



Practical Machine Learning

CS 472
Veselin Stoyanov
October 18, 2004

10/18/2004

1




Motivation

- Machine learning has practical applications
- When building real world applications we need well performing techniques
- In today's lecture:
 - Learn how to compare different ML methods (evaluation)
 - Three of the better performing techniques (models)
 - Results of a large scale empirical comparison of models

10/18/2004

2




Evaluation, Evaluation, Evaluation

- Why evaluation?
- Given two or more models we need to know which performs better
 - In the real world, we want to use the better performing model for our applications
 - In the scientific world, we want to compare new learning models to the ones that already exist

10/18/2004

3



Evaluation – basic assumptions

- Consider the straightforward case:
 - Data point \rightarrow 0/1 label
 - E.g., given an email \rightarrow spam or not?
- Given: a set of examples together with labels
 - x_i, y_i
 - Classifier $h: X \rightarrow Y$

10/18/2004

4

How to evaluate?

- Idea:
 - Train classifier h on the data set
 - Use h to classify the data set and count the errors
 - BAD!!!
- Instead:
 - Split the data into training and test sets
 - Train on the training set and count errors on the test set

10/18/2004

5

How do we compare different models?

- Accuracy:
 - Fraction of correct predictions
- Precision:
 - Fraction of examples the model classified as 1 that are actually 1
- Recall:
 - Fraction of all 1 examples that the classifier classifies as 1
- F-measure:
 - The harmonic mean of precision and recall $2PR/(P+R)$

10/18/2004

6

Other metrics

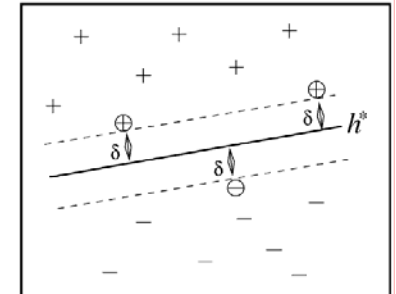
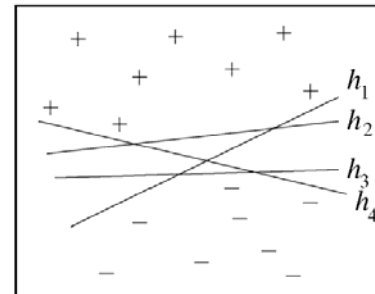
- Modify models to get confidence in addition to classification
 - Sort by confidence – Ordering metrics:
 - Average precision
 - ROC area, breakeven point
 - Interpret confidence as a probability that the example is class 1 – Probability metrics
 - Mean squared error
 - Cross entropy, calibration

10/18/2004

7

Support Vector Machine (SVM)

- Motivated by the fact that many rules can separate the data (almost) equally well
- Searches for the rule with largest margin
- Leads to an optimization problem



10/18/2004

Based on slide by Thorsten Joachims

8

Meta classifiers

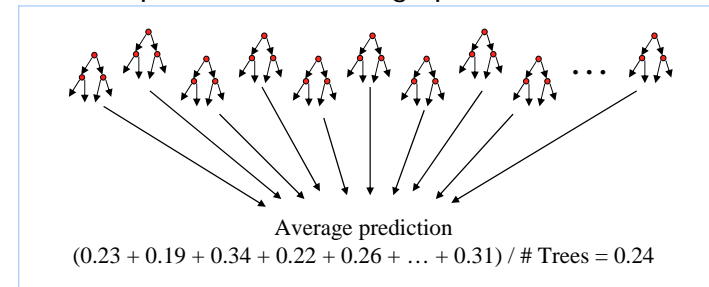
- Given a basic classifying method (kNN, DT) can we do better?
- Two ideas:
 - Instead of training one, train several of the base classifiers and average the outputs (bagging)
 - Force the algorithm to pay more attention to examples that it misclassifies (boosting)

10/18/2004

9

Bagged Models

- Draw N bootstrap samples of data
- Train a base model on each sample ==> N models
- Final prediction = average prediction of N models



10/18/2004

Based on Rich Caruana's presentation

10

Boosted Models

- Algorithm:
 - Train a base classifier and use it to classify the training set
 - Give more weight to examples on which the classifier made mistakes and retrain
 - Repeat for n iterations
- Use a combination of the n classifiers for classifying new examples
- *Fun fact: Boosting is a maximum margin method (like SVM)*

10/18/2004

11

A large scale empirical evaluation of ML models

- Work by Rich Caruana and Alexandru Niculescu-Mizil
- Next slides based on Rich Caruana's AI lunch presentation

10/18/2004

12

10 Binary Classification Performance Metrics

- Threshold Metrics:
 - Accuracy
 - F-Score
 - Lift
- Ordering/Ranking Metrics:
 - ROC Area
 - Average Precision
 - Precision/Recall Break-Even Point
- Probability Metrics:
 - Root-Mean-Squared-Error
 - Cross-Entropy
 - Probability Calibration
- $SAR = ((1 - \text{Squared Error}) + \text{Accuracy} + \text{ROC Area}) / 3$

10/18/2004

Based on Rich Caruana's presentation

13

Massive Empirical Comparison

7 base-level learning methods
 X
 100's of parameter settings per method
 =
 ~ 2000 models per problem
 X
 7 test problems
 =
 14,000 models
 X
 10 performance metrics
 =
 140,000 model performance evaluations

10/18/2004

Based on Rich Caruana's presentation

14

Normalized Scores

- Problem:
 - some metrics, 1.00 is best (e.g. ACC)
 - some metrics, 0.00 is best (e.g. RMS)
 - some metrics, baseline is 0.50 (e.g. AUC)
 - some problems/metrics, 0.60 is excellent performance
 - some problems/metrics, 0.99 is poor performance
- Solution: Normalized Scores:
 - baseline performance => 0.00
 - best observed performance => 1.00 (proxy for Bayes optimal)
 - puts all metrics on equal footing

10/18/2004

Based on Rich Caruana's presentation

15

Learning Model Comparison

Model	Threshold Metrics			Rank/Ordering Metrics			Probability Metrics			SAR	Mean
	Accuracy	F-Score	Lift	ROC Area	Average Precision	Break Even Point	Squared Error	Cross-Entropy	Calibration		
SVM	0.8134	0.9092	0.9480	0.9621	0.9335	0.9377	0.8767	0.8778	0.9824	0.9055	0.9156
ANN	0.8769	0.8752	0.9487	0.9552	0.9167	0.9142	0.8532	0.8634	0.9881	0.8956	0.9102
BAG-DT	0.8114	0.8609	0.9465	0.9674	0.9416	0.9220	0.8588	0.8942	0.9744	0.9036	0.9086
BST-DT	0.8904	0.8986	0.9574	0.9778	0.9597	0.9427	0.6066	0.6107	0.9241	0.8710	0.8631
KNN	0.7557	0.8463	0.9095	0.9370	0.8847	0.8890	0.7612	0.7354	0.9843	0.8470	0.8559
DT	0.5261	0.7891	0.8503	0.8678	0.7674	0.7954	0.5564	0.6243	0.9647	0.7445	0.7491
BST-STMP	0.7319	0.7903	0.9046	0.9187	0.8610	0.8336	0.3038	0.2861	0.9410	0.6589	0.7303

- SVM and ANN tied for first place; Bagged Trees nearly as good
- Boosted Trees win 5 of 6 Threshold & Rank metrics, but yield bad probabilities!
- KNN and Plain Decision Trees usually not competitive (with 4k train sets)
- Differences of about 0.01 are significant

10/18/2004

Based on Rich Caruana's presentation

16