CS522 Advanced Database Systems
Classification

Chengyu Sun
California State University, Los Angeles

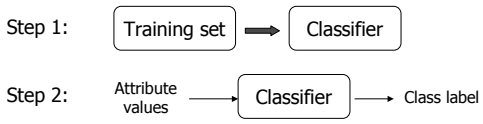
A Classification Problem

- Is a loan to a person who is 45 years old, divorced, renting an apartment, with two kids and annual income of 100K high risk or low risk?

Terminology and Concepts ...

- Record (or tuple)
  - Attributes
    - E.g. age, marital status, # of kids, owns home or not, credit score ...
  - Class label
    - E.g. high risk, low risk ...
- Classification: predict the class label with given attribute values

... Terminology and Concepts



- Classifier (or model)
- Training set: records with known class labels that are used to construct (i.e. *train*) the classifier

Classification vs. Regression

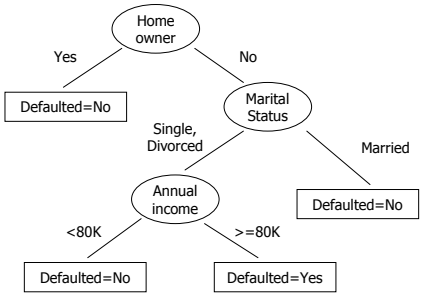
- Classification predicts categorical attribute values
- Regression predicts *continuous* numerical attribute values

SID	HW1	HW2	HW3	Final	Pass/Fail
1	40	60	70	95	Passed
2	10	15	11	65	Failed
3	30	45	40	75	Passed
4	35	50	35	?	?

A Training Set

TID	Home Owner	Marital Status	Annual Income	Defaulted Borrower
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

A Decision Tree



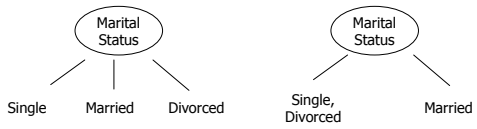
Decision Tree Induction

- Let D be the set of training record associated with current node
  - If all record in D belong to the same class C, current node is a leaf node and is labeled as C.
  - If D contains records that belong to more than one class, select an attribute to split D into subsets, and create a child node for each subset. *Apply the algorithm recursively on each child node.*

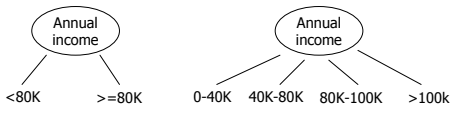
Decision Tree Induction Example

- Training set ➔ Decision tree

Split on Attributes with Discrete Values



Split on Attributes with Continuous Values



Terminating Conditions

- All records in D belong to the same class
- No more attribute to split
  - *Class label??*
- No records associated with the node
  - *Class label??*

Splitting Attribute Selection

- After a split, ideally each subset would "pure", i.e. contains only one class of records

Gender	Age	Preferred color
female	20	pink
male	20	black
female	15	pink
male	15	black

Attribute Selection Measures

- Entropy (Information Gain)
- Gini index
- Gain Ratio

Entropy

$$Entropy(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

- p_i is the fraction of records in `D` that belongs to class C_i
- `m` is the number of classes in `D`

Entropy Example

- Preferred color
  - 2 black and 2 pink??
  - 3 black and 1 pink??
  - 4 black??

Information Gain

- Suppose `D` is split into `v` subsets on attribute `A`

$$Gain(A) = Entropy(D) - \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Entropy(D_j)$$

Information Gain Example

- Preferred color
  - Gain(Gender)??
  - Gain(Age)??

Split Information

- Information gain favors attributes with lots of distinct values
- *Split information* can be used to "normalized" information gain

$$SplitInfo(A) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right)$$

Gain Ratio

$$GainRatio(A) = \frac{Gain(A)}{SplitInfo(A)}$$

Gini Index

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2$$

- Used in the CART algorithm for *binary split*

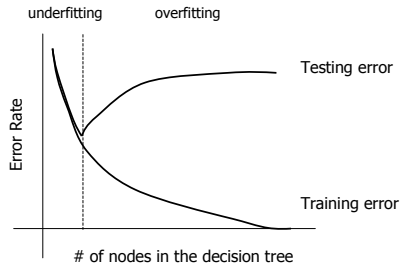
Gini Index Example

- Preferred color
  - 2 black and 2 pink??
  - 3 black and 1 pink??
  - 4 black??
  - Split on gender??
  - Split on age??

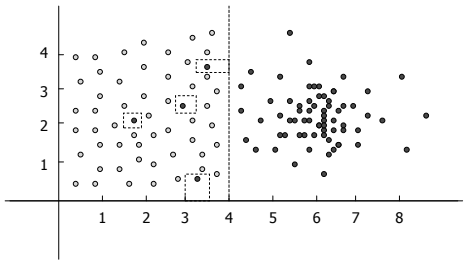
Training Error and Testing Error

- Training error
  - Misclassification of training records
- Testing (Generalization) error
  - Misclassification of testing records

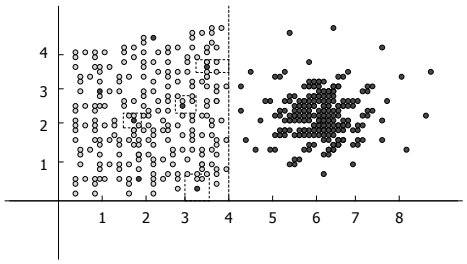
Model Overfitting and Underfitting



Overfitting Due to Outliers ...

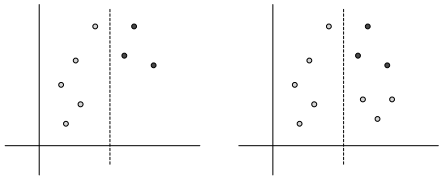


... Overfitting Due to Outliers



Other Reasons That Cause Overfitting

- Noise (mislabeled training records)
- Lack of representative samples



Tree Pruning – Prepruning

- Prune during decision tree construction
  - Number of records < threshold
  - "Purity gain" < threshold

Tree Pruning – Postpruning

- Buttom-up pruning of a fully constructed tree
  - Replace a subtree with a leaf node if it reduces testing error
    - *How do we know whether it reduces testing error or not??*
  - Pruning based on Minimum Description Length (MDL)

Estimate Testing Errors

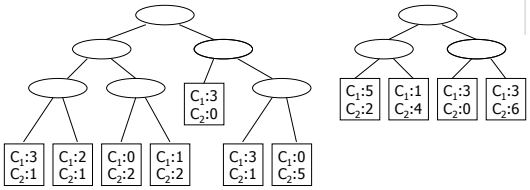
- Use a *pruning set* in addition to the training set
- Optimistic error estimation
  - The training set is a good representation of the overall data (optimistic!), so the training error is the testing error
- Pessimistic error estimation
  - Training error + penalty term for model complexity

Pessimistic Error Estimation

$$e_g(T) = \frac{\sum_{i=1}^{k} [e(t_i) + \Omega(t_i)]}{N_t}$$

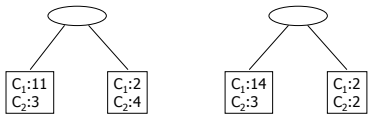
- N_t – Total number of records
- $e(t_i)$ – Training error at leaf node t_i
- $\Omega(t_i)$ – Penalty term associated with t_i

Example of Pessimistic Error Estimation



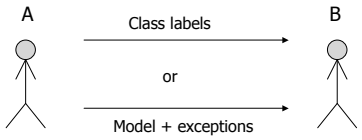
- $e_g(T)$ with $\Omega(t_i)=0.5$?? $\Omega(t_i)=1$??

Pruning with Estimated Testing Error



- With optimistic error estimation??
- With pessimistic error estimation??
- With a pruning set??

Minimum Description Length (MDL)

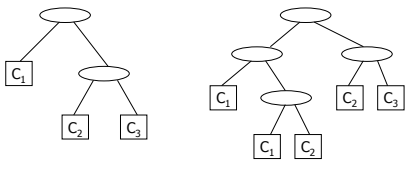


- The best model is the one that minimizes the number of bits to encode both the model and the exceptions to the model

MDL Example ...

- n records
- m binary attributes
- k classes
- Cost(Internal Node) = $\log_2 m$
- Cost(Leaf node) = $\log_2 k$
- Cost(Error) = $\log_2 n$
- Cost = Cost(All Nodes)+Cost(All Errors)

... MDL Example



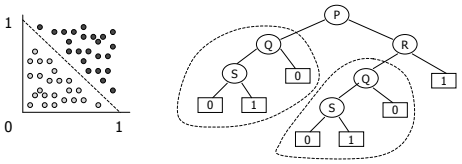
- 16 binary attributes and 3 classes
- Left tree has 7 errors and right tree has 4 errors

About Decision Tree Classification ...

- Inexpensive to construct
- Extremely fast at classifying unknown records
- Easy to interpret for small-sized trees
- Accuracy is comparable to other classification techniques for many simple data sets

... About Decision Tree Classification

- Data fragmentation
- Finding an optimal decision tree is NP-hard
- Limitation on expressiveness
- Tree replication



Bayes' Theorem

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

- Prior and posterior probabilities
  - P(A) and P(A|B)
  - P(B) and P(B|A)

Bayesian Classification

$$P(C_i \mid \mathbf{X}) = \frac{P(\mathbf{X} \mid C_i)P(C_i)}{P(\mathbf{X})}$$

- **X** is a given record with attribute values $(x_1, x_2, \ldots, x_n)$, and C_i is a class
- $P(C_i \mid \mathbf{X})$ is the probability of **X** belonging to class C_i given **X**'s attribute values
- We predict that **X** belong to C_i if $P(C_i \mid X) > P(C_j \mid X)$ for $j \neq i$

Calculate $P(C_i|\mathbf{X})$

- P(X) does not need to be calculated
  - *Why??*
- $P(C_i)$??
- $P(\mathbf{X}|C_i)$??

Naive Bayesian Classification

- $\mathbf{X}=(x_1,x_2,\ldots,x_n)$
- Assume the attribute values are conditionally independent of one another (the *naive* assumption)

$$P(\mathbf{X} \mid C_i) = \prod_{i=1}^{n} P(x_i \mid C_i)$$
$$= P(x_1 \mid C_i) \times P(x_2 \mid C_i) \times \cdots \times P(x_n \mid C_i)$$

Attribute A_k is Categorical

- $P(x_k | C_i)$ is the fraction of number of records in C_i with value x_k for attribute A_k

Attribute A_k is Continuous-valued ...

- Assume A_k follows a Gaussian distribution with a mean μ and standard deviation σ

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$\sigma = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu)^2}$$

... Attribute A_k is Continuous-valued

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$P(x_k | C_i) = g(x_k, \mu_{c_i}, \sigma_{c_i})$$

Sample Data

TID	Home Owner	Marital Status	Annual Income	Defaulted Borrower
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
11	No	Married	120K	??

Naive Bayesian Classification Example ...

- P(No|HO=No,MS=M,AI=120K) vs. P(Yes|HO=No,MS=M,AI=120K)

... Naive Bayesian Classification Example

- Annual Income, Default=No
  - μ=110, σ=54.54
  - P(AI=120K|No)=0.0072
- Annual Income, Default=Yes
  - μ=90, σ=5
  - P(AI=120K|Yes)=1.2×10^{-9}

Avoid Zero $P(x_k|C_i)$

- A zero $P(x_k|C_i)$ would make the whole $P(\mathbf{X}|C_i)$ zero
- To avoid this problem, add 1 to to each count, assuming the training set is sufficiently large that the effect of adding one is negligible
- Example
  - Low income:0
  - Medium income: 990
  - High income: 10

About Naive Bayesian Classification

- The most accurate classification *if the conditional independence assumption holds*
- In practice, some attributes may be correlated
  - E.g. education level and annual income

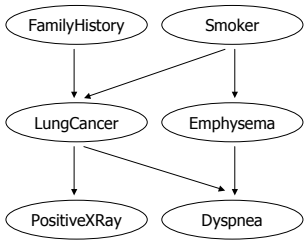
Bayesian Belief Network (BBN)

- A *directed acyclic graph* (dag) encoding the dependencies among a set of variables
- A *conditional probability table* (CPT) for each node given its immediate parent nodes

A BBN Example

CPT for LungCancer

	Yes	No
FH,S	0.8	0.2
FH,!S	0.5	0.5
!FH,S	0.7	0.3
!FH,!S	0.1	0.9



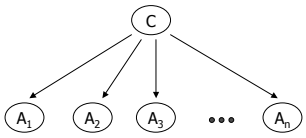
BBN Terminology

- If there is a directed *arc* from X to Y
  - X is a parent of Y
  - Y is a child of X
- If there is a directed *path* from X to Y
  - X is an ancestor of Y
  - Y is a descendent of X

Conditional Independence in BBN

- A node in a Bayesian network is conditionally independent of its non-descendants if its parents are known

Naive Bayesian is a Special Case of BBN



Construct a BBN

- Create the structure of the network
  - From domain knowledge
  - From training data
- Calculate the CPT for each node
  - All variables are observable
  - Some variable may be *hidden*

Another BBN Example

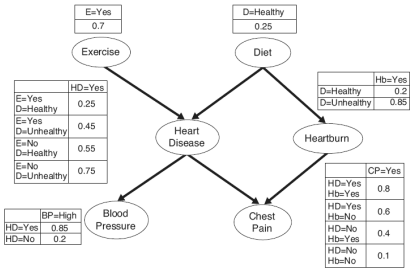


Figure 5.13. A Bayesian belief network for detecting heart disease and heartburn in patients.

Bayesian Classification Examples

- Output node – Heart Disease
- Testing data
  - ()
  - (BP=high)
  - (BP=high,D=Healthy,E=Yes)

Bayesian Classification Examples – 1

$$P(HD=Yes)=\sum_{i=1}^{n}\sum_{j=1}^{m}P(HD=Yes\mid E=a_i,D=b_j)P(E=a_i,D=b_j)$$

$$=\sum_{i=1}^{n}\sum_{j=1}^{m}P(HD=Yes\mid E=a_i,D=b_j)P(E=a_i)P(D=b_j)$$

$$=0.49$$

Bayesian Classification Examples – 2

$$P(HD=Yes\mid BP=High)$$

$$=\frac{P(BP=High\mid HD=Yes)P(HD=Yes)}{P(BP=High)}$$

$$=\frac{P(BP=High\mid HD=Yes)P(HD=Yes)}{\sum_{i=1}^{n}P(BP=High\mid HD=a_i)P(HD=a_i)}$$

$$=0.80$$

Bayesian Classification Examples – 3

$$P(HD = Yes \mid BP = High, D = Healthy, E = Yes)$$
$$= \frac{P(BP = High \mid HD = Yes, D = Healthy, E = Yes)P(HD = Yes \mid D = Healthy, E = Yes)}{P(BP = High \mid D = Healthy, E = Yes)}$$
$$= \frac{P(BP = High \mid HD = Yes)P(HD = Yes \mid D = Healthy, E = Yes)}{\sum_{i=1}^{n} P(BP = High \mid HD = a_i)P(HD = a_i \mid D = Healthy, E = Yes)}$$
$$= 0.59$$

Other Classification Methods

- Rule-based
- Artificial Neural Network (ANN)
- Support Vector Machine (SVM)
- Association rule analysis
- Nearest neighbor
- Genetic algorithms
- Rough Set and Fuzzy Set thoery
- …

Ensemble Methods

- Use a number of *base* classifiers, and make a predication by combining the predications of all the classifiers
- Example
  - Binary classification
  - 25 classifiers, each with error rate 35%
  - Predict by majority vote
  - *Error rate of the ensemble classifier??*

Construct an Ensemble Classifier

- Train k classifiers with one dataset
  - Use the same dataset for each classifier??
  - Divide the dataset into k subsets??
  - Bagging and Boosting

Bagging (Bootstrap Aggregation)

- A *bootstrap* sampling of |D|
  - Uniform sampling with replacement (vs. without replacement)
  - Allow duplicates in the sample
  - |D| samples
    - Roughly contains 63.2% of the original records. *Why??*
- Bagging
  - Use a bootstrap sample as the training set for each classifier

Bagging Example

- Record (x, y)
  - x: attribute
  - y: class label
- Base classifier: decision tree with one level $x \leq k$
- Ensemble classifier: 10 classifiers, majority vote

Bagging Example – Dataset

x	y
0.1	1
0.2	1
0.3	1
0.4	-1
0.5	-1
0.6	-1
0.7	-1
0.8	1
0.9	1
1.0	1

Bagging Example – Bagging

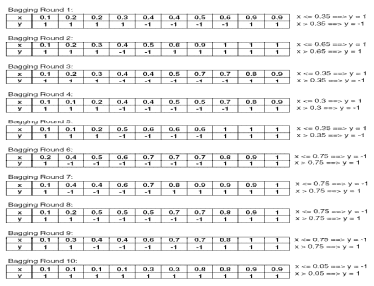


Figure 5.35. Example of bagging.

Bagging Example – Classification

Round	x=0.1	x=0.2	x=0.3	x=0.4	x=0.5	x=0.6	x=0.7	x=0.8	x=0.9	x=1.0
1	1	1	1	-1	-1	-1	-1	-1	-1	-1
2	1	1	1	1	1	1	1	1	1	1
3	1	1	1	-1	-1	-1	-1	-1	-1	-1
4	1	1	1	-1	-1	-1	-1	-1	-1	-1
5	1	1	1	-1	-1	-1	-1	-1	-1	-1
6	-1	-1	-1	-1	-1	-1	-1	1	1	1
7	-1	-1	-1	-1	-1	-1	-1	1	1	1
8	-1	-1	-1	-1	-1	-1	-1	1	1	1
9	-1	-1	-1	-1	-1	-1	-1	1	1	1
10	1	1	1	1	1	1	1	1	1	1
Sum	2	2	2	-6	-6	-6	-6	2	2	2
Sign	1	1	1	-1	-1	-1	-1	1	1	1
True Class	1	1	1	-1	-1	-1	-1	1	1	1

Figure 5.36. Example of combining classifiers constructed using the bagging approach.

About Bagging

- Reduces the errors associated with random fluctuations in the training data for *unstable classifiers*, e.g. decision trees, rule-based classifiers, ANN
- May degrade the performance of *stable classifiers*, e.g. Bayesian network, SVM, k-NN

Intuition for Boosting

- Sample with weights
  - hard-to-classify records should be chosen more often
- Combine the prediction of the base classifiers with weights
  - Classifiers with lower error rates get more voting power

Boosting – Training

- For k classifiers, do k rounds of
  - Assign a weight to each record
  - Sample with replacement according to the weights
  - Train a classifier M_i
  - Calculate $error(M_i)$
  - Update the weights of the records
    - Increase the weights of the misclassified records
    - Decrease the weights of the correctly classified records

Boosting – Classification

- For each class, sum up the weights of the classifiers that vote for that class
- The class that gets the highest sum is the predicted class

Boosting Implementation

- How the record weights are updated
- How the classifier weights are calculated

Adaboost

Error rate of classifier M_i: $error(M_i) = \sum w_j \times err(\mathbf{X_j})$

Update the weights of the correctly classified records: $w_j \times \frac{error(M_i)}{1-error(M_i)}$

Weight of classifier M_i: $\log \frac{1-error(M_i)}{error(M_i)}$

Evaluate the Accuracy of a Classifier

- Accuracy measures
  - Accuracy rate and error rate
  - Confusion matrix
  - Precision and Recall (for binary classification)

Example of Accuracy Measures

- Example
  - Two classes C_1 and C_2
  - 100 testing records with 50 C_1 records and 50 C_2 records
  - 20 C_1 records misclassified as C_2, and 10 C_2 records misclassified as C_1
- Accuracy measures
  - *Accuracy and error rates??*
  - *Confusion matrix??*
  - *Precision and Recall??*

Evaluate the Accuracy of a Classifier

- The Holdout Method
  - Given a set of records with known class labels, use half of them for training and the other half for testing (or 2/3 for training and 1/3 for testing)

Problems of the Holdout Method

- More records for training means less for testing, and vice versa
- Distribution of the data in the training/testing set may be different from the original dataset
- Some classifiers are sensitive to random fluctuations in the training data

Random Subsampling

- Repeat the holdout method k times
- Take the average accuracy over the k iterations
- Random subsampling methods
  - Cross-validation
  - Bootstrap

K-fold Cross-validation

- Divide the original dataset into k non-overlapping subsets
- Each iteration uses (k-1) subsets for training, and the remaining subset for testing
- Total errors are the sum of the errors in each iteration

Bootstrap (.632 Bootstrap)

- Each iteration uses a bootstrap sample to train the classifier, and the remaining records for testing
- Calculate the overall accuracy:

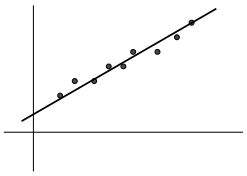
$$\frac{1}{k}\sum_{i=1}^{k}(0.632\times Acc(M_i)_{test\_set}+0.368\times Acc(M_i)_{all\_records})$$

Predicating Continuous Values

- Regression methods
  - Linear regression
  - Non-linear regression
- Other methods
  - Some classification methods can be adapted to predict continuous values

Linear Regression

- Record (x, y)
  - x: predictor variable
  - y: response variable
- Model
  - $y = w_0 + w_1 x$



Linear Regression Using Least-Squares Method

$$w_1 = \frac{\sum_{i=1}^{|D|}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{|D|}(x_i - \bar{x})^2}$$

$$w_0 = \bar{y} - w_1 \bar{x}$$

Multiple Linear Regression

- Record ($x_1, \ldots, x_n, y$)
- Model:

$$y = w_0 + \sum_{i=1}^{n} w_i x_i$$

Summary

- Classification
  - Problem definition and terminology
  - Decision Tree Induction
    - Information gain and gain ratio
    - Model overfitting
    - Tree pruning
  - Naive Bayesian classification and BBN
  - Ensemble methods
  - Evaluation of classification accuracy
- Linear regression