Performance Measures

- Accuracy
- Weighted (Cost-Sensitive) Accuracy
- Lift
- Precision/Recall
  - F
  - Break Even Point
- ROC
  - ROC Area
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

$$\text{accuracy} = \frac{\sum_{i=1}^{N} (1 - (\text{target}_i - \text{threshold}(f(\tilde{x}_i)))^2}{N}$$

• #right / #total
• \(p(\text{"correct"})\): \(p(\text{threshold}(f(x)) = \text{target})\)
Confusion Matrix

\[
\begin{array}{c|cc}
\text{True 1} & \text{Predicted 1} & \text{Predicted 0} \\
\hline
\text{True 1} & a & b \\
\text{True 0} & c & d \\
\end{array}
\]

accuracy = \frac{(a+d)}{(a+b+c+d)}
## Prediction Threshold

<table>
<thead>
<tr>
<th>Predicted 1</th>
<th>Predicted 0</th>
<th>Condition</th>
<th>Calculation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>True 1</strong></td>
<td>0</td>
<td>threshold &gt; MAX(f(x))</td>
<td>(b) = all cases predicted 0</td>
</tr>
<tr>
<td></td>
<td><strong>b</strong></td>
<td></td>
<td>((b+d) = \text{total})</td>
</tr>
<tr>
<td><strong>True 0</strong></td>
<td>0</td>
<td></td>
<td>(d) = all cases predicted 0</td>
</tr>
<tr>
<td></td>
<td><strong>d</strong></td>
<td></td>
<td>((b+d) = \text{total})</td>
</tr>
</tbody>
</table>

- **Accuracy** = \%False = \%0’s

<table>
<thead>
<tr>
<th>Predicted 1</th>
<th>Predicted 0</th>
<th>Condition</th>
<th>Calculation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>True 1</strong></td>
<td><strong>a</strong></td>
<td>threshold &lt; MIN(f(x))</td>
<td>0 = all cases predicted 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>((a+c) = \text{total})</td>
</tr>
<tr>
<td><strong>True 0</strong></td>
<td><strong>c</strong></td>
<td></td>
<td>0 = all cases predicted 0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>((b+d) = \text{total})</td>
</tr>
</tbody>
</table>

- **Accuracy** = \%True = \%1’s
optimal threshold

82% 0’s in data

18% 1’s in data
threshold demo
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)
Percent Reduction in Error

• 80% accuracy = 20% error
• suppose learning increases accuracy from 80% to 90%
• error reduced from 20% to 10%
• 50% reduction in error

• 99.90% to 99.99% = 90% reduction in error
• 50% to 75% = 50% reduction in error
• can be applied to many other measures
Costs (Error Weights)

<table>
<thead>
<tr>
<th></th>
<th>Predicted 1</th>
<th>Predicted 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>True 1</td>
<td>( w_a )</td>
<td>( w_b )</td>
</tr>
<tr>
<td>True 0</td>
<td>( w_c )</td>
<td>( w_d )</td>
</tr>
</tbody>
</table>

- Often \( w_a = w_d = \text{zero} \) and \( w_b \neq w_c \neq \text{zero} \)
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

\[
lift(\text{threshold}) = \frac{\% \text{positives} > \text{threshold}}{\% \text{dataset} > \text{threshold}}
\]

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

\[
\text{lift} = \frac{a/(a+b)}{(a+c)/(a+b+c+d)}
\]
lift = 3.5 if mailings sent to 20% of the customers
Lift and Accuracy do not always correlate well

Problem 1

Problem 2

(thresholds arbitrarily set at 0.5 for both lift and accuracy)
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

\[
\begin{array}{cc|cc}
\text{True 1} & \text{Predicted 1} & \text{Predicted 0} \\
\hline
a & b \\
\text{True 0} & c & d \\
\text{threshold} & & & \\
\end{array}
\]

\[
\text{PRECISION} = \frac{a}{a + c}
\]

\[
\text{RECALL} = \frac{a}{a + b}
\]
Summary Stats: F & BreakEvenPt

**PRECISION** = \( a / (a + c) \)

**RECALL** = \( a / (a + b) \)

**F** = \( \frac{2 \times (\text{PRECISION} \circ \text{RECALL})}{\text{PRECISION} + \text{RECALL}} \)

**BreakEvenPoint** = **PRECISION** = **RECALL**
Precision vs. Recall graph with F Scores: 0.6210 (0.5), 0.6627 (freq), 0.6190 (max_acc).

BreakEvenPoint = 0.6627

Better performance

Worse performance

y = x
F and BreakEvenPoint do not always correlate well
A confusion matrix is shown with the following terms:

- True Positive (TP): Correctly predicted positive outcome.
- False Positive (FP): Incorrectly predicted positive outcome.
- True Negative (TN): Correctly predicted negative outcome.
- False Negative (FN): Incorrectly predicted negative outcome.

The matrix is divided into four quadrants:

- Predicted 1
  - True Positive (TP)
  - False Positive (FP)

- Predicted 0
  - True Negative (TN)
  - False Negative (FN)

Additionally, the matrix includes:

- P(pr1|tr1): Probability of predicting 1 given that 1 is true.
- P(pr0|tr1): Probability of predicting 0 given that 1 is true.
- P(pr1|tr0): Probability of predicting 1 given that 0 is true.
- P(pr0|tr0): Probability of predicting 0 given that 0 is true.
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)
Sensitivity = True Positive Rate = $P(\text{pred true}|\text{true})$

ROC Area = 0.9049

diagonal line is random prediction

1-Specificity = False Positive Rate = $P(\text{pred true}|\text{false})$
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
Problem 1

Problem 2
Problem 1

Problem 2
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!