Data Mining

1.1 Introduction

Fall 2008

Instructor: Dr. Masoud Yaghini

Outline

- Definitions
- Simple Examples
- Practical Applications
- References

Definitions

Data vs. Information

- Society produces huge amounts of data
  - Sources: business, science, medicine, economics, geography, environment, sports, …
- Data vs. Information:
  - **Data**: recorded facts
  - **Information**: patterns underlying the data
- Raw data is useless: need techniques to extract information from it.

Data vs. Information

- People have been seeking patterns in data since human life began.
  - Hunters seek patterns in animal migration behavior
  - Farmers seek patterns in crop growth
  - Politicians seek patterns in voter opinion
  - Lovers seek patterns in their partners' responses

Data Mining

- **Data Mining** is the process of extracting implicit, previously unknown, potentially useful information from data.
- **Data Mining** is the process of discovering useful patterns in large quantities of data.
- Needed: programs that detect patterns and regularities in the data

Structural Pattern

- **Structural patterns** are descriptions that explicitly
  - Can be used to predict outcome in new situation
  - Can be used to understand and explain how prediction is derived
- **Concepts**:
  - things that can be learned
- **Concept description:**
  - output of learning process

Input

- Components of the input:
  - **Instances:** the individual, independent examples of a concept
  - **Attributes:** measuring aspects of an instance

Machine Learning

- **Machine Learning**: algorithms for finding and describing structural patterns in data.
- These structural patterns in data are used as a tool for helping to explain that data and make predictions from it.

Types of learning

- Types of learning:
  - **Classification learning**
    - predicting a discrete class
  - **Association learning**
    - detecting associations between features
  - **Clustering**
    - grouping similar instances into clusters
  - **Numeric prediction**
    - predicting a numeric quantity

Machine learning and statistics

- Historical difference (grossly oversimplified):
  - Statistics: testing hypotheses
  - Machine learning: finding the right hypothesis
- ML has arisen out of computer science.
- But: huge overlap
  - Decision trees
    - Breiman, 1984 as a statisticians
    - Quinlan, 1970s and early 1980s as a ML researcher

Machine learning and statistics

- ML researchers adapt the statistical techniques
  - to improve performance
  - to make the procedure more efficient computationally.
- Most ML researchers employ statistical techniques:
  - From the beginning, visualization of data, selection of attributes, discarding outliers, and so on.
  - Statistical tests are used to validate machine learning models and to evaluate machine learning algorithms.

Simple Examples

Example: The contact lens data

- This example gives the conditions under which an optician might want to prescribe
  - Soft contact lenses,
  - Hard contact lenses, or
  - No contact lenses at all
- Instances in a dataset are characterized by the values of features, or *attributes.*
- In this example there are four attributes: *age, spectacle prescription, astigmatism,* and *tear production rate.*

Example: The contact lens data

- There are 24 cases, representing
  - three possible values of age
  - two possible values of spectacle prescription
  - two possible values of astigmatism
  - two possible values of tear production rate
  - (3 * 2 * 2 * 2 = 24).
- All possible combinations of the attribute values are represented in the table.

Example: The contact lens data

Age	Spectacle prescription	Astigmatism	Tear production rate	Recommended lenses
young	myope	no	reduced	none
young	myope	no	normal	soft
young	myope	yes	reduced	none
young	myope	yes	normal	hard
young	hypermetrope	no	reduced	none
young	hypermetrope	no	normal	soft
young	hypermetrope	yes	reduced	none
young	hypermetrope	yes	normal	hard
pre-presbyopic	myope	no	reduced	none
pre-presbyopic	myope	no	normal	soft
pre-presbyopic	myope	yes	reduced	none
pre-presbyopic	myope	yes	normal	hard
pre-presbyopic	hypermetrope	no	reduced	none
pre-presbyopic	hypermetrope	no	normal	soft
pre-presbyopic	hypermetrope	yes	reduced	none
pre-presbyopic	hypermetrope	yes	normal	none
presbyopic	myope	no	reduced	none
presbyopic	myope	no	normal	none
presbyopic	myope	yes	reduced	none
presbyopic	myope	yes	normal	hard
presbyopic	hypermetrope	no	reduced	none
presbyopic	hypermetrope	no	normal	soft
presbyopic	hypermetrope	yes	reduced	none
presbyopic	hypermetrope	yes	normal	none

Example: The contact lens data

- Part of a structural pattern of this information might be as follows:

```
If tear production rate = reduced then recommendation = none
Otherwise, if age = young and astigmatic = no
               then recommendation = soft
```

Contact lenses problem

- Rules for the contact lenses dataset:

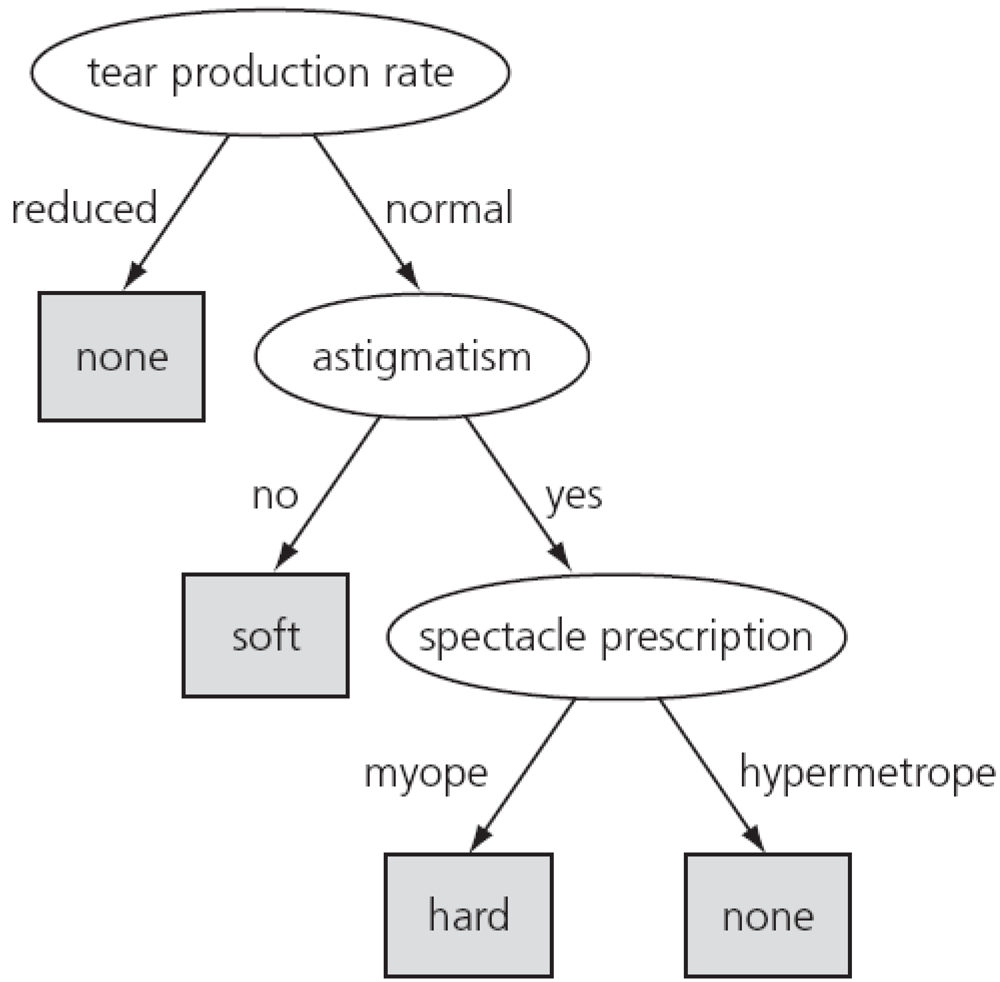
```
If tear production rate = reduced then recommendation = none
If age = young and astigmatic = no and
   tear production rate = normal then recommendation = soft
If age = pre-presbyopic and astigmatic = no and
   tear production rate = normal then recommendation = soft
If age = presbyopic and spectacle prescription = myope and
   astigmatic = no then recommendation = none
If spectacle prescription = hypermetrope and astigmatic = no and
   tear production rate = normal then recommendation = soft
If spectacle prescription = myope and astigmatic = yes and
   tear production rate = normal then recommendation = hard
If age = young and astigmatic = yes and
   tear production rate = normal then recommendation = hard
If age = pre-presbyopic and
   spectacle prescription = hypermetrope and astigmatic = yes
   then recommendation = none
If age = presbyopic and spectacle prescription = hypermetrope
   and astigmatic = yes then recommendation = none
```

Contact lenses problem

- In real-life datasets:
  - sometimes there are situations in which no rule applies;
  - Sometimes more than one rule may apply, resulting in conflicting recommendations.
  - Sometimes probabilities or weights may be associated with the rules themselves to indicate that some are more important, or more reliable, than others.

Contact lenses problem

- A decision tree for the contact lenses data



Weather problem

- This example supposedly concerns the conditions that are suitable for playing some unspecified game.
- There are four attributes: *outlook, temperature, humidity,* and *windy.*
- The four attributes have values that are symbolic categories rather than numbers.
  - *Outlook* can be *sunny, overcast,* or *rainy*
  - *Temperature* can be *hot, mild,* or *cool*
  - Humidity can be *high* or *normal*
  - Windy can be *true* or *false*

Weather problem

- The attributes create 36 possible combinations (3 * 3 * 2 * 2 = 36), of which 14 are present in the set of input examples.

Outlook	Temperature	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

Weather problem

- A set of rules learned from this information might look as follows:

```
If outlook = sunny and humidity = high then play = no
If outlook = rainy and windy = true    then play = no
If outlook = overcast                  then play = yes
If humidity = normal                   then play = yes
If none of the above                   then play = yes
```

- These rules are meant to be interpreted in order: the first one, then if it doesn't apply the second, and so on.

Weather problem

- A set of rules that are intended to be interpreted in sequence is called a **decision list**.
- Some of the rules are incorrect if they are taken individually. For example, the rule if humidity = normal then play = yes

Weather problem

- Weather data with some numeric attributes

Outlook	Temperature	Humidity	Windy	Play
sunny	85	85	false	no
sunny	80	90	true	no
overcast	83	86	false	yes
rainy	70	96	false	yes
rainy	68	80	false	yes
rainy	65	70	true	no
overcast	64	65	true	yes
sunny	72	95	false	no
sunny	69	70	false	yes
rainy	75	80	false	yes
sunny	75	70	true	yes
overcast	72	90	true	yes
overcast	81	75	false	yes
rainy	71	91	true	no

Weather problem

- The problem with numeric attributes is called a ***numeric-attribute problem***
- This case is a ***mixed-attribute problem*** because not all attributes are numeric.
- The first rule given might take the following form:

```
If outlook = sunny and humidity > 83 then play = no
```

Weather problem

- ***Classification rule:***
  - The rules we have seen so far are *classification rules:*
  - Rules predict value of a given attribute
  - In weather problem the rules predict the classification of the example in terms of whether to play or not.

- Example:

```
If outlook = sunny and humidity > 83 then play = no
```

Weather problem

- ***Association rule:***
  - look for any rules that strongly associate different attribute values.
  - predicts value of arbitrary attribute (or combination)

- Example:

```
If temperature = cool                         then humidity = normal
If humidity = normal and windy = false then play = yes
If outlook = sunny and play = no           then humidity = high
If windy = false and play = no              then outlook = sunny
                                                  and humidity = high.
```

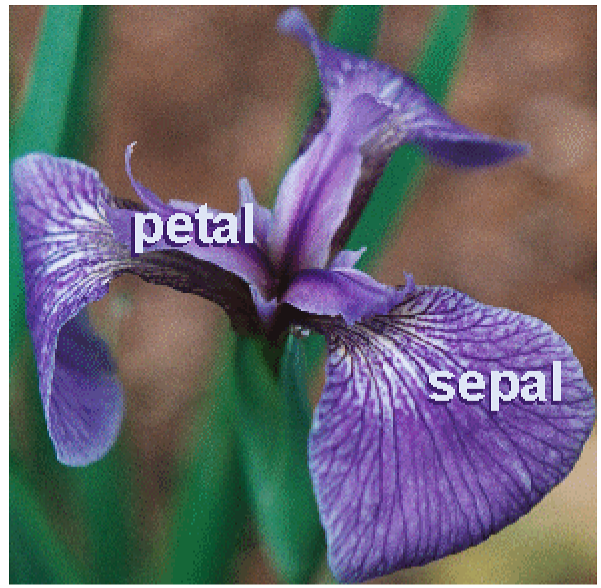
Iris flowers problem

- This dataset back to work by R.A. Fisher in the mid-1930s
- It contains 50 examples each of three types of iris.

Iris setosa *Iris versicolor* *Iris virginica*

Iris flowers problem

- There are four attributes: sepal length, sepal width, petal length, and petal width (all measured in centimeters)



- The iris dataset involves numeric attributes, the outcome—the type of iris—is a category

Iris flowers problem

	Sepal length (cm)	Sepal width (cm)	Petal length (cm)	Petal width (cm)	Type
1	5.1	3.5	1.4	0.2	*Iris setosa*
2	4.9	3.0	1.4	0.2	*Iris setosa*
3	4.7	3.2	1.3	0.2	*Iris setosa*
4	4.6	3.1	1.5	0.2	*Iris setosa*
5	5.0	3.6	1.4	0.2	*Iris setosa*
. . .					
51	7.0	3.2	4.7	1.4	*Iris versicolor*
52	6.4	3.2	4.5	1.5	*Iris versicolor*
53	6.9	3.1	4.9	1.5	*Iris versicolor*
54	5.5	2.3	4.0	1.3	*Iris versicolor*
55	6.5	2.8	4.6	1.5	*Iris versicolor*
. . .					
101	6.3	3.3	6.0	2.5	*Iris virginica*
102	5.8	2.7	5.1	1.9	*Iris virginica*
103	7.1	3.0	5.9	2.1	*Iris virginica*
104	6.3	2.9	5.6	1.8	*Iris virginica*
105	6.5	3.0	5.8	2.2	*Iris virginica*
. . .					

Classifying iris flowers

- The rules might be learned from this dataset:

```
If petal length < 2.45 then Iris setosa
If sepal width < 2.10 then Iris versicolor
If sepal width < 2.45 and petal length < 4.55 then Iris versicolor
If sepal width < 2.95 and petal width < 1.35 then Iris versicolor
If petal length ≥ 2.45 and petal length < 4.45 then Iris versicolor
If sepal length ≥ 5.85 and petal length < 4.75 then Iris versicolor
If sepal width < 2.55 and petal length < 4.95 and
   petal width < 1.55 then Iris versicolor
If petal length ≥ 2.45 and petal length < 4.95 and
   petal width < 1.55 then Iris versicolor
If sepal length ≥ 6.55 and petal length < 5.05 then Iris versicolor
If sepal width < 2.75 and petal width < 1.65 and
   sepal length < 6.05 then Iris versicolor
If sepal length ≥ 5.85 and sepal length < 5.95 and
   petal length < 4.85 then Iris versicolor
If petal length ≥ 5.15 then Iris virginica
If petal width ≥ 1.85 then Iris virginica
If petal width ≥ 1.75 and sepal width < 3.05 then Iris virginica
If petal length ≥ 4.95 and petal width < 1.55 then Iris virginica
```

CPU performance Problem

- In this example attributes and outcome are numeric.
- It concerns the relative performance of computer processing power on the basis of a number of relevant attributes

CPU performance Problem

- Attributes:
  - MYCT: machine cycle time in nanoseconds (integer)
  - MMIN: minimum main memory in kilobytes (integer)
  - MMAX: maximum main memory in kilobytes (integer)
  - CACH: cache memory in kilobytes (integer)
  - CHMIN: minimum channels in units (integer)
  - CHMAX: maximum channels in units (integer)
  - PRP: published relative performance

CPU performance Problem

- The CPU performance data: each row represents 1 of 209 different computer configurations.

	Cycle time (ns) MYCT	Main memory (KB) Min. MMIN	Main memory (KB) Max. MMAX	Cache (KB) CACH	Channels Min. CHMIN	Channels Max. CHMAX	Performance PRP
1	125	256	6000	256	16	128	198
2	29	8000	32000	32	8	32	269
3	29	8000	32000	32	8	32	220
4	29	8000	32000	32	8	32	172
5	29	8000	16000	32	8	16	132
...							
207	125	2000	8000	0	2	14	52
208	480	512	8000	32	0	0	67
209	480	1000	4000	0	0	0	45

CPU performance Problem

- **Linear regression equation**:

$$PRP = -55.9 + 0.0489\ MYCT + 0.0153\ MMIN + 0.0056\ MMAX \\ + 0.6410\ CACH - 0.2700\ CHMIN + 1.480\ CHMAX.$$

- The process of determining the weights is called **regression**
- Practical situations frequently present a mixture of numeric and nonnumeric attributes.

Labor negotiations problem

- The labor negotiations dataset is summarized the outcome of Canadian contract negotiations in 1987 and 1988.
- It includes agreements reached for organizations with at least 500 members (teachers, nurses, university staff, police, etc.).
- Each case concerns one contract, and the outcome is whether the contract is supposed ***acceptable*** or ***unacceptable***.

Labor negotiations problem

- The acceptable contracts are ones in which agreements were accepted by both labor and management.
- The unacceptable ones are either known offers that fell through because one party would not accept them.
- There are 40 examples in the dataset.
- Many of the values are unknown or missing, as indicated by question marks.

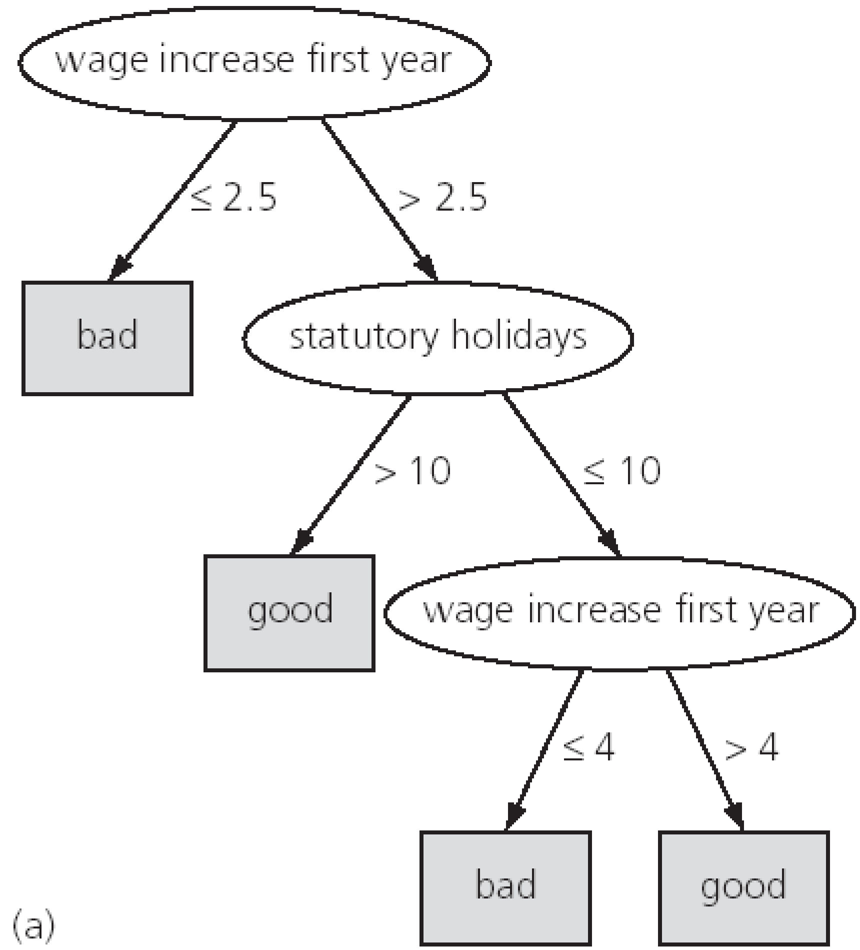
Labor negotiations problem

- **The labor negotiations data**

Attribute	Type	1	2	3	...	40
duration	years	1	2	3		2
wage increase 1st year	percentage	2%	4%	4.3%		4.5
wage increase 2nd year	percentage	?	5%	4.4%		4.0
wage increase 3rd year	percentage	?	?	?		?
cost of living adjustment	{none, tcf, tc}	none	tcf	?		none
working hours per week	hours	28	35	38		40
pension	{none, ret-allw, empl-cntr}	none	?	?		?
standby pay	percentage	?	13%	?		?
shift-work supplement	percentage	?	5%	4%		4
education allowance	{yes, no}	yes	?	?		?
statutory holidays	days	11	15	12		12
vacation	{below-avg, avg, gen}	avg	gen	gen		avg
long-term disability assistance	{yes, no}	no	?	?		yes
dental plan contribution	{none, half, full}	none	?	full		full
bereavement assistance	{yes, no}	no	?	?		yes
health plan contribution	{none, half, full}	none	?	full		half
acceptability of contract	{good, bad}	bad	good	good		good

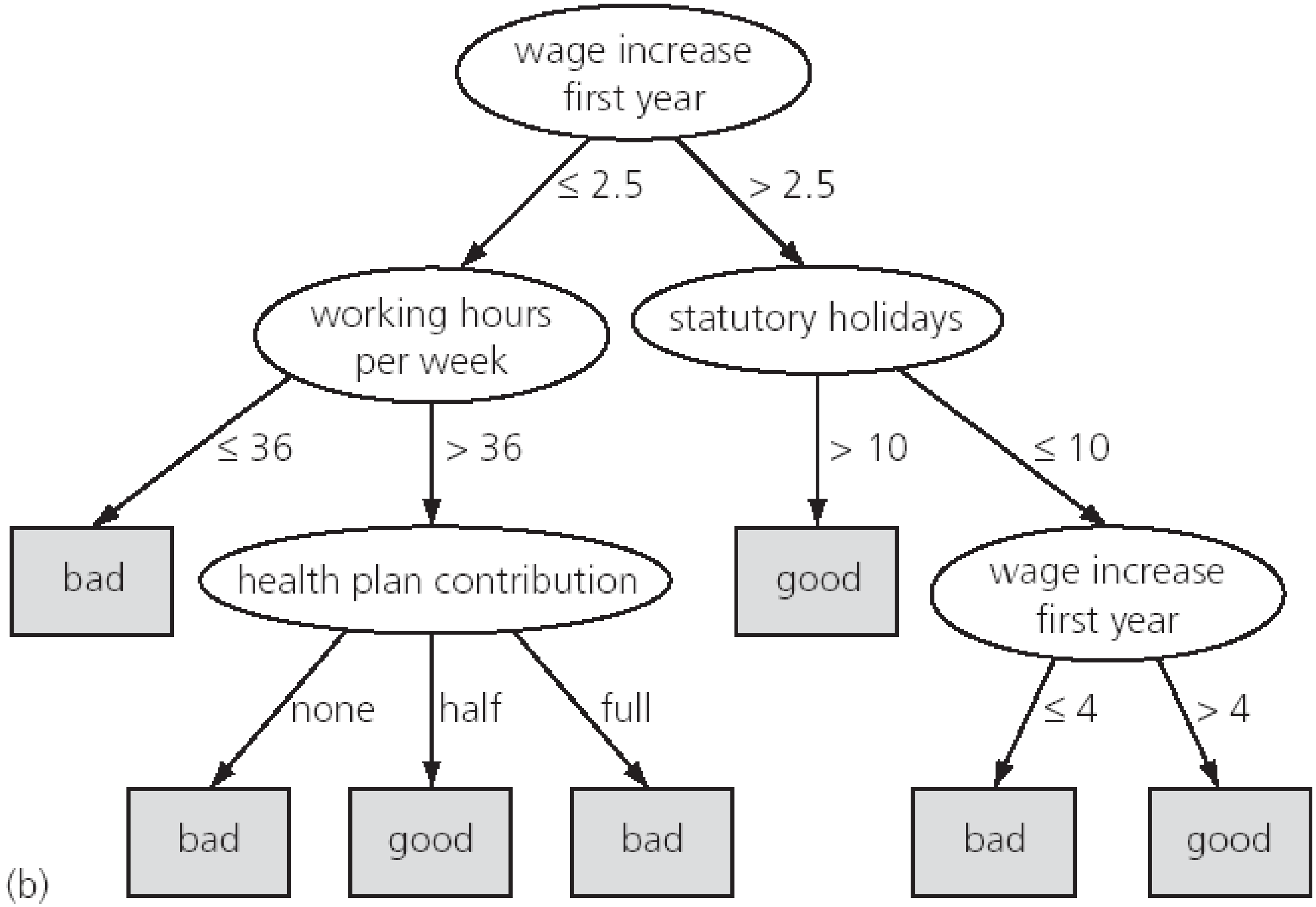
Labor negotiations problem

- A simple decision tree for the labor negotiations data.



Labor negotiations problem

- A more complex decision tree for the labor negotiations data.



Soybean diseases problem

- Identification of rules for diagnosing soybean diseases
- The data is taken from questionnaires describing plant diseases
- There are 680 examples
- Plants were measured on 35 attributes
- There are 19 disease categories

Soybean diseases problem

	Attribute	Number of values	Sample value
Environment	time of occurrence	7	July
	precipitation	3	above normal
....			
Seed	condition	2	normal
	mold growth	2	absent
....			
Fruit	condition of fruit pods	3	normal
	fruit spots	5	—
Leaf	condition	2	abnormal
	leaf spot size	3	—
....			
Stem	condition	2	abnormal
	stem lodging	2	yes
....			
Root	condition	3	normal
Diagnosis		19	diaporthe stem canker

Soybean diseases problem

- Here an example rule, learned from this data:

```
If    [leaf condition is normal and
      stem condition is abnormal and
      stem cankers is below soil line and
      canker lesion color is brown]
then
      diagnosis is rhizoctonia root rot
```

- **Domain knowledge** is necessary in data mining process

Soybean diseases problem

- Research on this problem in the late 1970s found that these diagnostic rules could be generated by a machine learning algorithm, along with rules for every other disease category, from about 300 training examples.
- These training examples were carefully selected from the amount of cases as being quite different from one another—“far apart” in the example space.
- At the same time, the plant pathologist who had produced the diagnoses was interviewed, and his expertise was translated into diagnostic rules.

Soybean diseases problem

- Surprisingly, the computer generated rules outperformed the expert-derived rules on the remaining test examples.
- They gave the correct disease top ranking 97.5% of the time compared with only 72% for the expert-derived rules.

Real-Life Applications

Processing loan applications

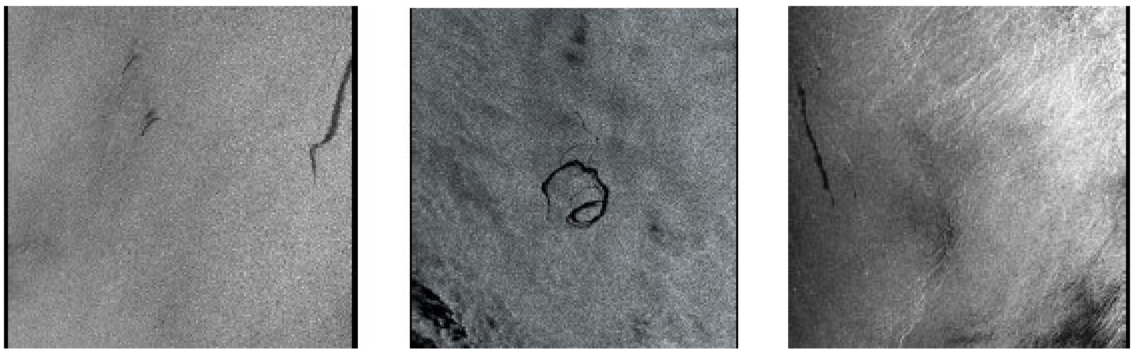
- Given: questionnaire with financial and personal information
- Question: should money be lent?
- Statistical methods are used to determine clear “accept” and “reject” cases
- Statistical method covers 90% of cases
- Borderline cases referred to loan officers
- But: 50% of accepted borderline cases defaulted!
- Solution: reject all borderline cases?

Processing loan applications

- 1000 training examples of borderline cases for which a loan had been made
- 20 attributes:
  - age
  - years with current employer
  - years at current address
  - years with the bank
  - other credit cards possessed,…
- Learned rules: correct on 70% of cases
- Rules could be used to explain decisions to customers

Screening images

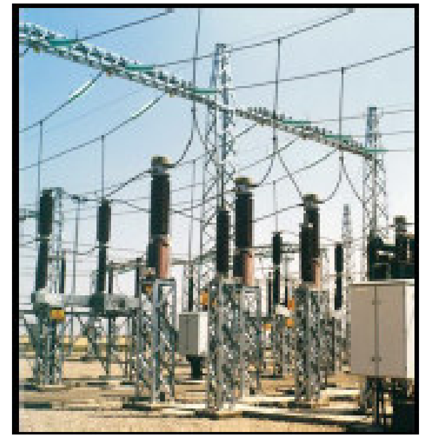
- Given: radar satellite images of coastal waters
- Problem: detect oil slicks in those images
- Oil slicks appear as dark regions
- Not easy: look-alike dark regions can be caused by weather conditions (e.g. high wind)
- Expensive process requiring highly trained personnel



Screening images

- Input Attributes:
  - size of region
  - shape
  - area
  - intensity
  - sharpness and jaggedness of boundaries
  - proximity of other regions
- Output:
  - Extract dark regions
- Some constraints:
  - Few training examples—oil slicks are rare!
  - Unbalanced data: most dark regions aren't slicks

Load forecasting



- Electricity supply companies need forecast of future demand for power
- Forecasts of min/max load for each hour
- Given: constructed load model using over the previous 15 years
- Static model consist of:
  - base load for the year
  - load periodicity over the year
  - effect of holidays
- It assumes "normal" climatic conditions
- Problem: adjust for weather conditions

Load forecasting

- Prediction corrected using “most similar” days
- Attributes:
  - temperature
  - humidity
  - wind speed
  - cloud cover readings
  - plus difference between actual load and predicted load
- Average difference among eight “most similar” days added to static model
- Linear regression coefficients form attribute weights in similarity function

Market basket analysis

- Companies precisely record massive amounts of marketing and sales data

- Special offers: identifying profitable customers and detecting their patterns of behavior that could benefit from new services (e.g. phone companies)

Market basket analysis

- Market basket analysis
  - Association techniques find groups of items that tend to occur together in a transaction
  - e.g. used to analyze supermarket checkout data may uncover the fact that on Thursdays, customers who buy diapers also buy chips

References

References

- Ian H. Witten and Eibe Frank, **Data Mining: Practical Machine Learning Tools and Techniques**, 2nd Edition, Elsevier Inc., 2005. (Chapter 1)

The end