Knowledge Discovery & Data Mining
— Classifier Evaluation, Model Selection —

Instructor: Yong Zhuang

yong.zhuang@gvsu.edu

Outline: Model Evaluation and Selection

- Evaluation metrics: How can we measure accuracy? Other metrics to consider?
- Use **validation set** of class-labeled tuples instead of training set when assessing accuracy
- Methods for estimating a classifier’s accuracy:
  - Holdout method, random subsampling
  - Cross-validation
  - Bootstrap
- Comparing classifiers:
  - Confidence intervals
  - Cost-benefit analysis and ROC Curves

Classifier Evaluation Metrics: Confusion Matrix

Confusion Matrix:

Actual class\Predicted class	C_1	$\neg C_1$
C_1	**True Positives (TP)**	**False Negatives (FN)**
$\neg C_1$	**False Positives (FP)**	**True Negatives (TN)**

Example of Confusion Matrix:

Actual class\Predicted class	buy_computer = yes	buy_computer = no	Total
buy_computer = yes	**6954**	**46**	7000
buy_computer = no	**412**	**2588**	3000
Total	7366	2634	10000

- Given *m* classes, an entry, $CM_{i,j}$ in a **confusion matrix** indicates # of tuples in class *i* that were labeled by the classifier as class *j*
- May have extra rows/columns to provide totals

Accuracy, Error Rate, Sensitivity and Specificity

A\P	C	¬C	
C	**TP**	**FN**	**P**
¬C	**FP**	**TN**	**N**
	P’	**N’**	**All**

- **Classifier Accuracy,** or recognition rate: percentage of test set tuples that are correctly classified

  Accuracy = (TP + TN)/All

- **Error rate(misclassification rate):** *1 – accuracy*, or

  Error rate = (FP + FN)/All

- **Class Imbalance Problem**:
  - One class may be *rare*, e.g. fraud, or HIV-positive
  - Significant *majority of the negative class* and minority of the positive class
  - **Sensitivity**: True Positive recognition rate (the proportion of positive tuples that are correctly identified)
    - **Sensitivity = TP/P**
  - **Specificity**: True Negative recognition rate (the proportion of negative tuples that are correctly identified)
    - **Specificity = TN/N**

Precision and Recall, and F-measures

- **Precision**: exactness – what % of tuples that the classifier labeled as positive are actually positive

$$precision = \frac{TP}{TP + FP}$$

- **Recall:** completeness – what % of positive tuples did the classifier label as positive?
- Perfect score is 1.0
- Inverse relationship between precision & recall
- ***F* measure (*F*$_1$ or *F*-score)**: harmonic mean of precision and recall,

$$recall = \frac{TP}{TP + FN} = \frac{TP}{P}$$

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

- ***F*$_ß$:** weighted measure of precision and recall
  - assigns ß times as much weight to recall as to precision

$$F_\beta = \frac{(1 + \beta^2) \times precision \times recall}{\beta^2 \times precision + recall}$$

Classifier Evaluation Metrics: Example

Actual Class\Predicted class	cancer = yes	cancer = no	Total	Recognition(%)
cancer = yes	**90**	**210**	300	30.00 (*sensitivity*
cancer = no	**140**	**9560**	9700	98.56 (*specificity*)
Total	230	9770	10000	96.40 (*accuracy*)

Measure	**Formula**
accuracy, recognition rate	$\frac{TP+TN}{P+N}$
error rate, misclassification rate	$\frac{FP+FN}{P+N}$
sensitivity, true positive rate, recall	$\frac{TP}{P}$
specificity, true negative rate	$\frac{TN}{N}$
precision	$\frac{TP}{TP+FP}$
F, F_1, F-score, harmonic mean of precision and recall	$\frac{2 \times precision \times recall}{precision + recall}$

Predicted class

Actual class		*yes*	*no*	Total
	yes	TP	FN	P
	no	FP	TN	N
	Total	P'	N'	$P+N$

- *Precision* = 90/230 = 39.13%
- *Recall* = 90/300 = 30.00%

Issues Affecting Model Selection

- **Speed**
  - time to construct the model (training time)
  - time to use the model (classification/prediction time)
- **Robustness**: handling noise and missing values
- **Scalability**: is typically assessed with a series of data sets of increasing size.
- **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

Evaluation: Holdout & Cross-Validation Methods

- **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 sampling: a variation of holdout
    - Repeat holdout k times, accuracy = avg. of the accuracies obtained
- **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***: folds are stratified so that class dist. in each fold is approx. the same as that in the initial data

Evaluation: Bootstrap

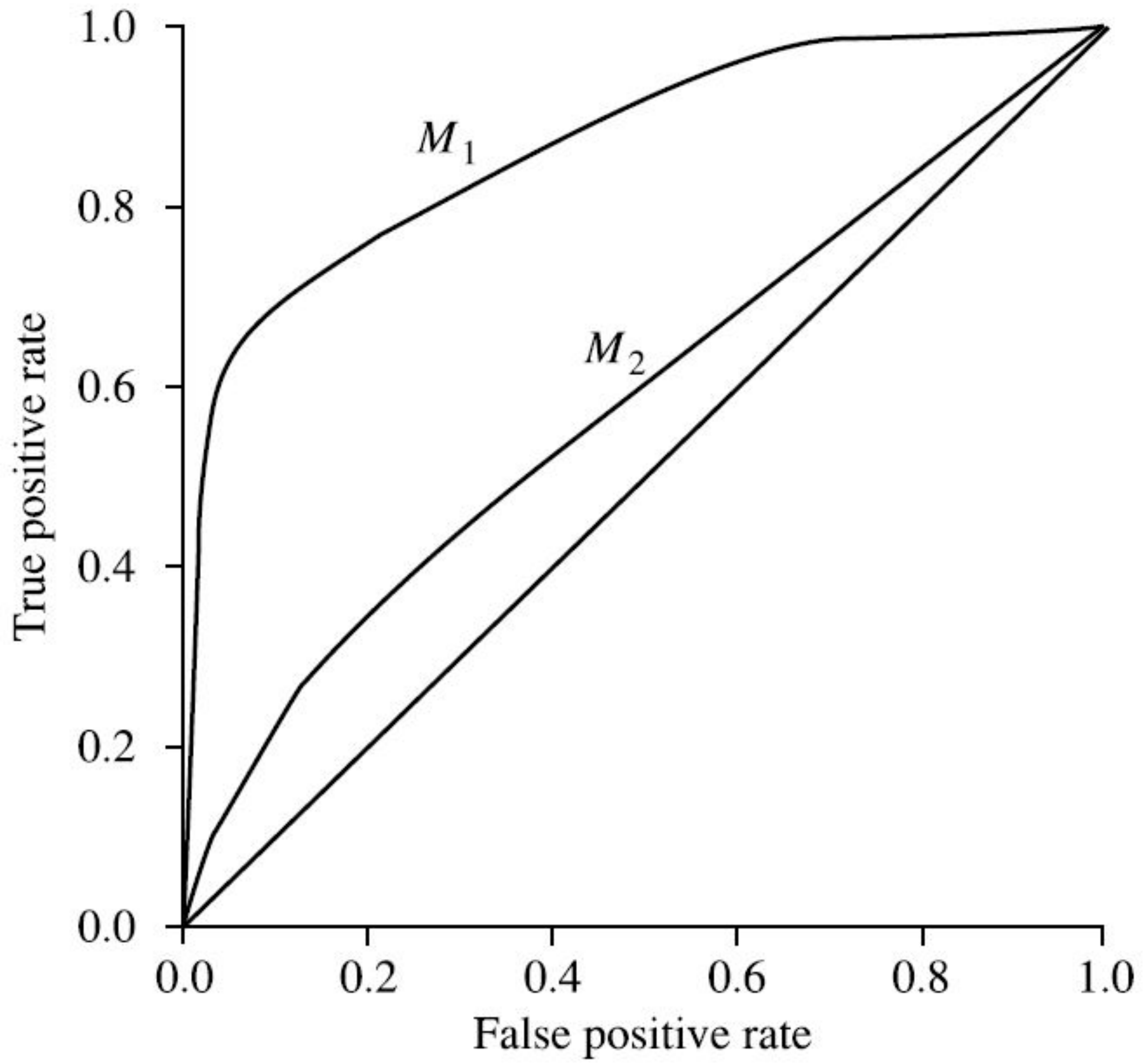
- **Bootstrap**
  - Works well with small data sets
  - Samples the given training tuples uniformly *with replacement*
    - i.e., each time a tuple is selected, it is equally likely to be selected again and re-added to the training set
- Several bootstrap methods, and a common one is **.632 boostrap**
  - 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:

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

where Acc(Mi)test_set is the accuracy of the model obtained with bootstrap sample i when it is applied to test set i. Acc(Mi)train_set is the accuracy of the model obtained with bootstrap sample i when it is applied to the original set of data tuples.

Model Selection: ROC Curves

- **ROC** (Receiver Operating Characteristics) curves: for visual comparison of classification models
- Originated from signal detection theory
- Shows the trade-off between the true positive rate and the false positive rate
- The area under the ROC curve is a measure of the accuracy of the model
- Rank the test tuples in decreasing order: the one that is most likely to belong to the positive class appears at the top of the list.
- The closer to the diagonal line (i.e., the closer the area is to 0.5), the less accurate is the model



- Vertical axis represents the **true positive rate**
- Horizontal axis rep. the **false positive rate**
- The plot also shows a diagonal line
- A model with perfect accuracy will have an area of 1.0

Model Selection: ROC Curves

Example. Plotting a ROC curve. The following figure shows the probability value (column 3) returned by a probabilistic classifier for each of the 10 tuples in a test set, sorted in the decreasing probability order.

Tuple #	Class	Prob.	*TP*	*FP*	*TN*	*FN*	*TPR*	*FPR*
1	*P*	0.90	1	0	5	4		
2	*P*	0.80	2	0	5	3		
3	*N*	0.70	2	1	4	3		
4	*P*	0.60	3	1	4	2		
5	*P*	0.55	4	1	4	1		
6	*N*	0.54	4	2	3	1		
7	*N*	0.53	4	3	2	1		
8	*N*	0.51	4	4	1	1		
9	*P*	0.50	5	4	1	0		
10	*N*	0.40	5	5	0	0		

Predicted class

Actual class		*yes*	*no*	Total
	yes	TP	FN	P
	no	FP	TN	N
	Total	P'	N'	$P+N$

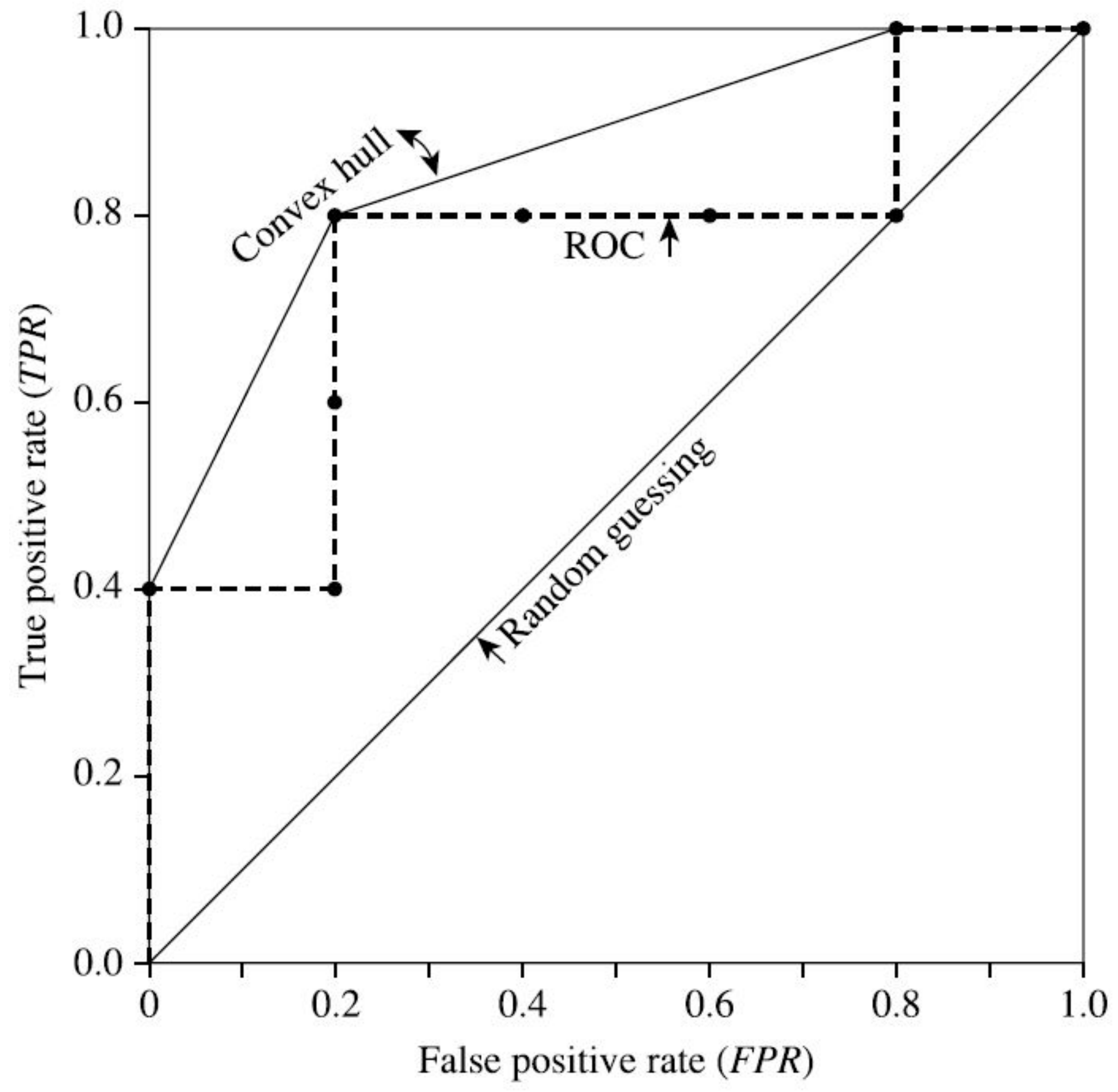
TPR = TP / (TP + FN)

FPR = FP / (FP + TN)

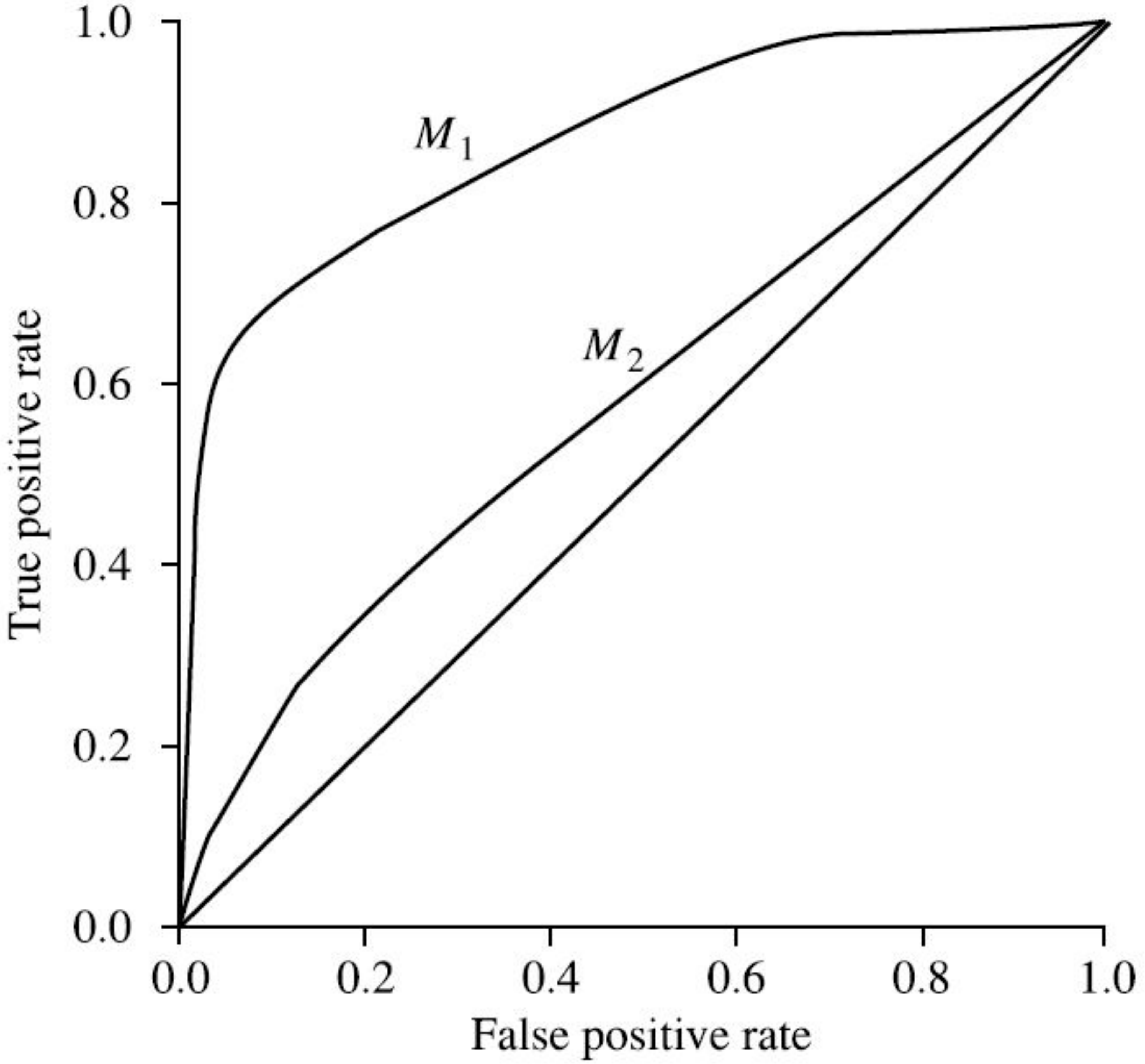
Model Selection: ROC Curves

Example. Plotting a ROC curve. The following figure shows the probability value (column 3) returned by a probabilistic classifier for each of the 10 tuples in a test set, sorted in the decreasing probability order.

Tuple #	Class	Prob.	*TP*	*FP*	*TN*	*FN*	*TPR*	*FPR*
1	*P*	0.90	1	0	5	4	0.2	0
2	*P*	0.80	2	0	5	3	0.4	0
3	*N*	0.70	2	1	4	3	0.4	0.2
4	*P*	0.60	3	1	4	2	0.6	0.2
5	*P*	0.55	4	1	4	1	0.8	0.2
6	*N*	0.54	4	2	3	1	0.8	0.4
7	*N*	0.53	4	3	2	1	0.8	0.6
8	*N*	0.51	4	4	1	1	0.8	0.8
9	*P*	0.50	5	4	1	0	1.0	0.8
10	*N*	0.40	5	5	0	0	1.0	1.0



Model Selection: ROC Curves



Summary

- Evaluation metrics
  - Confusion Matrix, Accuracy, Error Rate,
  - Sensitivity and Specificity
  - Precision and Recall, and F-measures
  - Issues Affecting Model Selection
- Methods for estimating a classifier’s accuracy:
  - Holdout method, random subsampling
  - Cross-validation
  - Bootstrap
- Comparing classifiers:
  - Cost-benefit analysis and ROC Curves