Chapter 6: Classification

Dong-Kyu Chae

PI of the Data Intelligence Lab @HYU
Department of Computer Science & Data Science
Hanyang University

Data
Intelligence
LAB

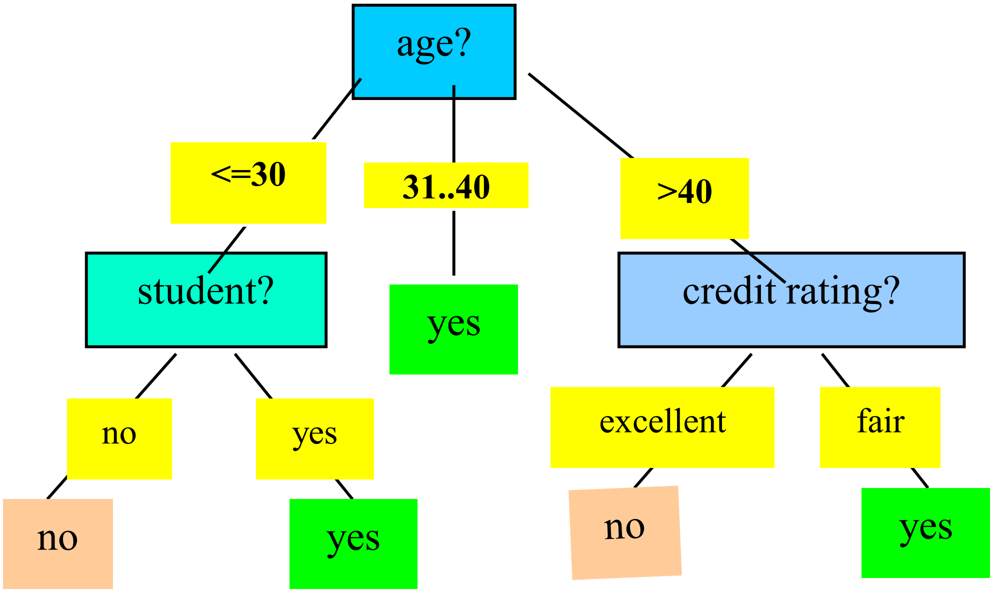
Topics

- What is classification?
- Issues regarding classification
- Classification by decision tree induction
- Random Forest
- Rule-based classification
- Associative classification
- Lazy learners (or learning from your neighbors)
- Accuracy and error measures
- Ensemble methods
- Summary

Rule-based Classification

- **Basic idea: using IF-THEN rules**
  - Rule (**R**) example: **IF** *age* = youth **AND** *student* = no
    THEN *buys_computer* = no
- **It is employed when there is little amount of (or no) data**
  - Rules can be made by domain experts
- **If more than one rule is triggered, need conflict resolution**
  - **Size ordering**: assign the highest priority to the triggering rules that have the "toughest" requirement (i.e., with the *most features to test*)
  - **Class-based ordering**: decreasing order of misclassification cost
  - **Rule-based ordering** (decision list): rules are organized into one long priority list
    - According to some measure of rule quality or by domain experts

Rule Extraction from a Decision Tree



IF *age* <=30 **AND** *student* = *no*,
THEN *buys_computer* = *no*

IF *age* <=30 **AND** *student* = *yes*,
THEN *buys_computer* = *yes*

IF *age* is in 31~40,
THEN *buys_computer* = *yes*

.....

- Rules are easier to understand than a large tree
- One **rule** is created for each path **from the root to a leaf**
- Each feature-value pair along a path forms a conjunction, the leaf holds the class prediction
- Rules are mutually exclusive (no conflict)

Rule Extraction from Association Rule Mining

- **Also known as associative classification** : rule based classification,
  - rules can be generated based on the frequent pattern mining
  - Association rules are generated and analyzed for use in classification

feature name feature value

$$P_1 \wedge p_2 \ldots \wedge p_l \rightarrow \text{"}A_{class} = C\text{"} \quad (conf, sup)$$

  - By controlling min_sup. and min_conf., we can search for strong associations between conjunctions of feature-value pairs (the condition part) and the class label
  - Rules are not mutually exclusive: need conflict management
- **Benefits and limits**
  - It explores highly confident associations with considering multiple attributes, by setting higher min_conf.
  - May have **low coverage** w.r.t. the values for min_conf. and min_sup.

Rule Extraction from Association Rule Mining

❑ Coverage and accuracy of a rule R

- ❑ n_{covers} = # of data *covered* by R
- ❑ $n_{correct}$ = # of data *correctly classified* by R

coverage(R) = n_{covers} /|D| /* D: training data set */

accuracy(R) = $n_{correct}$ / n_{covers}

measure in training phase

Accuracy

Coverage

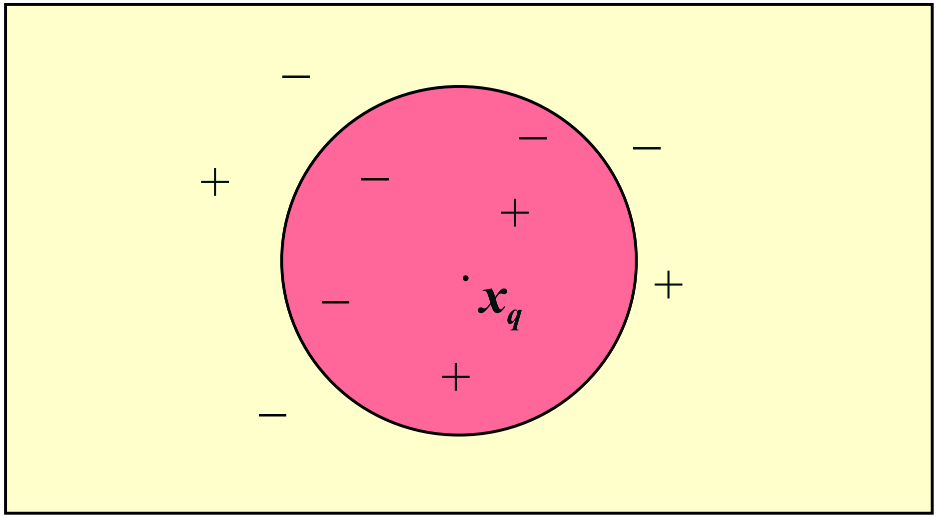
Lazy vs. Eager Learning

- **Lazy vs. Eager learning**
  - Eager learning (most machine learning methods)
    - Given a set of training set, constructs a classification model *before receiving a* new test data to classify
  - Lazy learning
    - Simply stores training data (or only minor processing) and *just waits* until a test data is given
- **Lazy: much less time in training but more time in predicting**

The *k*-Nearest Neighbor (KNN) Algorithm

(lazy learning)

- *k*-nearest neighbors are retrieved in terms of the distance
  - All data points are expressed in an *n*-dimensional space
  - Distance, **dist**($\mathbf{X}_1$, $\mathbf{X}_2$), is defined over the space (like **Euclidean**)



- The test sample is classified by the class label of a **majority** of the *k*-nearest neighbors (i.e., voting)
  - Or, weighted voting can be adopted depending on their distance

Evaluation of a Classification Model

Accuracy evaluation

- Goal: to evaluate the accuracy of the model by using a test data
- Test data
  - A set of data / data points / tuples / samples used for accuracy evaluation
  - Each data: <feat-1, feat-2, …., feat-n, class label> (attribute / feature)
  - Each data has a predefined class
- The known label of a test sample is compared with the classified result from the model
- Accuracy = # of correctly classified / # of entire test set data
- The test set must be independent of the training set!

Evaluation of a Classification Model

Confusion matrix

$CM_{i,j}$, an entry, indicates # of data in class i that are predicted by the classifier as class j

Classified

	classes	C = yes	C = no
Ground truth	C = yes	True positive	False negative
	C = no	False positive	True negative

classes	Covid = yes	Covid = no	total	recognition(%)
Covid = yes	2588	412	3000	86.27
Covid = no	46	6954	7000	99.34
total	2634	7366	10,000	95.52

- **Accuracy of a classifier: percentage of ~~tuples~~ data (in a test set) that are correctly classified by the model (=(2588+6954)/10,000)**
  - Error rate (misclassification rate) = (1.00 – accuracy)

Evaluation of a Classification Model

Classified

Ground truth

classes	Covid = yes	Covid = no	total	recognition(%)
Covid = yes	2588	412	3000	86.27
Covid = no	46	6954	7000	99.34
total	2634	7366	10000	95.52

❑ Alternative accuracy measures

sensitivity = true-positive / positive (*recall*) /* true positive recognition rate */

specificity = true-negative / negative /* true negative recognition rate */

precision = true-positive / (true-positive + false-positive)

❑ Precision and recall are dependent on each other; they are complementary. We thus use F1-score:

$$2 \times \frac{recall \times precision}{recall + precision}$$

Evaluation Protocols

❑ Holdout method

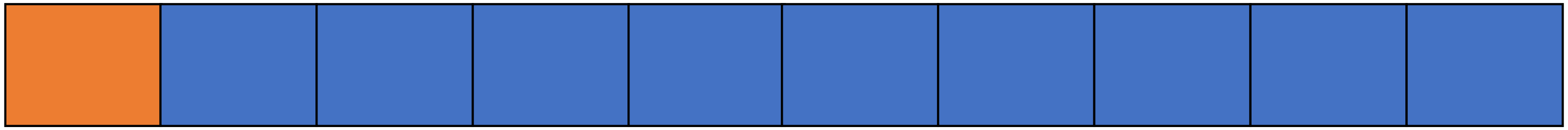
Split data

- ❑ Given data is randomly partitioned into two independent sets
  - **Training set** (e.g., 4/5) for model construction
  - **Test set** (e.g., 1/5) for accuracy estimation
- ❑ Repeat holdout *k* times
  - Just one time is not sufficient
  - Accuracy = avg. of the k accuracies obtained

Evaluation Protocols

Cross-validation (or, *k*-fold cross validation)

- Randomly partition the data into ***k* subsets**, each having approximately equal size
  - k is typically chosen as 5 or 10
- At *i*-th iteration, use D_i as a test set and others as a training set



- Accuracy = avg. of the k accuracies obtained

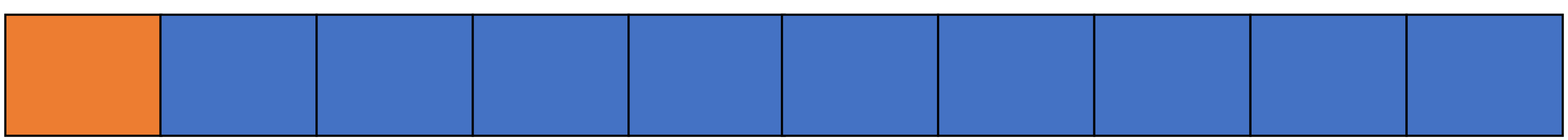
Evaluation Protocols

Leave-one-out:

- Extreme case of the *k*-fold cross-validation, for small sized data
- *k* folds where *k* = # of data points!

Stratified cross-validation

- Another special case of the *k*-fold cross-validation
- It aims to maintain the class distribution
  - Folds are stratified so that class distribution in each fold is approximately the same as that in the original data



Ensemble

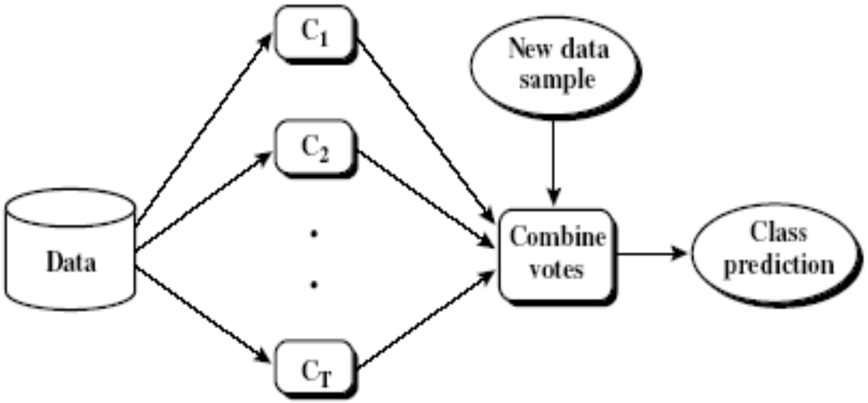
Ensemble methods

- Use a combination of models to **increase accuracy**
- Combine a series of *k* learned models, C_1, C_2, …, C_k, with the aim of aggregating multiple opinions

Popular ensemble methods

Single type of classifier

- Bagging: simply averaging the prediction over a collection of classifiers
  → Majority Voting
- Boosting: weight is considered within a collection of classifiers
- Model ensemble: combining a set of heterogeneous classifiers
  - E.g., SVM + Decision tree + Neural network + etc.…



Ensemble

ex) Random Forest
→ aggregate the opinions of all the decision trees by majority voting → Bag
or considering weight → Boosting

- **Bagging: Bootstrap Aggregation**
- **Training**
  - At each iteration *i*, a training set D_i is sampled with replacement from the original D (i.e., bootstrap)
  - A classifier model M_i is learned for each training set D_i
- **Classification: classify an unknown sample X**
  - Each classifier M_i returns its class prediction
  - The bagged classifier M* counts the votes and assigns the class with the most votes to **X**

Ensemble

Boosting

- Weight is considered based on each model's accuracy

Process

- Initial **weights (1/d)** are assigned to each data point
  - Weights are also used in sampling for building i-th training set (using the bootstrap sampling)
- In each iteration from 1 to k, when each classifier M_i is learned, examine it to the i-th test set (composed of non-sampled data)
- Then, the weights are **updated** to allow the subsequent classifier, M_{i+1}, to pay more attention (in sampling) to the data points that were misclassified by M_i → 다음 boostrap sample 에 뽑힐 확률 높아짐
- The final result is obtained from votes of each individual classifier, where the weight of each classifier's vote is equal to its accuracy

Summary

- **What is classification?**
- **Decision tree induction**
  - Random Forest
- **Rule-based classification**
  - Associative classification
- **Lazy learners (*k*-NN classifiers)**
- **Accuracy and error measures, evaluation protocols**
- **Ensemble**

Thank You

Data
Intelligence
Lab