Quantitative Evaluation

Adapted in part from:

http://www.cs.cornell.edu/Courses/cs578/2003fa/performance_measures.pdf

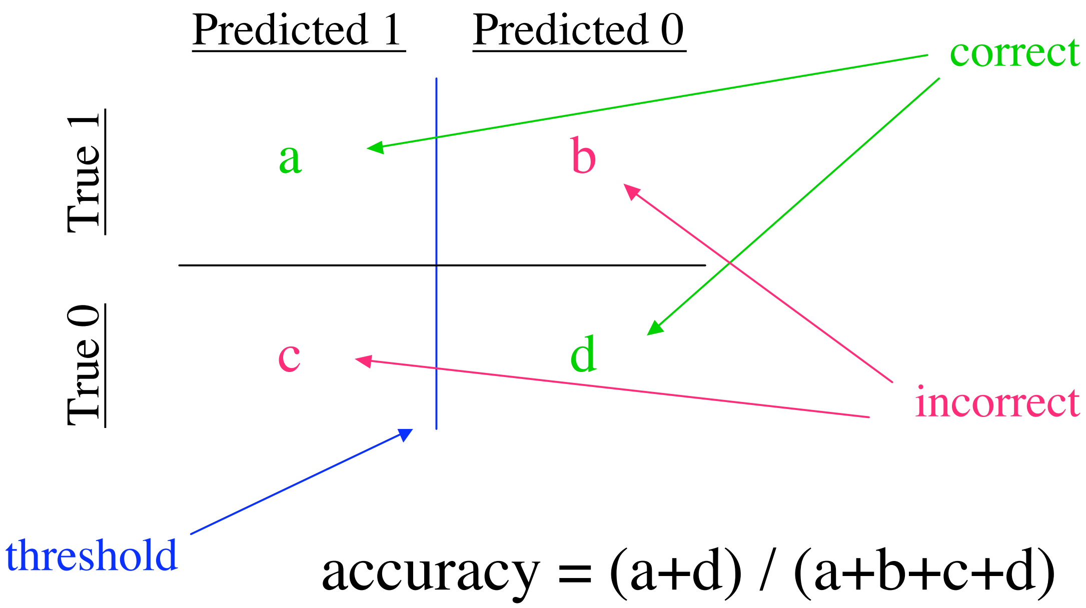
Accuracy

- Target: 0/1, -1/+1, True/False, …
- Prediction = f(inputs) = f(x): 0/1 or Real
- Threshold: f(x) > thresh => 1, else => 0
- threshold(f(x)): 0/1

$$accuracy = \frac{\sum_{i=1\ldots N} \left(1 - (target_i - threshold(f(\vec{x}_i)))\right)^2}{N}$$

- #right / #total
- p("*correct*"): p(threshold(f(x)) = target)

Confusion Matrix



	Predicted 1	Predicted 0
True 1	a	b
True 0	c	d

$$\text{accuracy} = (a+d) \,/\, (a+b+c+d)$$

	Predicted 1	Predicted 0
True 1	true positive	false negative
True 0	false positive	true negative

	Predicted 1	Predicted 0
True 1	TP	FN
True 0	FP	TN

	Predicted 1	Predicted 0
True 1	hits	misses
True 0	false alarms	correct rejections

	Predicted 1	Predicted 0
True 1	$P(pr1 \mid tr1)$	$P(pr0 \mid tr1)$
True 0	$P(pr1 \mid tr0)$	$P(pr0 \mid tr0)$

Problems with Accuracy

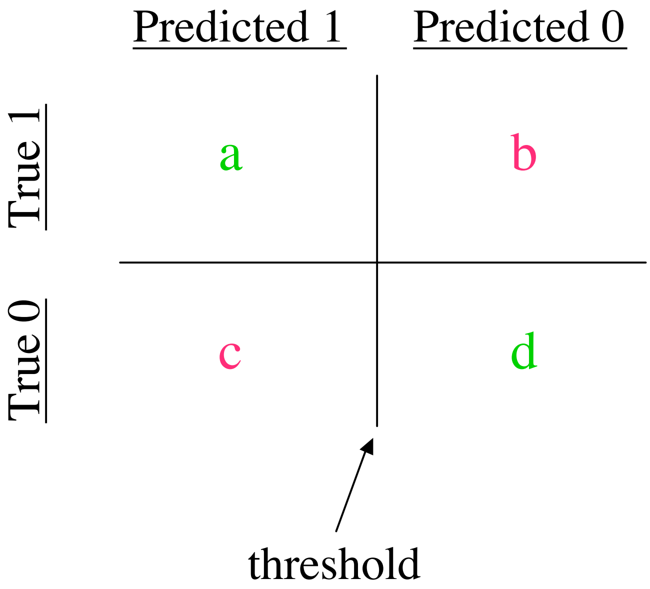
- Assumes equal cost for both kinds of errors
  - cost(b-type-error) = cost (c-type-error)

- is 99% accuracy good?
  - can be excellent, good, mediocre, poor, terrible
  - depends on problem
- is 10% accuracy bad?
  - information retrieval
- BaseRate = accuracy of predicting predominant class
  (on most problems obtaining BaseRate accuracy is easy)

Precision and Recall

- typically used in document retrieval
- Precision:
  - how many of the returned documents are correct
  - precision(threshold)
- Recall:
  - how many of the positives does the model return
  - recall(threshold)
- Precision/Recall Curve: sweep thresholds

Precision/Recall

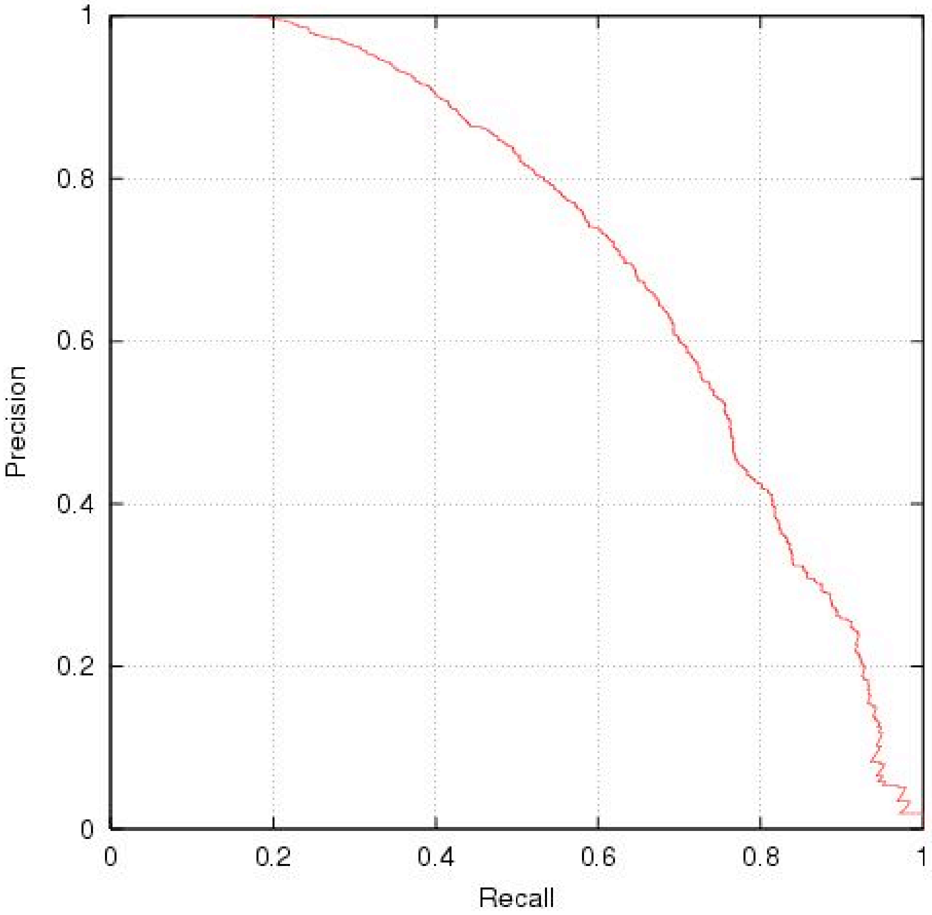


	Predicted 1	Predicted 0
True 1	a	b
True 0	c	d

threshold

$$PRECISION = a/(a+c)$$

$$RECALL = a/(a+b)$$


Precision
Recall
0
0.2
0.4
0.6
0.8
1

Summary Stats: F & BreakEvenPt

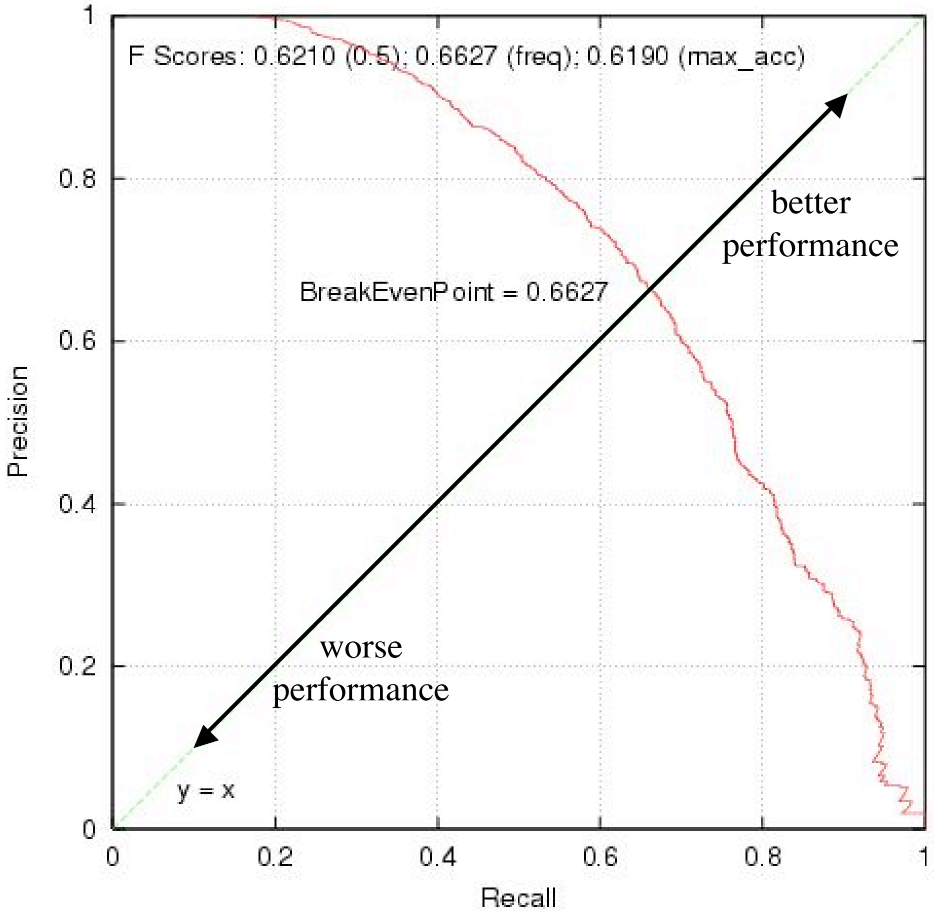
$$PRECISION = a/(a+c)$$

$$RECALL = a/(a+b)$$

$$F = \frac{2*(PRECISION \square RECALL)}{(PRECISION + RECALL)}$$

harmonic average of precision and recall

$$BreakEvenPo\mathrm{int} = PRECISION = RECALL$$

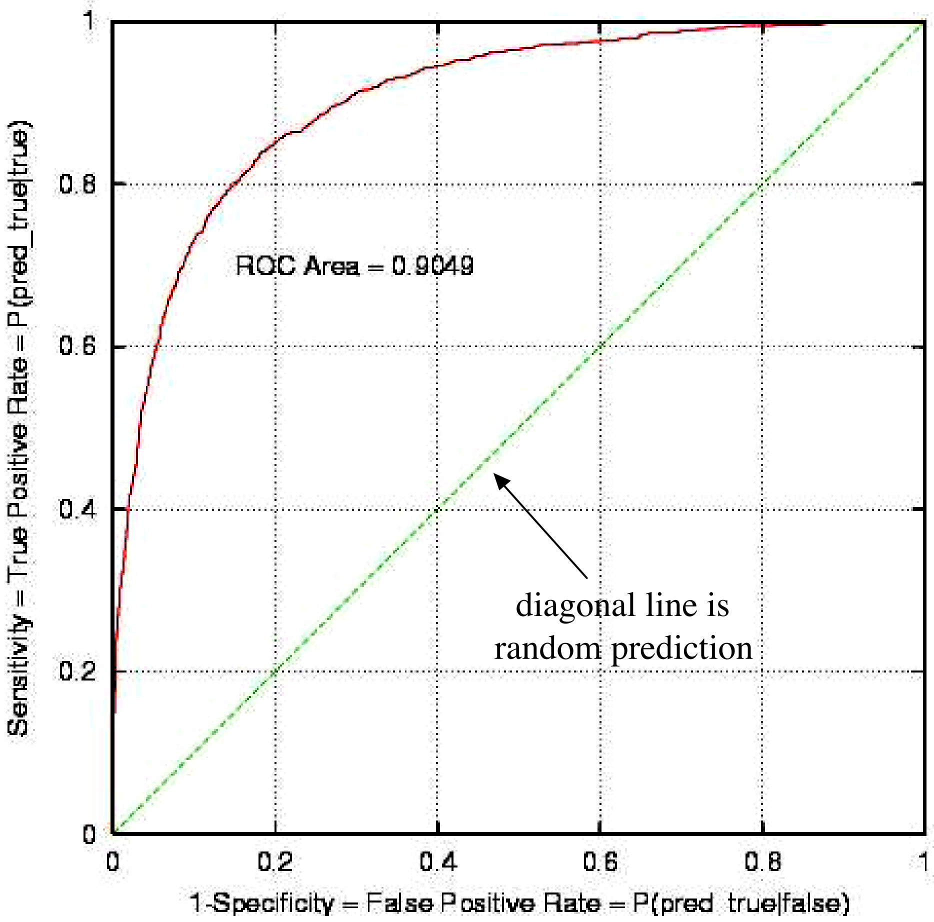

F Scores: 0.6210 (0.5); 0.6627 (freq); 0.6190 (max_acc)
BreakEvenPoint = 0.6627
better performance
worse performance
y = x
Precision
Recall
0
0.2
0.4
0.6
0.8
1

ROC Plot and ROC Area

- Receiver Operator Characteristic
- Developed in WWII to statistically model false positive and false negative detections of radar operators
- Better statistical foundations than most other measures
- Standard measure in medicine and biology
- Becoming more popular in ML

ROC Plot

- Sweep threshold and plot
  - TPR vs. FPR
  - Sensitivity vs. 1-Specificity
  - P(true|true) vs. P(true|false)
- Sensitivity = a/(a+b) = Recall = LIFT numerator
- 1 - Specificity = 1 - d/(c+d)



diagonal line is random prediction

Properties of ROC

- ROC Area:
  - 1.0: perfect prediction
  - 0.9: excellent prediction
  - 0.8: good prediction
  - 0.7: mediocre prediction
  - 0.6: poor prediction
  - 0.5: random prediction
  - <0.5: something wrong!

Properties of ROC

- Slope is non-increasing
- Each point on ROC represents different tradeoff (cost ratio) between false positives and false negatives
- Slope of line tangent to curve defines the cost ratio
- ROC Area represents performance averaged over all possible cost ratios
- If two ROC curves do not intersect, one method dominates the other
- If two ROC curves intersect, one method is better for some cost ratios, and other method is better for other cost ratios

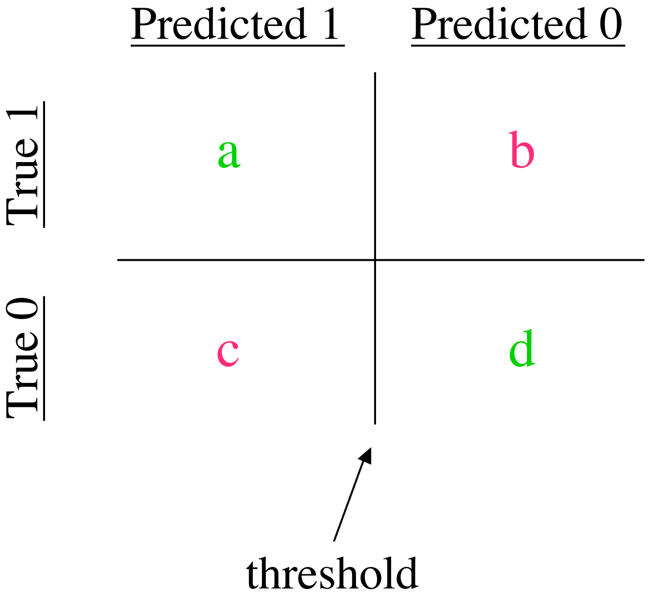
Lift

- not interested in accuracy on entire dataset
- want accurate predictions for 5%, 10%, or 20% of dataset
- don’t care about remaining 95%, 90%, 80%, resp.
- typical application: marketing

$$lift(threshold) = \frac{\%positives > threshold}{\%\,dataset > threshold}$$

- how much better than random prediction on the fraction of the dataset predicted true (f(x) > threshold)

Lift

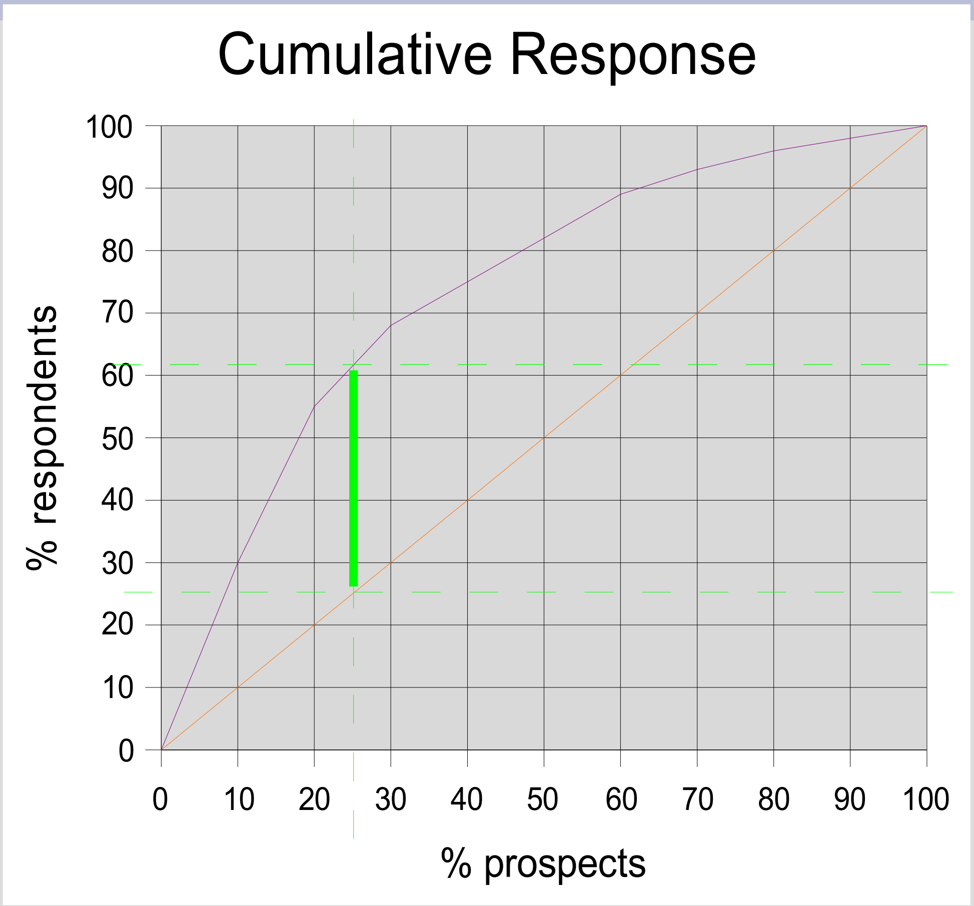


	Predicted 1	Predicted 0
True 1	a	b
True 0	c	d

threshold

$$lift = \frac{a/(a+b)}{(a+c)/(a+b+c+d)}$$

Visualizing Lift



Lift(c) = CR(c) / c

Example:
Lift(25%)= CR(25%) / 25%
= 62% / 25%
= 2.5

If we send to 25% of our prospects using the model, they are 2.5 times as likely to respond than if we were to select them randomly.

Computing Profit

- Assume cut-off at some value *c*
- Let:
  - T = total number of prospects
  - H = total number of respondents
  - n = cost per mailing
  - p = profit per response
- Then:
  - $$\begin{aligned} \text{Profit}(c) = {} & \text{CR}(c).H.p && \text{revenue generated by respondents} \\ & - c.T.n && \text{cost of sending the mailings} \\ & + (1-c).T.n && \text{saving from not sending mailings} \\ & - (1-\text{CR}(c)).H.p && \text{cost of missed revenue} \end{aligned}$$

Understanding Profit (I)

- Profit(c)

  $= 2.\mathrm{CR}(c).H.p - 2.c.T.n + T.n - H.p$

  $= 2.[\mathrm{CR}(c).H.p - c.T.n] - [H.p - T.n]$

- Since:
  - 2 is a constant (scaling)
  - $H.p - T.n$ is a constant (translation)
- Then,
  - $\mathrm{Profit}(c) \sim \mathrm{CR}(c).H.p - c.T.n$
- Let
  - $E = H / T$ response rate
  - $\underline{\mathrm{Profit}(c) \sim \mathrm{CR}(c).E.p - c.n}$

Understanding Profit (II)

- Note that:
  - Lift(c) = CR(c)/c
  - Lift would be maximum if we could send to only exactly all of the respondents; we would then have $c = E$ (=H/T) and CR(E) = 100%
  - The maximum value for lift is thus: 1/E
- Returning to profit:
  - Case 1: $p < n$
    - Profit(c) < 0 => not viable
  - Case 2: $p = n$
    - Profit(c) $\geq$ 0 only if Lift(c) $\geq$ 1/E => impossible
  - Case 3: $p > n$
    - Profit(c) $\geq$ 0 => OK

Summary

- the measure you optimize to makes a difference
- the measure you report makes a difference
- use measure appropriate for problem/community
- accuracy often is not sufficient/appropriate
- ROC is gaining popularity in the ML community
- only accuracy generalizes to >2 classes!