Statistical Learning and Data Mining

Lecture: Classification - Model Evaluation

Prof. Pradeep Ravikumar
pradeepr@cs.utexas.edu

Adapted From: Pang-Ning Tan, Steinbach, Kumar

Model Evaluation

- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?
- Methods for Model Comparison
  - How to compare the relative performance among competing models?

Model Evaluation

- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?

- Methods for Performance Evaluation
  - How to obtain reliable estimates?

- Methods for Model Comparison
  - How to compare the relative performance among competing models?

Metrics for Performance Evaluation

- Focus on the predictive capability of a model
  - Rather than how fast it takes to classify or build models, scalability, etc.

Metrics for Performance Evaluation

- Focus on the predictive capability of a model
  - Rather than how fast it takes to classify or build models, scalability, etc.
- Confusion Matrix:

	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a	b
	Class=No	c	d

a: TP (true positive)

b: FN (false negative)

c: FP (false positive)

d: TN (true negative)

Metrics for Performance Evaluation

	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

Metrics for Performance Evaluation

	PREDICTED CLASS	

		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

- Most widely-used metric:

$$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

Limitation of Accuracy

- Consider a 2-class problem
  - Number of Class 0 examples = 9990
  - Number of Class 1 examples = 10

- If model predicts everything to be class 0, accuracy is 9990/10000 = 99.9 %

Limitation of Accuracy

- Consider a 2-class problem
  - Number of Class 0 examples = 9990
  - Number of Class 1 examples = 10

- If model predicts everything to be class 0, accuracy is 9990/10000 = 99.9 %
  - Accuracy is misleading because model does not detect any class 1 example

Cost Matrix

	PREDICTED CLASS		
ACTUAL CLASS	C(i|j)	**Class=Yes**	**Class=No**
	Class=Yes	C(Yes|Yes)	C(No|Yes)
	Class=No	C(Yes|No)	C(No|No)

C(i|j): Cost of misclassifying class j example as class i

Computing Cost of Classification

Cost Matrix	PREDICTED CLASS		
ACTUAL CLASS	C(i|j)	**+**	**-**
	+	-1	100
	-	1	0

Computing Cost of Classification

Cost Matrix	PREDICTED CLASS		
ACTUAL CLASS	C(i|j)	+	-
	+	-1	100
	-	1	0

Model M_1	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	150	40
	-	60	250

Computing Cost of Classification

Cost Matrix	PREDICTED CLASS		
ACTUAL CLASS	C(i|j)	**+**	**-**
	+	-1	100
	-	1	0

Model M_1	PREDICTED CLASS		
ACTUAL CLASS		**+**	**-**
	+	150	40
	-	60	250

Accuracy = 80%

Cost = 3910

Computing Cost of Classification

Cost Matrix	PREDICTED CLASS		
ACTUAL CLASS	C(i|j)	+	-
	+	-1	100
	-	1	0

Model M_1	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	150	40
	-	60	250

Model M_2	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	250	45
	-	5	200

Accuracy = 80%

Cost = 3910

Computing Cost of Classification

Cost Matrix	PREDICTED CLASS		
ACTUAL CLASS	C(i|j)	+	-
	+	-1	100
	-	1	0

Model M_1	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	150	40
	-	60	250

Accuracy = 80%

Cost = 3910

Model M_2	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	250	45
	-	5	200

Accuracy = 90%

Cost = 4255

Cost vs Accuracy

Count	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a	b
	Class=No	c	d

Accuracy is proportional to cost if
1. C(Yes|No)=C(No|Yes) = q
2. C(Yes|Yes)=C(No|No) = p

Cost vs Accuracy

Count	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a	b
	Class=No	c	d

Accuracy is proportional to cost if
1. C(Yes|No)=C(No|Yes) = q
2. C(Yes|Yes)=C(No|No) = p

N = a + b + c + d

Cost vs Accuracy

Count	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a	b
	Class=No	c	d

Accuracy is proportional to cost if
1. C(Yes|No)=C(No|Yes) = q
2. C(Yes|Yes)=C(No|No) = p

N = a + b + c + d

Accuracy = (a + d)/N

Cost vs Accuracy

Count	PREDICTED CLASS	
ACTUAL CLASS	Class=Yes	Class=No
Class=Yes	a	b
Class=No	c	d

Cost	PREDICTED CLASS	
ACTUAL CLASS	Class=Yes	Class=No
Class=Yes	p	q
Class=No	q	p

Accuracy is proportional to cost if

1. C(Yes|No)=C(No|Yes) = q
2. C(Yes|Yes)=C(No|No) = p

N = a + b + c + d

Accuracy = (a + d)/N

Cost vs Accuracy

Count	PREDICTED CLASS	
	Class=Yes	Class=No
ACTUAL CLASS: Class=Yes	a	b
ACTUAL CLASS: Class=No	c	d

Cost	PREDICTED CLASS	
	Class=Yes	Class=No
ACTUAL CLASS: Class=Yes	p	q
ACTUAL CLASS: Class=No	q	p

Accuracy is proportional to cost if

1. C(Yes|No)=C(No|Yes) = q
2. C(Yes|Yes)=C(No|No) = p

N = a + b + c + d

Accuracy = (a + d)/N

Cost = p (a + d) + q (b + c)

Cost vs Accuracy

Count	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a	b
	Class=No	c	d

Cost	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	p	q
	Class=No	q	p

Accuracy is proportional to cost if

1. C(Yes|No)=C(No|Yes) = q
2. C(Yes|Yes)=C(No|No) = p

$N = a + b + c + d$

$\text{Accuracy} = (a + d)/N$

$$
\begin{aligned}
\text{Cost} &= p\,(a + d) + q\,(b + c) \\
&= p\,(a + d) + q\,(N - a - d) \\
&= q\,N - (q - p)(a + d) \\
&= N\,[q - (q\text{-}p) \times \text{Accuracy}]
\end{aligned}
$$

Cost-Sensitive Measures

Count	PREDICTED CLASS	
	Class=Yes	Class=No
ACTUAL CLASS: Class=Yes	a	b
ACTUAL CLASS: Class=No	c	d

$$\text{Precision (p)} = \frac{a}{a+c}$$

Among examples predicted as positive, fraction correctly predicted

- Precision is biased towards C(Yes|Yes) & C(Yes|No)

Cost-Sensitive Measures

Count	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a	b
	Class=No	c	d

$$\text{Precision (p)} = \frac{a}{a+c}$$

Among examples predicted as positive, fraction correctly predicted

$$\text{Recall (r)} = \frac{a}{a+b}$$

Among examples that are actually positive, fraction correctly predicted

- Precision is biased towards C(Yes|Yes) & C(Yes|No)
- Recall is biased towards C(Yes|Yes) & C(No|Yes)

Cost-Sensitive Measures

Count	PREDICTED CLASS	
	Class=Yes	Class=No
ACTUAL CLASS Class=Yes	a	b
ACTUAL CLASS Class=No	c	d

$$\text{Precision (p)} = \frac{a}{a+c}$$

Among examples predicted as positive, fraction correctly predicted

$$\text{Recall (r)} = \frac{a}{a+b}$$

Among examples that are actually positive, fraction correctly predicted

$$\text{F - measure (F)} = \frac{2rp}{r+p} = \frac{2a}{2a+b+c}$$

- Precision is biased towards C(Yes|Yes) & C(Yes|No)
- Recall is biased towards C(Yes|Yes) & C(No|Yes)
- F-measure is biased towards all except C(No|No)

Cost-Sensitive Measures

Count	PREDICTED CLASS	
	Class=Yes	Class=No
ACTUAL CLASS: Class=Yes	a	b
ACTUAL CLASS: Class=No	c	d

$$\text{Precision (p)} = \frac{a}{a+c}$$

Among examples predicted as positive, fraction correctly predicted

$$\text{Recall (r)} = \frac{a}{a+b}$$

Among examples that are actually positive, fraction correctly predicted

$$\text{F - measure (F)} = \frac{2rp}{r+p} = \frac{2a}{2a+b+c}$$

- Precision is biased towards C(Yes|Yes) & C(Yes|No)
- Recall is biased towards C(Yes|Yes) & C(No|Yes)
- F-measure is biased towards all except C(No|No)

$$\text{Weighted Accuracy} = \frac{w_1 a + w_4 d}{w_1 a + w_2 b + w_3 c + w_4 d}$$

Other Measures

- **True Positive Rate (TPR)** = TP/(TP + FN)

Other Measures

- **True Positive Rate (TPR)** = TP/(TP + FN)
  - Among actual positive examples, fraction correctly predicted (cf. Recall)

Other Measures

- **True Positive Rate (TPR)** = TP/(TP + FN)
  - Among actual positive examples, fraction correctly predicted (cf. Recall)
- **True Negative Rate (TNR)** = TN/(TN + FP)

Other Measures

- **True Positive Rate (TPR)** = TP/(TP + FN)
  - Among actual positive examples, fraction correctly predicted (cf. Recall)
- **True Negative Rate (TNR)** = TN/(TN + FP)
  - Among actual negative examples, fraction correctly predicted

Other Measures

- **True Positive Rate (TPR)** = TP/(TP + FN)
  - Among actual positive examples, fraction correctly predicted (cf. Recall)
- **True Negative Rate (TNR)** = TN/(TN + FP)
  - Among actual negative examples, fraction correctly predicted
- **False Positive Rate (FPR)** = FP/(TN + FP)

Other Measures

- **True Positive Rate (TPR)** = TP/(TP + FN)
  - Among actual positive examples, fraction correctly predicted (cf. Recall)
- **True Negative Rate (TNR)** = TN/(TN + FP)
  - Among actual negative examples, fraction correctly predicted
- **False Positive Rate (FPR)** = FP/(TN + FP)
  - Among actual negative examples, fraction incorrectly predicted

Other Measures

- **True Positive Rate (TPR)** = TP/(TP + FN)
  - Among actual positive examples, fraction correctly predicted (cf. Recall)
- **True Negative Rate (TNR)** = TN/(TN + FP)
  - Among actual negative examples, fraction correctly predicted
- **False Positive Rate (FPR)** = FP/(TN + FP)
  - Among actual negative examples, fraction incorrectly predicted
- **False Negative Rate (TPR)** = FN/(TP + FN)

Other Measures

- **True Positive Rate (TPR)** = TP/(TP + FN)
  - Among actual positive examples, fraction correctly predicted (cf. Recall)
- **True Negative Rate (TNR)** = TN/(TN + FP)
  - Among actual negative examples, fraction correctly predicted
- **False Positive Rate (FPR)** = FP/(TN + FP)
  - Among actual negative examples, fraction incorrectly predicted
- **False Negative Rate (TPR)** = FN/(TP + FN)
  - Among actual positive examples, fraction incorrectly predicted

Model Evaluation

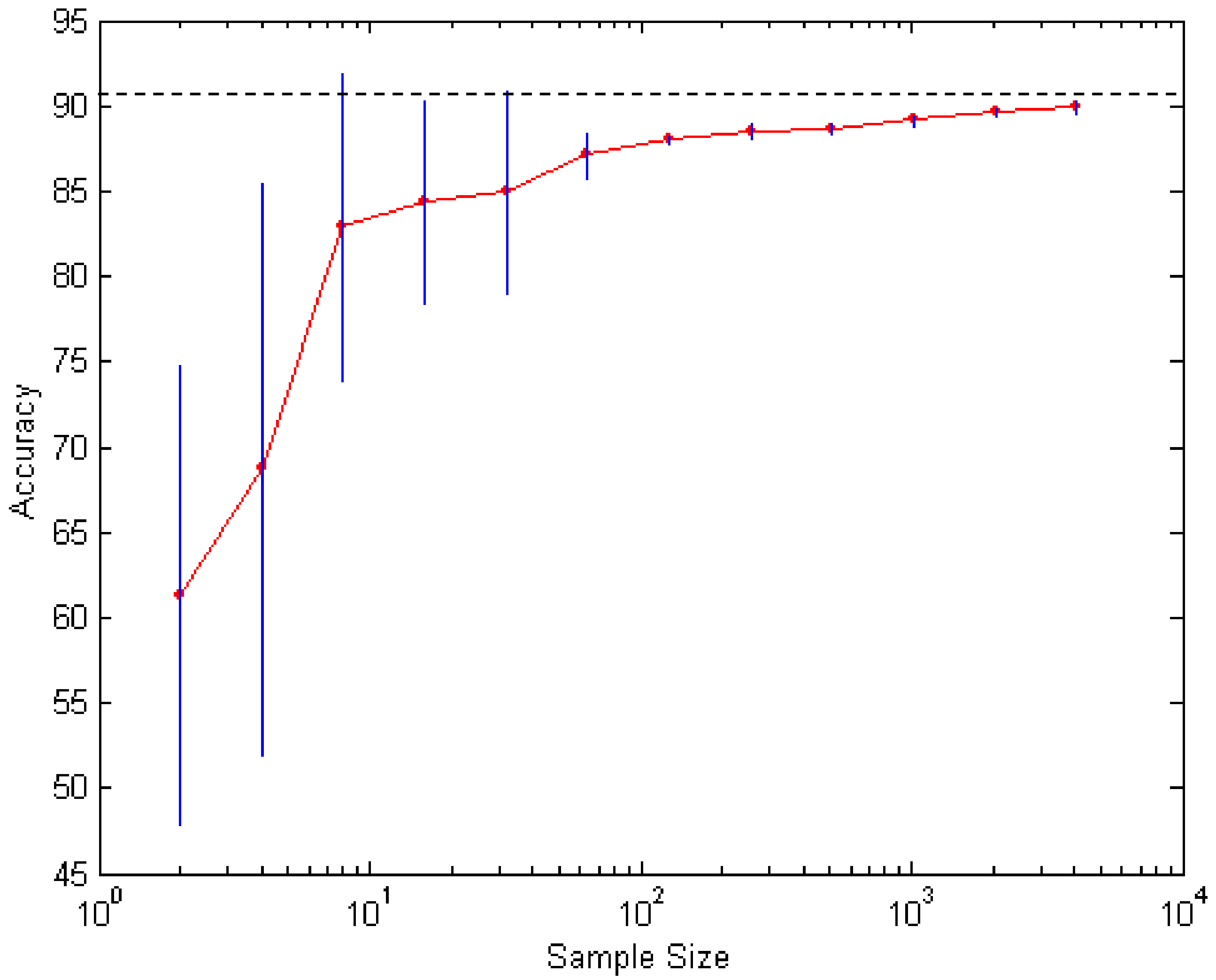
- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?
- Methods for Model Comparison
  - How to compare the relative performance among competing models?

Methods for Performance Evaluation

- How to obtain a reliable estimate of performance?

- Performance of a model may depend on other factors besides the learning algorithm:
  - Class distribution
  - Cost of misclassification
  - Size of training and test sets

Learning Curve



- Learning curve shows how accuracy changes with varying sample size

Methods of Estimation

- Holdout
  - Reserve 2/3 for training and 1/3 for testing

Methods of Estimation

- Holdout
  - Reserve 2/3 for training and 1/3 for testing

Caveats:

> Less data available for training

> If more for training, then test error is unreliable;
if more for testing, model might be fit less well

> Training, test datasets no longer independent
e.g. a class over-represented in training, will be
under-represented in test

Methods of Estimation

- Holdout
  - Reserve 2/3 for training and 1/3 for testing
- Random subsampling
  - Repeated holdout

Methods of Estimation

- Holdout
  - Reserve 2/3 for training and 1/3 for testing
- Random subsampling
  - Repeated holdout

Caveats:

> Less data available for training

> No control over records used for training: some records may show up in multiple sample-draws

Methods of Estimation

- Holdout
  - Reserve 2/3 for training and 1/3 for testing
- Random subsampling
  - Repeated holdout
- Cross validation
  - Partition data into k disjoint subsets
  - k-fold: train on k-1 partitions, test on the remaining one
  - Leave-one-out: $k=n$

Methods of Estimation

- Holdout
  - Reserve 2/3 for training and 1/3 for testing
- Random subsampling
  - Repeated holdout
- Cross validation
  - Partition data into k disjoint subsets
  - k-fold: train on k-1 partitions, test on the remaining one
  - Leave-one-out: k=n

Caveats:

> Computation

Methods of Estimation

- Holdout
  - Reserve 2/3 for training and 1/3 for testing
- Random subsampling
  - Repeated holdout
- Cross validation
  - Partition data into k disjoint subsets
  - k-fold: train on k-1 partitions, test on the remaining one
  - Leave-one-out: k=n

- Bootstrap
  - Sampling with replacement

Bootstrap

- If number of records is N, then N records are sampled ***with replacement***
- A bootstrap samples of size N will contain about 63.2% of original records
  - Prob.(record is chosen in N draws) $= 1 - (1 - 1/N)^N \approx 1 - e^{-1} = 0.632$

Bootstrap

- If number of records is N, then N records are sampled ***with replacement***
- A bootstrap samples of size N will contain about 63.2% of original records
  - Prob.(record is chosen in N draws) $= 1 - (1 - 1/N)^N \approx 1 - e^{-1} = 0.632$
- Variant: **0.632 bootstrap**:

Combine accuracies of each bootstrap sample ϵ_i with the accuracy computed from a training set that contains all the samples acc_s:

$$\text{Accuracy, } acc_{boot} = \frac{1}{b}\sum_{i=1}^{b}(0.632 \times \epsilon_i + 0.368 \times acc_s).$$

Model Evaluation

- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?
- Methods for Model Comparison
  - How to compare the relative performance among competing models?

ROC (Receiver Operating Characteristic)

- Developed in 1950s for signal detection theory to analyze noisy signals
  - Characterize the trade-off between positive hits and false alarms

ROC (Receiver Operating Characteristic)

- Developed in 1950s for signal detection theory to analyze noisy signals
  - Characterize the trade-off between positive hits and false alarms
- ROC curve plots **TPR** (on the y-axis) against **FPR** (on the x-axis)

ROC (Receiver Operating Characteristic)

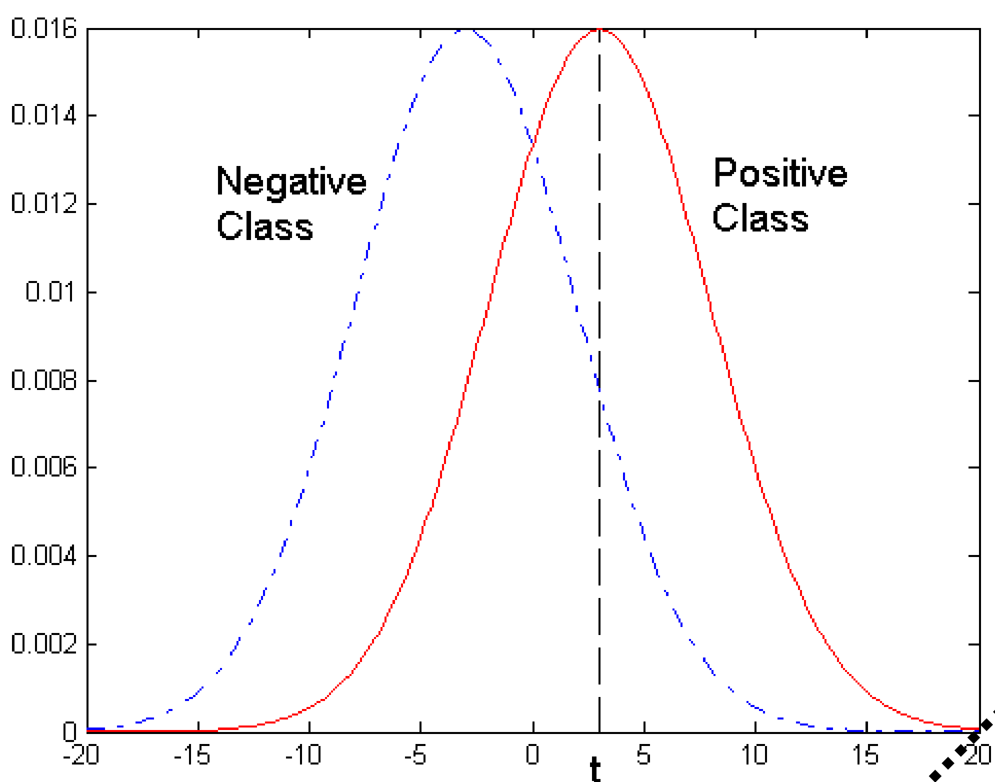
- Developed in 1950s for signal detection theory to analyze noisy signals
  - Characterize the trade-off between positive hits and false alarms
- ROC curve plots **TPR** (on the y-axis) against **FPR** (on the x-axis)
- Performance of each classifier represented as a point on the ROC curve

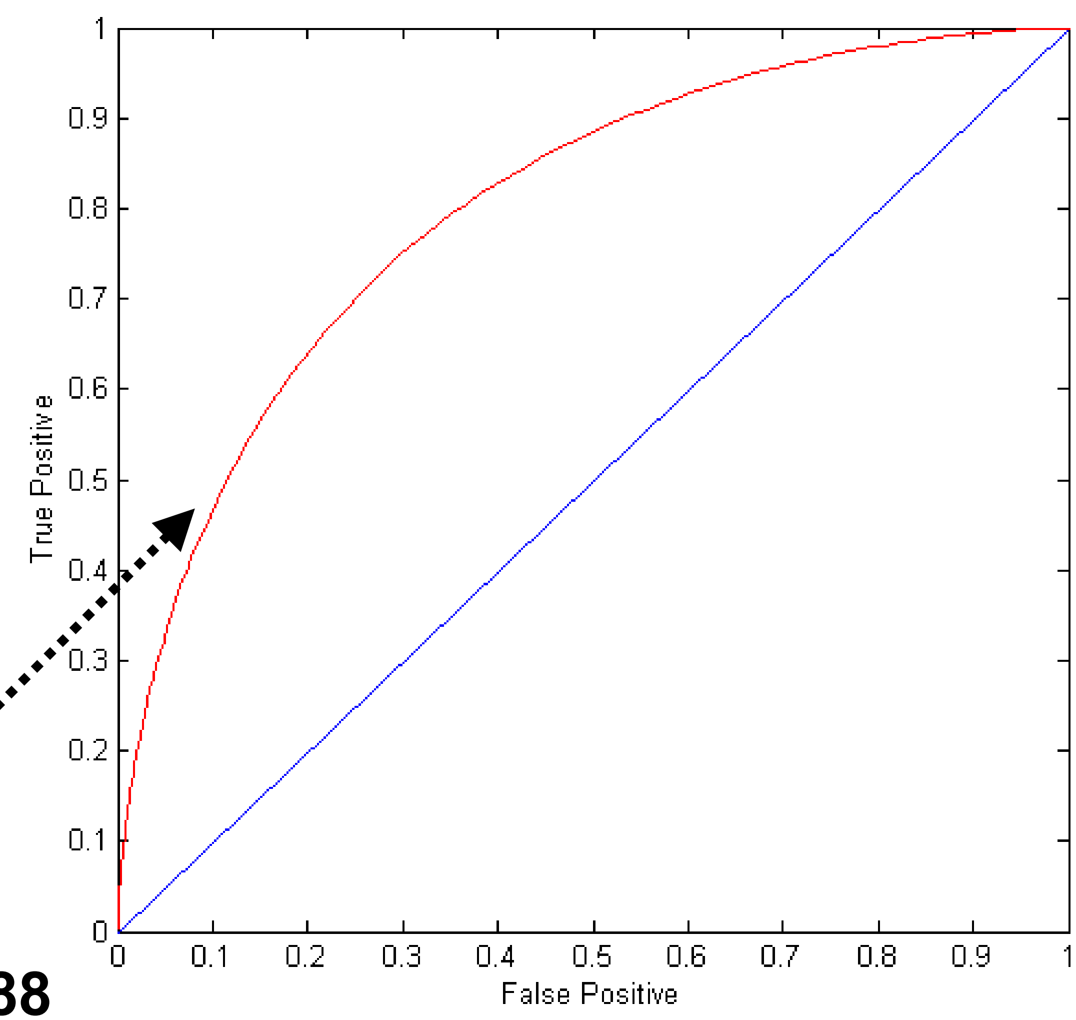
ROC (Receiver Operating Characteristic)

- Developed in 1950s for signal detection theory to analyze noisy signals
  - Characterize the trade-off between positive hits and false alarms
- ROC curve plots **TPR** (on the y-axis) against **FPR** (on the x-axis)
- Performance of each classifier represented as a point on the ROC curve
  - changing the threshold of algorithm, sample distribution or cost matrix changes the location of the point

ROC Curve

- 1-dimensional data set containing 2 classes (positive and negative)
- any points located at x > t is classified as positive





At threshold t:

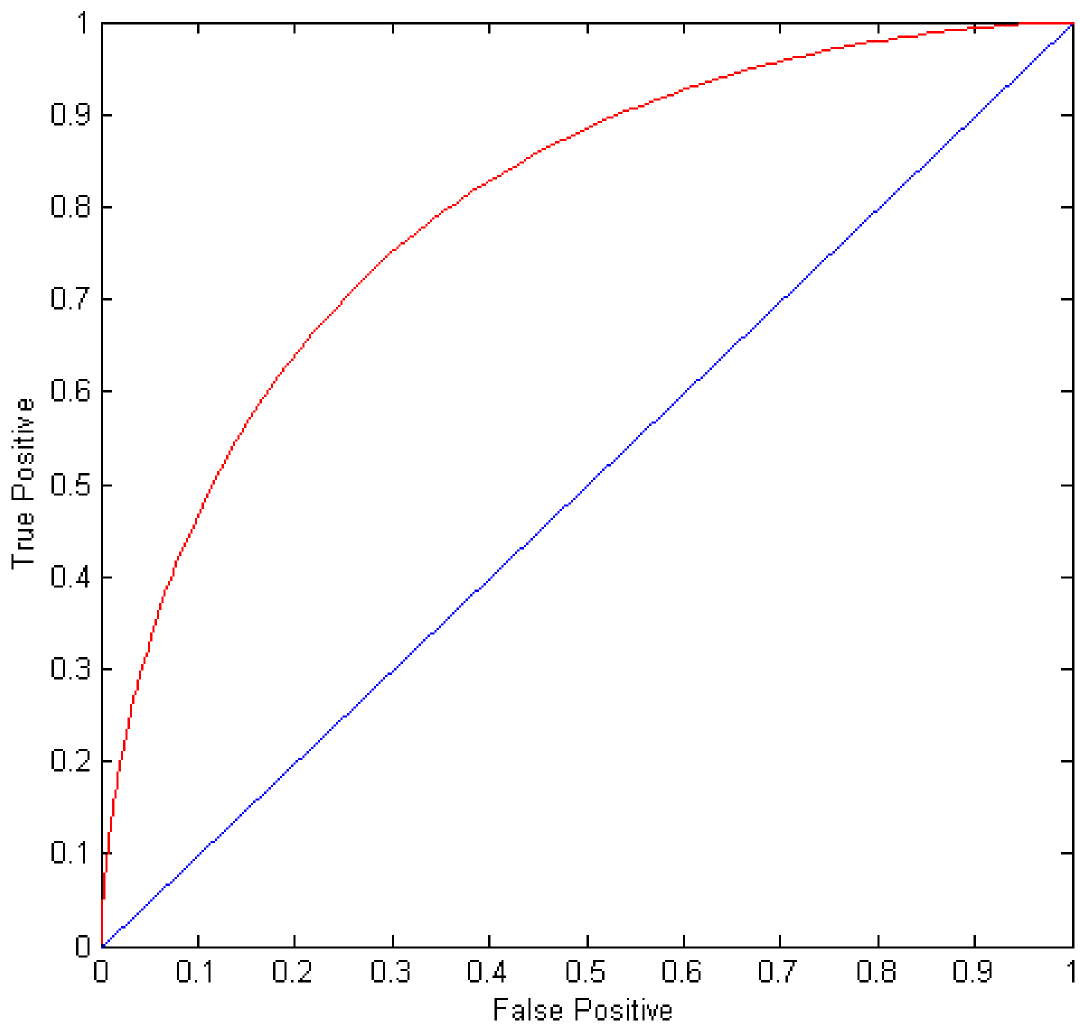
TP=0.5, FN=0.5, FP=0.12, FN=0.88

ROC Curve

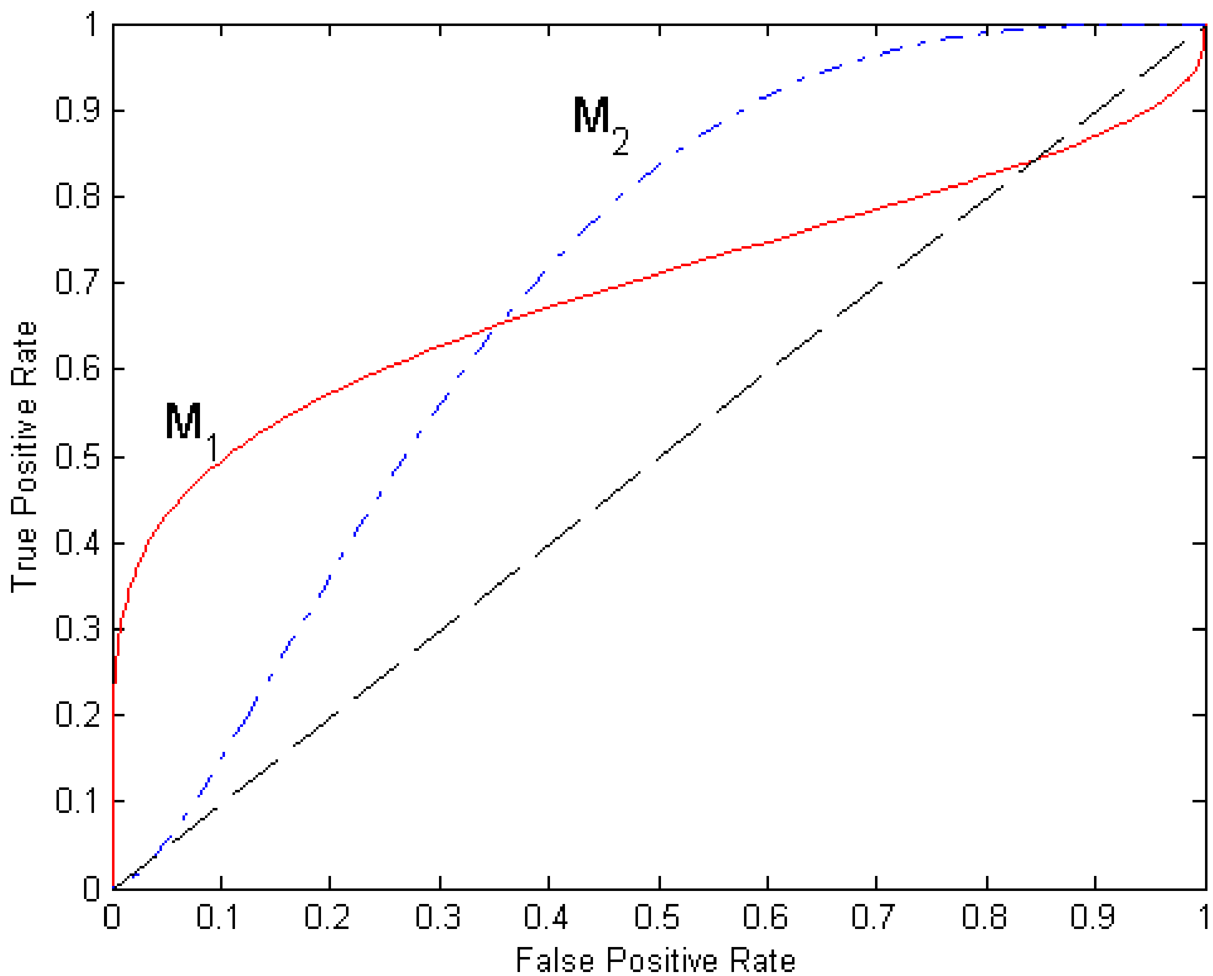
(TP,FP):

- (0,0): declare everything to be negative class
- (1,1): declare everything to be positive class
- (1,0): ideal

- Diagonal line:
  - Random guessing
  - Below diagonal line:
    - prediction is opposite of the true class



Using ROC for Model Comparison



- No model consistently outperform the other
  - M_1 is better for small FPR
  - M_2 is better for large FPR
- Area Under the ROC curve
  - Ideal:
    - Area = 1
  - Random guess:
    - Area = 0.5

How to Construct an ROC Curve

Instance	P(+|A)	True Class
1	0.95	+
2	0.93	+
3	0.87	-
4	0.85	-
5	0.85	-
6	0.85	+
7	0.76	-
8	0.53	+
9	0.43	-
10	0.25	+

- Use classifier that produces posterior probability for each test instance P(+|A)
- Sort the instances according to P(+|A) in decreasing order
- Apply threshold at each unique value of P(+|A)
- Count the number of TP, FP, TN, FN at each threshold
- TP rate, TPR = TP/(TP+FN)
- FP rate, FPR = FP/(FP + TN)