

SMC versus Jaccard: Example

p = 1 0 0 0 0 0 0 0 0 0

q = 0 0 0 0 0 0 1 0 0 1

$M_{01} = 2$ (the number of attributes where p was 0 and q was 1)

$M_{10} = 1$ (the number of attributes where p was 1 and q was 0)

$M_{00} = 7$ (the number of attributes where p was 0 and q was 0)

$M_{11} = 0$ (the number of attributes where p was 1 and q was 1)

SMC = $(M_{11} + M_{00})/(M_{01} + M_{10} + M_{11} + M_{00})$ = (0+7) / (2+1+0+7) = 0.7

J = $(M_{11}) / (M_{01} + M_{10} + M_{11})$ = 0 / (2 + 1 + 0) = 0

Cosine Similarity

- If d_1 and d_2 are two vectors, then

  $$\cos(d_1, d_2) = (d_1 \bullet d_2) / ||d_1|| \, ||d_2|| ,$$

where • indicates vector dot product and $|| d ||$ is the length of vector d.

- Example:

  $d_1 =$ **3 2 0 5 0 0 0 2 0 0**

  $d_2 =$ **1 0 0 0 0 0 0 1 0 2**

$d_1 \bullet d_2 = 3*1 + 2*0 + 0*0 + 5*0 + 0*0 + 0*0 + 0*0 + 2*1 + 0*0 + 0*2 = 5$

$||d_1|| = (3*3+2*2+0*0+5*5+0*0+0*0+0*0+2*2+0*0+0*0)^{\mathbf{0.5}} = (42)^{\mathbf{0.5}} = 6.481$

$||d_2|| = (1*1+0*0+0*0+0*0+0*0+0*0+0*0+1*1+0*0+2*2)^{\mathbf{0.5}} = (6)^{\mathbf{0.5}} = 2.245$

$$\cos(d_1, d_2) = .3150$$

Based on Tan,Steinbach, Kumar, Han & Kamber

Correlation

- Correlation measures the linear relationship between objects
- To compute correlation, we standardize data objects, p and q, and then take their dot product

$$p'_k = (p_k - mean(p)) / std(p)$$

$$q'_k = (q_k - mean(q)) / std(q)$$

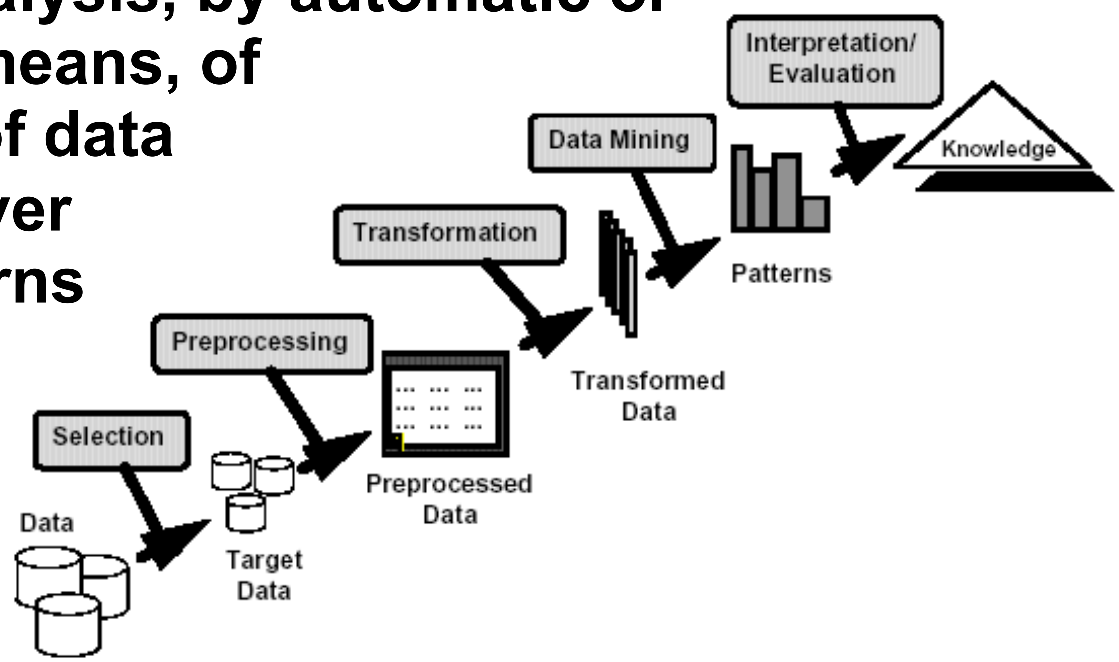
$$correlation(p, q) = p' \bullet q'$$

Data Mining: Pipelines & Tasks

Based on Tan,Steinbach, Kumar, Han & Kamber

What is Data Mining?

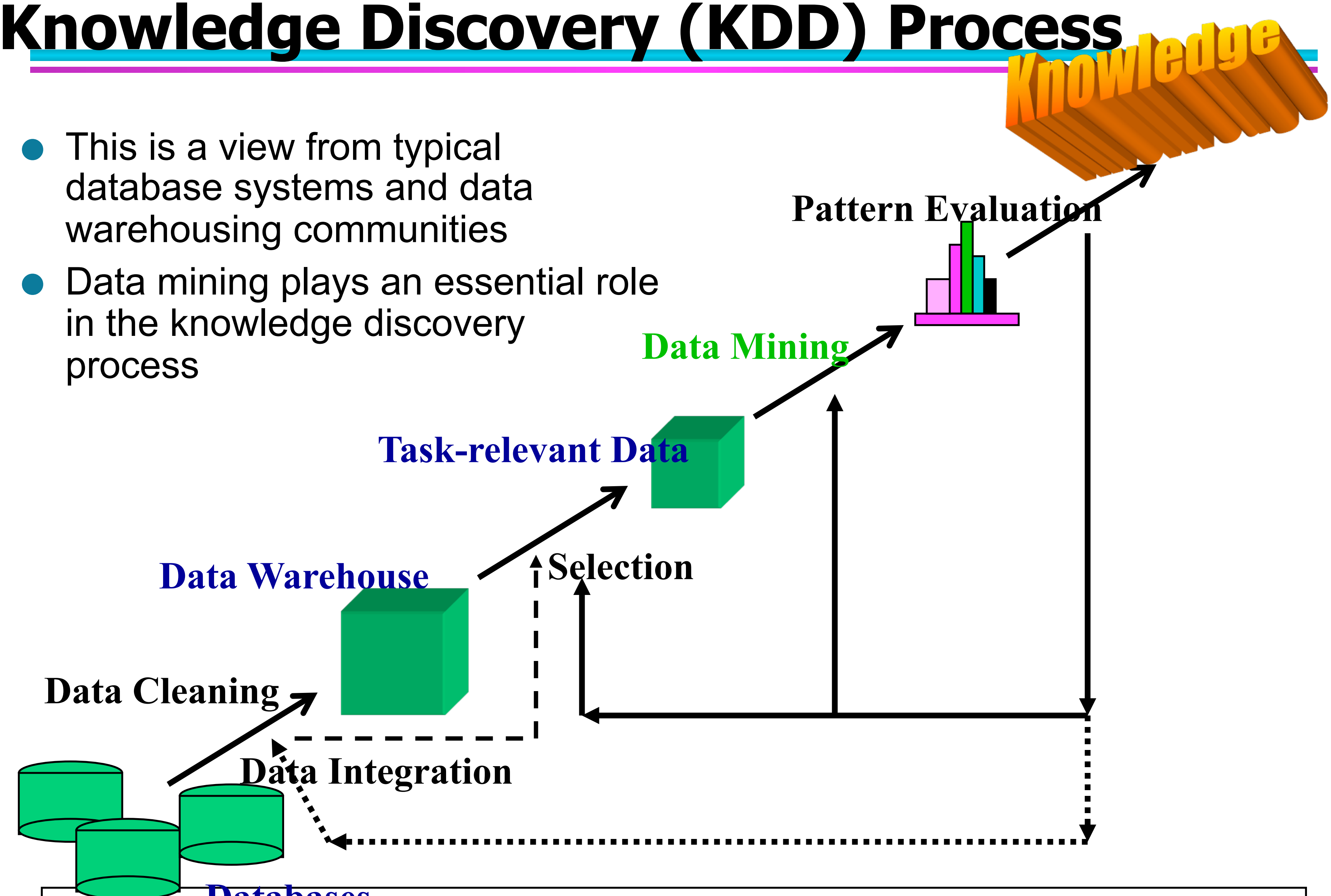
- **Many Definitions**
  - **Non-trivial extraction of implicit, previously unknown and potentially useful information from data**
  - **Exploration & analysis, by automatic or semi-automatic means, of large quantities of data in order to discover meaningful patterns**



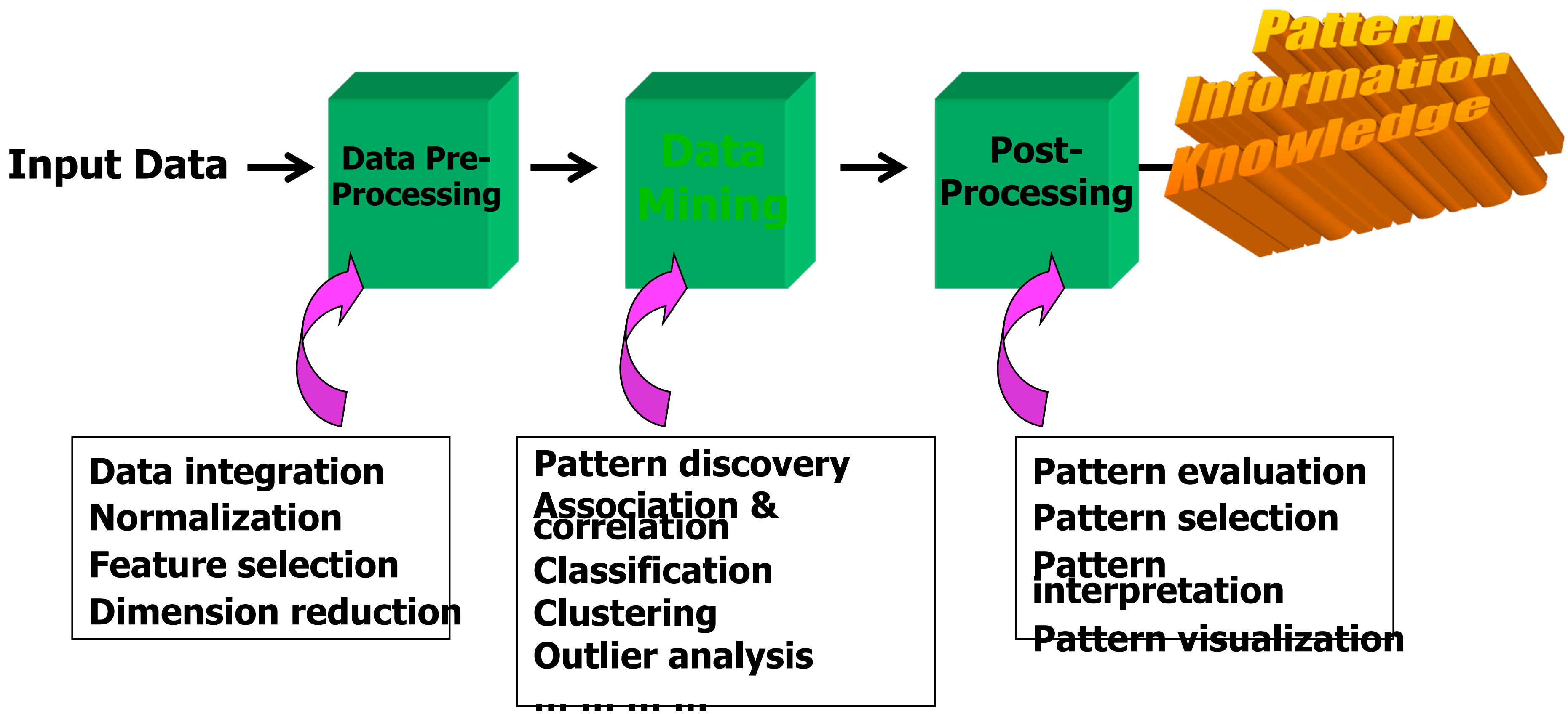
Based on Tan,Steinbach, Kumar, Han & Kamber

Knowledge Discovery (KDD) Process

- This is a view from typical database systems and data warehousing communities
- Data mining plays an essential role in the knowledge discovery process

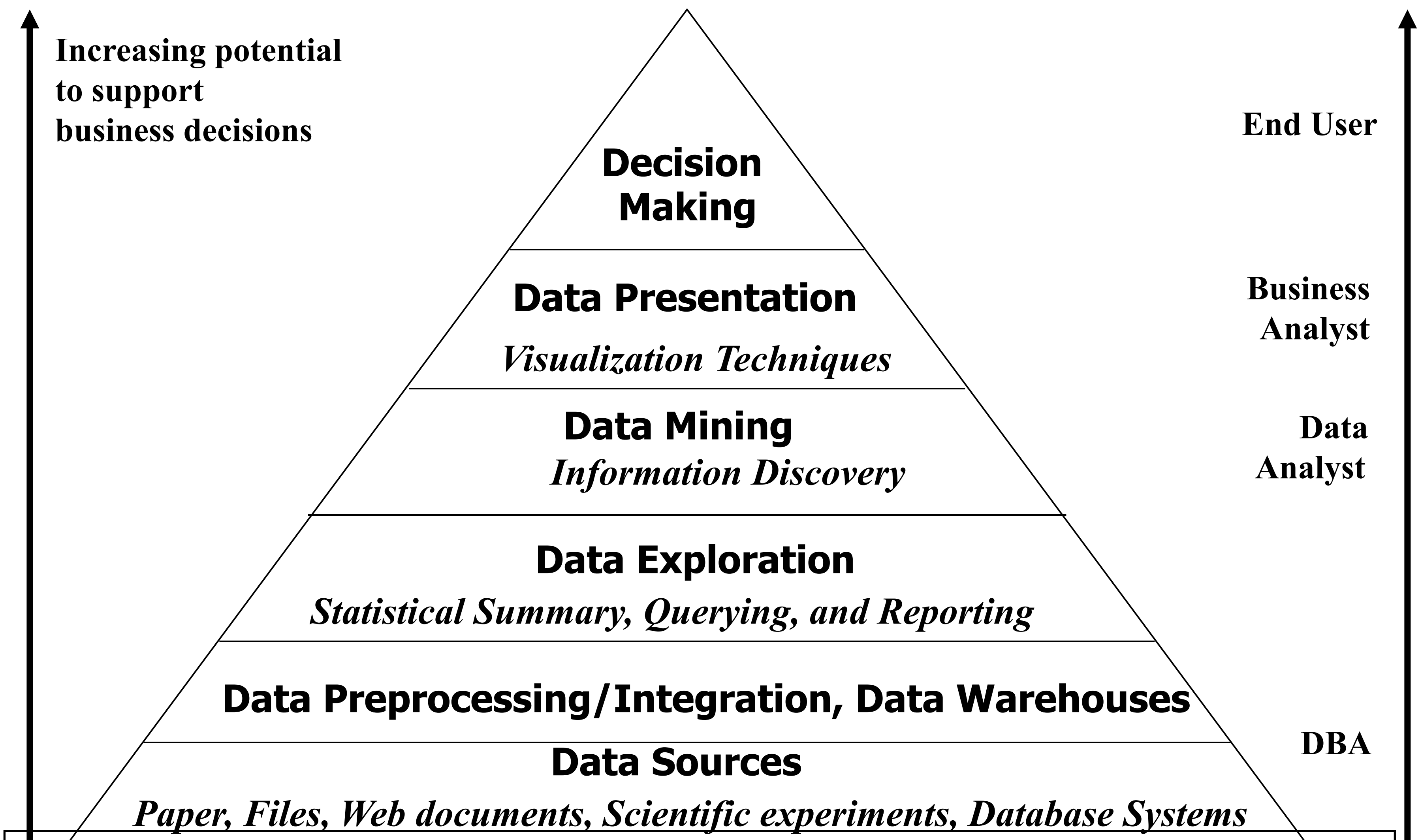


KDD Process: A Typical View from ML and Statistics



- This is a view from typical machine learning and statistics communities

Data Mining in Business Intelligence



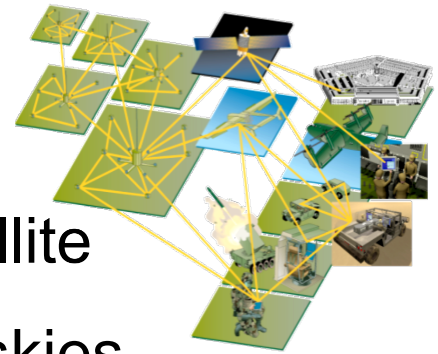
Why Mine Data? Commercial Viewpoint

- Lots of data is being collected and warehoused
  - Web data, e-commerce
  - purchases at department/ grocery stores
  - Bank/Credit Card transactions

- Data mining enables better, customized services
  - In Customer Relationship Management
  - Providing competitive advantages

Based on Tan,Steinbach, Kumar, Han & Kamber

Why Mine Data? Scientific Viewpoint



- Data collected and stored at enormous speeds (PB/day)
  - remote sensors on a satellite
  - telescopes scanning the skies
  - gene expression data
  - scientific simulations
- Data mining may help scientists
  - in classifying and segmenting data
  - in Hypothesis Formation
- Traditional techniques may not be computationally efficient, Data mining focuses on efficiency as well as effectiveness

What is (not) Data Mining?

- **What is not Data Mining?**
  - Look up phone number in phone directory
  - Query a Web search engine for information about "Amazon"

- **What is Data Mining?**
  - Certain names are more prevalent in certain US locations (O'Brien, O'Rurke, O'Reilly… in Boston area)
  - Group together similar documents returned by search engine according to their context (e.g. Amazon rainforest, Amazon.com,)

Discussion: What is (not) Data Mining?

- Dividing the customers of a company according to their gender.
- Dividing the customers of a company according to their profitability.
- Computing the total sales of a company.
- Sorting a student database based on student identification numbers.
- Predicting the outcomes of tossing a (fair) pair of dice.
- Predicting the future stock price of a company using historical records.
- Monitoring the heart rate of a patient for abnormalities.
- Monitoring seismic waves for earthquake activities.
- Extracting the frequencies of a sound wave.

Discussion

- Please list at least 2 data mining tasks

Data Mining Tasks

- Prediction Methods
  – Use some variables to predict unknown or future values of other variables.

- Description Methods
  – Find human-interpretable patterns that describe the data.

From [Fayyad, et.al.] Advances in Knowledge Discovery and Data Mining, 1996

Data Mining Tasks...

Descriptive:

- Generalization
- Clustering
- Sequential Pattern Discovery
- Causal Discovery

Predictive:

- Classification
- Regression
- Sequential Pattern Discovery
- Association Rule Discovery
- Outlier Detection
- Deviation Detection

Data Mining Tasks...

Descriptive:

- Generalization
- Clustering
- Sequent
  Discov
- Causa

Predictive:

- Classification
- ssion
- al Pattern
- n Rule
- etection
- on Detection

A data mining process usually includes both descriptive analysis and predictive modeling

Generalization

- Information integration and data warehouse construction
  - Data cleaning, transformation, integration, and multidimensional data model
- Data cube technology
  - Scalable methods for computing (i.e., materializing) multidimensional aggregates
  - OLAP (online analytical processing)
- Multidimensional concept description: Characterization and discrimination
  - Generalize, summarize, and contrast data characteristics, e.g., dry vs. wet region

Association Rule Discovery: Definition

- Given a set of records each of which contain some number of items from a given collection;
  - Produce dependency rules which will predict occurrence of an item based on occurrences of other items.

TID	*Items*
1	**Bread, Coke, Milk**
2	**Beer, Bread**
3	**Beer, Coke, Diaper, Milk**
4	**Beer, Bread, Diaper, Milk**
5	**Coke, Diaper, Milk**

A typical association rule:

Diaper → Beer [0.5%, 75%] (support, confidence)

Based on Tan,Steinbach, Kumar, Han & Kamber

Association Rule Discovery: Application 1

- Marketing and Sales Promotion:
  - Let the rule discovered be

    {Bagels, … } --> {Potato Chips}
  - Potato Chips as consequent => Can be used to determine what should be done to boost its sales.
  - Bagels in the antecedent => Can be used to see which products would be affected if the store discontinues selling bagels.
  - Bagels in antecedent *and* Potato chips in consequent => Can be used to see what products should be sold with Bagels to promote sale of Potato chips!

Association Rule Discovery: Application 2

- Supermarket shelf management.
  - Goal: To identify items that are likely to be bought together.
  - Approach: Process the point-of-sale data collected with barcode scanners to find dependencies among items.
  - A classic rule --
    - If a customer buys diaper and milk, then he is very likely to buy beer.
    - So, don’t be surprised if you find six-packs stacked next to diapers!

Association Rule Discovery: Application 3

- Inventory Management:
  - Goal: repair company anticipates repairs and keep the service vehicles equipped with right parts to reduce on number of visits to consumer households.
  - Approach: Process the data on tools and parts required in previous repairs and discover the co-occurrence patterns.

Classification: Definition

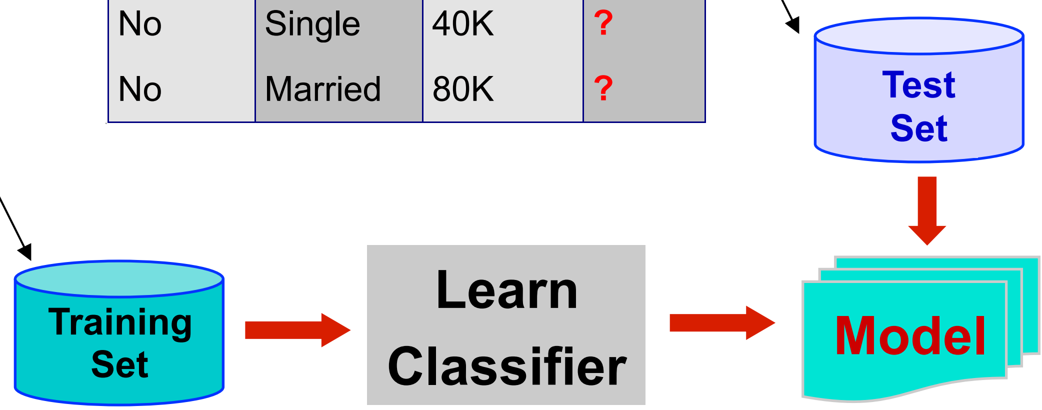
- **Given** a collection of records (*training set*)
  - Each record contains a set of *attributes/features*, one of the attributes is the *class*.
- **Find** a *model* for class attribute as a function of the values of other attributes.
- **Goal**: previously unseen records should be assigned a class as accurately as possible.
  - A *test set* is used to determine the accuracy of the model.

Classification Example

categorical | categorical | continuous | class

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Refund	Marital Status	Taxable Income	Cheat
No	Single	75K	?
Yes	Married	50K	?
No	Married	150K	?
Yes	Divorced	90K	?
No	Single	40K	?
No	Married	80K	?



Based on Tan,Steinbach, Kumar, Han & Kamber

Classification: Application 1

- Direct Marketing
  - Goal: Reduce cost of mailing by *targeting* a set of consumers likely to buy a new cell-phone product.
  - Approach:
    - Use the data for a similar product introduced before.
    - We know which customers decided to buy and which decided otherwise. This *{buy, don't buy}* decision forms the *class attribute*.
    - Collect various demographic, lifestyle, and company-interaction related information about all such customers.
      - Type of business, where they stay, how much they earn, etc.
    - Use this information as input attributes to learn a classifier model.

From [Berry & Linoff] Data Mining Techniques, 1997

Classification: Application 2

- Fraud Detection
  - Goal: Predict fraudulent cases in credit card transactions.
  - Approach:
    - Use credit card transactions and the information on its account-holder as attributes.
      - When does a customer buy, what does he buy, how often he pays on time, etc
    - Label past transactions as fraud or fair transactions. This forms the class attribute.
    - Learn a model for the class of the transactions.
    - Use this model to detect fraud by observing credit card transactions on an account.

Classification: Application 3

- Customer Attrition/Churn:
  - Goal: To predict whether a customer is likely to be lost to a competitor.
  - Approach:
    - Use detailed record of transactions with each of the past and present customers, to find attributes.
      - How often the customer calls, where he calls, what time-of-the day he calls most, his financial status, marital status, etc.
    - Label the customers as loyal or disloyal.
    - Find a model for loyalty.

From [Berry & Linoff] Data Mining Techniques, 1997

Classification: Application 4

- Sky Survey Cataloging
  - Goal: To predict class (star or galaxy) of sky objects, especially visually faint ones, based on the telescopic survey images (from Palomar Observatory).
    - 3000 images with 23,040 x 23,040 pixels per image.
  - Approach:
    - Segment the image.
    - Measure image attributes (features) - 40 of them per object.
    - Model the class based on these features.
    - Success Story: Could find 16 new high red-shift quasars, some of the farthest objects that are difficult to find!

From [Fayyad, et.al.] Advances in Knowledge Discovery and Data Mining, 1996

Regression

- Predict a value of a given continuous valued variable based on the values of other variables, assuming a linear or nonlinear model of dependency.
- Greatly studied in statistics, neural network fields.
- Examples:
  - Predicting sales amounts of new product based on advertising expenditure.
  - Predicting wind velocities as a function of temperature, humidity, air pressure, etc.
  - Time series prediction of stock market indices.

Classifying Galaxies

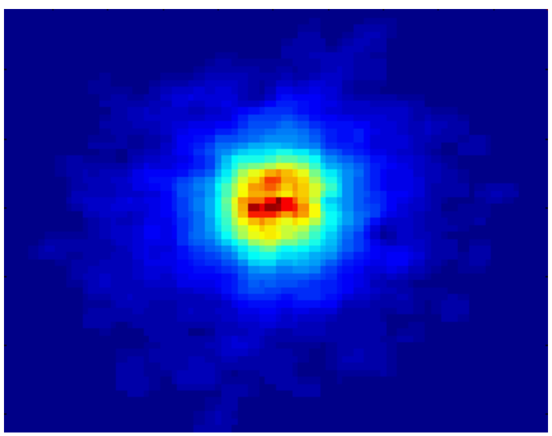
Courtesy: http://aps.umn.edu

Data Size: **72 million stars, 20 million galaxies, Object Catalog: 9 GB**
- **Image Database: 150 GB**

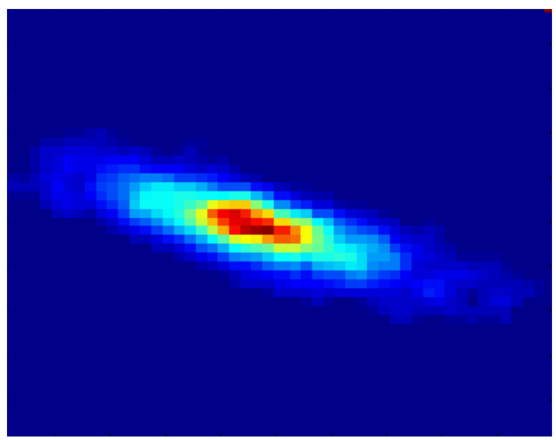
Class:
- **Stages of Formation**

Attributes:
- **Image features,**
- **Characteristics of light waves received, etc.**

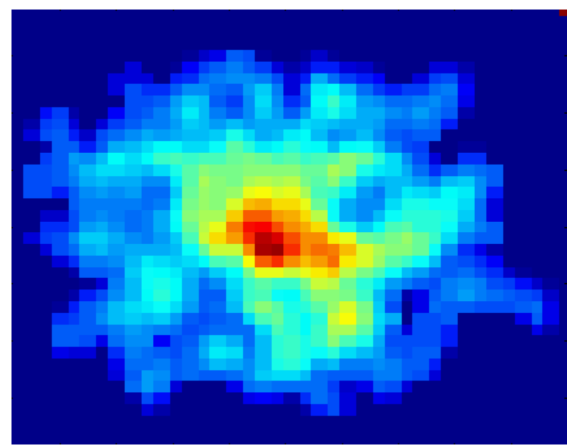
Early



Intermediate



Late



Based on Tan,Steinbach, Kumar, Han & Kamber

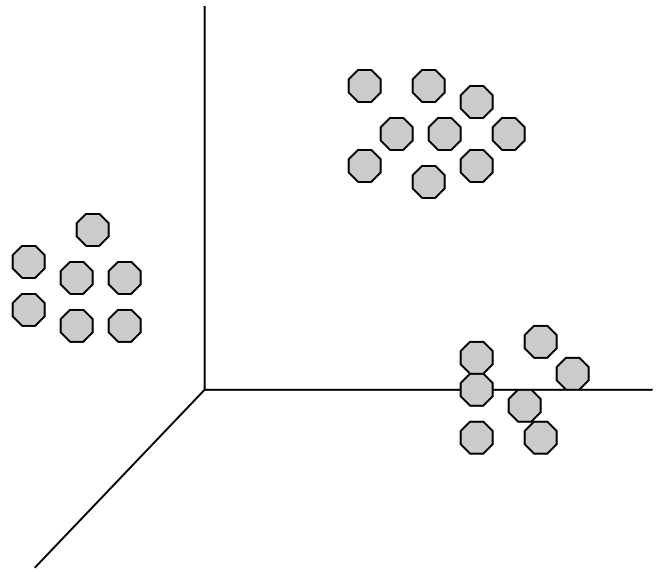
Clustering Definition

- Given a set of data points, each having a set of attributes, and (maybe) a similarity measure among them, find clusters such that
  - Data points in one cluster are more similar to one another.
  - Data points in separate clusters are less similar to one another.
- Similarity Measures:
  - Manhattan Distance, Cosine Distance, Distributional Similarity, …
  - Other Problem-specific Measures

Based on Tan,Steinbach, Kumar, Han & Kamber

Clustering Example: Euclidean Distance

- How data points should be grouped together?



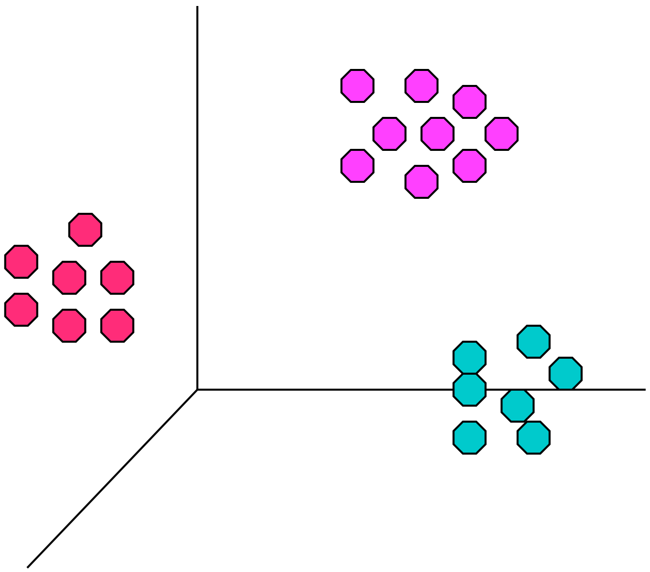
Based on Tan,Steinbach, Kumar, Han & Kamber

Clustering Example: Euclidean Distance

- Euclidean Distance Based Clustering in 3-D space.

Intracluster distances are minimized

Intercluster distances are maximized



Based on Tan,Steinbach, Kumar, Han & Kamber

Clustering: Application 1

- Market Segmentation:
  - Goal: subdivide a market into distinct subsets of customers where any subset may conceivably be selected as a market target to be reached with a distinct marketing mix.
  - Approach:
    - Collect different attributes of customers based on their geographical and lifestyle related information.
    - Find clusters of similar customers.
    - Measure the clustering quality by observing buying patterns of customers in same cluster vs. those from different clusters.

Clustering: Application 2

- Document Clustering:
  - Goal: To find groups of documents that are similar to each other based on the important terms appearing in them.
  - Approach: To identify frequently occurring terms in each document. Form a similarity measure based on the frequencies of different terms. Use it to cluster.
  - Gain: Information Retrieval can utilize the clusters to relate a new document or search term to clustered documents.

Illustrating D… nt Clustering

- Clu … geles Times.
- S … mmon in

Reminder:
This is one possible approach.
You will learn many
other clustering
approaches later
in this class

Entertainment	[illegible]	[illegible]

Clustering of S&P 500 Stock Data

- Observe Stock Movements every day.
- Clustering points: Stock-{UP/DOWN}
- Similarity Measure: Two points are more similar if the events described by them frequently happen together on the same day.
  - We used association rules to quantify a similarity measure.

	Discovered Clusters	*Industry Group*
1	Applied-Matl-DOWN,Bay-Network-Down,3-COM-DOWN, Cabletron-Sys-DOWN,CISCO-DOWN,HP-DOWN, DSC-Comm-DOWN,INTEL-DOWN,LSI-Logic-DOWN, Micron-Tech-DOWN,Texas-Inst-Down,Tellabs-Inc-Down, Natl-Semiconduct-DOWN,Oracl-DOWN,SGI-DOWN, Sun-DOWN	Technology1-DOWN
2	Apple-Comp-DOWN,Autodesk-DOWN,DEC-DOWN, ADV-Micro-Device-DOWN,Andrew-Corp-DOWN, Computer-Assoc-DOWN,Circuit-City-DOWN, Compaq-DOWN, EMC-Corp-DOWN, Gen-Inst-DOWN, Motorola-DOWN,Microsoft-DOWN,Scientific-Atl-DOWN	Technology2-DOWN
3	Fannie-Mae-DOWN,Fed-Home-Loan-DOWN, MBNA-Corp-DOWN,Morgan-Stanley-DOWN	Financial-DOWN
4	Baker-Hughes-UP,Dresser-Inds-UP,Halliburton-HLD-UP, Louisiana-Land-UP,Phillips-Petro-UP,Unocal-UP, Schlumberger-UP	Oil-UP

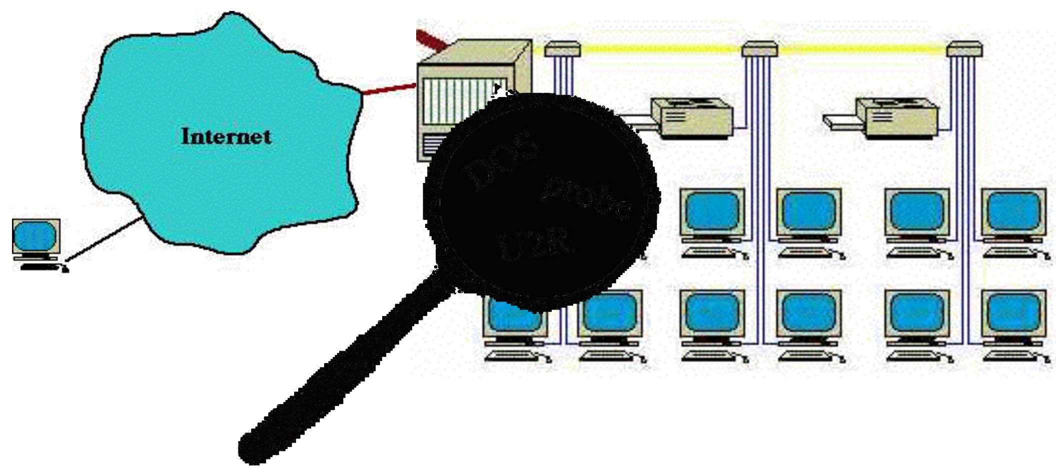
Outlier Analysis

- Outlier: A data object that does not comply with the general behavior of the data
- Noise or exception? — One person's garbage could be another person's treasure
- Methods: by product of clustering or regression analysis, …
- Useful in fraud detection, rare events analysis

Outlier/Deviation/Anomaly Detection

- Applications:
  - Credit Card Fraud Detection

  - Network Intrusion Detection



Based on Tan,Steinbach, Kumar, Han & Kamber

Data Mining

Practical Machine Learning Tools and Techniques

Slides for Chapter 3 of *Data Mining* by I. H. Witten and E. Frank

Output: Knowledge representation

- Decision tables
- Decision trees
- Decision rules
- Association rules
- Rules with exceptions
- Rules involving relations
- Linear regression
- Trees for numeric prediction
- Instance-based representation
- Clusters

Output: representing structural patterns

- Many different ways of representing patterns
  - Decision trees, rules, instance-based, …
- Also called “knowledge” representation
- Representation determines inference method
- Understanding the output is the key to understanding the underlying learning methods
- Different types of output for different learning problems (e.g. classification, regression, …)

Decision tables

- Simplest way of representing output:
  - Use the same format as input!
- Decision table for the weather problem:

Outlook	Humidity	Play
Sunny	High	No
Sunny	Normal	Yes
Overcast	High	Yes
Overcast	Normal	Yes
Rainy	High	No
Rainy	Normal	No

- Main problem: selecting the right attributes

Decision trees

- "Divide-and-conquer" approach produces tree
- Nodes involve testing a particular attribute
- Usually, attribute value is compared to constant
- Other possibilities:
  - Comparing values of two attributes
  - Using a function of one or more attributes
- Leaves assign classification, set of classifications, or probability distribution to instances
- Unknown instance is routed down the tree

Nominal and numeric attributes

- Nominal:
  number of children usually equal to number values
  ⇒ attribute won't get tested more than once
  - Other possibility: division into two subsets
- Numeric:
  test whether value is greater or less than constant
  ⇒ attribute may get tested several times
  - Other possibility: three-way split (or multi-way split)
    - Integer: *less than, equal to, greater than*
    - Real: *below, within, above*

Missing values

- Does absence of value have some significance?
- Yes ⇒ "missing" is a separate value
- No ⇒ "missing" must be treated in a special way
  - Solution A: assign instance to most popular branch
  - Solution B: split instance into pieces
    - Pieces receive weight according to fraction of training instances that go down each branch
    - Classifications from leave nodes are combined using the weights that have percolated to them

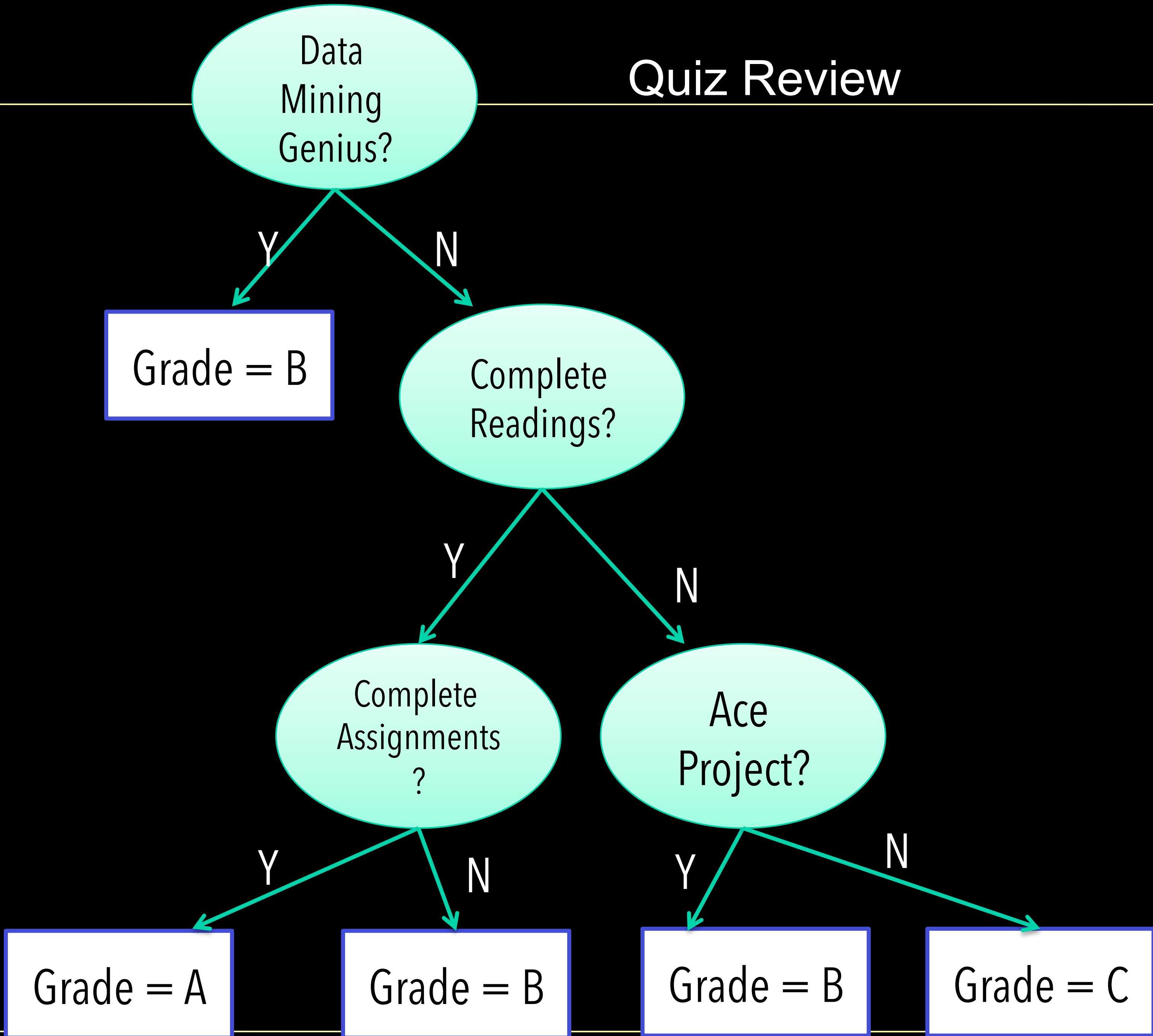
Classification rules

- Popular alternative to decision trees
- *Antecedent* (pre-condition): a series of tests (just like the tests at the nodes of a decision tree)
- Tests are usually logically ANDed together (but may also be general logical expressions)
- *Consequent* (conclusion): classes, set of classes, or probability distribution assigned by rule
- Individual rules are often logically ORed together
  - Conflicts arise if different conclusions apply

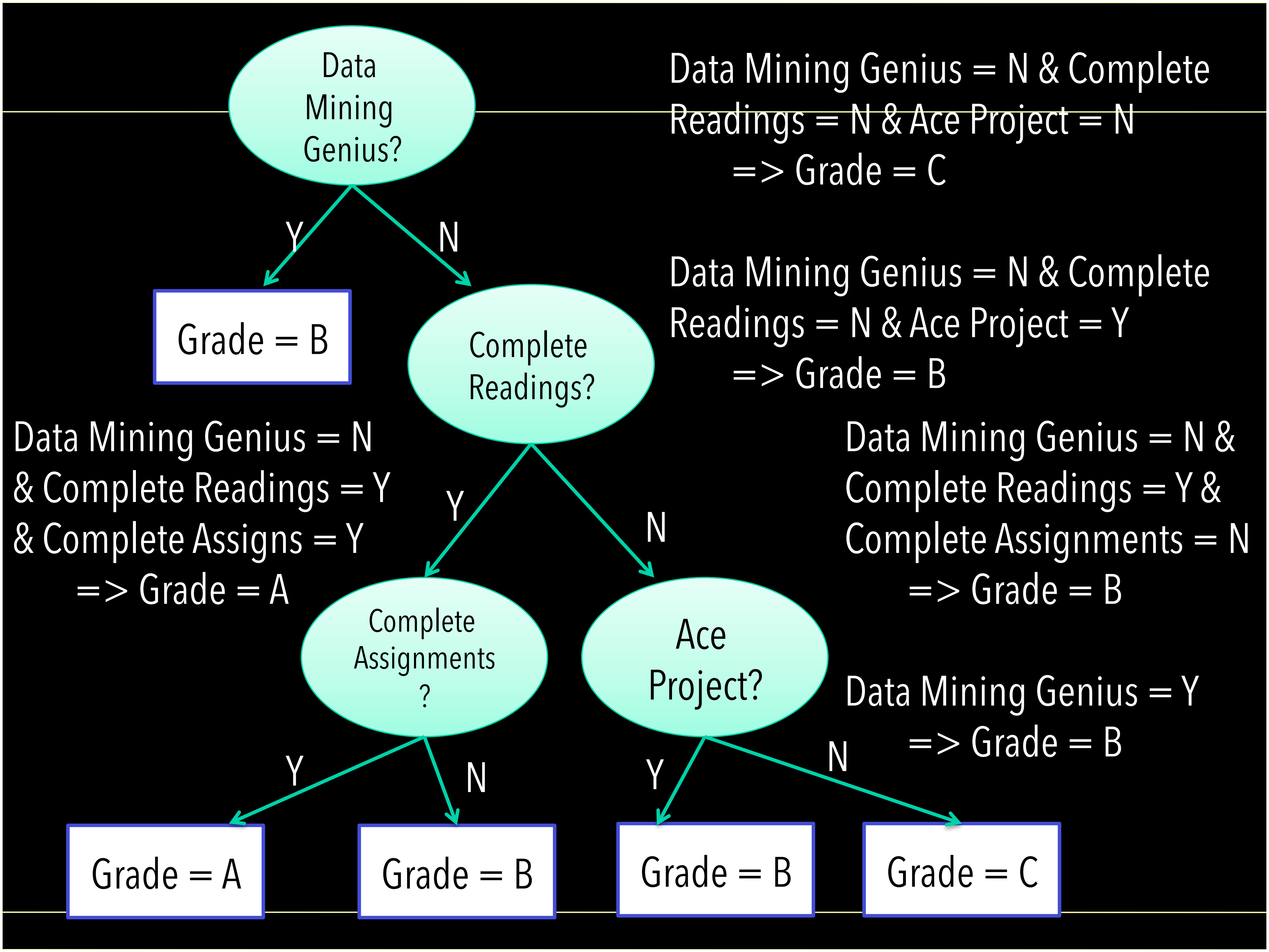
From trees to rules

- Easy: converting a tree into a set of rules
  - One rule for each leaf:
    - Antecedent contains a condition for every node on the path from the root to the leaf
    - Consequent is class assigned by the leaf
- Produces rules that are unambiguous
  - Doesn’t matter in which order they are executed
- But: resulting rules are unnecessarily complex
  - Pruning to remove redundant tests/rules

Quiz Review



Data Mining Genius?
Y
N
Grade = B
Complete Readings?
Y
N
Complete Assignments ?
Ace Project?
Y
N
Y
N
Grade = A
Grade = B
Grade = B
Grade = C


Data Mining Genius?
Y
N
Grade = B
Complete Readings?
Y
N
Complete Assignments ?
Ace Project?
Y
N
Y
N
Grade = A
Grade = B
Grade = B
Grade = C
Data Mining Genius = N & Complete Readings = N & Ace Project = N
=> Grade = C
Data Mining Genius = N & Complete Readings = N & Ace Project = Y
=> Grade = B
Data Mining Genius = N & Complete Readings = Y & Complete Assigns = Y
=> Grade = A
Data Mining Genius = N & Complete Readings = Y & Complete Assignments = N
=> Grade = B
Data Mining Genius = Y
=> Grade = B

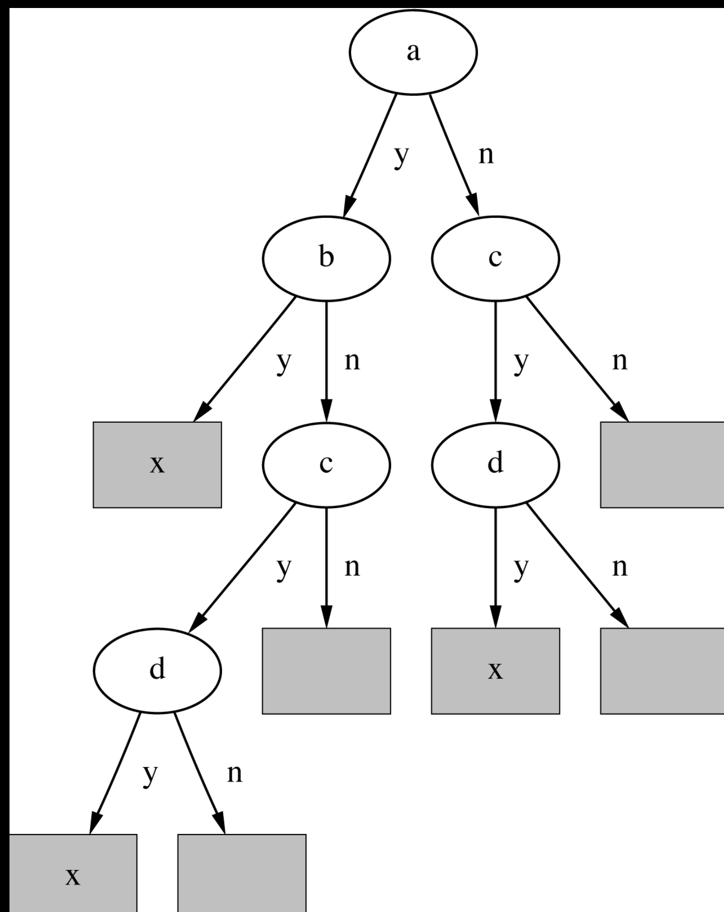
From rules to trees

- More difficult: transforming a rule set into a tree
  - Tree cannot easily express disjunction between rules
- Example: rules which test different attributes

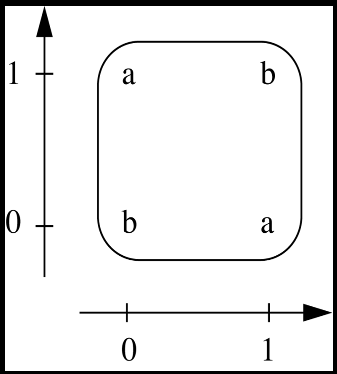
```
If a and b then x
If c and d then x
```

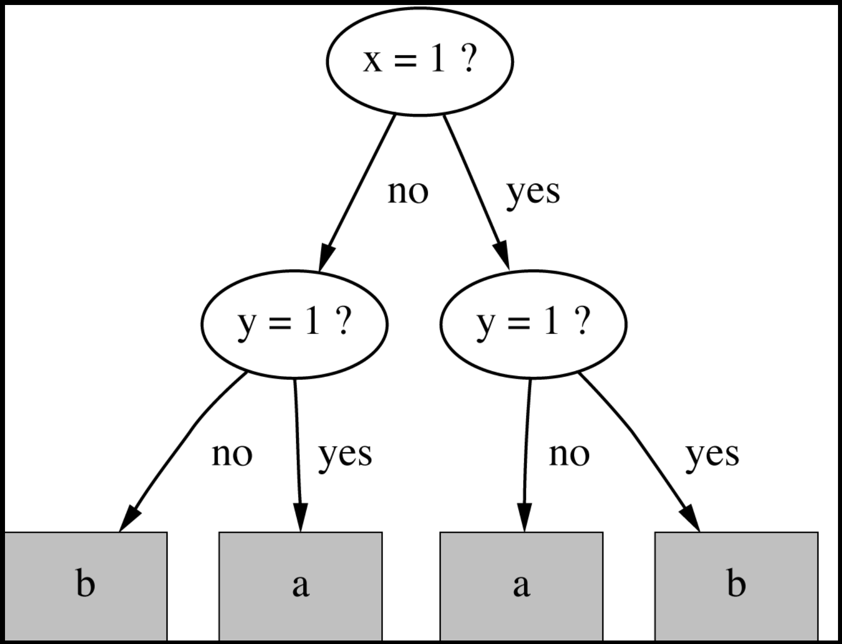
- Symmetry needs to be broken
- Corresponding tree contains identical subtrees
  (⇒ "replicated subtree problem")

A tree for a simple disjunction



The exclusive-or problem

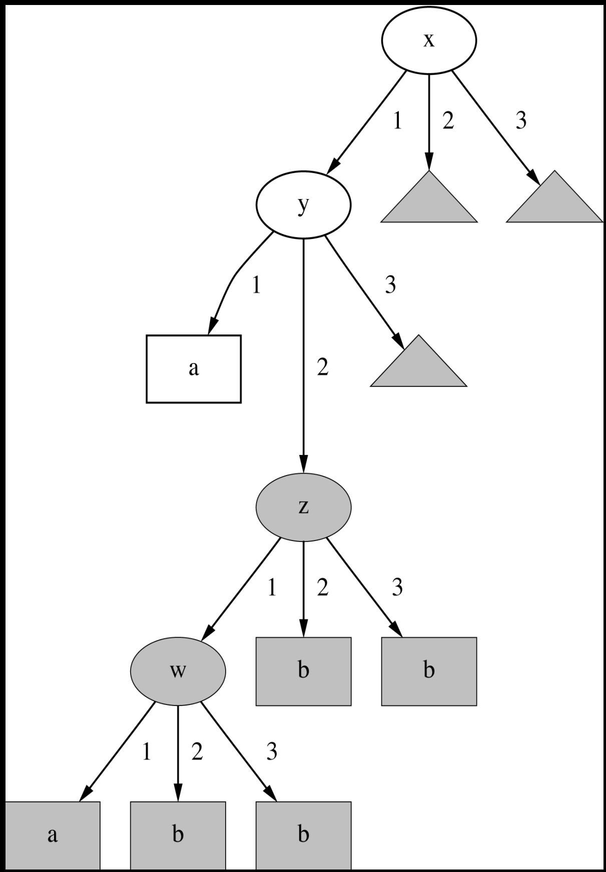





```
If x = 1 and y = 0
   then class = a
If x = 0 and y = 1
   then class = a
If x = 0 and y = 0
   then class = b
If x = 1 and y = 1
   then class = b
```

A tree with a replicated subtree

```
If x = 1 and y = 1
   then class = a
If z = 1 and w = 1
   then class = a
Otherwise class = b
```



“Nuggets” of knowledge

- Are rules independent pieces of knowledge? (It seems easy to add a rule to an existing rule base.)
- Problem: ignores how rules are executed
- Two ways of executing a rule set:
  - Ordered set of rules (“decision list”)
    - Order is important for interpretation
  - Unordered set of rules
    - Rules may overlap and lead to different conclusions for the same instance

Interpreting rules

- What if two or more rules conflict?
  - Give no conclusion at all?
  - Go with rule that is most popular on training data?
  - …
- What if no rule applies to a test instance?
  - Give no conclusion at all?
  - Go with class that is most frequent in training data?
  - …

Special case: boolean class

- Assumption: if instance does not belong to class "yes", it belongs to class "no"
- Trick: only learn rules for class "yes" and use default rule for "no"

```
If x = 1 and y = 1 then class = a
If z = 1 and w = 1 then class = a
Otherwise class = b
```

- Order of rules is not important. No conflicts!
- Rule can be written in *disjunctive normal form*

Association rules

- Association rules…
  - … can predict any attribute and combinations of attributes
  - … are not intended to be used together as a set
- Problem: immense number of possible associations
  - Output needs to be restricted to show only the most predictive associations ⇒ only those with high *support* and high *confidence*

Support and confidence of a rule

- Support: number of instances predicted correctly
- Confidence: number of correct predictions, as proportion of all instances that rule applies to
- Example: 4 cool days with normal humidity

```
If temperature = cool then humidity = normal
```

$\Rightarrow$ Support = 4, confidence = 100%

- Normally: minimum support and confidence pre-specified (e.g. 58 rules with support ≥ 2 and confidence ≥ 95% for weather data)

Interpreting association rules

- Interpretation is not obvious:

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

is *not* the same as

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

- It means that the following also holds:

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

Rules with exceptions

- Idea: allow rules to have *exceptions*
- Example: rule for iris data

```
If petal-length ≥ 2.45 and petal-length < 4.45 then Iris-versicolor
```

- New instance:

Sepal length	Sepal width	Petal length	Petal width	Type
5.1	3.5	2.6	0.2	Iris-setosa

- Modified rule:

```
If petal-length ≥ 2.45 and petal-length < 4.45 then Iris-versicolor
  EXCEPT if petal-width < 1.0 then Iris-setosa
```

A more complex example

- Exceptions to exceptions to exceptions …

```
default: Iris-setosa
except if petal-length ≥ 2.45 and petal-length < 5.355
          and petal-width < 1.75
       then Iris-versicolor
            except if petal-length ≥ 4.95 and petal-width < 1.55
                   then Iris-virginica
                   else if sepal-length < 4.95 and sepal-width ≥ 2.45
                        then Iris-virginica
       else if petal-length ≥ 3.35
            then Iris-virginica
                 except if petal-length < 4.85 and sepal-length < 5.95
                        then Iris-versicolor
```

Advantages of using exceptions

- Rules can be updated incrementally
  - Easy to incorporate new data
  - Easy to incorporate domain knowledge
- People often think in terms of exceptions
- Each conclusion can be considered just in the context of rules and exceptions that lead to it
  - Locality property is important for understanding large rule sets
  - “Normal” rule sets don’t offer this advantage

More on exceptions

- `Default...except if...then...`

  is logically equivalent to

  `if...then...else`

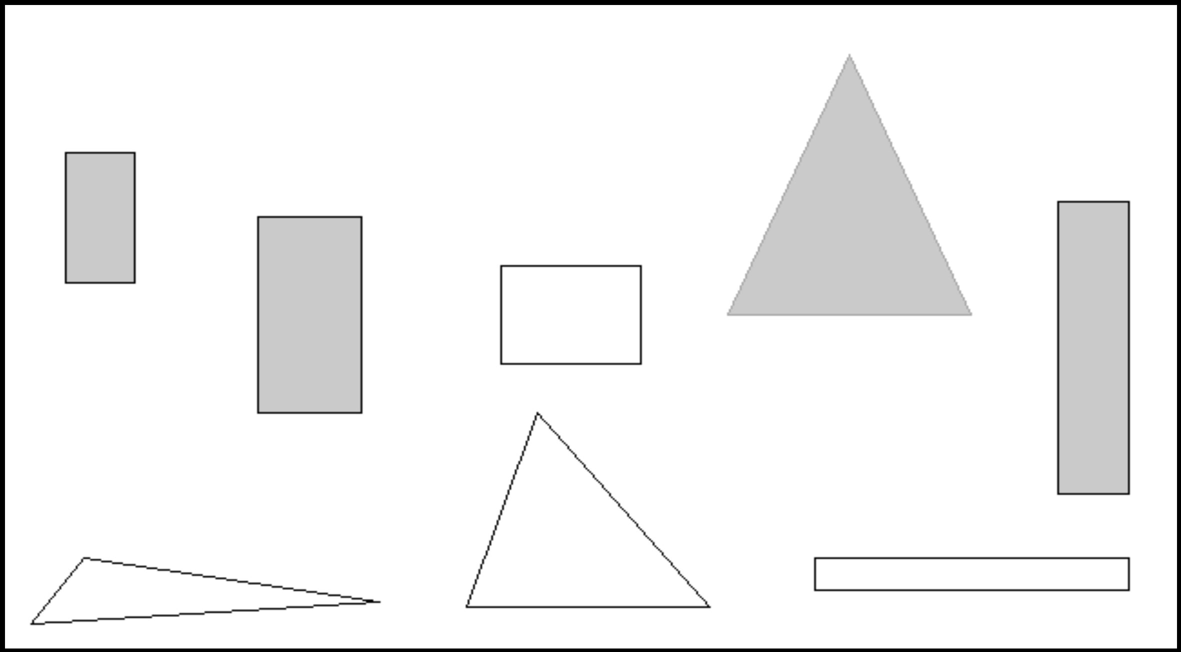
  (where the else specifies what the default did)
- But: exceptions offer a psychological advantage
  - Assumption: defaults and tests early on apply more widely than exceptions further down
  - Exceptions reflect special cases

Rules involving relations

- So far: all rules involved comparing an attribute-value to a constant (e.g. temperature < 45)
- These rules are called “propositional” because they have the same expressive power as propositional logic
- What if problem involves relationships between examples (e.g. family tree problem from above)?
  - Can’t be expressed with propositional rules
  - More expressive representation required

The shapes problem

- Target concept: *standing up*
- Shaded: *standing*
  Unshaded: *lying*



A propositional solution

Width	Height	Sides	Class
2	4	4	Standing
3	6	4	Standing
4	3	4	Lying
7	8	3	Standing
7	6	3	Lying
2	9	4	Standing
9	1	4	Lying
10	2	3	Lying

```
If width ≥ 3.5 and height < 7.0
   then lying
If height ≥ 3.5 then standing
```

A relational solution

- Comparing attributes with each other

```
If width > height then lying
If height > width then standing
```

- Generalizes better to new data
- Standard relations: =, <, >
- But: learning relational rules is costly
- Simple solution: add extra attributes
  (e.g. a binary attribute *is width < height?*)

Rules with variables

- Using variables and multiple relations:

```
If height_and_width_of(x,h,w) and h > w
   then standing(x)
```

- The top of a tower of blocks is standing:

```
If height_and_width_of(x,h,w) and h > w
     and is_top_of(y,x)
   then standing(x)
```

- The whole tower is standing:

```
If is_top_of(x,z) and
   height_and_width_of(z,h,w) and h > w
   and is_rest_of(x,y)and standing(y)
   then standing(x)
If empty(x) then standing(x)
```

- Recursive definition!

Inductive logic programming

- Recursive definition can be seen as logic program
- Techniques for learning logic programs stem from the area of “inductive logic programming” (ILP)
- But: recursive definitions are hard to learn
  - Also: few practical problems require recursion
  - Thus: many ILP techniques are restricted to non-recursive definitions to make learning easier

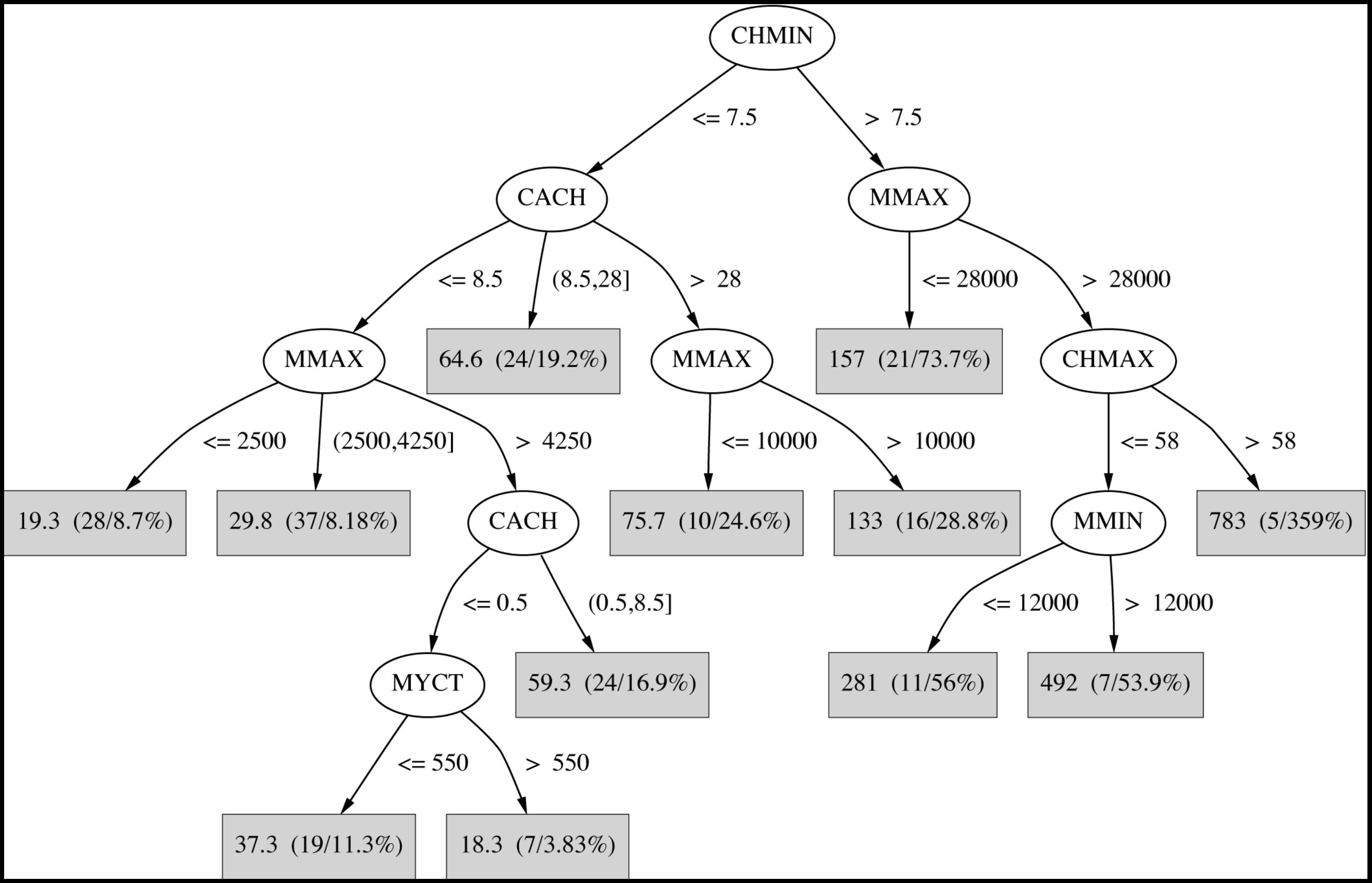
Trees for numeric prediction

- *Regression*: the process of computing an expression that predicts a numeric quantity
- *Regression tree*: “decision tree” where each leaf predicts a numeric quantity
  - Predicted value is average value of training instances that reach the leaf
- *Model tree:* “regression tree” with linear regression models at the leaf nodes
  - Linear patches approximate continuous function

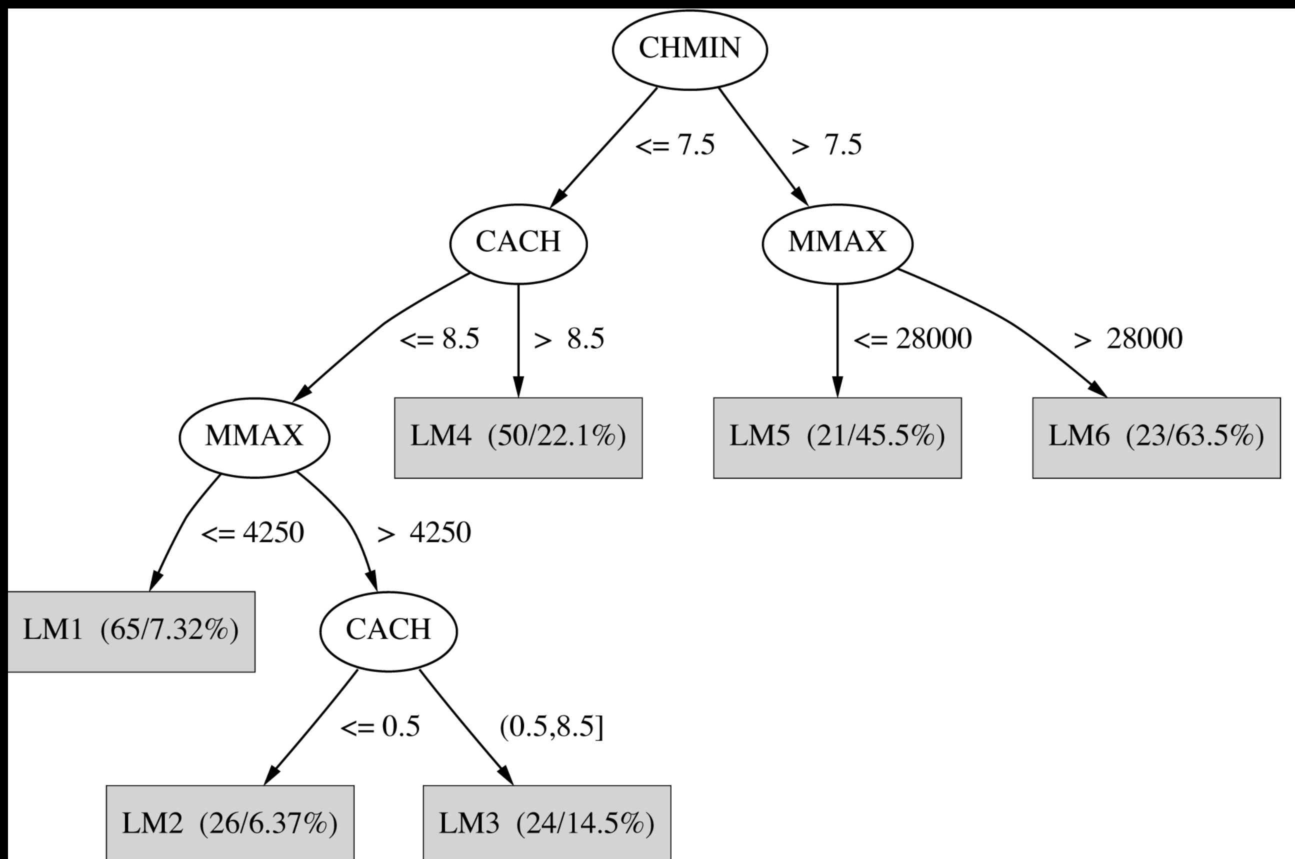
Linear regression for the CPU data

```
PRP =
   - 56.1
    + 0.049 MYCT
    + 0.015 MMIN
    + 0.006 MMAX
    + 0.630 CACH
    - 0.270 CHMIN
    + 1.46 CHMAX
```

Regression tree for the CPU data



Model tree for the CPU data



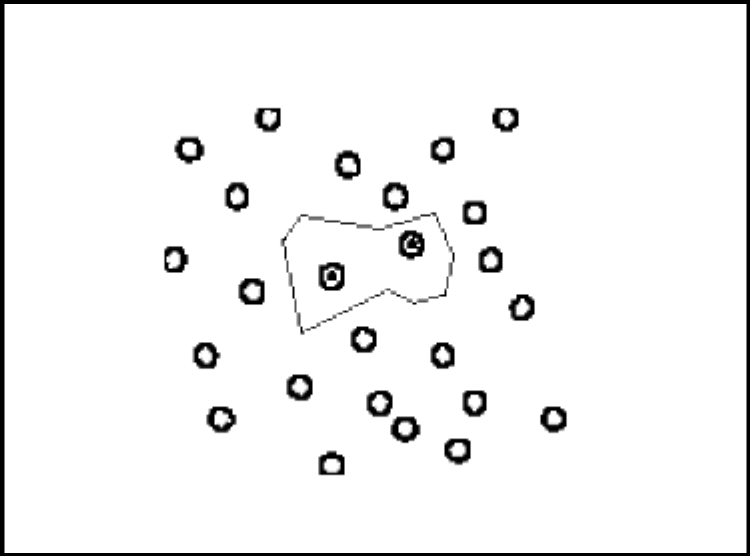
Instance-based representation

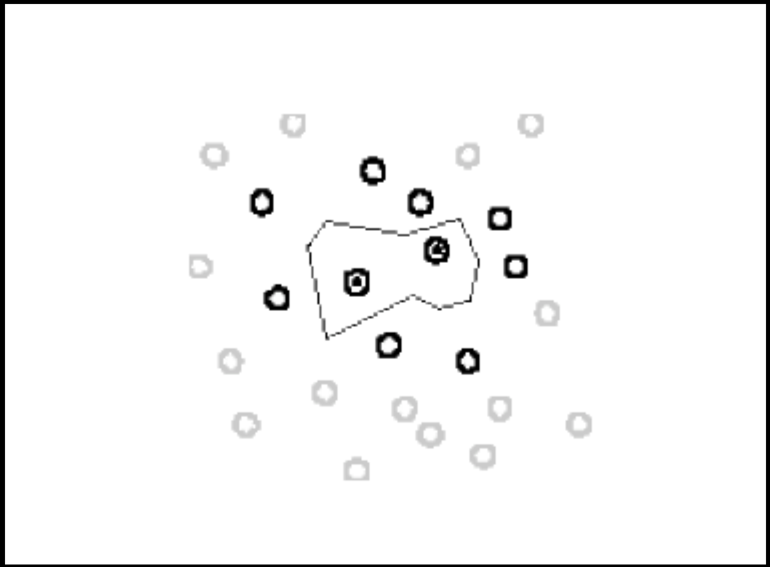
- Simplest form of learning: *rote learning*
  - Training instances are searched for instance that most closely resembles new instance
  - The instances themselves represent the knowledge
  - Also called *instance-based* learning
- Similarity function defines what's "learned"
- Instance-based learning is *lazy* learning
- Methods: *nearest-neighbor, k-nearest-neighbor, …*

The distance function

- 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
- Are all attributes equally important?
  - Weighting the attributes might be necessary

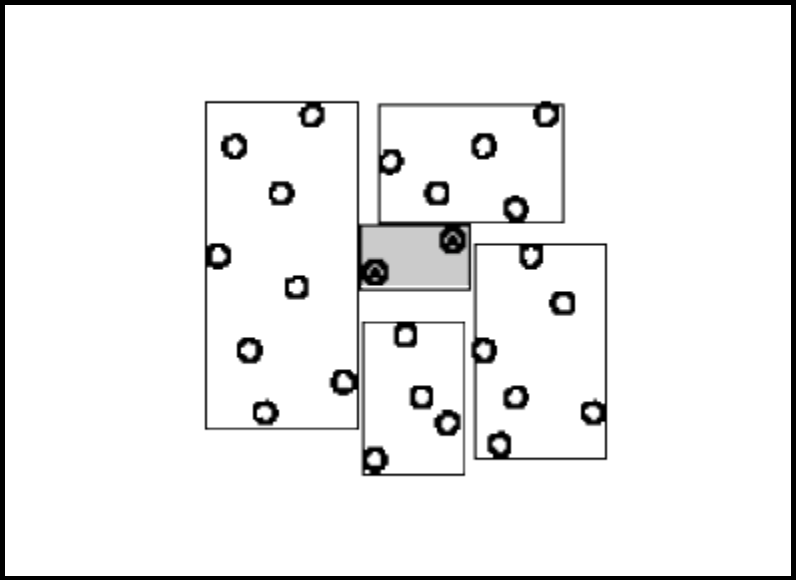
Learning prototypes

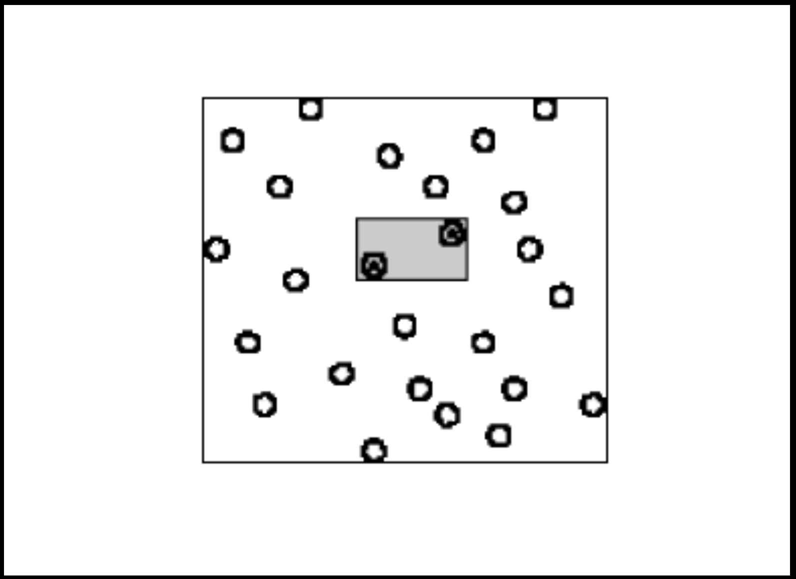




- Only those instances involved in a decision need to be stored
- Noisy instances should be filtered out
- Idea: only use *prototypical* examples

Rectangular generalizations

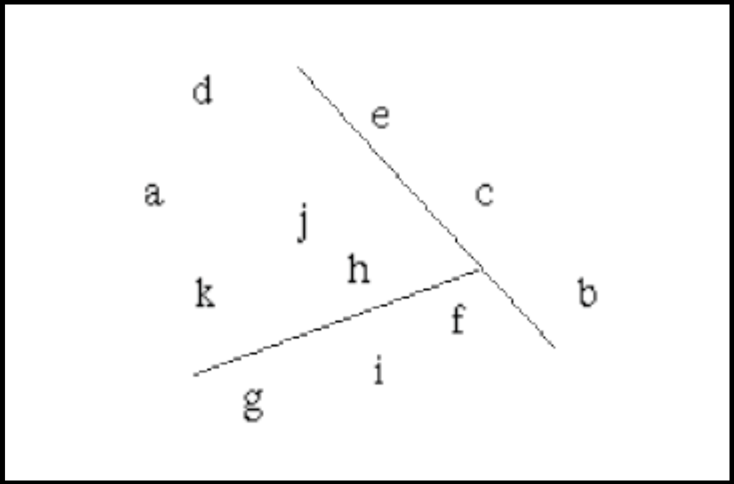




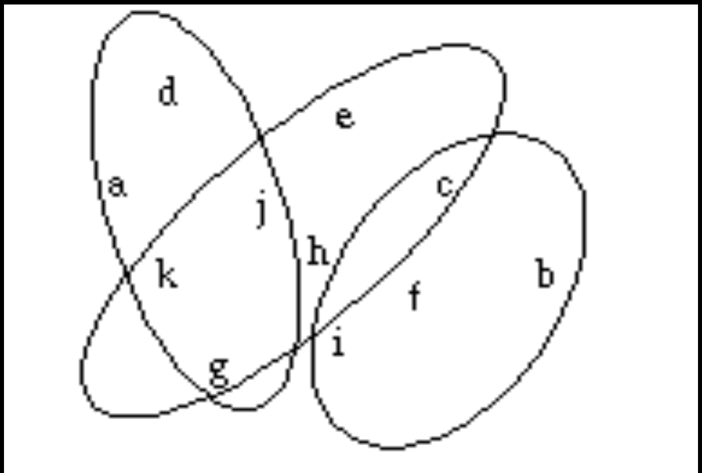
- Nearest-neighbor rule is used outside rectangles
- Rectangles are rules! (But they can be more conservative than “normal” rules.)
- Nested rectangles are rules with exceptions

Representing clusters I

Simple 2-D representation



Venn diagram



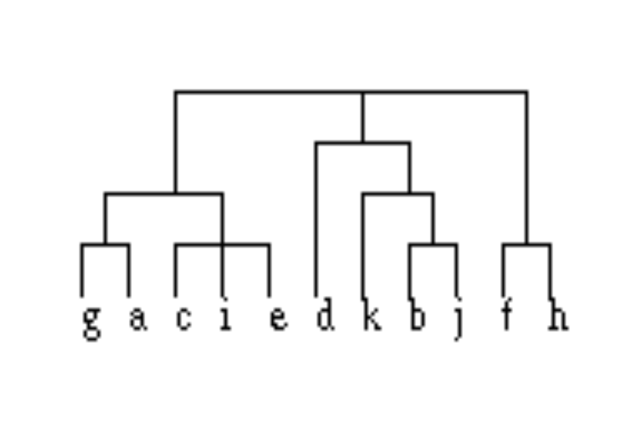
Overlapping clusters

Representing clusters II

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
…			

Dendrogram



NB: dendron is the Greek word for tree