CSDS 440: Machine Learning

Soumya Ray (he/him, sray@case.edu)

Olin 516

Office hours T, Th 11:15-11:45 or by appointment

Zoom recording here

Announcements

- Written 1 due this week
- Quiz 1 next Thursday, in class, 30-45 minutes, closed book/notes
  - Topics: everything up to and including decision trees
  - Remember to review probability and statistics

Recap

- What is the geometry of the tree's decision boundary?
- Tree learners don't need a m___ s___ representation, can represent c_____ concepts, are human i______ and easy to e_____.
- But they have trouble with features that have lots of v_____, features that i_____, and o______ easily.
- What is goal of learning algorithm performance evaluation?
- Given a finite dataset, we want the training set to an algorithm to be as l_____ as possible. We also want the test sets to be i______.
- These goals are achieved by __-___ ___ ___.
- In this procedure, we p_____ the data into f___. Each iteration we use ___ as the train set and _____ as the test set.
- What is leave one out cross validation?
- What is stratified CV?
- What is internal CV?

Today

- Metrics

Contingency Table

		Class according to Target Concept / Oracle (Correct Answer)	
		Positive	Negative
Class according to Learned Classifier (Predicted Answer)	Positive	True Positives (TP)	False Positives (FP) (Type I error)
	Negative	False Negatives (FN) (Type II error)	True Negatives (TN)

Accuracy

- Most commonly used measure for comparing classification algorithms

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Error Rate

- Inverse of Accuracy

$$ErrorRate = \frac{FP + FN}{TP + TN + FP + FN}$$

Weaknesses of Accuracy

- Does not account for:
  - Skewed class distributions
  - Differential misclassification costs
  - Confidence estimates from learning algorithms

Weighted/Balanced Accuracy

- Corrects for skewed class distributions

$$WAcc = \frac{1}{2}\left(\frac{TP}{Allpos} + \frac{TN}{Allneg}\right)$$

$$= \frac{1}{2}\left(\underbrace{\frac{TP}{TP+FN}}_{\text{True Positive Rate}} + \underbrace{\frac{TN}{TN+FP}}_{\text{True Negative Rate}}\right)$$

Measuring one class

- Often, just a single class is “interesting”
  - Call this the “positive” class

	Positive	Negative
Positive	True Positives (TP)	False Positives (FP) (Type I error)
Negative	False Negatives (FN) (Type II error)	~~True Negatives (TN)~~

Precision

- Of the examples the learner predicted positive, how many were actually positive?

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

Recall/TP rate/Sensitivity

- Of the examples that were actually positive, how many did the learner predict correctly?

$$Recall = \frac{TP}{TP+FN} = \frac{TP}{Allpos}$$

Specificity/TN rate

- Counterpart of recall for the negative class

$$Specificity = \frac{TN}{TN + FP} = \frac{TN}{Allneg}$$

- So:

$$WAcc = \frac{1}{2}\left(Sensitivity + Specificity\right)$$

F_1 score

- Combines precision and recall into a single measure, giving each equal weight

$$\frac{1}{F_1} = \frac{1}{2}\left(\frac{1}{Precision} + \frac{1}{Recall}\right)$$

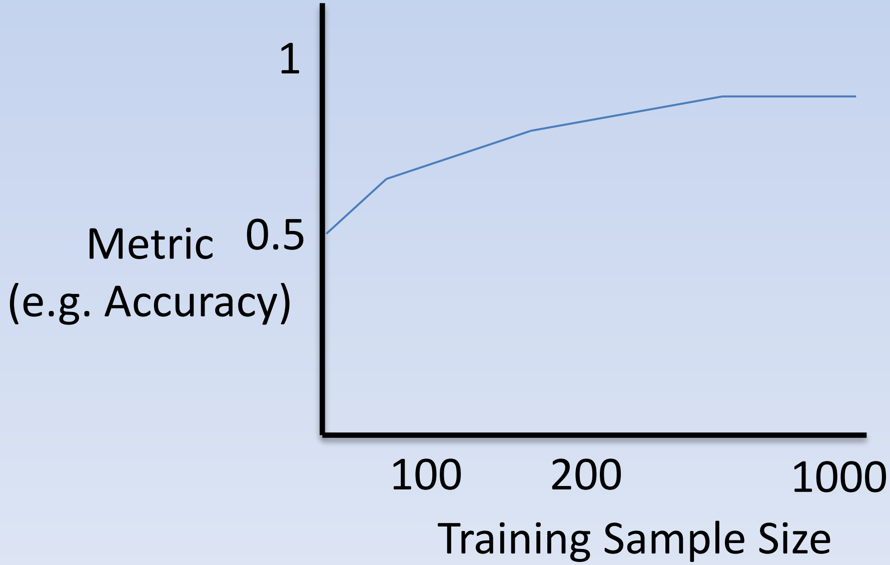
$$F_1 = \frac{2}{\dfrac{1}{Precision} + \dfrac{1}{Recall}}$$

Beyond point estimates

- Everything above is a "point estimate"
- Because they will be computed on the basis of a sample, we can also compute variance estimates for each quantity
- Important to show "stability" of solutions, and when comparing across algorithms (later)

Learning Curves

- Often useful to plot each metric as a function of training sample size
- Provides insight into how many examples the algorithm needs to become effective



Metrics with Confidence Measures

- Many learning algorithms can produce models that can provide estimates of how *confident* they are about a prediction
- Example: Pruned Decision Trees

Metrics with Confidence Measures

	True Class	Confidence On +
Example 1	+	0.9
Example 2	−	0.8
Example 3	+	0.4
Example 4	−	0.3

- We can create *multiple classifiers* by *thresholding* the confidence

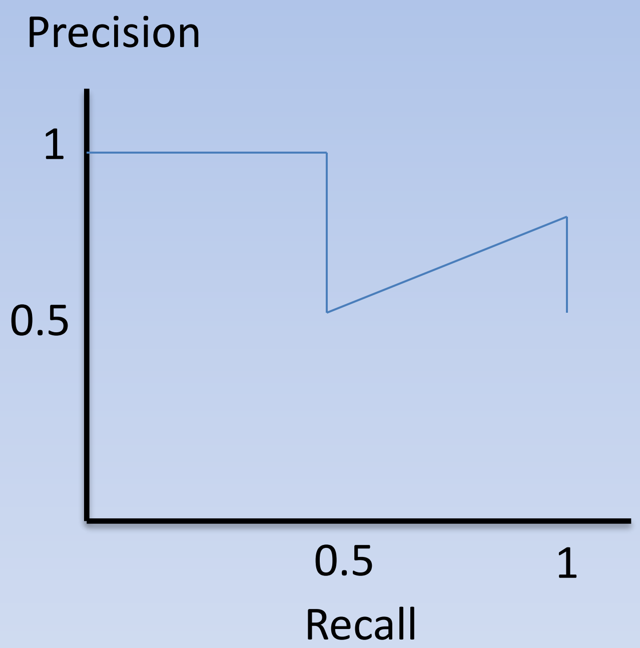
- In this case, we can plot Precision-Recall (PR) and Receiver Operating Characteristic (ROC) graphs tracking *all* of the classifiers

Precision-Recall graphs

	True Class	Confidence On +	Recall (x axis)	Precision (y axis)
Example 1	+	0.9		
Example 2	–	0.8		
Example 3	+	0.4		
Example 4	–	0.3		

Precision-Recall graphs

	True Class	Confidence On +	Recall (x axis)	Precision (y axis)
Example 1	+	0.9	0.5	1
Example 2	–	0.8	0.5	0.5
Example 3	+	0.4	1	0.67
Example 4	–	0.3	1	0.5

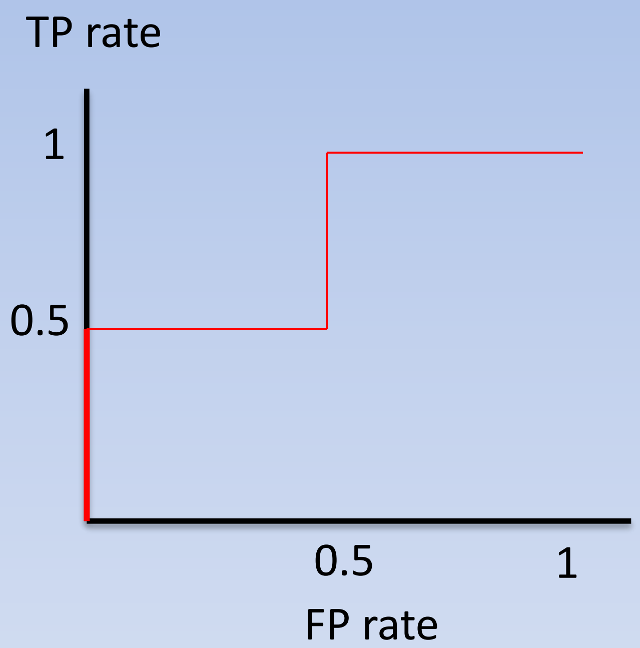


ROC graphs

	True Class	Confidence On +	FP Rate (1-Spec.) (x axis)	Sens./Recall (y axis)
Example 1	+	0.9		
Example 2	–	0.8		
Example 3	+	0.4		
Example 4	–	0.3		

ROC graphs

	True Class	Confidence On +	FP Rate (x axis)	Sens./Recall (y axis)
Example 1	+	0.9	0	0.5
Example 2	–	0.8	0.5	0.5
Example 3	+	0.4	0.5	1
Example 4	–	0.3	1	1



Properties of ROC graphs

- Random guessing is a diagonal line
  - Also majority class classifier
  - If your classifier is any good its ROC must lie above the diagonal
- Monotonically increasing
- Often use “AUC”/ “AROC” as comparison statistic (later)
- Can be misleading if class distribution is too skewed (use PR graphs instead)