DS 4400

Machine Learning and Data Mining I
Spring 2022

Alina Oprea
Associate Professor
Khoury College of Computer Science
Northeastern University

February 23 2022

Outline

- Evaluation of classifiers
  - Accuracy, error, precision, recall
  - ROC curves and the AUC metric
  - Why multiple metrics
- Generative classifiers
- Linear Discriminant Analysis (LDA)
- Midterm exam review

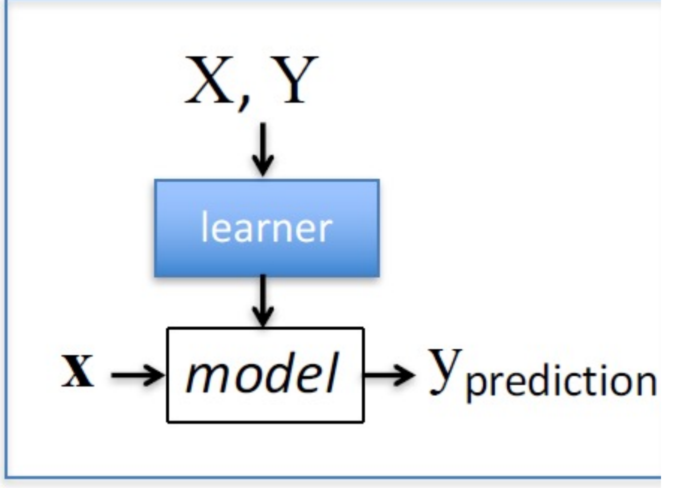
Classifier Evaluation

- Classification is a supervised learning problem
  - Prediction is binary or multi-class
- Classification techniques
  - Linear classifiers
    - Logistic regression (probabilistic interpretation)
  - Instance learners
    - kNN: need to store entire training data
- Cross-validation should be used for parameter selection and estimation of model error

Evaluation of classifiers

Given: labeled training data $X, Y = \{\langle \boldsymbol{x}_i, y_i \rangle\}_{i=1}^{n}$

- Assumes each $\boldsymbol{x}_i \sim \mathcal{D}(\mathcal{X})$



Train the model:

model ← *classifier*.train(X, Y)

Apply the model to new data:

- Given: new unlabeled instance $\boldsymbol{x} \sim \mathcal{D}(\mathcal{X})$

$\mathrm{y}_{\text{prediction}}$ ← *model*.predict($\mathbf{x}$)

Classification Metrics

$$\text{accuracy} = \frac{\#\ \text{correct predictions}}{\#\ \text{instances}}$$

$$\text{error} = 1 - \text{accuracy} = \frac{\#\ \text{incorrect predictions}}{\#\ \text{instances}}$$

- Can evaluate on both training or testing
- Training set accuracy and error
- Testing set accuracy and error

Confusion Matrix

Given a dataset of P positive instances and N negative instances:

		Predicted Class	
		Yes	No
Actual Class	Yes	TP	FN
	No	FP	TN

Accuracy and Error

Given a dataset of P positive instances and N negative instances:

		Predicted Class	
		Yes	No
Actual Class	Yes	TP	FN
	No	FP	TN

$$\text{accuracy} = \frac{TP + TN}{P + N}$$

		Predicted Class	
		Yes	No
Actual Class	Yes	TP	FN
	No	FP	TN

$$\text{error} = 1 - \frac{TP + TN}{P + N}$$

$$= \frac{FP + FN}{P + N}$$

Confusion Matrix

- Given a dataset of P positive instances and N negative instances:

		Predicted Class	
		Yes	No
Actual Class	Yes	TP	FN
	No	FP	TN

$$\text{accuracy} = \frac{TP + TN}{P + N}$$

- Imagine using classifier to identify positive cases (i.e., for information retrieval)

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

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

Probability that classifier predicts positive correctly

Probability that actual class is predicted correctly

Why One Metric is Not Enough

Assume that in your training data, Spam email is 1% of data, and Ham email is 99% of data

- Scenario 1
  - Have classifier always output HAM!
  - What is the accuracy? 99%
- Scenario 2
  - Predict one SPAM email as SPAM, all other emails as legitimate
  - What is the precision? 100%
- Scenario 3
  - Output always SPAM!
  - What is the recall? 100%

Precision & Recall

Precision

- the fraction of positive predictions that are correct
- P(is pos|predicted pos)

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

Recall

- fraction of positive instances that are identified
- P(predicted pos|is pos)

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

- You can get high recall (but low precision) by only predicting positive
- Recall is a non-decreasing function of the # positive predictions
- Typically, precision decreases as either the number of positive predictions or recall increases
- Precision & recall are widely used in information retrieval

F-Score

- Combined measure of precision/recall tradeoff

$$F_1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

  - This is the harmonic mean of precision and recall
  - In the F_1 measure, precision and recall are weighted evenly
  - Can also have biased weightings that emphasize either precision or recall more (F_2 = 2 × recall; $F_{0.5}$ = 2 × precision)

- Limitations:
  - F-measure can exaggerate performance if balance between precision and recall is incorrect for application
    - Don't typically know balance ahead of time

A Word of Caution

- Consider binary classifiers A, B, C:

		A	.	B	.	C	.
		1	0	1	0	1	0
Predictions	1	0.9	0.1	0.8	0	0.78	0
	0	0	0	0.1	0.1	0.12	0.1

A Word of Caution

- Consider binary classifiers A, B, C:

		A	.	B	.	C	.
		1	0	1	0	1	0
Predictions	1	0.9	0.1	0.8	0	0.78	0
	0	0	0	0.1	0.1	0.12	0.1

- Clearly A is useless, since it always predicts 1
- B is slightly better than C
  - less probability mass wasted on the off-diagonals
- But, here are the performance metrics:

Metric	A	B	C
Accuracy	0.9	0.9	0.88
Precision	0.9	1.0	1.0
Recall	1.0	0.888	0.8667
F-score	0.947	0.941	0.9286

Classifiers can be tuned

- Logistic regression sets by default the threshold at 0.5 for classifying positive and negative instances
- Some applications have strict constraints on false positives (or other metrics)
  - Example: very low false positives in security (spam)

Classifiers can be tuned

- Logistic regression sets by default the threshold at 0.5 for classifying positive and negative instances
- Some applications have strict constraints on false positives (or other metrics)
  - Example: very low false positives in security (spam)
- Solution: choose different threshold

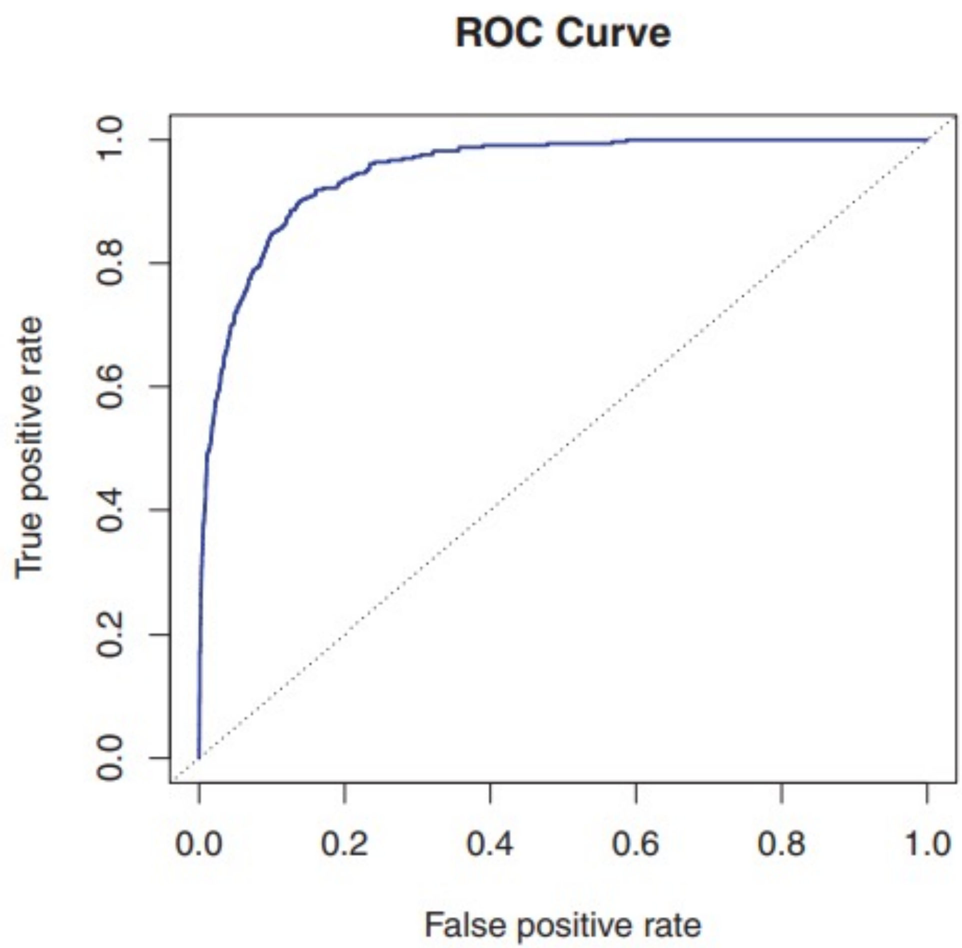
Probabilistic model $h_{\theta(x)} = \mathrm{P}[y = 1|x; \theta]$

– Predict y = 1 if $h_\theta(x) \geq$ T

– Predict y = 0 if $h_\theta(x) <$ T

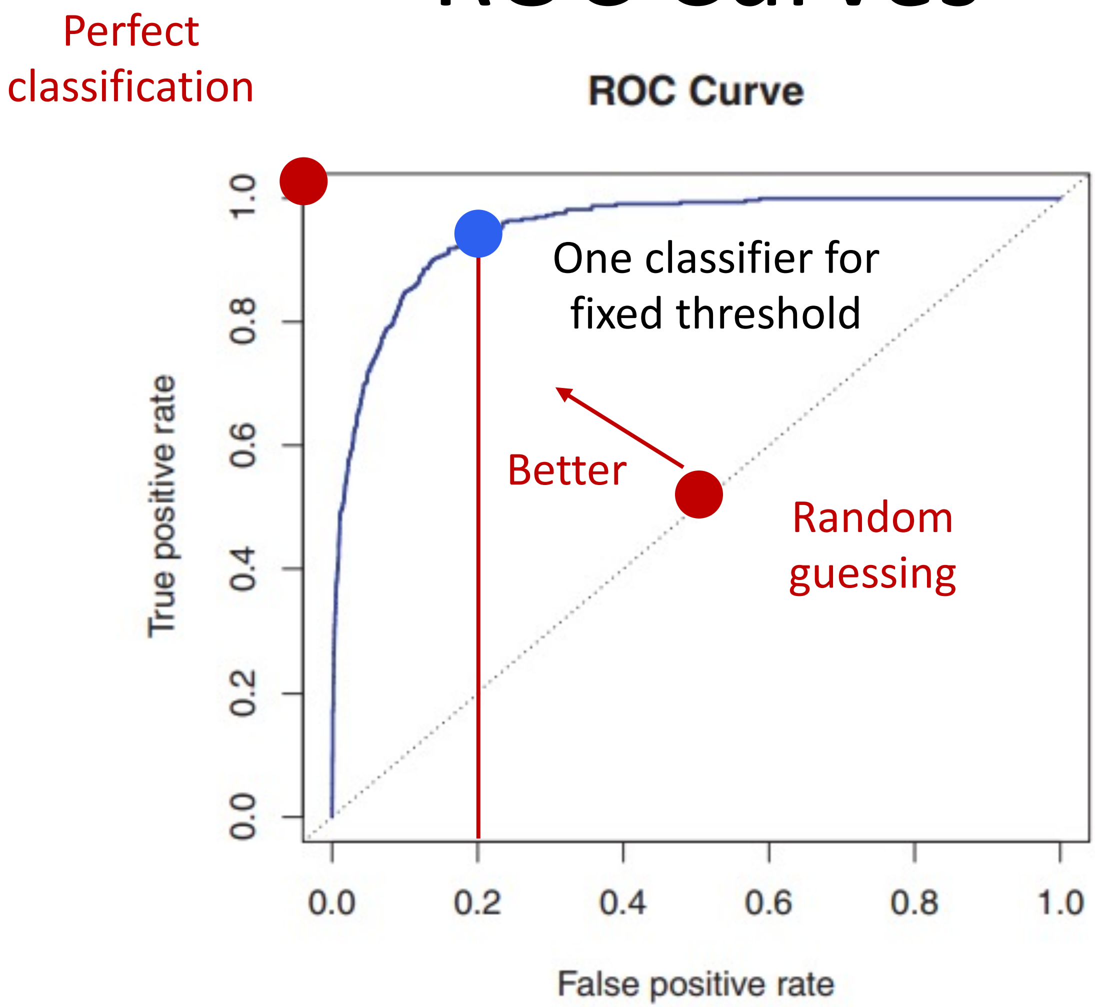
Higher T, lower FP
Lower T, lower FN

ROC Curves



- Receiver Operating Characteristic (ROC)
- Determine operating point (e.g., by fixing false positive rate)

ROC Curves

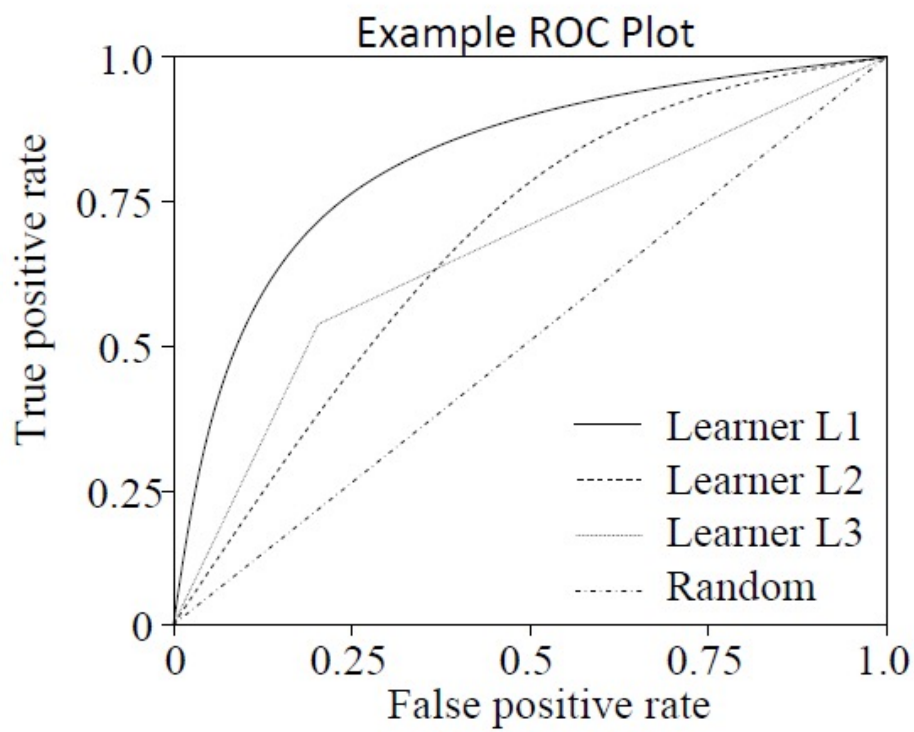


- Receiver Operating Characteristic (ROC)
- Determine operating point (e.g., by fixing false positive rate)

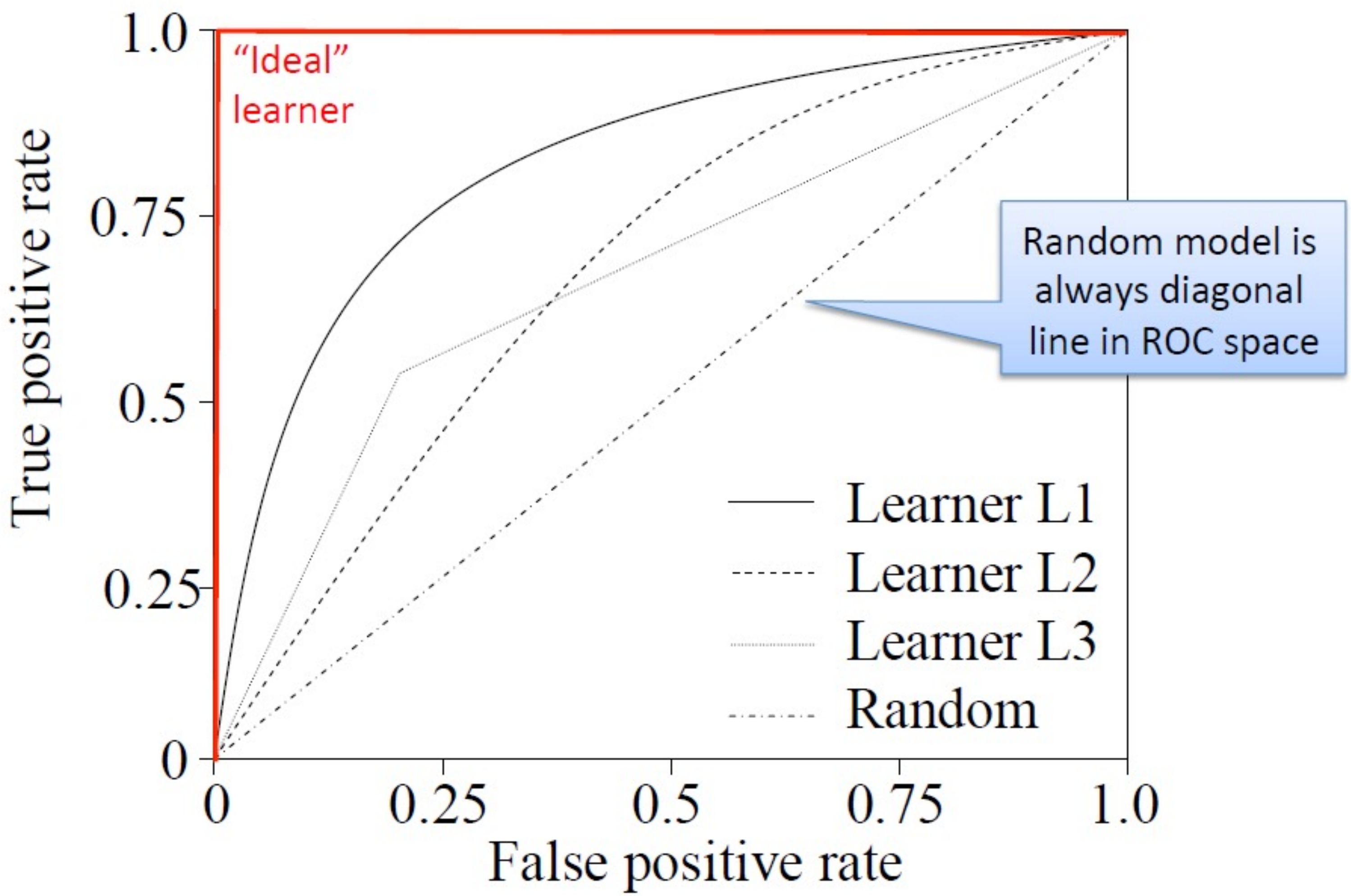
Performance Depends on Threshold

Predict positive if $\mathrm{P}(y = 1 \mid \mathbf{x}) > T$ otherwise negative

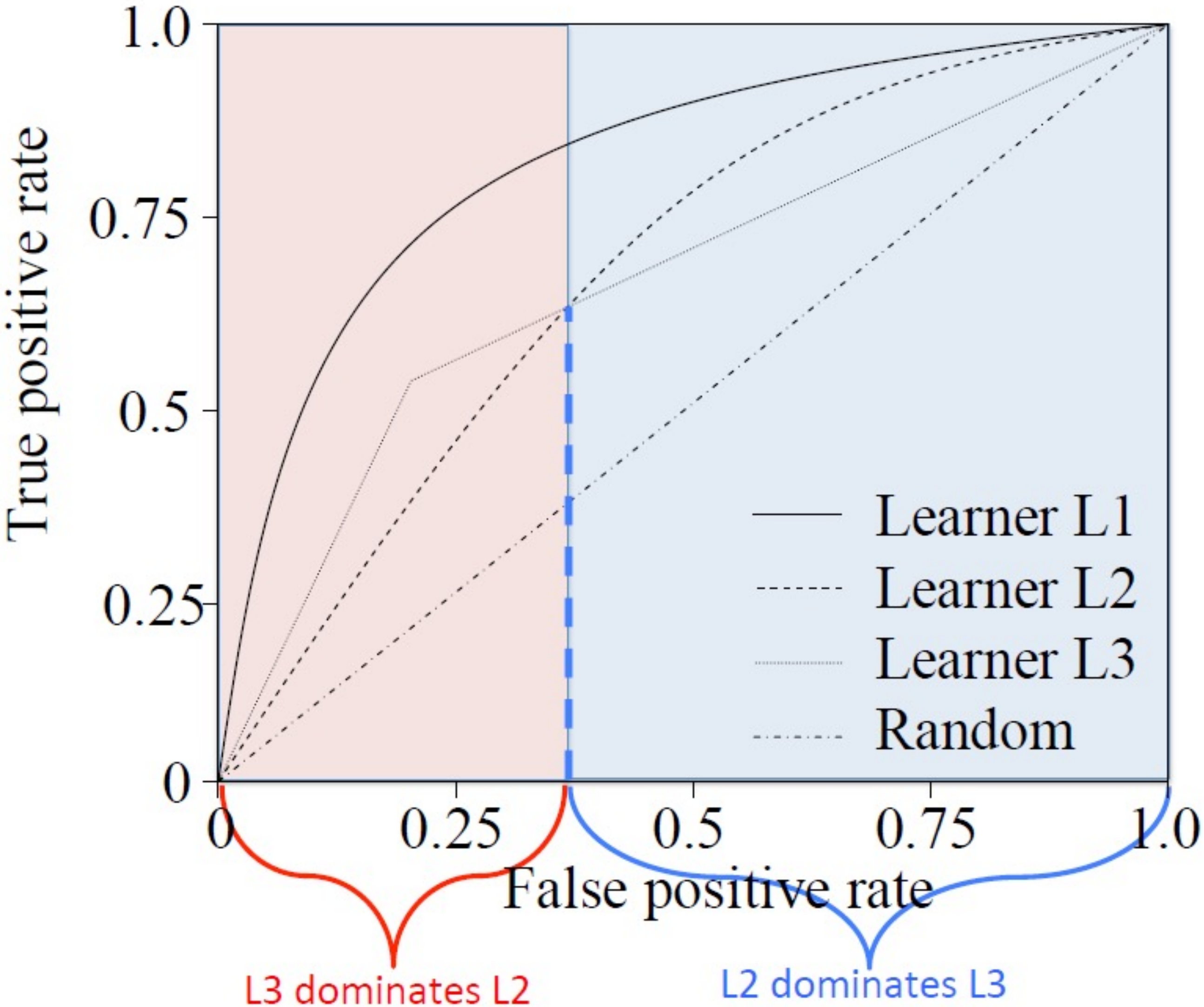
- Number of TPs and FPs depend on threshold T
- As we vary T we get different (TPR, FPR) points



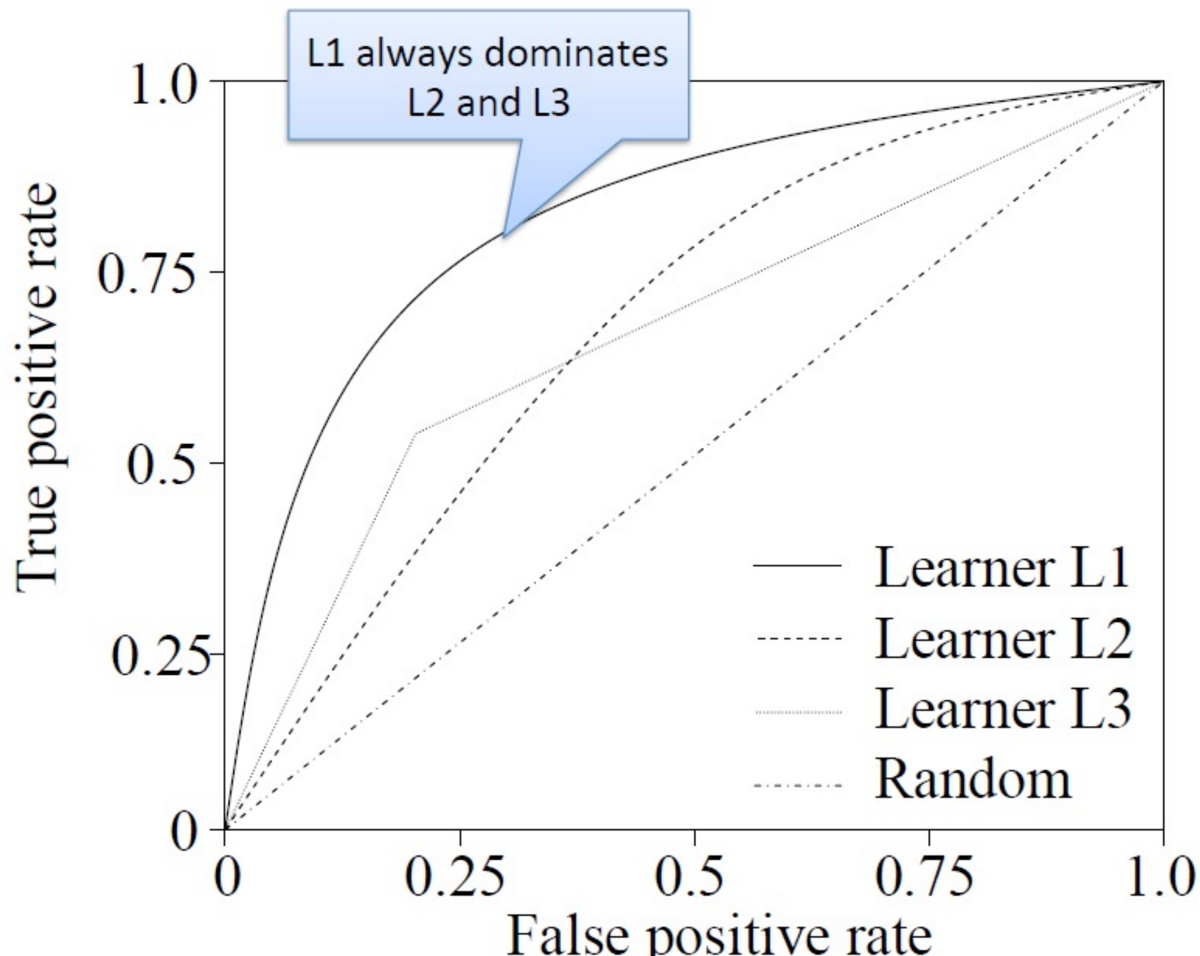
ROC Curve


1.0
0.75
0.5
0.25
0
True positive rate
"Ideal" learner
Random model is always diagonal line in ROC space
Learner L1
Learner L2
Learner L3
Random
0
0.25
0.5
0.75
1.0
False positive rate

ROC Curve

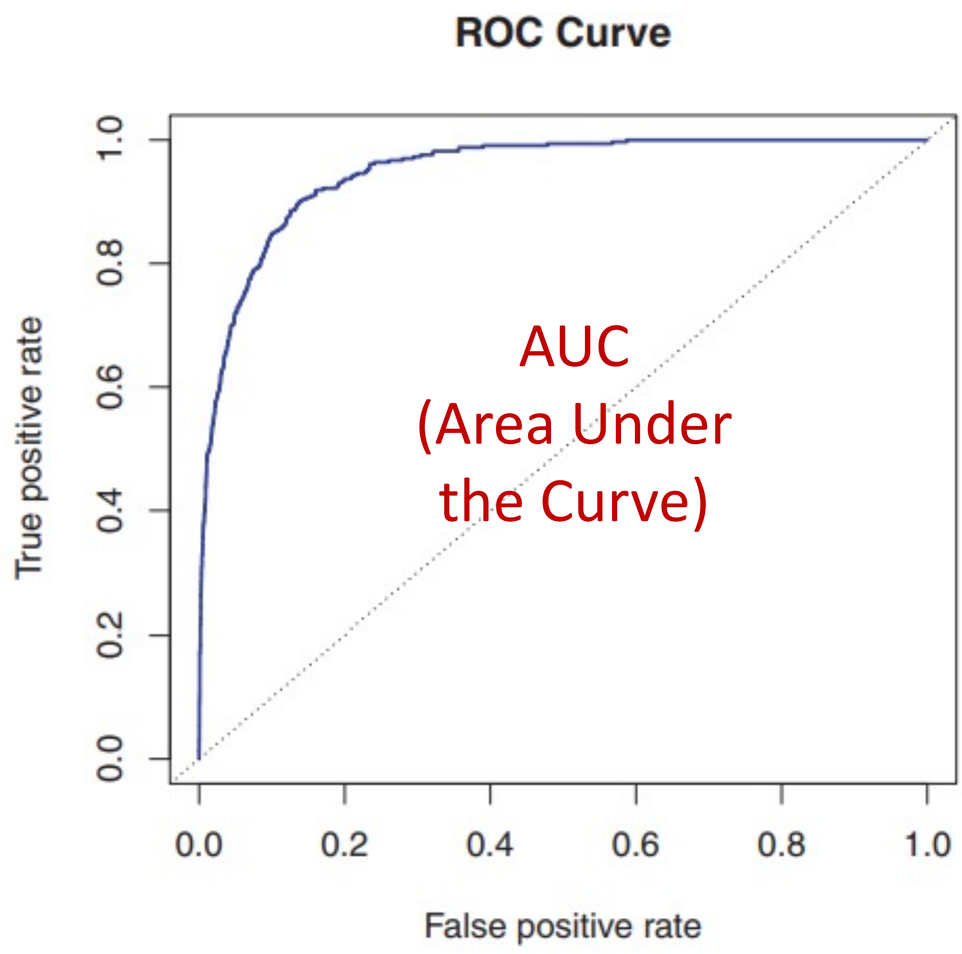

1.0
0.75
0.5
0.25
0
True positive rate
0
0.25
0.5
0.75
1.0
False positive rate
Learner L1
Learner L2
Learner L3
Random
L3 dominates L2
L2 dominates L3

ROC Curve



L1 always dominates L2 and L3
1.0
0.75
0.5
0.25
0
True positive rate
0
0.25
0.5
0.75
1.0
False positive rate
Learner L1
Learner L2
Learner L3
Random

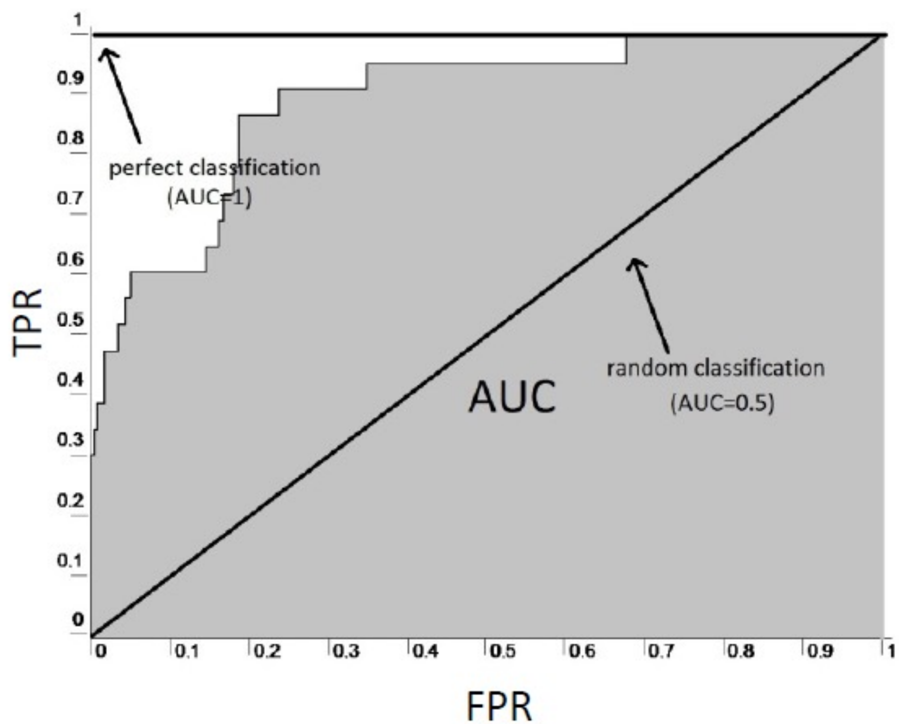
ROC Curves

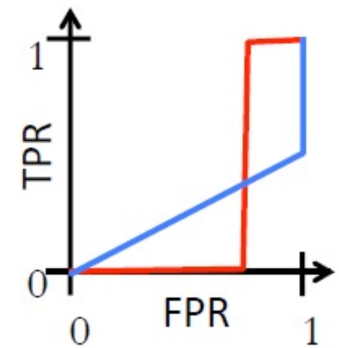


- Another useful metric: Area Under the Curve (AUC)
- The closest to 1, the better!

Area Under the ROC Curve

- Can take area under the ROC curve to summarize performance as a single number
  - Be cautious when you see only AUC reported without a ROC curve; AUC can hide performance issues





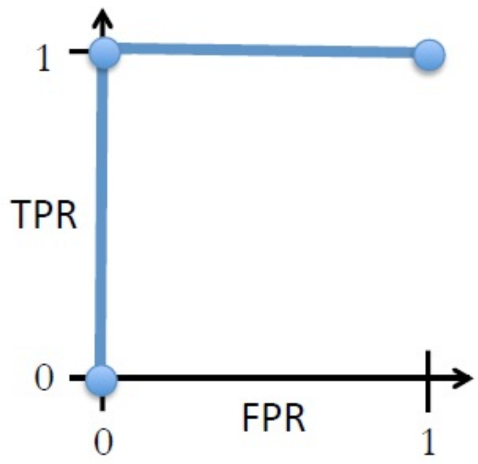
Same AUC, very different performance

ROC Example

i	y_i	$p(y_i = 1 \mid \mathbf{x}_i)$	$h(\mathbf{x_i} \mid T = 0)$	$h(\mathbf{x_i} \mid T = 0.5)$	$h(\mathbf{x_i} \mid T = 1)$
1	1	0.9	1	1	0
2	1	0.8	1	1	0
3	1	0.7	1	1	0
4	1	0.6	1	1	0
5	1	0.5	1	1	0
6	0	0.4	1	0	0
7	0	0.3	1	0	0
8	0	0.2	1	0	0
9	0	0.1	1	0	0
			$TPR =$	$TPR =$	$TPR =$
			$FPR =$	$FPR =$	$FPR =$

ROC Example

i	y_i	$p(y_i = 1 \mid \mathbf{x}_i)$	$h(\mathbf{x_i} \mid \theta = 0)$	$h(\mathbf{x_i} \mid \theta = 0.5)$	$h(\mathbf{x_i} \mid \theta = 1)$
1	1	0.9	1	1	0
2	1	0.8	1	1	0
3	1	0.7	1	1	0
4	1	0.6	1	1	0
5	1	0.5	1	1	0
6	0	0.4	1	0	0
7	0	0.3	1	0	0
8	0	0.2	1	0	0
9	0	0.1	1	0	0
			$TPR = 5/5 = 1$ $FPR = 4/4 = 1$	$TPR = 5/5 = 1$ $FPR = 0/4 = 0$	$TPR = 0/5 = 0$ $FPR = 0/4 = 0$

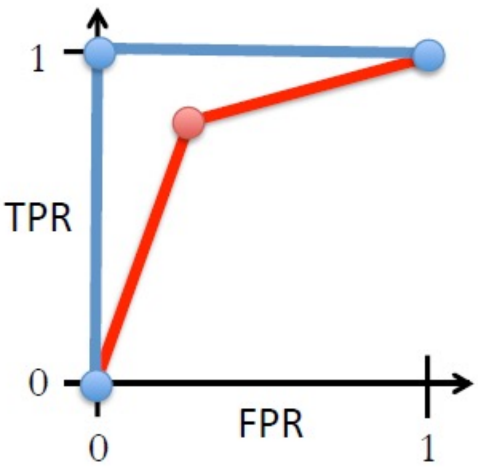


ROC Example

i	y_i	$p(y_i = 1 \mid \mathbf{x}_i)$	$h(\mathbf{x_i} \mid T = 0)$	$h(\mathbf{x_i} \mid T = 0.5)$	$h(\mathbf{x_i} \mid T = 1)$
1	1	0.9	1	1	0
2	1	0.8	1	1	0
3	1	0.7	1	1	0
4	1	0.6	1	1	0
5	1	**0.2**	1	**0**	0
6	0	**0.6**	1	**1**	0
7	0	0.3	1	0	0
8	0	0.2	1	0	0
9	0	0.1	1	0	0
			$TPR =$	$TPR =$	$TPR =$
			$FPR =$	$FPR =$	$FPR =$

ROC Example

i	y_i	$p(y_i = 1 \mid \mathbf{x}_i)$	$h(\mathbf{x_i} \mid \theta = 0)$	$h(\mathbf{x_i} \mid \theta = 0.5)$	$h(\mathbf{x_i} \mid \theta = 1)$
1	1	0.9	1	1	0
2	1	0.8	1	1	0
3	1	0.7	1	1	0
4	1	0.6	1	1	0
5	1	**0.2**	1	**0**	0
6	0	**0.6**	1	**1**	0
7	0	0.3	1	0	0
8	0	0.2	1	0	0
9	0	0.1	1	0	0
			$TPR = 5/5 = 1$ $FPR = 4/4 = 1$	$TPR = 4/5 = 0.8$ $FPR = 1/4 = 0.25$	$TPR = 0/5 = 0$ $FPR = 0/4 = 0$



Classifier Evaluation

- Accuracy and error are not sufficient for classifier evaluation
- Always include multiple metrics
  - Precision and recall for imbalance cases (plot precision-recall curve by varying the thresholds)
  - F1 score is a single metric averaging precision and recall
- Classifiers can be tuned by changing the threshold for prediction
  - Respect application constraints (e.g., low false positives, high recall)
  - Plot ROC curve and report AUC
- Confusion matrix and metrics can be extended to multi-class classification

Midterm Exam Review

What we covered: I

- Bias-Variance Tradeoff
- Linear Regression
  - Closed form simple and multiple Linear Regression
  - Correlation and regression
- Gradient Descent (GD)
  - General algorithm
  - GD for Linear Regression; comparison to closed form
- Non-linear regression: polynomial, spline regression
- Regularization
  - Ridge and Lasso regularization
  - GD for Ridge regression

What we covered: II

- Classifiers
  – Linear vs non-linear classification
  – Generative vs Discriminative models
- kNN classifier
- Logistic regression
  – Maximum Likelihood Estimation (MLE)
  – Cross-entropy objective
  – GD for logistic regression
- Linear Discriminant Analysis (LDA)
- Cross-validation
- Evaluation of classifiers
  – Metrics: precision, recall, F1 score, accuracy, error, confusion matrix
  – ROC curves, AUC

ML Models

- Categorization
  - Is it a linear or non-linear?
  - Is it generative or discriminative?
- For each ML model
  - Understand how training is done
  - Take a small example and train a model
    - E.g., linear regression, LDA
  - Once you have a model know how to evaluate a point and generate a prediction
    - Example: predict probability by logistic regression model or kNN

How to measure performance

- Regression: MSE
- Why we need multiple metrics
  - Accuracy, error
  - Precision, recall
  - Confusion matrix
  - F1 score
  - ROC curves, AUC
- Compute these metrics on small examples

Type I: Conceptual

- Example 1: Describe difference between classification and regression
- Example 2: List two methods for regularization and compare them
- Example 3: Provide advantages and disadvantages, and compare the following:
  - Linear regression with polynomial regression
  - Gradient descent vs closed form solution for linear regression

Type II: Computational

- Example 1: Given a small dataset, train a particular ML model
  - E.g., linear regression, LDA, etc.
  - Evaluate model on some small training and testing data
- Example 2: Compute different metrics: accuracy, precision, recall, etc.

Type III: Case Study

- Example: Consider the problem of predicting a patient’s risk to a disease. The features include demographic information (address, zip code), as well as measurements from blood test results in the last 2 years. Assume there is a datasets including patients with and without the disease.

Describe the process to:

1. Represent the features in a format suitable for ML
2. Describe what models you would use and why

Acknowledgements

- Slides made using resources from:
  - Andrew Ng
  - Eric Eaton
  - David Sontag
- Thanks!