

COMP 345: Data Mining

More Classification Basics

Slides Adapted From : Jiawei Han, Micheline Kamber & Jian Pei
Data Mining: Concepts and Techniques, 3rd ed.



Reminders

- Assignment 4 has been assigned
 - details on Course Website
 - Due Mon. Oct. 1st /Tues. Oct. 2nd at beginning of class
- Group Project Proposal
 - Due Mon. Oct. 1st /Tues. Oct. 2nd at beginning of class
- Midterm Exam
 - Wed. Oct. 3rd /Thurs. Oct. 4th
 - You may bring a 1 page (front & back) 8.5" x 11" sheet of paper with whatever you want on it
 - You may also bring a calculator

Classifier Evaluation Metrics: Example

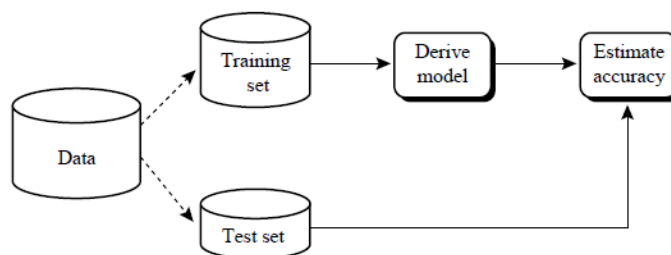
Actual Class\Predicted class	cancer = yes	cancer = no	Total	Recognition(%)
cancer = yes	90	210	300	30.00 (<i>sensitivity</i>)
cancer = no	140	9560	9700	98.56 (<i>specificity</i>)
Total	230	9770	10000	96.50 (<i>accuracy</i>)

- $Precision = 90/230 = 39.13\%$ $Recall = 90/300 = 30.00\%$

3

Evaluating Classifier Accuracy: Holdout

- **Holdout method**
 - Given data is randomly partitioned into two independent sets
 - Training set (e.g., 2/3) for model construction
 - Test set (e.g., 1/3) for accuracy estimation
 - Random subsampling: a variation of holdout
 - Repeat holdout k times, accuracy = avg. of the accuracies obtained from each iteration



4

Evaluating Classifier Accuracy: Cross-Validation Methods

- **Cross-validation** (k -fold, where $k = 10$ is most popular)
 - Randomly partition the data into k *mutually exclusive* subsets, each approximately equal size
 - At i -th iteration, use D_i as test set and others as training set
 - Leave-one-out: k folds where $k = \#$ of tuples, for small sized data
 - ***Stratified cross-validation*** = *Recommended*
 - folds are stratified so that class distribution in each fold is approximately the same as that in the initial data

5

Evaluating Classifier Accuracy: Bootstrap

- **Bootstrap**
 - Works well with small data sets
 - Samples the given training tuples uniformly *with replacement*
 - i.e., each time a tuple is selected, it is equally likely to be selected again and re-added to the training set
- Several bootstrap methods, and a common one is **.632 bootstrap**
 - A data set with d tuples is sampled d times, with replacement, resulting in a training set of d samples. The data tuples that did not make it into the training set end up forming the test set. About 63.2% of the original data end up in the bootstrap, and the remaining 36.8% form the test set (since $(1 - 1/d)^d \approx e^{-1} = 0.368$)
 - Repeat the sampling procedure k times, overall accuracy of the model:

$$Acc(M) = \frac{1}{k} \sum_{i=1}^k (0.632 \times Acc(M_i)_{test_set} + 0.368 \times Acc(M_i)_{train_set})$$

6

Estimating Confidence Intervals: Classifier Models M_1 vs. M_2

- Suppose we have 2 classifiers, M_1 and M_2 , which one is better?
- Use 10-fold cross-validation to obtain $\overline{err}(M_1)$ and $\overline{err}(M_2)$
- These mean error rates are just *estimates* of error on the true population of *future* data cases
- What if the difference between the 2 error rates is just attributed to *chance*?
 - Use a **test of statistical significance**
 - Obtain **confidence limits** for our error estimates

7

Estimating Confidence Intervals: Null Hypothesis

- Perform 10-fold cross-validation
- Assume samples follow a **t distribution** with $k-1$ **degrees of freedom** (here, $k=10$)
- Use **t-test** (or **Student's t-test**)
- **Null Hypothesis**: M_1 & M_2 are the same
- If we can **reject** null hypothesis, then
 - we conclude that the difference between M_1 & M_2 is **statistically significant**
 - Chose model with lower error rate

8

Estimating Confidence Intervals: t-test

- If only 1 test set available: **pairwise comparison**
 - For i^{th} round of 10-fold cross-validation, the same cross partitioning is used to obtain $err(M_1)_i$ and $err(M_2)_i$
 - Average over 10 rounds to get $\overline{err}(M_1)$ and $\overline{err}(M_2)$
- t-test** computes **t-statistic** with $k-1$ **degrees of freedom**:

$$t = \frac{\overline{err}(M_1) - \overline{err}(M_2)}{\sqrt{var(M_1 - M_2)/k}} \quad \text{where}$$

$$var(M_1 - M_2) = \frac{1}{k} \sum_{i=1}^k [err(M_1)_i - err(M_2)_i - (\overline{err}(M_1) - \overline{err}(M_2))]^2$$

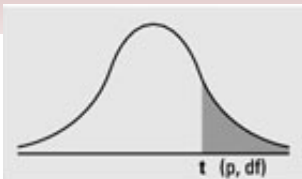
- If two test sets available: use **non-paired t-test**

$$\text{where } var(M_1 - M_2) = \sqrt{\frac{var(M_1)}{k_1} + \frac{var(M_2)}{k_2}},$$

where k_1 & k_2 are # of cross-validation samples used for M_1 & M_2 resp.

9

Estimating Confidence Intervals: Table for t-distribution



- Symmetric
- Significance level**, e.g., $sig = 0.05$ or 5% means M_1 & M_2 are *significantly different* for 95% of population
- Confidence limit**
 $z = sig/2$

Table for t-distribution

df/p	0.40	0.25	0.10	0.05	0.025	0.01	0.005	0.0005
1	0.324920	1.000000	3.077684	6.313752	12.70620	31.82052	63.65674	636.6192
2	0.288675	0.816497	1.885618	2.919986	4.30265	6.96456	9.92484	31.5991
3	0.276671	0.764892	1.637744	2.353363	3.18245	4.54070	5.84091	12.9240
4	0.270722	0.740697	1.533206	2.131847	2.77645	3.74695	4.60409	8.6103
5	0.267181	0.726687	1.475884	2.015048	2.57058	3.36493	4.03214	6.8688
6	0.264835	0.717558	1.439756	1.943180	2.44691	3.14267	3.70743	5.9588
7	0.263167	0.711142	1.414924	1.894579	2.36462	2.99795	3.49948	5.4079
8	0.261921	0.706387	1.396815	1.859548	2.30600	2.89646	3.35539	5.0413
9	0.260955	0.702722	1.383029	1.833113	2.26216	2.82144	3.24984	4.7809
10	0.260185	0.699812	1.372184	1.812461	2.22814	2.76377	3.16927	4.5869
...								
30	0.255605	0.682756	1.310415	1.697261	2.04227	2.45726	2.75000	3.6480
z	0.253347	0.674490	1.281552	1.644854	1.95996	2.32635	2.57583	3.2905
CI	——	——	80%	90%	95%	98%	99%	99.9%

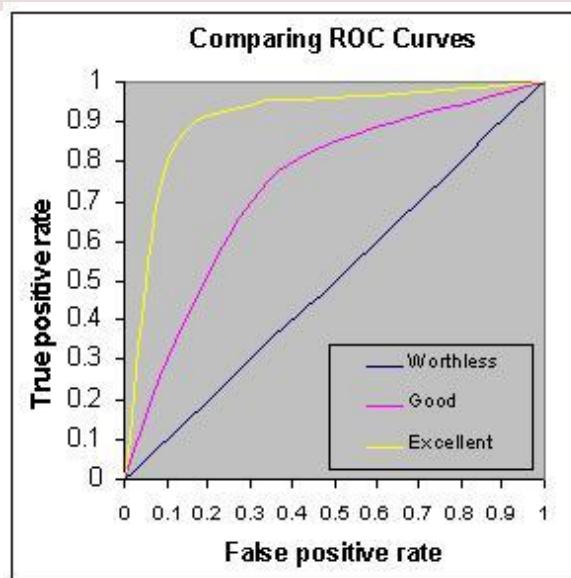
10

Estimating Confidence Intervals: Statistical Significance

- Are M_1 & M_2 **significantly different**?
 - Compute t . Select *significance level* (e.g. $\text{sig} = 5\%$)
 - Consult table for t -distribution: Find t value corresponding to $k-1$ *degrees of freedom* (here, 9)
 - t -distribution is symmetric: typically upper % points of distribution shown \rightarrow look up value for **confidence limit** $z = \text{sig}/2$ (here, 0.025)
 - If $t > z$ or $t < -z$, then t value lies in rejection region:
 - **Reject null hypothesis** that mean error rates of M_1 & M_2 are same
 - Conclude: statistically significant difference between M_1 & M_2
 - **Otherwise**, conclude that any difference is **chance**

11

Model Selection: ROC Curves (Receiver Operating Characteristics)



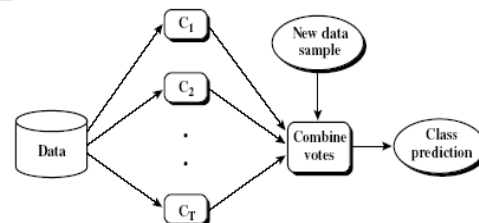
12

Issues Affecting Model Selection

- **Accuracy**
 - classifier accuracy: predicting class label
- **Speed**
 - time to construct the model (training time)
 - time to use the model (classification/prediction time)
- **Robustness**: handling noise and missing values
- **Scalability**: efficiency in disk-resident databases
- **Interpretability**
 - understanding and insight provided by the model
- Other measures, e.g., goodness of rules, such as decision tree size or compactness of classification rules

13

Ensemble Methods: Increasing the Accuracy



- Ensemble methods
 - Use a combination of models to increase accuracy
 - Combine a series of k learned models, M_1, M_2, \dots, M_k , with the aim of creating an improved model M^*
- Popular ensemble methods
 - Bagging: averaging the prediction over a collection of classifiers
 - Boosting: weighted vote with a collection of classifiers
 - Ensemble: combining a set of heterogeneous classifiers

14

Bagging: Bootstrap Aggregation

- Analogy: Diagnosis based on multiple doctors' majority vote
- Training
 - Given a set D of d tuples, at each iteration i , a training set D_i of d tuples is sampled with replacement from 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
- Prediction: can be applied to the prediction of continuous values by taking the average value of each prediction for a given test tuple
- Accuracy
 - Often significantly better than a single classifier derived from D
 - For noisy data: not considerably worse, more robust
 - Proved improved accuracy in prediction

15

Boosting

- Analogy: Consult several doctors, based on a combination of weighted diagnoses—weight assigned based on the previous diagnosis accuracy
- How boosting works?
 - **Weights** are assigned to each training tuple
 - A series of k classifiers is iteratively learned
 - After a classifier M_i is learned, the weights are updated to allow the subsequent classifier, M_{i+1} , to **pay more attention to the training tuples that were misclassified** by M_i
 - The final **M^* combines the votes** of each individual classifier, where the weight of each classifier's vote is a function of its accuracy
- Boosting algorithm can be extended for numeric prediction
- Comparing with bagging: Boosting tends to have greater accuracy, but it also risks overfitting the model to misclassified data

16

Random Forest (Breiman 2001)

- Random Forest:
 - Each classifier in the ensemble is a *decision tree* classifier and is generated using a random selection of attributes at each node to determine the split
 - During classification, each tree votes and the most popular class is returned
- Two Methods to construct Random Forest:
 - Forest-RI (*random input selection*): Randomly select, at each node, F attributes as candidates for the split at the node. The CART methodology is used to grow the trees to maximum size
 - Forest-RC (*random linear combinations*): Creates new attributes (or features) that are a linear combination of the existing attributes (reduces the correlation between individual classifiers)
- Comparable in accuracy to boosting, but more robust to errors and outliers
- Insensitive to the number of attributes selected for consideration at each split, and faster than bagging or boosting

17

Classification of Class-Imbalanced Data Sets

- Class-imbalance problem: Rare positive example but numerous negative ones, e.g., medical diagnosis, fraud, oil-spill, fault, etc.
- Traditional methods assume a balanced distribution of classes and equal error costs: not suitable for class-imbalanced data
- Typical methods for imbalance data in 2-class classification:
 - **Oversampling**: re-sampling of data from positive class
 - **Under-sampling**: randomly eliminate tuples from negative class
 - **Threshold-moving**: moves the decision threshold, t , so that the rare class tuples are easier to classify, and hence, less chance of costly false negative errors
- Still difficult for class imbalance problem on multiclass tasks

18

Summary (I)

- **Classification** is a form of data analysis that extracts **models** describing important data classes.
- Effective and scalable methods have been developed for **decision tree induction**, **Naive Bayesian classification**, **rule-based classification**, and many other classification methods.
- **Evaluation metrics** include: accuracy, sensitivity, specificity, precision, recall, F measure, and F_β measure.
- **Stratified k-fold cross-validation** is recommended for accuracy estimation. **Bagging** and **boosting** can be used to increase overall accuracy by learning and combining a series of individual models.

19

Summary (II)

- **Significance tests** and **ROC curves** are useful for model selection.
- There have been numerous **comparisons of the different classification** methods; the matter remains a research topic
- No single method has been found to be superior over all others for all data sets
- Issues such as accuracy, training time, robustness, scalability, and interpretability must be considered and can involve trade-offs, further complicating the quest for an overall superior method

20