Evaluation and Writing

Basic Measures

How do we determine whether a hypothesis is any good or not?

- Predictive ability! How well does it perform on new data?

Classification: count the number of misclassified points
Regression: mean squared error, absolute value error …

No single ML algorithm is optimal.

Must find suitable methods for comparing ML algorithms.

Cross-validation

- Infinite data is best, but…
- N (N=10) Fold cross validation
  - Create N folds or subsets from the training data (approximately equally distributed with approximately the same number of instances).
  - Build N models, each with a different set of N-1 folds, and evaluate each model on the remaining fold
  - Error estimate is average error over all N models

Training, Testing and Validation

• **What data do we use for testing?**
– Performance on training data is not sufficient:

Why not?

If we have unlimited data …
– Randomly sample enough to construct hypothesis
  • too little: bad hypothesis
  • too much: slow!
  • how much? use a Learning Curve …
– Randomly sample more to test hypothesis
– If you change hypothesis, re-sample more data

Training, Testing and Validation

• **Normally, data set is limited …**
– Need to divide into statistically identical subsets

• **Good practice**
– Randomly sample a validation set and hide it
– Divide the remainder into training sets and test sets
– Training sets: used to construct hypotheses
– Test sets: used to evaluate/compare hypotheses
– When you're happy you've found the best algorithm and parameter settings you can, construct hypothesis using training/test data together
– Finally, unlock the drawer with the validation set; evaluate (objectively) using it

Learning Curves

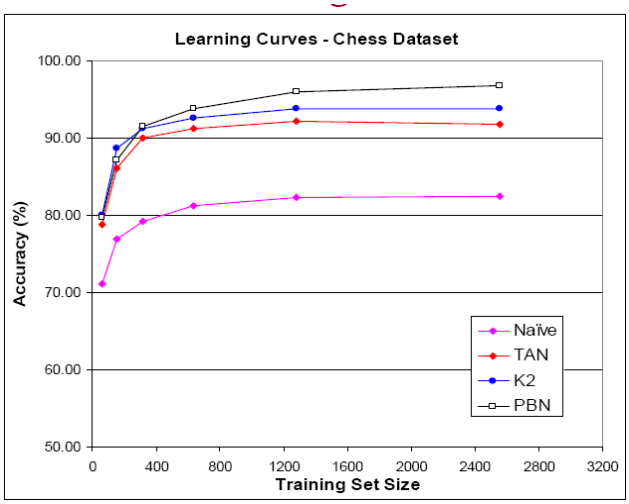
• **Repeat Training/Testing for various percentages of data from 0% to 100%:**

- Train on X% of data (randomly sampled)
- Test on the rest
- Average the results at X%

• **Popular because…**

– Useful for comparing techniques
– Insight into how much data is sufficient for a technique
– Indicative of error in the limit

Learning Curves



Cross-validation

• **This is useful if the data is somewhat scarce …**

First:
– Divide into N 'folds': subsets of near-equal size
– Often use N=10: want fair no. of folds, with >30 examples each
– Stratified: have similar distribution of data in each fold
(eg same proportion of data from each class)

Then:
For each fold i from 1 to N
- keep fold i for testing
- construct a hypothesis using the rest
- test on fold i

Result:
• We constructed (and discarded) N hypotheses, using all data N–1 times for training
• All the data was used as testing data once

• **Extreme case: Leave-One-Out Cross-Validation**

Confusion Matrix

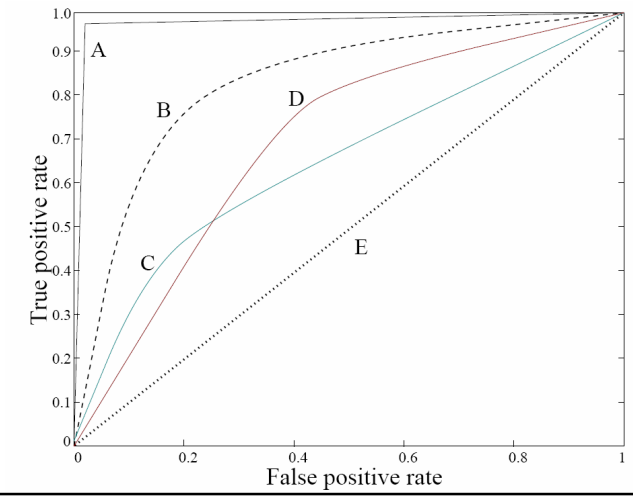
A confusion matrix (Kohavi and Provost, 1998) contains information about actual and predicted classifications done by a classification system.
For a two class classifier,

- *a* = number of **correct** predictions for -1, **True negatives**
- *b* = number of **incorrect** predictions for +1, **False positives**
- *c* = number of **incorrect** of predictions for -1, **False negatives**
- *d* = number of **correct** predictions for +1. **True positives**

	Negative	Positive
Negative	a	b
Positive	c	d

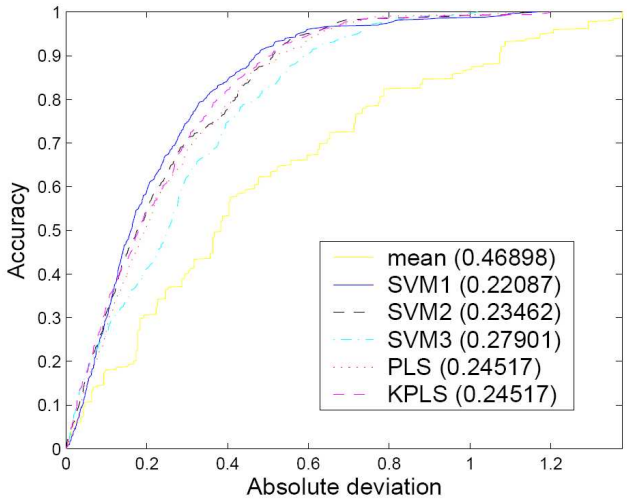
ROC curves

Receiver Operating Characteristic curve – vary the threshold or probability used to discriminate between classes for that function.
AUC (area under the curve):
AUC → 1 … method performs well, AUC → 0.5 … method no better than random



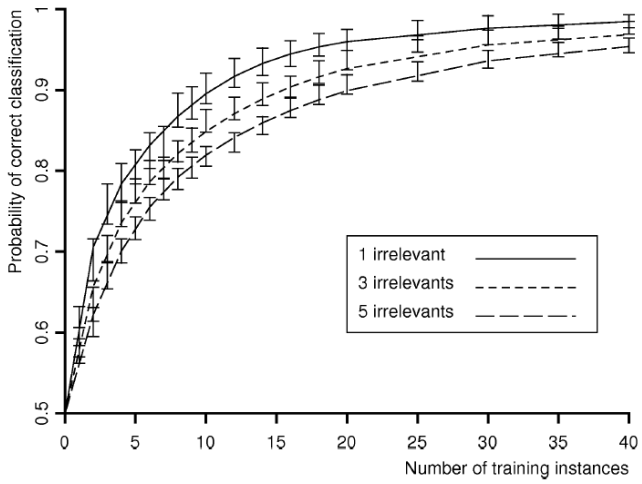
REC curves

Regression Error Characteristic curve – plot the error tolerance on the x-axis versus the percentage of points predicted within the tolerance on the y-axis. Larger AUC implies better method



Convergence curves

How fast does a learning algorithm converge to its optimal performance?



How many datasets?

Artificial data – can be used to demonstrate theoretical advantages of your method.

> ???Best ways to generate artificial data???
> Must have some noise/randomness to simulate real data which is never perfect (especially in Biology!)

Otherwise, use as many real datasets as possible!

Crafting papers on ML

•State the goals of the research
•Specify the performance and learning tasks
•Describe the representation and organization
•Explain both the performance and learning components of the system
•Evaluate the approach to learning
•Relate the approach to other methods
•State the limitations of the approach and suggest directions for future research.

Experimental Evaluation

Use real data to validate your model. *(artificially generated data can help highlight the advantages of your model, but is not usually sufficient)*

Clearly state dependent and independent variables.

ROC curves
– to show algorithm performance on new data

Learning curves
– to show algorithm convergence or computational cost

Use statistical tests to ensure that the differences between your algorithm and those you are comparing it to are statistically significant.

Alternate Evaluation Methods

- Running your algorithm on different tasks to see how it performs under different conditions
- Formal analysis of learning algorithms or tasks – solid proofs needed here!

Things to keep in mind when WRITING

- What kind of audience are you targeting?
- Where would you like to submit?

- Title and abstract
- Partitioning your text
- Continuity and flow
- Figures and Tables
- Describing your system
- Terminology and Notation
- Concluding remarks

- Highlight your NOVEL CONTRIBUTION(S)

Performance Metrics

With reference to Caruana and Niculescu-Mizil
'**Data Mining in Metric Space: An Empirical Analysis of Supervised Learning Performance Criteria**'
http://www.cs.cornell.edu/~caruana/perfs.kdd04.revised.rev1.pdf

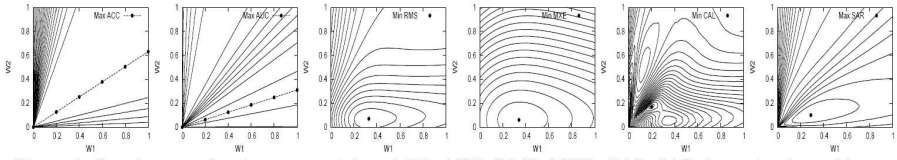


Figure 1: Level curves for six error metrics: ACC, AUC, RMS, MXE, CAL, SAR for a simple problem.

ACC – Accuracy
AUC – area under curve
RMS – root mean square error
MXE – mean cross entropy
CAL – probability calibration
SAR – square error accuracy + ROC
ROC – Receiving Operator Curve

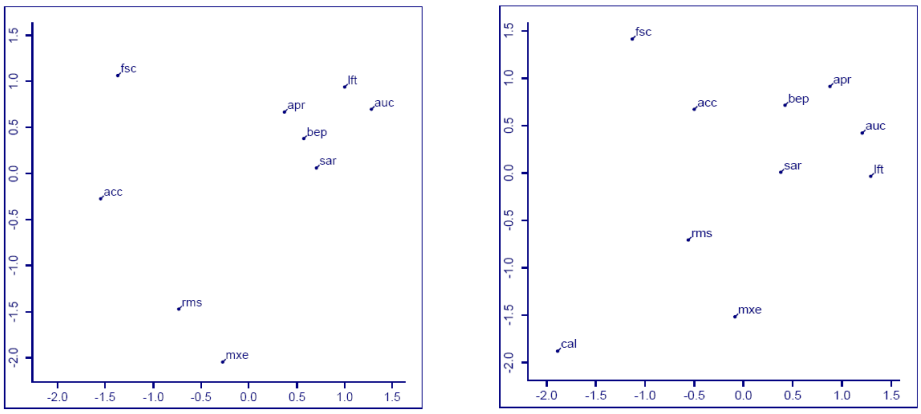


Figure 3 shows two MDS plots for the metrics that result when dimensionality is reduced to two dimensions. The plot on the left is MDS using normalized scores when CAL is excluded. The plot on the right is MDS using standard deviation scaled scores when CAL is included.

$$RMSE = \sqrt{\frac{1}{N}\sum(Pred(C) - True(C))^2}$$

accuracy: probably the most widely used performance metric in Machine Learning. It is defined as the proportion of correct predictions the classifier makes relative to the size of the dataset. If a classifier has continuous

ACC – Accuracy
AUC – area under curve
RMS – root mean square error
MXE – mean cross entropy
CAL – probability calibration
SAR – square error accuracy + ROC
ROC – Receiving Operator Curve

$$MXE = -\frac{1}{N}\sum(True(C) * ln(Pred(C)) + (1 - True(C)) * ln(1 - Pred(C)))$$

CAL is based on reliability diagrams [2]. It is calculated as follows: order all cases by their predicted value, and put cases 1-100 in the same bin. Calculate the percentage of these cases that are true positives. This approximates the true probability that these cases are positive. Then calculate the mean prediction for these cases. The absolute value of the difference between the observed frequency and the mean prediction is the calibration error for this bin. Now take cases 2-101, 3-102, ... and compute the errors in the same way for each of these bins. CAL is the mean of these binned calibration errors.

P. Agius – L9, Spring 2008