Chapter 7. Classification and Prediction

- What is classification? What is prediction?
- Issues regarding classification and prediction
- Classification by decision tree induction
- Bayesian Classification
- Classification by backpropagation
- Classification based on concepts from association rule mining
- Other Classification Methods
- Prediction
- Classification accuracy
- Summary
Classification vs. Prediction

- **Classification:**
  - predicts categorical class labels
  - classifies data (constructs a model) based on the training set and the values (class labels) in a classifying attribute and uses it in classifying new data

- **Prediction:**
  - models continuous-valued functions, i.e., predicts unknown or missing values

- **Typical Applications**
  - credit approval
  - target marketing
  - medical diagnosis
  - treatment effectiveness analysis
Classification—A Two-Step Process

• Model construction: describing a set of predetermined classes
  – Each tuple/sample is assumed to belong to a predefined class, as determined by the class label attribute
  – The set of tuples used for model construction: training set
  – The model is represented as classification rules, decision trees, or mathematical formulae
• Model usage: for classifying future or unknown objects
  – Estimate accuracy of the model
    • The known label of test sample is compared with the classified result from the model
    • Accuracy rate is the percentage of test set samples that are correctly classified by the model
    • Test set is independent of training set, otherwise over-fitting will occur
    • A third set, validation set, is also possible
Classification Process (1): Model Construction

**Training Data**

<table>
<thead>
<tr>
<th>NAME</th>
<th>RANK</th>
<th>YEARS</th>
<th>TENURED</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mike</td>
<td>Assistant Prof</td>
<td>3</td>
<td>no</td>
</tr>
<tr>
<td>Mary</td>
<td>Assistant Prof</td>
<td>7</td>
<td>yes</td>
</tr>
<tr>
<td>Bill</td>
<td>Professor</td>
<td>2</td>
<td>yes</td>
</tr>
<tr>
<td>Jim</td>
<td>Associate Prof</td>
<td>7</td>
<td>yes</td>
</tr>
<tr>
<td>Dave</td>
<td>Assistant Prof</td>
<td>6</td>
<td>no</td>
</tr>
<tr>
<td>Anne</td>
<td>Associate Prof</td>
<td>3</td>
<td>no</td>
</tr>
</tbody>
</table>

**Classification Algorithms**

- **IF rank = ‘professor’ OR years > 6**
- **THEN tenured = ‘yes’**

**Classifier (Model)**
Classification Process (2): Use the Model in Prediction

Testing Data

Classifier

Unseen Data

<table>
<thead>
<tr>
<th>NAME</th>
<th>RANK</th>
<th>YEARS</th>
<th>TENURED</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tom</td>
<td>Assistant Prof</td>
<td>2</td>
<td>no</td>
</tr>
<tr>
<td>Merlisa</td>
<td>Associate Prof</td>
<td>7</td>
<td>no</td>
</tr>
<tr>
<td>George</td>
<td>Professor</td>
<td>5</td>
<td>yes</td>
</tr>
<tr>
<td>Joseph</td>
<td>Assistant Prof</td>
<td>7</td>
<td>yes</td>
</tr>
</tbody>
</table>

(Jeff, Professor, 4)

Tenured? Yes
Supervised vs. Unsupervised Learning

- **Supervised learning (classification)**
  - Supervision: The training data (observations, measurements, etc.) are accompanied by labels indicating the class of the observations
  - New data is classified based on the training set

- **Unsupervised learning (clustering)**
  - The class labels of training data is unknown
  - Given a set of measurements, etc. with the aim of establishing the existence of classes in the data

- **Enforced learning**
Chapter 7. Classification and Prediction

- What is classification? What is prediction?
- Issues regarding classification and prediction
- Classification by decision tree induction
- Bayesian Classification
- Classification by backpropagation
- Classification based on concepts from association rule mining
- Other Classification Methods
- Prediction
- Classification accuracy
- Summary
Issues regarding classification and prediction (1): Data Preparation

• Data cleaning
  – Preprocess data in order to reduce noise and handle missing values

• Relevance analysis (feature selection)
  – Remove the irrelevant or redundant attributes

• Data transformation
  – Generalize and/or normalize data
Issues regarding classification and prediction (2): Evaluating Classification Methods

- Predictive accuracy
- Speed and scalability
  - time to construct the model
  - time to use the model
- Robustness
  - handling noise and missing values
- Scalability
  - efficiency in disk-resident databases
- Interpretability:
  - understanding and insight provided by the model
- Goodness of rules
  - decision tree size
  - compactness of classification rules
Chapter 7. Classification and Prediction

• What is classification? What is prediction?
• Issues regarding classification and prediction
• Classification by decision tree induction
• Bayesian Classification
• Classification by backpropagation
• Classification based on concepts from association rule mining
• Other Classification Methods
• Prediction
• Classification accuracy
• Summary
Classification by Decision Tree
Induction

- Decision tree
  - A flow-chart-like tree structure
  - Internal node denotes a test on an attribute
  - Branch represents an outcome of the test
  - Leaf nodes represent class labels or class distribution
- Decision tree generation consists of two phases
  - Tree construction
    - At start, all the training examples are at the root
    - Partition examples recursively based on selected attributes
  - Tree pruning
    - Identify and remove branches that reflect noise or outliers
- Use of decision tree: Classifying an unknown sample
  - Test the attribute values of the sample against the decision tree
This follows an example from Quinlan’s ID3.

<table>
<thead>
<tr>
<th>age</th>
<th>income</th>
<th>student</th>
<th>credit_rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;=30</td>
<td>high</td>
<td>no</td>
<td>fair</td>
</tr>
<tr>
<td>&lt;=30</td>
<td>high</td>
<td>no</td>
<td>excellent</td>
</tr>
<tr>
<td>31…40</td>
<td>high</td>
<td>no</td>
<td>fair</td>
</tr>
<tr>
<td>&gt;40</td>
<td>medium</td>
<td>no</td>
<td>fair</td>
</tr>
<tr>
<td>&gt;40</td>
<td>low</td>
<td>yes</td>
<td>fair</td>
</tr>
<tr>
<td>&gt;40</td>
<td>low</td>
<td>yes</td>
<td>excellent</td>
</tr>
<tr>
<td>31…40</td>
<td>low</td>
<td>yes</td>
<td>excellent</td>
</tr>
<tr>
<td>&lt;=30</td>
<td>medium</td>
<td>no</td>
<td>fair</td>
</tr>
<tr>
<td>&lt;=30</td>
<td>low</td>
<td>yes</td>
<td>fair</td>
</tr>
<tr>
<td>&gt;40</td>
<td>medium</td>
<td>yes</td>
<td>fair</td>
</tr>
<tr>
<td>&lt;=30</td>
<td>medium</td>
<td>yes</td>
<td>excellent</td>
</tr>
<tr>
<td>31…40</td>
<td>medium</td>
<td>no</td>
<td>excellent</td>
</tr>
<tr>
<td>31…40</td>
<td>high</td>
<td>yes</td>
<td>fair</td>
</tr>
<tr>
<td>&gt;40</td>
<td>medium</td>
<td>no</td>
<td>excellent</td>
</tr>
</tbody>
</table>
Output: A Decision Tree for “buys_computer”

```
          age?
         /   \
  <=30   30..40  >40
     /     \
student? yes  credit rating?
   /  \
no yes

no  yes

excellent no

yes fair
```
Algorithm for Decision Tree Induction

• Basic algorithm (a greedy algorithm)
  – Tree is constructed in a **top-down recursive divide-and-conquer manner**
  – At start, all the training examples are at the root
  – Attributes are categorical (if continuous-valued, they are discretized in advance)
  – Examples are partitioned recursively based on selected attributes
  – Test attributes are selected on the basis of a heuristic or statistical measure (e.g., **information gain**)

• Conditions for stopping partitioning
  – All samples for a given node belong to the same class
  – There are no remaining attributes for further partitioning – **majority voting** is employed for classifying the leaf
  – There are no samples left
Attribute Selection Measure

- **Information gain** (ID3/C4.5)
  - All attributes are assumed to be categorical
  - Can be modified for continuous-valued attributes
- **Gini index** (IBM IntelligentMiner)
  - All attributes are assumed continuous-valued
  - Assume there exist several possible split values for each attribute
  - May need other tools, such as clustering, to get the possible split values
  - Can be modified for categorical attributes
Information Gain
(ID3/C4.5)

- Select the attribute with the highest information gain
- Assume there are two classes, $P$ and $N$
  - Let the set of examples $S$ contain $p$ elements of class $P$ and $n$ elements of class $N$
  - The amount of information, needed to decide if an arbitrary example in $S$ belongs to $P$ or $N$ is defined as

$$I(p, n) = -\frac{p}{p+n} \log_2 \frac{p}{p+n} - \frac{n}{p+n} \log_2 \frac{n}{p+n}$$
Information Gain in Decision Tree Induction

• Assume that using attribute A a set S will be partitioned into sets \( \{S_1, S_2, ..., S_v\} \)
  
  – If \( S_i \) contains \( p_i \) examples of \( P \) and \( n_i \) examples of \( N \), the entropy, or the expected information needed to classify objects in all subtrees \( S_i \) is

  \[
  E(A) = \sum_{i=1}^{v} \frac{p_i + n_i}{p + n} I(p_i, n_i)
  \]

• The encoding information that would be gained by branching on A

  \[
  Gain(A) = I(p, n) - E(A)
  \]
Attribute Selection by Information Gain Computation

- Class P: buys_computer = “yes”
- Class N: buys_computer = “no”
- \( I(p, n) = I(9, 5) = 0.940 \)
- Compute the entropy for age:

\[
E(\text{age}) = \frac{5}{14} I(2,3) + \frac{4}{14} I(4,0) + \frac{5}{14} I(3,2) = 0.69
\]

Hence

\[
Gain(\text{age}) = I(p,n) - E(\text{age})
\]

Similarly

\[
Gain(\text{income}) = 0.029
\]
\[
Gain(\text{student}) = 0.151
\]
\[
Gain(\text{credit_rating}) = 0.048
\]
Gini Index (IBM IntelligentMiner)

- If a data set $T$ contains examples from $n$ classes, gini index, $gini(T)$ is defined as
  \[ gini(T) = 1 - \sum_{j=1}^{n} p_j^2 \]
  where $p_j$ is the relative frequency of class $j$ in $T$.

- If a data set $T$ is split into two subsets $T_1$ and $T_2$ with sizes $N_1$ and $N_2$ respectively, the gini index of the split data contains examples from $n$ classes, the gini index $gini(T)$ is defined as
  \[ gini_{split}(T) = \frac{N_1}{N} gini(T_1) + \frac{N_2}{N} gini(T_2) \]

- The attribute provides the smallest gini split is chosen to split the node (need to enumerate all possible splitting points for each attribute).
Extracting Classification Rules from Trees

- Represent the knowledge in the form of **IF-THEN** rules
- One rule is created for each path from the root to a leaf
- Each attribute-value pair along a path forms a conjunction
- The leaf node holds the class prediction
- Rules are easier for humans to understand
- Example

  IF $age = "\leq 30"$ AND $student = "no"$ THEN $buys\_computer = "no"$
  
  IF $age = "\leq 30"$ AND $student = "yes"$ THEN $buys\_computer = "yes"$
  
  IF $age = "31...40"$ THEN $buys\_computer = "yes"
  
  IF $age = ">40"$ AND $credit\_rating = "excellent"$ THEN $buys\_computer = "yes"
  
  IF $age = ">40"$ AND $credit\_rating = "fair"$ THEN $buys\_computer = "no"$
Avoid Overfitting in Classification

- The generated tree may overfit the training data
  - Too many branches, some may reflect anomalies due to noise or outliers
  - Result is in poor accuracy for unseen samples
- Two approaches to avoid overfitting
  - Prepruning: Halt tree construction early—do not split a node if this would result in the goodness measure falling below a threshold
    - Difficult to choose an appropriate threshold
  - Postpruning: Remove branches from a “fully grown” tree—get a sequence of progressively pruned trees
    - Use a set of data different from the training data to decide which is the “best pruned tree”
Approaches to Determine the Final Tree Size

- Separate training (2/3) and testing (1/3) sets
- Use cross validation, e.g., 10-fold cross validation
- Use all the data for training
  - but apply a statistical test (e.g., chi-square) to estimate whether expanding or pruning a node may improve the entire distribution
- Use minimum description length (MDL) principle:
Enhancements to basic decision tree induction

• Allow for continuous-valued attributes
  – Dynamically define new discrete-valued attributes that partition the continuous attribute value into a discrete set of intervals
• Handle missing attribute values
  – Assign the most common value of the attribute
  – Assign probability to each of the possible values
• Attribute construction
  – Create new attributes based on existing ones that are sparsely represented
  – This reduces fragmentation, repetition, and replication
Classification in Large Databases

- Classification—a classical problem extensively studied by statisticians and machine learning researchers
- Scalability: Classifying data sets with millions of examples and hundreds of attributes with reasonable speed
- Why decision tree induction in data mining?
  - relatively faster learning speed (than other classification methods)
  - convertible to simple and easy to understand classification rules
  - can use SQL queries for accessing databases
  - comparable classification accuracy with other methods
Scalable Decision Tree Induction Methods in Data Mining Studies

- **SLIQ (EDBT’96 — Mehta et al.)**
  - builds an index for each attribute and only class list and the current attribute list reside in memory

- **SPRINT (VLDB’96 — J. Shafer et al.)**
  - constructs an attribute list data structure

- **PUBLIC (VLDB’98 — Rastogi & Shim)**
  - integrates tree splitting and tree pruning: stop growing the tree earlier

- **RainForest (VLDB’98 — Gehrke, Ramakrishnan & Ganti)**
  - separates the scalability aspects from the criteria that determine the quality of the tree
  - builds an AVC-list (attribute, value, class label)
Data Cube-Based Decision-Tree Induction

- Integration of generalization with decision-tree induction (Kamber et al’97).
- Classification at primitive concept levels
  - E.g., precise temperature, humidity, outlook, etc.
  - Low-level concepts, scattered classes, bushy classification-trees
  - Semantic interpretation problems.
- Cube-based multi-level classification
  - Relevance analysis at multi-levels.
  - Information-gain analysis with dimension + level.
Presentation of Classification Results

Classification attribute: product
- Environmental Line
- GO Sport Line
- Outdoor Products

Dim: cost
Level: Level 0
Cost: 85%
Chapter 7. Classification and Prediction

• What is classification? What is prediction?
• Issues regarding classification and prediction
• Classification by decision tree induction
• Bayesian Classification
• Classification by backpropagation
• Classification based on concepts from association rule mining
• Other Classification Methods
• Prediction
• Classification accuracy
• Summary
Bayesian Classification: Why?

- **Probabilistic learning**: Calculate explicit probabilities for hypothesis, among the most practical approaches to certain types of learning problems.
- **Incremental**: Each training example can incrementally increase/decrease the probability that a hypothesis is correct. Prior knowledge can be combined with observed data.
- **Probabilistic prediction**: Predict multiple hypotheses, weighted by their probabilities.
- **Standard**: Even when Bayesian methods are computationally intractable, they can provide a standard of optimal decision making against which other methods can be measured.
Bayesian Theorem

- Given training data $D$, *posteriori probability of a hypothesis* $h$, $P(h|D)$ follows the Bayes theorem

\[ P(h|D) = \frac{P(D|h)P(h)}{P(D)} \]

- MAP (maximum posteriori) hypothesis

\[ h_{MAP} \equiv \arg\max_{h\in H} P(h|D) = \arg\max_{h\in H} P(D|h)P(h). \]

- Practical difficulty: require initial knowledge of many probabilities, significant computational cost
Naïve Bayes Classifier (I)

• A simplified assumption: attributes are conditionally independent:

\[ P(C_j|V) \propto P(C_j)\prod_{i=1}^{n} P(v_i|C_j) \]

• Greatly reduces the computation cost, only count the class distribution.
Naive Bayesian Classifier (II)

- Given a training set, we can compute the probabilities

<table>
<thead>
<tr>
<th>Outlook</th>
<th>P</th>
<th>N</th>
<th>Humidity</th>
<th>P</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>sunny</td>
<td>2/9</td>
<td>3/5</td>
<td>high</td>
<td>3/9</td>
<td>4/5</td>
</tr>
<tr>
<td>overcast</td>
<td>4/9</td>
<td>0</td>
<td>normal</td>
<td>6/9</td>
<td>1/5</td>
</tr>
<tr>
<td>rain</td>
<td>3/9</td>
<td>2/5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Temperature</th>
<th>P</th>
<th>N</th>
<th>Windy</th>
<th>P</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>hot</td>
<td>2/9</td>
<td>2/5</td>
<td>true</td>
<td>3/9</td>
<td>3/5</td>
</tr>
<tr>
<td>mild</td>
<td>4/9</td>
<td>2/5</td>
<td>false</td>
<td>6/9</td>
<td>2/5</td>
</tr>
<tr>
<td>cool</td>
<td>3/9</td>
<td>1/5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Bayesian classification

• The classification problem may be formalized using \textit{a-posteriori probabilities}:

\[ P(C|X) = \text{prob. that the sample tuple } X = <x_1, \ldots, x_k> \text{ is of class } C. \]

• E.g. \( P(\text{class}=N \mid \text{outlook}=\text{sunny}, \text{windy}=\text{true}, \ldots) \)

• Idea: assign to sample \( X \) the class label \( C \) such that \( P(C|X) \) is maximal
Estimating a-posteriori probabilities

- **Bayes theorem:**
  \[ P(C|X) = \frac{P(X|C) \cdot P(C)}{P(X)} \]

- \( P(X) \) is constant for all classes
- \( P(C) = \) relative freq of class C samples
- \( C \) such that \( P(C|X) \) is maximum = \( C \) such that \( P(X|C) \cdot P(C) \) is maximum

- **Problem:** computing \( P(X|C) \) is difficult!
Naïve Bayesian Classification

- Naïve assumption: attribute independence
  \[ P(x_1,\ldots,x_k \mid C) = P(x_1 \mid C) \cdot \ldots \cdot P(x_k \mid C) \]
- If i-th attribute is **categorical**: 
  \[ P(x_i \mid C) \] is estimated as the relative freq of samples having value \( x_i \) as i-th attribute in class \( C \)
- If i-th attribute is **continuous**: 
  \[ P(x_i \mid C) \] is estimated thru a Gaussian density function
- Computationally easy in both cases
### Play-tennis example: estimating

<table>
<thead>
<tr>
<th>Outlook</th>
<th>Temperature</th>
<th>Humidity</th>
<th>Windy</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>sunny</td>
<td>hot</td>
<td>high</td>
<td>false</td>
<td>N</td>
</tr>
<tr>
<td>sunny</td>
<td>hot</td>
<td>high</td>
<td>true</td>
<td>N</td>
</tr>
<tr>
<td>overcast</td>
<td>hot</td>
<td>high</td>
<td>false</td>
<td>P</td>
</tr>
<tr>
<td>rain</td>
<td>mild</td>
<td>high</td>
<td>false</td>
<td>P</td>
</tr>
<tr>
<td>rain</td>
<td>cool</td>
<td>normal</td>
<td>false</td>
<td>P</td>
</tr>
<tr>
<td>rain</td>
<td>cool</td>
<td>normal</td>
<td>true</td>
<td>N</td>
</tr>
<tr>
<td>overcast</td>
<td>cool</td>
<td>normal</td>
<td>true</td>
<td>N</td>
</tr>
<tr>
<td>sunny</td>
<td>mild</td>
<td>high</td>
<td>false</td>
<td>P</td>
</tr>
<tr>
<td>sunny</td>
<td>cool</td>
<td>normal</td>
<td>false</td>
<td>P</td>
</tr>
<tr>
<td>sunny</td>
<td>mild</td>
<td>normal</td>
<td>false</td>
<td>P</td>
</tr>
<tr>
<td>overcast</td>
<td>mild</td>
<td>high</td>
<td>true</td>
<td>P</td>
</tr>
<tr>
<td>overcast</td>
<td>hot</td>
<td>normal</td>
<td>false</td>
<td>P</td>
</tr>
<tr>
<td>rain</td>
<td>mild</td>
<td>high</td>
<td>true</td>
<td>N</td>
</tr>
</tbody>
</table>

| outlook | \(P(sunny|p) = 2/9\) | \(P(sunny|n) = 3/5\) |
|----------|---------------------|---------------------|
| P(overcast| p) = 4/9 | P(overcast| n) = 0 |
| P(rain| p) = 3/9 | P(rain| n) = 2/5 |

| temperature | \(P(\text{hot}|p) = 2/9\) | \(P(\text{hot}|n) = 2/5\) |
|-------------|-------------------|-------------------|
| P(mild| p) = 4/9 | P(mild| n) = 2/5 |
| P(cool| p) = 3/9 | P(cool| n) = 1/5 |

| humidity | \(P(\text{high}|p) = 3/9\) | \(P(\text{high}|n) = 4/5\) |
|----------|------------------|------------------|
| P(normal| p) = 6/9 | P(normal| n) = 2/5 |

| windy | \(P(\text{true}|p) = 3/9\) | \(P(\text{true}|n) = 3/5\) |
|--------|-----------------|-----------------|
| P(false| p) = 6/9 | P(false| n) = 2/5 |

\(P(p) = 9/14\)

\(P(n) = 5/14\)
Play-tennis example: classifying X

- An unseen sample $X = \langle\text{rain}, \text{hot}, \text{high}, \text{false}\rangle$

- $P(X|p) \cdot P(p) = P(\text{rain}|p) \cdot P(\text{hot}|p) \cdot P(\text{high}|p) \cdot P(\text{false}|p) \cdot P(p) = \frac{3}{9} \cdot \frac{2}{9} \cdot \frac{3}{9} \cdot \frac{6}{9} \cdot \frac{9}{14} = 0.010582$

- $P(X|n) \cdot P(n) = P(\text{rain}|n) \cdot P(\text{hot}|n) \cdot P(\text{high}|n) \cdot P(\text{false}|n) \cdot P(n) = \frac{2}{5} \cdot \frac{2}{5} \cdot \frac{4}{5} \cdot \frac{2}{5} \cdot \frac{5}{14} = 0.018286$

- Sample $X$ is classified in class $n$ (don’t play)
The independence hypothesis...

- ... makes computation possible
- ... yields optimal classifiers when satisfied
- ... but is seldom satisfied in practice, as attributes (variables) are often correlated.

- Attempts to overcome this limitation:
  - **Bayesian networks**, that combine Bayesian reasoning with causal relationships between attributes
  - **Decision trees**, that reason on one attribute at the time, considering most important attributes first
Bayesian Belief Networks (I)

Bayesian Belief Networks = acyclic directed graphical models

The conditional probability table (CPT) for the variable LungCancer

<table>
<thead>
<tr>
<th></th>
<th>LC</th>
<th>~LC</th>
</tr>
</thead>
<tbody>
<tr>
<td>(FH, S)</td>
<td>0.8</td>
<td>0.2</td>
</tr>
<tr>
<td>(FH, ~S)</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>(~FH, S)</td>
<td>0.7</td>
<td>0.3</td>
</tr>
<tr>
<td>(~FH, ~S)</td>
<td>0.1</td>
<td>0.9</td>
</tr>
</tbody>
</table>
Bayesian belief network is a factorization of the overall joint distribution function into a product of conditional distributions. It allows a subset of the variables conditionally independent.

A graphical model of causal relationships

Several cases of learning Bayesian belief networks

- Given both network structure and all the variables: easy
- Given network structure but only some variables: gradient descent strategy
- When the network structure is not known in advance: more tricky
Chapter 7. Classification and Prediction

- What is classification? What is prediction?
- Issues regarding classification and prediction
- Classification by decision tree induction
- Bayesian Classification
- Classification by backpropagation
- Classification based on concepts from association rule mining
- Other Classification Methods
- Prediction
- Classification accuracy
- Summary
Neural Networks

• Advantages
  – prediction accuracy is generally high
  – robust, works when training examples contain errors
  – output may be discrete, real-valued, or a vector of several discrete or real-valued attributes
  – fast evaluation of the learned target function

• Criticism
  – long training time
  – difficult to understand the learned function (weights)
  – not easy to incorporate domain knowledge
A Neuron

In a neuron, the \( n \)-dimensional input vector \( x \) is mapped into variable \( y \) by means of the scalar product and a nonlinear function mapping.

- The \( n \)-dimensional input vector \( x \) is mapped into variable \( y \) by means of the scalar product and a nonlinear function mapping.

\[
\sum w_i x_i \rightarrow f(\sum w_i x_i) \rightarrow \mu_k 
\]
Network Training

- The ultimate objective of training
  - obtain a set of weights that makes almost all the tuples in the training data classified correctly

- Steps
  - Initialize weights with random values
  - Feed the input tuples into the network one by one
  - For each unit
    - Compute the net input to the unit as a linear combination of all the inputs to the unit
    - Compute the output value using the activation function
    - Compute the error
    - Update the weights and the bias
Multi-Layer Perceptron

Output vector

Output nodes

Hidden nodes

Input nodes

Input vector: \( x_i \)

\[
\text{Err}_j = O_j (1 - O_j) \sum_k \text{Err}_k w_{jk} \\
\theta_j = \theta_j + (l) \text{Err}_j \\
w_{ij} = w_{ij} + (l) \text{Err}_j O_i \\
\text{Err}_j = O_j (1 - O_j) (T_j - O_j) \\
O_j = \frac{1}{1 + e^{-I_j}} \\
I_j = \sum_i w_{ij} O_i + \theta_j
\]
Network Pruning and Rule Extraction

• Network pruning
  – Fully connected network will be hard to articulate
  – $N$ input nodes, $h$ hidden nodes and $m$ output nodes lead to $h(m+N)$ weights
  – Pruning: Remove some of the links without affecting classification accuracy of the network

• Extracting rules from a trained network
  – Discretize activation values; replace individual activation value by the cluster average maintaining the network accuracy
  – Enumerate the output from the discretized activation values to find rules between activation value and output
  – Find the relationship between the input and activation value
  – Combine the above two to have rules relating the output to input
Chapter 7. Classification and Prediction

- What is classification? What is prediction?
- Issues regarding classification and prediction
- Classification by decision tree induction
- Bayesian Classification
- Classification by backpropagation
- Classification based on concepts from association rule mining
- Other Classification Methods
- Prediction
- Classification accuracy
- Summary
Association-Based Classification

- An association rule is a simple probabilistic statement about the co-occurrence of certain events in a database.
- Several methods for association-based classification
  - **AssociationRuleClusteringSystem**: Quantitative association mining and clustering of association rules (Lent et al’97): \( A_{\text{quant1}} \land A_{\text{quant2}} \Rightarrow A_{\text{cat}} \)
    - It beats C4.5 in (mainly) scalability and also accuracy
  - **Associative classification**: (Liu et al’98)
    - It mines high support and high confidence rules in the form of “cond_set => y”, where y is a class label
    - frequent, support, confidence
  - **CAEP (Classification by aggregating emerging patterns)**: (Dong et al’99)
    - Emerging patterns (EPs): the itemsets whose support increases significantly from one class to another, Mine Eps based on minimum support and growth rate
Chapter 7. Classification and Prediction

• What is classification? What is prediction?
• Issues regarding classification and prediction
• Classification by decision tree induction
• Bayesian Classification
• Classification by backpropagation
• Classification based on concepts from association rule mining
• Other Classification Methods
• Prediction
• Classification accuracy
• Summary
Other Classification Methods

- k-nearest neighbor classifier
- case-based reasoning
- Genetic algorithm
- Rough set approach
- Fuzzy set approaches
Instance-Based Methods

• Instance-based learning:
  – Store training examples and delay the processing (“lazy evaluation”) until a new instance must be classified

• Typical approaches
  – $k$-nearest neighbor approach
    • Instances represented as points in a Euclidean space.
  – Locally weighted regression
    • Constructs local approximation
  – Case-based reasoning
    • Uses symbolic representations and knowledge-based inference
The $k$-Nearest Neighbor Algorithm

- All instances correspond to points in the n-D space.
- The nearest neighbor are defined in terms of Euclidean distance.
- The target function could be discrete- or real- valued.
- For discrete-valued, the $k$-NN returns the most common value among the $k$ training examples nearest to $x_q$.
- Voronoi diagram: the decision surface induced by 1-NN for a typical set of training examples.
Discussion on the \textit{k-NN Algorithm}

- The \textit{k-NN} algorithm for continuous-valued target functions
  - Calculate the mean values of the \textit{k} nearest neighbors

- Distance-weighted nearest neighbor algorithm
  - Weight the contribution of each of the \textit{k} neighbors according to their distance to the query point \( x_q \)
  - Giving greater weight to closer neighbors
  - Similarly, for real-valued target functions

- Robust to noisy data by averaging \textit{k}-nearest neighbors

- Curse of dimensionality: distance between neighbors could be dominated by irrelevant attributes.
  - To overcome it, axes stretch or elimination of the least relevant attributes.

\[ w \equiv \frac{1}{d(x_q, x_i)^2} \]
Case-Based Reasoning

• **Also uses:** lazy evaluation + analyze similar instances
• **Difference:** Instances are not “points in a Euclidean space”
• **Example:** Water faucet problem in CADET (Sycara et al’92)
• **Methodology**
  – Instances represented by rich symbolic descriptions (e.g., function graphs)
  – Multiple retrieved cases may be combined
  – Tight coupling between case retrieval, knowledge-based reasoning, and problem solving
• **Research issues**
  – Indexing based on syntactic similarity measure, and when failure, backtracking, and adapting to additional cases
Remarks on Lazy vs. Eager Learning

- **Instance-based learning:** lazy evaluation
- **Decision-tree and Bayesian classification:** eager evaluation
- **Key differences**
  - Lazy method may consider query instance $x_q$ when deciding how to generalize beyond the training data $D$
  - Eager method cannot since they have already chosen global approximation when seeing the query
- **Efficiency:** Lazy - less time training but more time predicting
- **Accuracy**
  - Lazy method effectively uses a richer hypothesis space since it uses many local linear functions to form its implicit global approximation to the target function
  - Eager: must commit to a single hypothesis that covers the entire instance space
Genetic Algorithms

- GA: based on an analogy to biological evolution
- Each rule is represented by a string of bits
- An initial population is created consisting of randomly generated rules
  - e.g., IF $A_1$ and Not $A_2$ then $C_2$ can be encoded as 100
- Based on the notion of survival of the fittest, a new population is formed to consists of the fittest rules and their offsprings
- The fitness of a rule is represented by its classification accuracy on a set of training examples
- Offsprings are generated by crossover and mutation
Rough Set Approach

- Rough sets are used to approximately or “roughly” define equivalent classes.
- A rough set for a given class C is approximated by two sets: a **lower approximation** (certain to be in C) and an **upper approximation** (cannot be described as not belonging to C).
- Finding the minimal subsets (reducts) of attributes (for feature reduction) is NP-hard but a discernibility matrix is used to reduce the computation intensity.
Fuzzy Set Approaches

- Fuzzy logic uses truth values between 0.0 and 1.0 to represent the degree of membership (such as using fuzzy membership graph)
- Attribute values are converted to fuzzy values
  - e.g., income is mapped into the discrete categories \{low, medium, high\} with fuzzy values calculated
- For a given new sample, more than one fuzzy value may apply
- Each applicable rule contributes a vote for membership in the categories
- Typically, the truth values for each predicted category are summed
Chapter 7. Classification and Prediction

- What is classification? What is prediction?
- Issues regarding classification and prediction
- Classification by decision tree induction
- Bayesian Classification
- Classification by backpropagation
- Classification based on concepts from association rule mining
- Other Classification Methods
- Prediction
- Classification accuracy
- Summary
What Is Prediction?

- Prediction is similar to classification
  - First, construct a model
  - Second, use model to predict unknown value
    - Major method for prediction is regression
      - Linear and multiple regression
      - Non-linear regression
- Prediction is different from classification
  - Classification refers to predict categorical class label
  - Prediction models continuous-valued functions
Predictive Modeling in Databases

- Predictive modeling: Predict data values or construct generalized linear models based on the database data.
- One can only predict value ranges or category distributions
- Method outline:
  - Minimal generalization
  - Attribute relevance analysis
  - Generalized linear model construction
  - Prediction
- Determine the major factors which influence the prediction
  - Data relevance analysis: uncertainty measurement, entropy analysis, expert judgement, etc.
- Multi-level prediction: drill-down and roll-up analysis
Regress Analysis and Log-Linear Models in Prediction

- **Linear regression**: \( Y = \alpha + \beta X \)
  - Two parameters, \( \alpha \) and \( \beta \), specify the line and are to be estimated by using the data at hand.
  - Using the least squares criterion to the known values of \( Y_1, Y_2, \ldots, X_1, X_2, \ldots \).

- **Multiple regression**: \( Y = b_0 + b_1 X_1 + b_2 X_2 \).
  - Many nonlinear functions can be transformed into the above.

- **Log-linear models**:
  - The multi-way table of joint probabilities is approximated by a product of lower-order tables.
  - Probability: \( p(a, b, c, d) = \alpha_{ab} \beta_{ac} \gamma_{ad} \delta_{bcd} \)
Locally Weighted Regression

- Construct an explicit approximation to $f$ over a local region surrounding query instance $x_q$.

- Locally weighted linear regression:
  - The target function $f$ is approximated near $x_q$ using the linear function:
    $$\hat{f}(x) = w_0 + w_1 a_1(x) + \cdots + w_n a_n(x)$$
  - minimize the squared error: distance-decreasing weight $K$
    $$E(x_q) \equiv \frac{1}{2} \sum_{x \in k \text{ nearest neighbors of } x_q} (f(x) - \hat{f}(x))^2 K(d(x_q,x))$$
  - the gradient descent training rule:
    $$\Delta w_j \equiv \eta \sum_{x \in k \text{ nearest neighbors of } x_q} K(d(x_q,x))((f(x) - \hat{f}(x))a_j(x))$$

- In most cases, the target function is approximated by a constant, linear, or quadratic function.
Prediction: Numerical Data

Relevance Analysis

<table>
<thead>
<tr>
<th>#</th>
<th>Predictive Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Sale_Price</td>
<td>-4950.00 ~ 51950.000</td>
</tr>
<tr>
<td>2</td>
<td>Channel</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Cost_of_Goods_Sold</td>
<td>-3500.00 ~ 85900.000</td>
</tr>
<tr>
<td>4</td>
<td>Advertising_Cost</td>
<td>0.000 ~ 1715.000</td>
</tr>
<tr>
<td>5</td>
<td>Average_Sales_Area</td>
<td>1130.000 ~ 4230.000</td>
</tr>
</tbody>
</table>

Profit:

- Red: -365.00~480.00
- Blue: 805.00~1000.00
- Magenta: 1260.00~6005.00
- Green: 480.00~805.00
- Yellow: 1130.00~1250.00

For Help, press F1
Prediction: Categorical Data

Relevance Analysis

<table>
<thead>
<tr>
<th>#</th>
<th>Predictive Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Sale_Price</td>
<td>(-91950.000 ~ 91950.000)</td>
</tr>
<tr>
<td>2</td>
<td>Channel</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Cost_of_Goods_Sold</td>
<td>(-85900.000 ~ -85900.000)</td>
</tr>
<tr>
<td>4</td>
<td>Advertising_Cost</td>
<td>(0.000 ~ 1715.000)</td>
</tr>
<tr>
<td>5</td>
<td>Average_Sales_Area</td>
<td>(1130.000 ~ 4230.000)</td>
</tr>
</tbody>
</table>

Profit:
- Red: -365.00 ~ 480.00
- Green: 480.00 ~ 805.00
- Yellow: 805.00 ~ 1000.00
- Pink: 1260.00 ~ 6005.00

Channels:
- Camping Chain
- GO Outlet
- Independent

Mass Marketer

Sports Chain
Chapter 7. Classification and Prediction

- What is classification? What is prediction?
- Issues regarding classification and prediction
- Classification by decision tree induction
- Bayesian Classification
- Classification by backpropagation
- Classification based on concepts from association rule mining
- Other Classification Methods
- Prediction
  - Classification accuracy
- Summary
Classification Accuracy: Estimating Error Rates

- **Partition: Training-and-testing**
  - Use two independent data sets, e.g., training set (2/3), test set (1/3)
  - Used for data set with large number of samples
- **Cross-validation**
  - Divide the data set into $k$ subsamples
  - Use $k-1$ subsamples as training data and one sub-sample as test data --- $k$-fold cross-validation
  - For data set with moderate size
- **Bootstrapping and jackknife (leave-one-out)**
  - For small size data
Boosting and Bagging

• Boosting increases classification accuracy
  – Applicable to decision trees or Bayesian classifier

• Learn a series of classifiers, where each classifier in the series pays more attention to the examples misclassified by its predecessor

• Boosting requires only linear time and constant space
Boosting Technique (II) — Algorithm

- Assign every example an equal weight \( \frac{1}{N} \)
- For \( t = 1, 2, \ldots, T \) Do
  - Obtain a hypothesis (classifier) \( h^{(t)} \) under \( w^{(t)} \)
  - Calculate the error of \( h(t) \) and re-weight the examples based on the error
  - Normalize \( w^{(t+1)} \) to sum to 1
- Output a weighted sum of all the hypothesis, with each hypothesis weighted according to its accuracy on the training set
Chapter 7. Classification and Prediction

• What is classification? What is prediction?
• Issues regarding classification and prediction
• Classification by decision tree induction
• Bayesian Classification
• Classification by backpropagation
• Classification based on concepts from association rule mining
• Other Classification Methods
• Prediction
• Classification accuracy
• Summary
Classification is an extensively studied problem (mainly in statistics, machine learning & neural networks).

Classification is probably one of the most widely used data mining techniques with a lot of extensions.

Scalability is still an important issue for database applications: thus combining classification with database techniques should be a promising topic.

Research directions: classification of non-relational data, e.g., text, spatial, multimedia, etc..
References (I)

References (II)