Evaluating classifier performance

Lecture 13

We are familiar with one classifier

- Exercise: Naïve Bayes classifier

A1	A2	Class
T	T	Yes
T	T	Yes
T	F	No
F	F	Yes
F	T	No
F	T	No
F	F	No
T	F	Yes
F	T	No

Now we can use it to classify: [T,F]

How do we know if this classifier is good?

Intuition → numeric evaluation

- How to measure the quality of the classifier
- How to statistically quantify the confidence
- How to compare the quality of two different classifiers

Natural performance measure: *error rate*

- *Success*: instance's class is predicted correctly
- *Error*: instance's class is predicted incorrectly
- *Error rate*: proportion of errors made over the whole set of test instances

A1	A2	Actual class	Predicted by classifier
T	T	Yes	Yes
T	T	Yes	Yes
T	F	No	Yes
F	F	Yes	No
F	T	No	No
F	T	No	No
F	F	No	Yes
T	F	Yes	Yes
F	T	No	No

Example:
Error rate: 3/9
Success rate: 6/9

Resubstitution (training) error

- Training error - error rate obtained from training data

  Training error is (hopelessly) optimistic!

Error for a test set

- *Test set*: independent **labeled** instances that played no part in formation of classifier
  - Assumption: both training data and test data are representative samples of the underlying problem

- Generally, the larger the training data, the better the classifier

- The larger the test data the more accurate the error estimate

Where to get the test set?

- Simple solution if lots of (labeled) data is available:
  - Split data into training and test set

- However: (labeled) data is usually limited
  - More sophisticated techniques need to be used
  - We need to make the most from the available data

Holdout

- *Holdout procedure*: method of splitting original data into training and test set

- Holdout reserves a certain amount for testing and uses the remainder for training
  - Usually: 1/3 for testing, the rest for training

Repeated holdout

- Holdout estimate can be made more reliable by repeating the process with different subsamples
  - In each iteration, a certain proportion is randomly selected for training (possibly with stratification)
  - The error rates on the different iterations are averaged to yield an overall error rate

- This is called the *repeated holdout* method

Cross-validation

- *Cross-validation* avoids overlapping test sets
  - **First step:** split data into *k* subsets of equal size
  - **Second step:** use each subset in turn for testing, the remainder for training

k-fold cross-validation

Bootstrap

- Cross-Validation uses sampling *without replacement*
  - The same instance, once selected, can not be selected again for a particular training/test set
- The *bootstrap* uses sampling *with replacement* to form the training set:
  - Randomly sample a dataset of *n* instances *n* times *with replacement* to form a new dataset of *n* instances
  - Use this data as the training set
  - Use the instances from the original dataset that don't occur in the new training set for testing
- Also called the *0.632 bootstrap (Why?)*

The 0.632 bootstrap

- A particular instance has a probability of 1–1/*n* of *not* being picked
- Thus, its probability of ending up in the test data is:

$$\left(1-\frac{1}{n}\right)^n \approx e^{-1} = 0.368$$

- This means the training data will contain approximately 63.2% of the instances

Estimating error with bootstrap

- The error estimate will be very pessimistic: after all we trained classifier on just ~63% of the instances
- Therefore, combine it with the optimistic training error:

$$err = 0.632 \cdot e_{\text{test instances}} + 0.368 \cdot e_{\text{training instances}}$$

The training error gets less weight than the error on the test data

- Repeat process several times with different replacement samples and average the results
- This is the best way of estimating performance for very small datasets

Statistics!

CONFIDENCE INTERVAL FOR ERROR (SUCCESS) RATE

Predicting true performance

- Assume the estimated from a test set success rate of a classifier is 75%. How close is this to the true success rate on an unknown future population?

- We can treat this as a regular proportion and compute confidence interval of success in the population at a given level of confidence

Recap: confidence intervals

- Estimate standard deviation of sample means distribution by computing standard deviation of a sample *s*
- For confidence interval C, find z-value for C/2+0.5 (from the z-table)

- The margin of error E from each side:

$$E = z\frac{s}{\sqrt{n}} \quad \text{or} \quad E = t\frac{s}{\sqrt{n}}$$

Depending on a size of a sample (size of a test set)

- Real μ is within:

$$\mu = \bar{x} \pm E$$

Example

- The test set contained 100 records. For 75 of them the class was predicted correctly
- Sample mean $\bar{x} = \hat{p} = 0.75$
- Sample standard deviation is $s = \sqrt{\hat{p}(1-\hat{p})} = \sqrt{0.75 * 0.25} = 0.43$

LOC (C)	z_c
0.80	1.28
0.85	1.44
0.90	1.645
0.95	1.96
0.98	2.33
0.99	2.575

- The margin of error *E*:

$$E = z_{C=0.80}\frac{s}{\sqrt{n}} = \frac{1.28 * 0.43}{10} = 0.055$$

- Real μ is within:
  $\mu = 0.75 \pm 0.055$

The predicted interval for the success rate of this classifier is: [0.695, 0.805]

From 69.5 to 80.5% accuracy with confidence 80%