Evaluating ML Models

Rishabh Iyer

Based on slides from Jude Shavlik and Tom Dietterich

Evaluating ML Methods

- What we have seen so far?
  - Accuracy
  - Root Mean Square Error

- Are there other evaluation criteria?

Contingency Tables

(special case of ‘confusion matrices’)

		True Answer +	**True Answer** -
Algorithm Answer	+	n(1,1) [true pos]	n(1,0) [false pos]
Algorithm Answer	-	n(0,1) [false neg]	n(0,0) [true neg]

Counts of occurrences

TPR and FPR

True Positive Rate (TPR) $= n(1,1) \,/\, (\, n(1,1) + n(0,1)\,)$

$=$ correctly categorized +'s / total positives

$\sim$ P(algo outputs + | + is correct)

False Positive Rate (FPR) $= n(1,0) \,/\, (\, n(1,0) + n(0,0)\,)$

$=$ incorrectly categorized –'s / total neg's

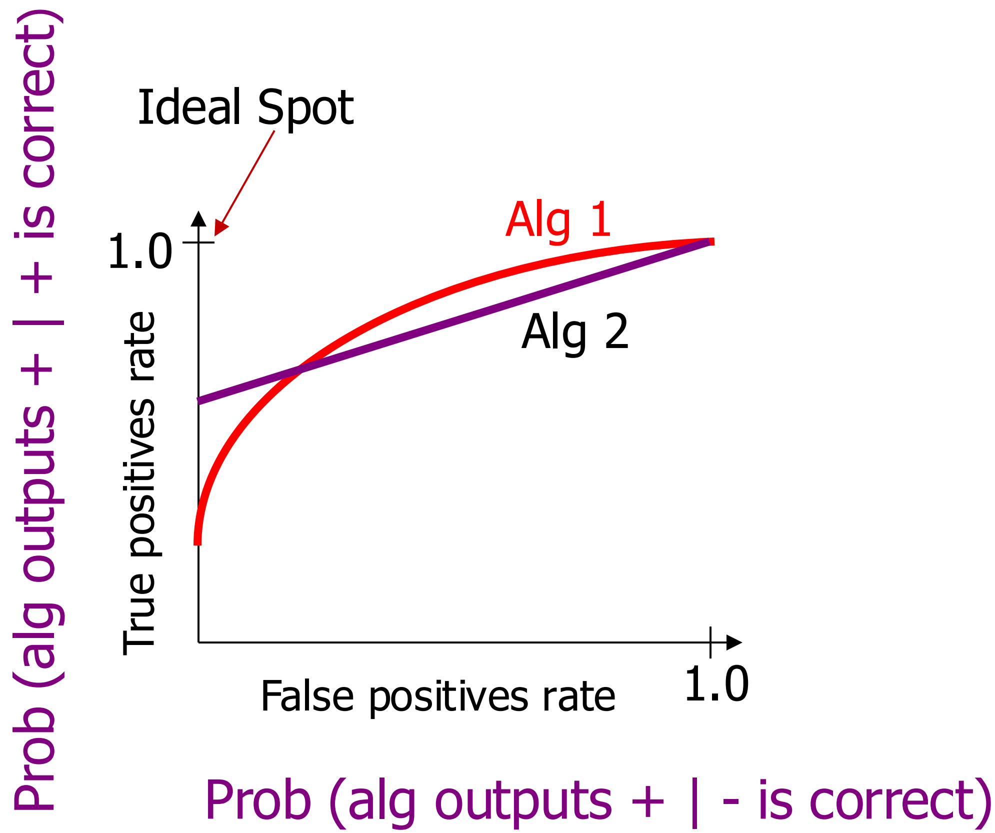
$\sim$ P(algo outputs + | - is correct)

Can similarly define False Negative Rate and True Negative Rate

ROC Curves

- ROC: *Receiver Operating Characteristics*
- Started for radar research during WWII
- Judging algorithms on accuracy alone may not be good enough when **getting a positive wrong** costs more than **getting a negative wrong** (or vice versa)
    - e.g., medical tests for serious diseases
    - e.g., a movie-recommender system

ROC Curves Graphically



Different algorithms can work better in different parts of ROC space. This depends on cost of false + vs false -

Creating an ROC Curve

The Standard Approach:

- You need an ML algorithm that outputs NUMERIC results such as prob(example is +)

- You can use ensemble methods to get this from a model that only provides Boolean outputs
  - e.g., have 100 models vote & count votes

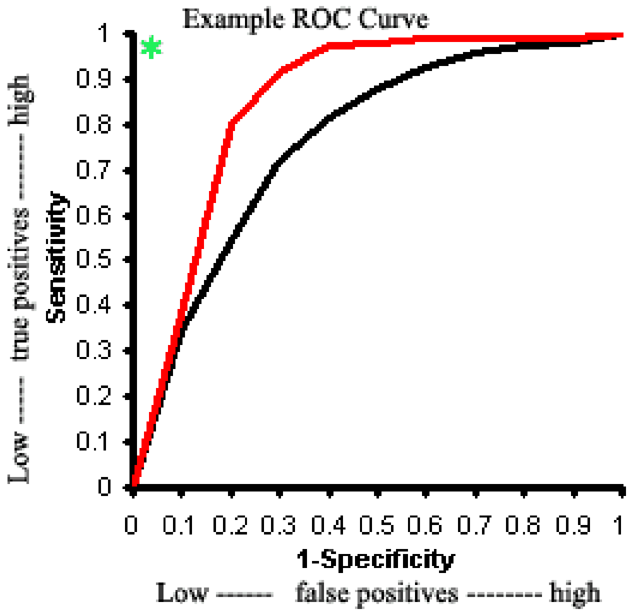
Alg. for Creating ROC Curves

Step 1: Sort predictions on test set

Step 2: Locate a *threshold* between examples with opposite categories

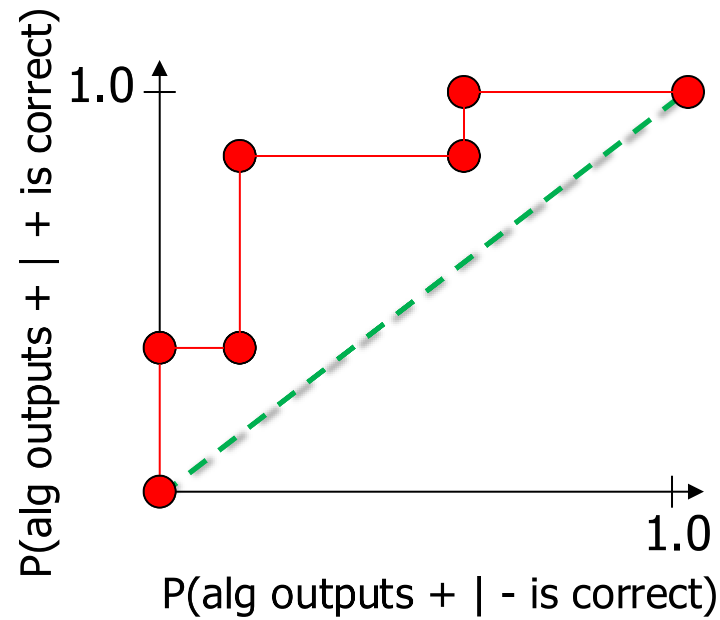
Step 3: Compute TPR & FPR for each threshold of Step 2

Step 4: Connect the dots



Plotting ROC Curves - Example

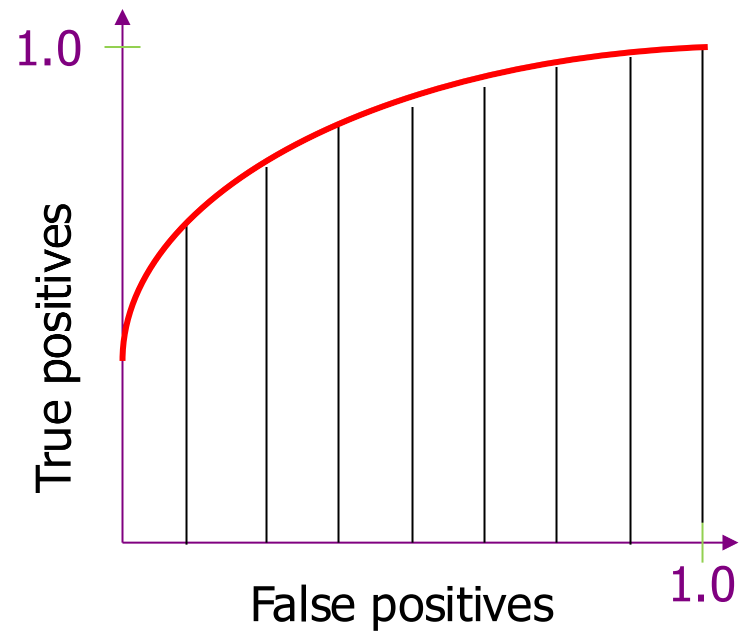
ML Algo Output (Sorted)			Correct Category
Ex 9	.99		+
Ex 7	.98	TPR=(2/5), FPR=(0/5)	+
Ex 1	.72	TPR=(2/5), FPR=(1/5)	-
Ex 2	.70		+
Ex 6	.65	TPR=(4/5), FPR=(1/5)	+
Ex 10	.51		-
Ex 3	.39	TPR=(4/5), FPR=(3/5)	-
Ex 5	.24	TPR=(5/5), FPR=(3/5)	+
Ex 4	.11		-
Ex 8	.01	TPR=(5/5), FPR=(5/5)	-



Algorithm predicts + if its output is ≥ 0

Area Under ROC Curve

- A common metric for experiments is to numerically integrate the ROC Curve
  - Usually called AUC
  - Probability that ML alg. will "rank" a randomly chosen positive instance higher than a randomly chosen negative one
  - Can summarize the curve **too much** in practice



Asymmetric Error Costs

- Assume that cost(FP) ≠ cost(FN)
- You would like to pick a threshold that minimizes

$$
\begin{aligned}
&E(total\ cost) \\
&= \ cost(FP) \times \ \mathrm{pr}(FP) \times (\#\ of\ neg\ ex's) + \\
&\ cost(FN) \times \ \mathrm{pr}(FN) \times (\#\ of\ pos\ ex's)
\end{aligned}
$$

- You could also have (maybe negative) costs for TP and TN (assumed zero in above)

ROC's & Skewed Data

- One strength of ROC curves is that they are a good way to deal with skewed data (|+| >> |-|) since the axes are fractions (rates) independent of the # of examples

- You must be careful though!
  - Low FPR * (many negative ex) = sizable number of FP
  - Possibly more than # of TP

Precision vs. Recall

- Think about search engines...

- **Precision** = (# of relevant items retrieved) / (total # of items retrieved)

  $= n(1,1) \,/\, (\, n(1,1) + n(1,0)\,)$

  $\cong$ P(is pos | called pos)

- **Recall** = (# of relevant items retrieved) / (# of relevant items that exist)

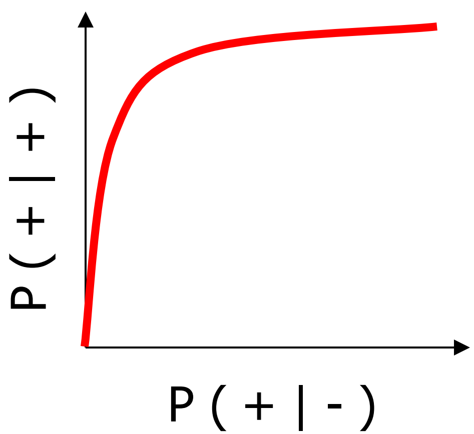
  $= n(1,1) \,/\, (\, n(1,1) + n(0,1)\,)$ = <u>TPR</u>

  $\cong$ P(called pos | is pos)

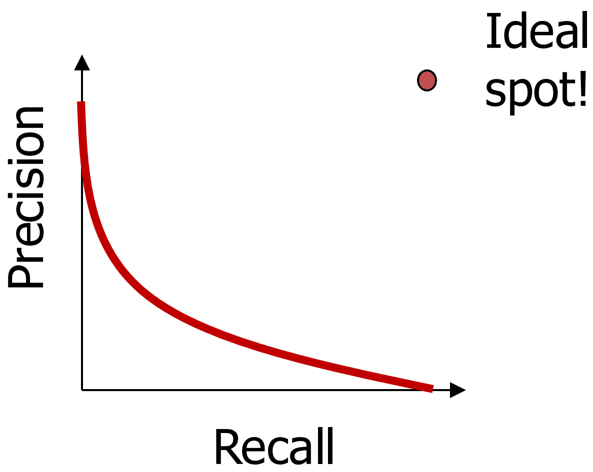
- Notice that n(0,0) is not used in either formula
  Therefore you get <u>no</u> credit for filtering out <u>ir</u>relevant items

ROC vs. Precision-Recall

You can get very different visual results on the same data!



vs.



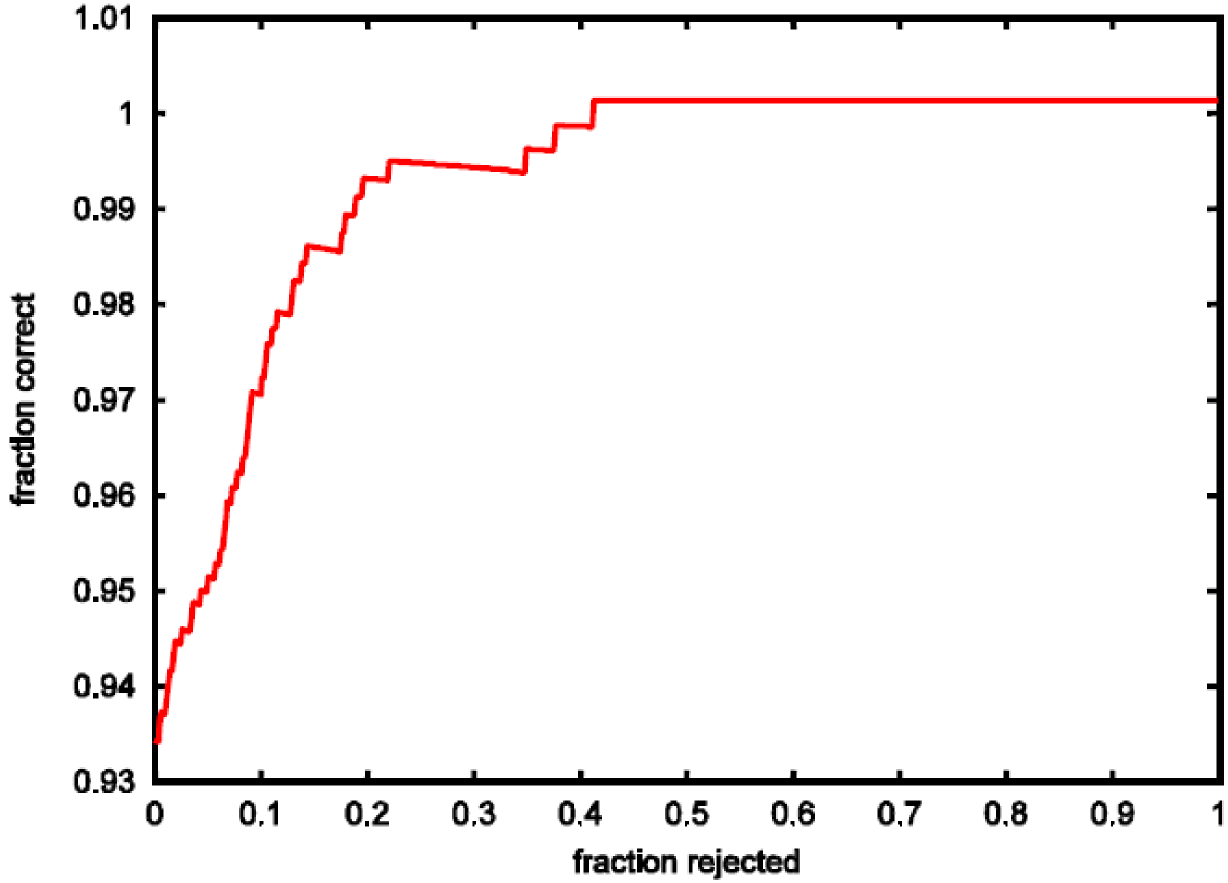
The reason for this is that there may be lots of – ex's
(e.g., might need to include 100 neg's to get 1 more pos)

Rejection Curves

- In most learning algorithms, we can specify a threshold for making a rejection decision
  - Probabilistic classifiers: adjust cost of rejecting versus cost of FP and FN
  - Decision-boundary method: if a test point $\boldsymbol{x}$ is within θ of the decision boundary, then reject
    - Equivalent to requiring that the “activation” of the best class is larger than the second-best class by at least θ

Rejection Curves

- Vary θ and plot fraction correct versus fraction rejected



The F1 Measure

- Figure of merit that combines precision and recall

$$F_1 = 2 \cdot \frac{P \cdot R}{P + R}$$

where P = precision; R = recall. This is twice the harmonic mean of P and R.

- We can plot $F1$ as a function of the classification threshold θ