APPENDIX

The Binary Classification Trade-off

A

A *binary classification trade-off* arises when we wish to classify a set of things into two categories (call them *In* and *Out*), but we do not have a direct way of doing the classifying. On the other hand, there is a *proxy* for those things that is relatively easy to classify. The problem is that the proxy is only approximate. Because it is only approximate, there are four classification outcomes:

- *True positive:* The proxy classifies things as *In* that should be *In*.
- *True negative:* The proxy classifies things as *Out* that should be *Out*.
- *False negative:* The proxy classifies things as *Out* that should be *In*.
- *False positive:* The proxy classifies things as *In* that should be *Out*.

The trade-off is that it may be possible to reduce the frequency of one of the false outcomes by adjusting some parameter of the proxy, but that adjustment will probably increase the frequency of the other false outcome*.

A common example is an e-mail spam filter, which is a proxy for the division between wanted e-mail and spam. The filter correctly classifies e-mail most of the time, but it occasionally misclassifies a wanted message as spam, with the undesirable outcome that you may never see that message. It may also misclassify some spam as wanted e-mail, with the undesirable outcome that the spam clutters up your mailbox. The trade-off appears when someone tries to adjust the spam filter. If the filter becomes more aggressive, more wanted e-mail is likely to end up misclassified as spam. If the spam filter becomes less aggressive, more spam is likely to end up in your mailbox.

Reducing both undesirable outcomes simultaneously usually requires discovering a better proxy, but a better one may be hard to find or may not exist at all.

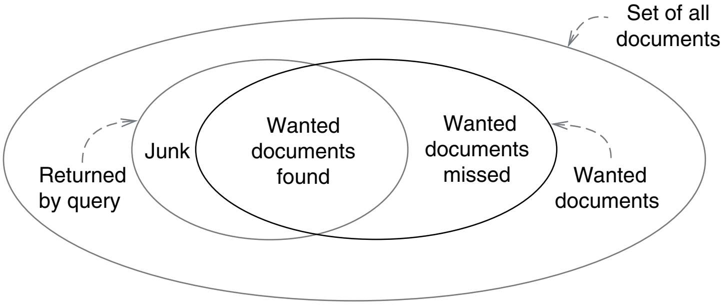
Representations: One can conveniently represent a binary classification trade-off with a 2 × 2 matrix such as the one on the next page by answering two questions: (1) What are the real categories? and (2) What are the proxy categories? The example describes a smoke detector. The real categories are {fire, no fire}. The proxy categories are {smoke detector signals, smoke detector is quiet}. A too-sensitive smoke detector

*In some areas, such as computer security and biometrics, the words "acceptance" and "rejection" replace "positive" and "negative", respectively.

		Real categories	
		fire	no fire
Proxy categories	detector signals	TA: fire extinguished	FA: false alarm
	detector quiet	FR: house burns down	TR: all quiet

may signal more false alarms, but an insensitive one may miss more real fires. When someone replaces the labels with numbers of actual events, this representation is called a *confusion matrix*.

A Venn diagram, such as the one below, can be another useful representation of a binary classification trade-off. Take, for example, document retrieval (e.g., a Google search) The real categories are wanted and unwanted documents. The proxy is a query, for which the categories are that the query matches or the query misses.



Measures: Sometimes one can identify the true categorizations and compare them with the proxy classifications. When that is possible, it can be useful to calculate ratios to measure proxy quality. Unfortunately, there are too many possible ratios. The confusion matrix contains four numbers, which may be used singly or may be added up to use as either a numerator or a denominator in 14 ways, so it is possible to calculate 14 × 13 = 182 different ratios. Not all of these ratios are interesting, but one can usually find at least one ratio among the 182 that seems to support his or her position in a debate.

Nine of these ratios are popular enough to have names, although three of the nine are just complements of other named ratios. The information retrieval community uses one set of labels for these ratios, whereas the medical and bioinformatics communities use another, with other communities developing their own nomenclature. As will be seen, all of the labels can be confusing.

Suppose that there is a population of *In* + *Out* = *N* items and that we have run the classifier and counted the number of true and false positives and negatives. Here are the nine ratios:

1. *Prevalance*: The fraction of the population that is *In*.

$$Prevalance = In/N$$

2. *Efficiency*, *Accuracy*, or *Hit Rate*: The fraction of the population the proxy classifies correctly.

$$Efficiency = (True\,Positives + True\,Negatives)/N$$

3. *Precision* (information retrieval) or *Positive Predictive Value* (medical): The fraction of things that the proxy classifies as *In* that are actually *In*.

$$Precision = (True\,Positives)/(True\,Positives + False\,Positives)$$

4. *Recall* (information retrieval), *Sensitivity* (medical), or *True acceptance rate* (biometrics): The fraction of things in the population that are *In* that the proxy classifies as *In*.

$$Recall = (True\,Positives)/In$$

5. *Specificity* (medical) or *True rejection rate* (biometrics): The fraction of things in the population that are *Out* that the proxy classifies as *Out*.

$$Specificity = (True\,Negatives)/Out$$

6. *Negative Predictive Value*: The fraction of things that the proxy classifies as *Out* that are actually *Out*.

$$Negative\,Predictive\,Value = \frac{True\,Negatives}{True\,Negatives + False\,Negatives}$$

7. *Misclassification Rate* or *Miss Rate*: The fraction of the population the proxy classifies wrong.

$$Miss\,Rate = (False\,Negatives + False\,Positives)/N = (1 - Efficiency)$$

8. *False Acceptance Rate*:The fraction of *Out* items that are falsely classified as *In*.

$$Fasle\ Acceptance\,Rate = (False\,Positives)/Out = (1 - Specificity)$$

9. *False Rejection Rate*: The fraction of *In* items that are falsely classified as *Out*.

$$Fasle\,Rejection\,Rate = (False\,Negatives)/In = (1 - Sensitivity)$$