Evaluation

Dan Roth
danroth@seas.upenn.edu|http://www.cis.upenn.edu/~danroth/|461C, 3401 Walnut

Slides were created by Dan Roth (for CIS519/419 at Penn or CS446 at UIUC), Eric Eaton for CIS519/419 at Penn, or from other authors who have made their ML slides available.

Administration

- HW1 is out. Due on October 7.
  - Please start working on it.
  - Go to office hours and recitations.
- Recall our late policy
  - 4 days
- Piazza
  - Be active on Piazza
  - We will reward highly active students

Metrics
Methodologies
Statistical Significance

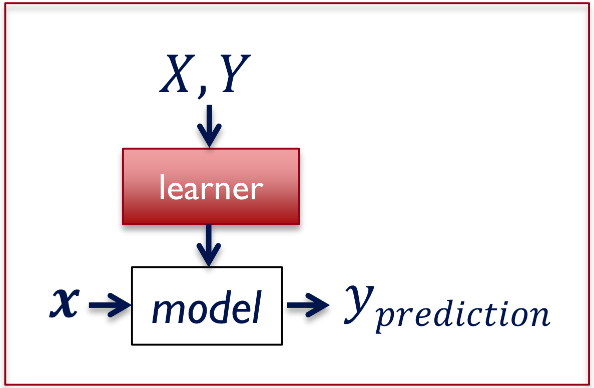
Flow of Batch Machine Learning

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

- Assumes each $x_i \sim D(X)$ with $y_i = f_{target}(\boldsymbol{x}_i)$

Train the model:

$model \leftarrow classifier.train(X, Y)$



Apply the model to new data:

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

$y_{prediction} \leftarrow model.predict(\boldsymbol{x})$

Key questions:

How to determine the quality of the model?

(i) measuring performance

(ii) understanding the significance of the results (is it better the other models?)

Metrics

- We train on our training data Train = $\{x_i, y_i\}_{1,m}$
- We test on Test data.
- We often set aside part of the training data as a development set, especially when the algorithms require tuning.
  - In the HW we asked you to present results also on the Training; why?
- When we deal with binary classification we often measure performance simply using Accuracy:

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

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

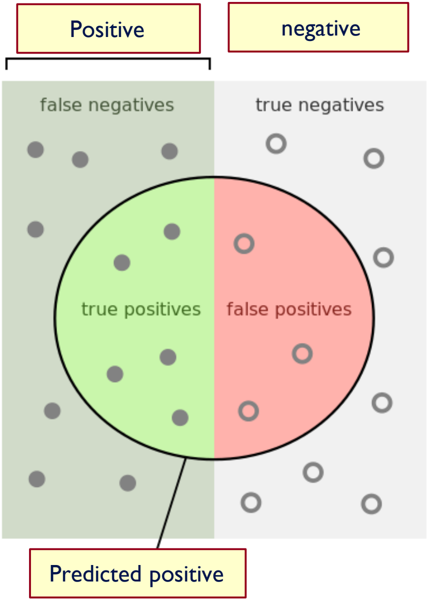
- Any possible problems with it?

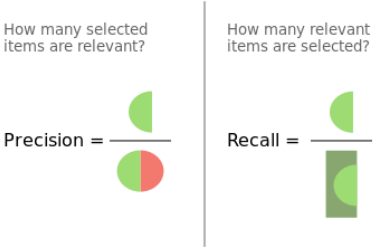
Alternative Metrics

- If the Binary classification problem is biased
  - In many problems most examples are negative
- Or, in multiclass classification
  - The distribution over labels is often non-uniform
- Simple accuracy is not a useful metric.
  - Often we resort to task specific metrics
- However one important example that is being used often involves Recall and Precision

- **Recall:** $$\frac{\#\,(\text{positive identified = true positives})}{\#\,(\text{all positive})}$$

- **Precision:** $$\frac{\#\,(\text{positive identified = true positives})}{\#\,(\text{predicted positive})}$$

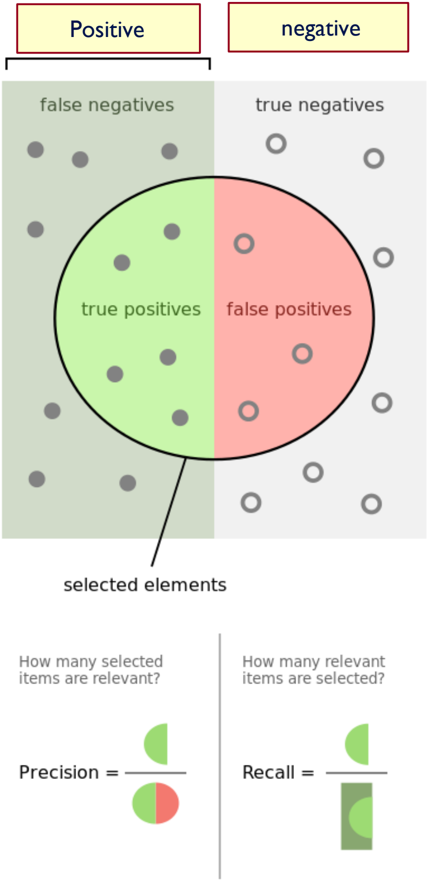




Example

- 100 examples, 5% are positive.

- Just say NO: your accuracy is 95%
  - Recall = precision = 0
- Predict 4+, 96-; 2 of the +s are indeed positive
  - Recall:2/5; Precision: 2/4



- **Recall:** $\dfrac{\textbf{\# (positive identified = true positives)}}{\textbf{\# (all positive)}}$

- **Precision:** $\dfrac{\textbf{\# (positive identified = true positives)}}{\textbf{\# (predicted positive)}}$

Confusion Matrix

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

Actual Class \ Predicted Class	Yes	No
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)

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

Probability that a randomly selected positive prediction is indeed positive

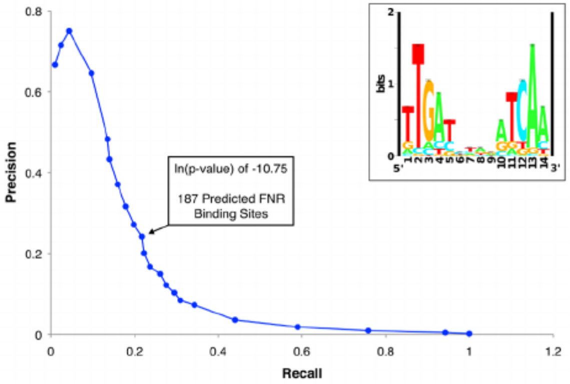
$$\text{recall} = \frac{TP}{TP + FN}$$

Probability that a randomly selected positive is identified

The notion of a confusion matrix can be usefully extended to the multiclass case (i,j) cell indicate how many of the i-labeled examples were predicted to be j

Relevant Metrics

- It makes sense to consider Recall and Precision together or combine them into a single metric.
- Recall-Precision Curve:



- F-Measure:
  - A measure that combines precision and recall is the harmonic mean of precision and recall.

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

  - F1 is the most commonly used metric.

Comparing Classifiers

Say we have two classifiers, *C1* and *C2*, and want to choose the best one to use for future predictions

Can we use training accuracy to choose between them?

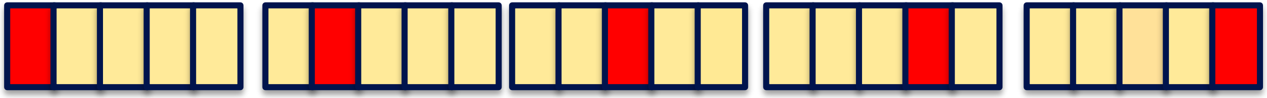
- No!

- What about accuracy on test data?
- Yes, but…
  - We basically want to look at more than a single number; gather some statistical evidence.

N-fold cross validation

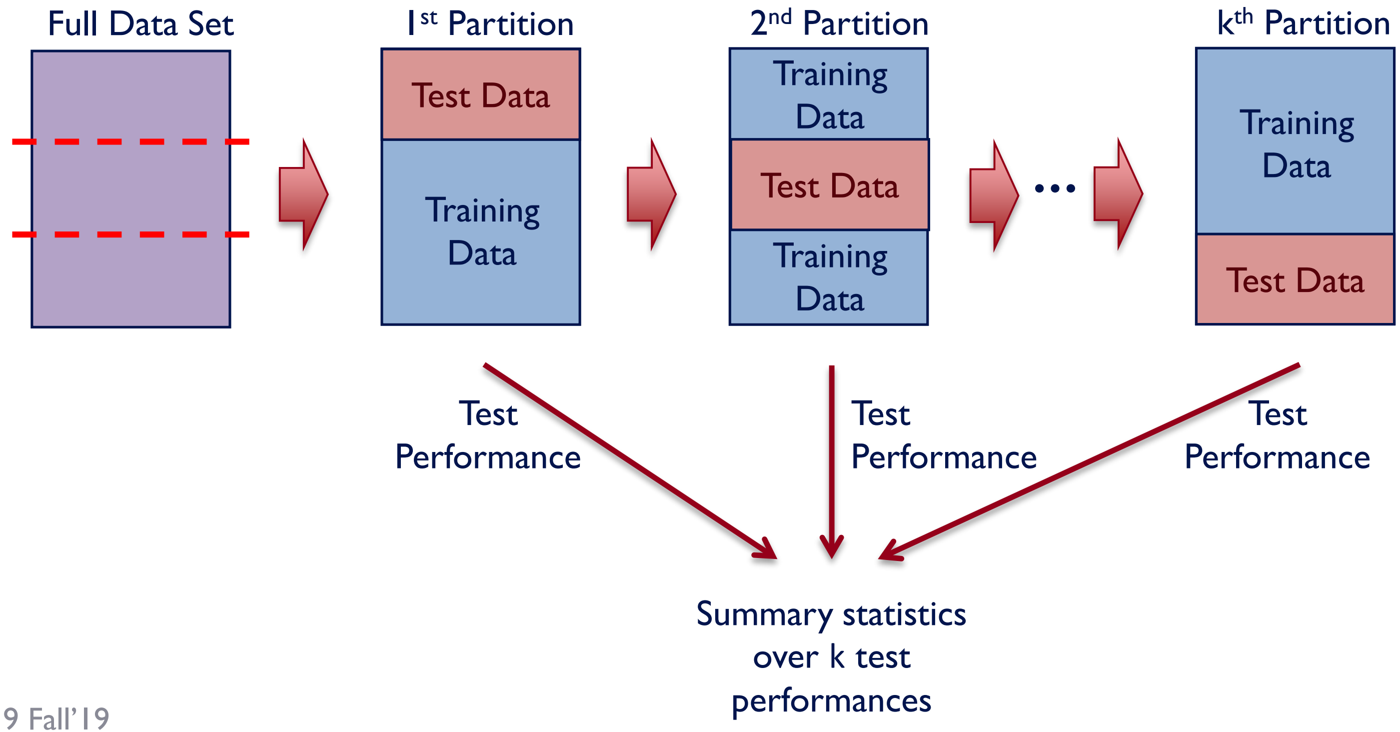
- Instead of a single test-training split:

- Split data into N equal-sized parts



- Train and test N different classifiers
- Report average accuracy and standard deviation of the accuracy

Example 3-Fold CV



More on Cross-Validation

- Cross-validation generates an approximate estimate of how well the classifier will do on "unseen" data
  - As $k \to n$, the model becomes more accurate (more training data)
  - ...but, CV becomes more computationally expensive
  - Choosing $k < n$ is a compromise. k=5 is often used.
  - k=n is called "leave-one-out";

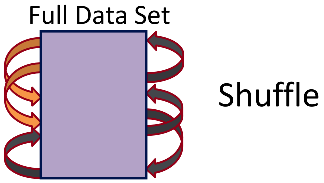
- Averaging over different partitions is more robust than just a single train/validate partition of the data

- It is an even better idea to do CV repeatedly!

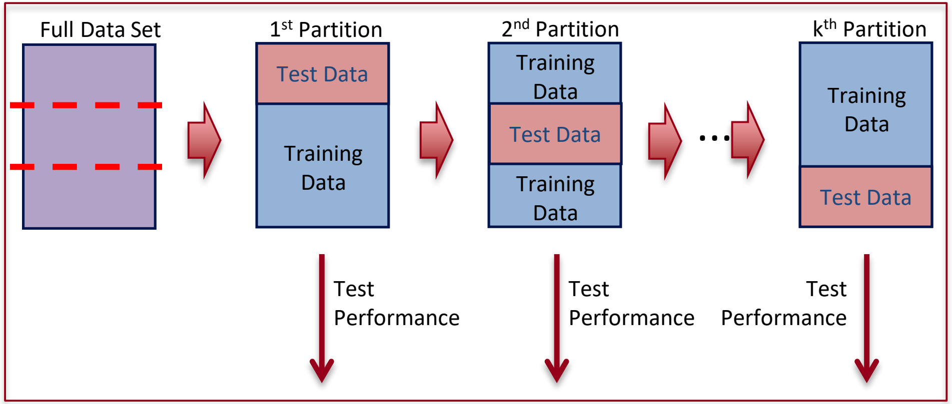
Multiple Trials of k-Fold CV

1.) Loop for t trials:

a.) Randomize Data Set



b.) Perform k-fold CV

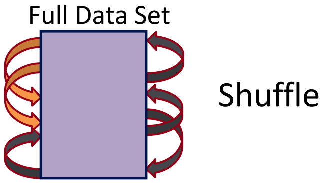


2.) Compute statistics over $t \times k$ test performances

Comparing Multiple Classifiers

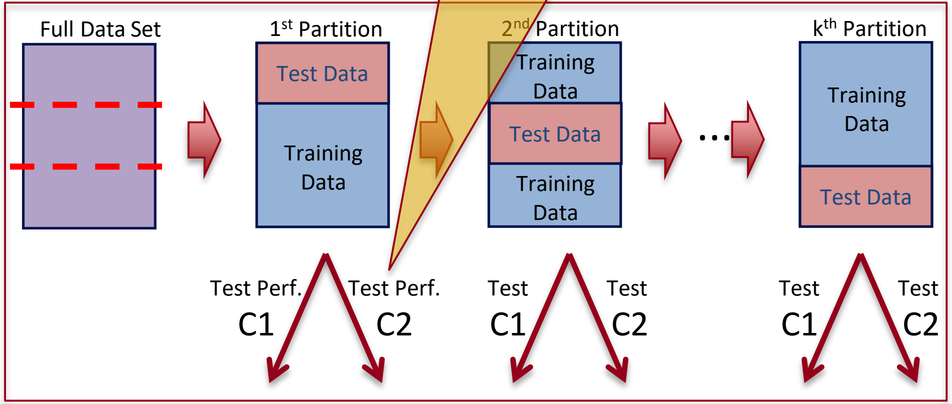
1.) Loop for t trials:

a.) Randomize Data Set



Test each candidate learner on same training/testing splits

b.) Perform k-fold CV



2.) Compute statistics over $t \times k$ test performances

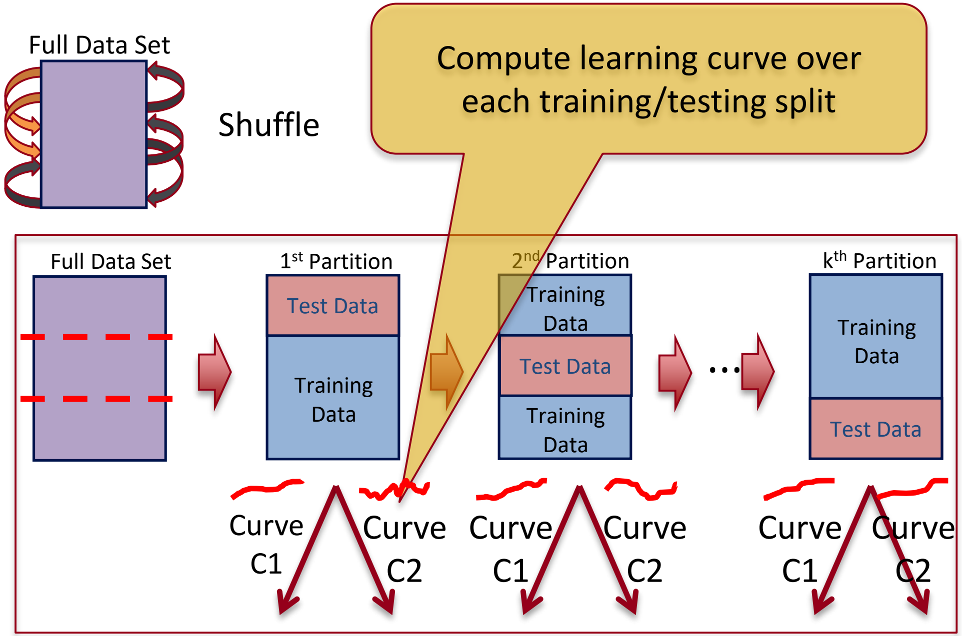
Allows us to do paired summary statistics (e.g., paired t-test)

Building Learning Curves

1.) Loop for t trials:

a.) Randomize Data Set

b.) Perform k-fold CV



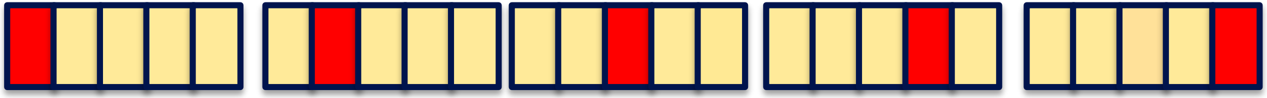
2.) Compute statistics over $t \times k$ learning curves

Evaluation: significance tests

- You have two different classifiers, A and B
- You train and test them on the same data set using N-fold cross-validation
- For the n-th fold:

  accuracy(A, n), accuracy(B, n)

  p_n = accuracy(A, n) - accuracy(B, n)
- Is the difference between A and B's accuracies significant?



Hypothesis testing

- [Next we are introducing a methodology for answering the question: can we distinguish two models? Which one is better?]
- You want to show that hypothesis H is true, based on your data
  - (e.g. H = “classifier A and B are different”)
- Define a null hypothesis H_0
  - (H_0 is the contrary of what you want to show)
- H_0 defines a distribution $P(m\ /H_0)$ over some statistic
  - e.g. a distribution over the difference in accuracy between A and B
- Can you refute (reject) H_0?

Rejecting H_0

- H_0 defines a distribution $P(M\ /H_0)$ over some statistic M
  - (e.g. M= the difference in accuracy between A and B)
- Select a significance value S
  - (e.g. 0.05, 0.01, etc.)
  - You can only reject H_0 if $P(m\ /H_0) \leq S$
- Compute the test statistic m from your data
  - e.g. the average difference in accuracy over your N folds
- Compute $P(m\ /H_0)$
- Refute H_0 with $p \leq S$ if $P(m\ /H_0) \leq S$

Paired t-test

- A paired t-test is used to compare two population means where you have two samples in which observations in one sample can be paired with observations in the other sample.

Paired t-test

- Null hypothesis (H_0; to be refuted):
  - There is no difference between A and B, i.e. the expected accuracies of A and B are the same
- That is, the expected difference (over all possible data sets) between their accuracies is 0:

  H_0: $E[p_D] = 0$
- We don't know the true $E[p_D]$
- N-fold cross-validation gives us N samples of p_D

Paired t-test

- Null hypothesis H_0: $E[diff_D] = \mu = 0$
- *m:* our estimate of μ based on *N* samples of $diff_D$

  $m = 1/N \sum_n diff_n$

- The estimated variance S^2:

  $S^2 = 1/(N-1) \sum_{1,N} (diff_n - m)^2$

- Accept Null hypothesis at significance level *a* if the following statistic lies in $(-t_{a/2, N-1}, +t_{a/2, N-1})$

$$\frac{\sqrt{N}m}{S} \sim t_{N-1}$$

Procedure for carrying out Paired t-test

- Calculate the difference between the two observations on each pair.
- Calculate the mean difference
- Calculate the standard deviation of the differences
- Calculate the error of the mean difference
- Calculate the t-statistic

McNemar's Test

The test is often used for the situation where one tests for the presence (1) or absence (0) of something and variable A is the state at the first observation (i.e., pretest) and variable B is the state at the second observation (i.e., post-test).

McNemar's Test

- An alternative to Cross Validation, when the test can be run only once, but you have access to predictions on individual examples.
- Divide the sample S into a training set R and a test set T.
- Train algorithms A and B on R, yielding classifiers A, B
- Record how each example in T is classified and compute the number of

Examples misclassified by both A and B N_{00}	Examples misclassified by A but not B N_{01}
Examples misclassified by B but not A N_{10}	Examples misclassified by neither A nor B N_{11}

where N is the total number of examples in the test set T

$$N_{00} + N_{10} + N_{01} + N_{11} = N$$

McNemar's Test

- The hypothesis: the two learning algorithms have the same error rate on a randomly drawn sample. That is, we expect that

$$N_{10} = N_{01}$$

- The statistics we use to measure deviation from the expected counts:

$$\frac{(|N_{01} - N_{10}| - 1)^2}{N_{01} + N_{10}}$$

- This statistics is distributed (approximately) as t-distribution with 1 degree of freedom