Lecture 4

Evaluating Classifiers

Mladen Kolar (mkolar@chicagobooth.edu)

Classification and Evaluating Classifiers

Reminder: Supervised learning

Training experience: a set of labeled examples (samples, observations) of the form $(x_1, y_1), (x_2, y_2), \ldots, (x_i, y_i), \ldots, (x_n, y_n)$ where $x_i = (1, x_{i1}, \ldots, x_{ip})$ are vectors of input variables (covariates, predictors) and y is the output

What to learn: A function f which maps input X into output Y

Goal: minimize the error (loss) function

What should be the loss function for classification?

Loss should be problem specific, however, it is often useful to have generic loss functions.

- Often need surrogate loss as the ultimate goal if not perfectly measurable.

We use loss measures to assess our out-of-sample performance (e.g. when doing cross-validation).

We also use loss measures in estimation.

Missclassification rate or accuracy

For classification, an obvious loss is the **missclassification rate** or **accuracy**.

If $Y, \hat{Y} \in \{1, 2, \ldots, C\}$ we can let

$$L(Y, \hat{Y}) = \begin{cases} 0, & Y = \hat{Y} \\ 1, & Y \neq \hat{Y} \end{cases}$$

If you guess right, you lose nothing, if you are wrong, you lose 1.

Then

$$MR = \frac{1}{m} \sum_{i=1}^{m} L(Y_i, \hat{Y}_i)$$

is the fraction miss-classified. Accuracy is simply $1 - MR$.

For example: CART algorithm uses miss-classification rate to grow trees.

Example: Accidents data

42,183 observations have been collected on automobile accidents.

We will predict whether an accident resulted in a bodily injury or not.

For each accident you have additional type of information, such as day of week, weather conditions, and road type.

Problems with missclassification rate

It is not easy to minimize — leads to a hard optimization problem.

- This is a technical point, but an important one. We will investigate alternative surrogates next.

Reduces performance of a classifier to a single number, which is a simplistic summary.

- Sometimes it is important to understand types of correct and incorrect decisions made by a classifier. For that we use the confusion matrix.

Does not work well with unbalanced classes.

- Often a classifier will put everything in a majority class (this is surprising).

Example: Tabloid data

Y is 1 if a customer responds to a promotion (mailed a "tabloid") and 0 otherwise.

4 x's from the database (selected from many other similar ones).

- nTab: number of past orders.
- moCbook: months since last order.
- iRecMer1 : 1/ months since last order in merchandise category 1.
- llDol: log of the dollar value of past purchases

10,000 in train; 5,000 in test.

Note: in the train, only 2.58% respond.

Surrogate Losses

For computational reasons, it is common to minimize a surrogate loss to miss-classification rate during training.

Commonly used losses in machine learning are:

- deviance loss (can be used for any probabilistic classifier)
- hinge loss (support vector machines — more next week!)
- exponential loss (AdaBoost algorithm)

Keep in mind that all these are generic losses, which are not tailored for a specific problem at hand.

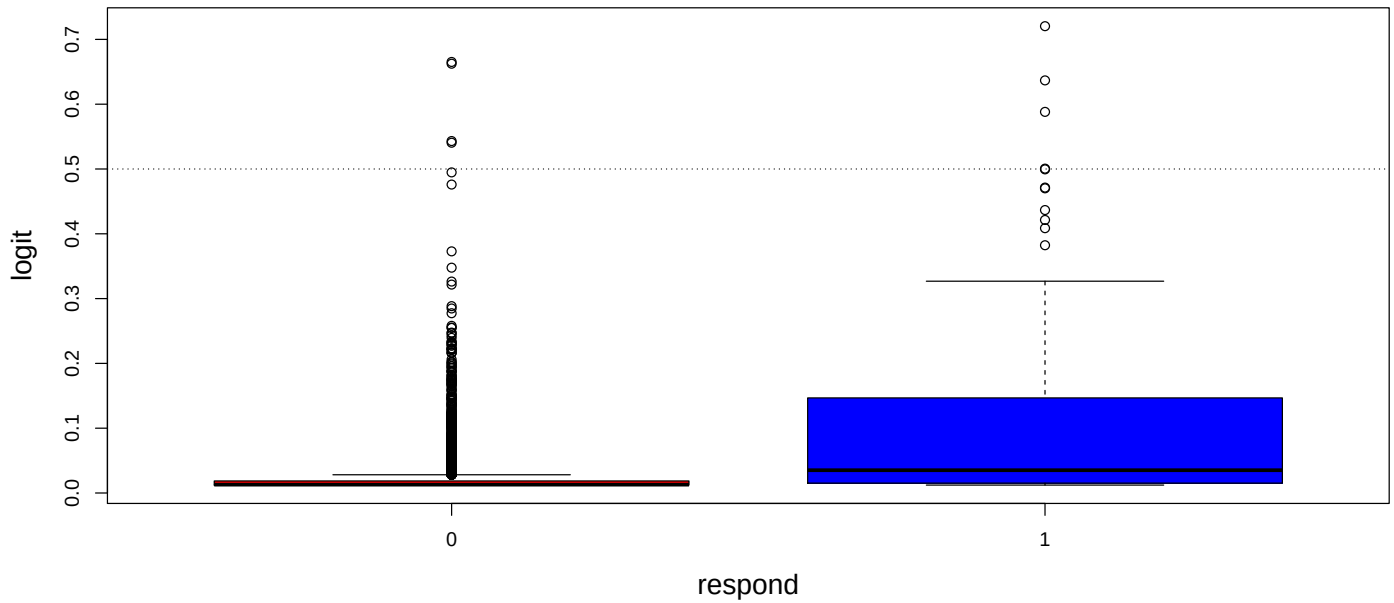
The hope is that a classifier trained to minimize these generic losses will also perform well in terms of miss-classification on out-of-sample data, as well as on a problem specific metric.

Another problem with miss-classification is that, in many cases, the model with best miss-classification rate predicts that all instances belong to the majority class.

- not useful for making decisions in business applications

Recall that in our target marketing example, the probability of a response is almost always less than 0.5. If you just predict "they won't respond" you do pretty good in terms of miss-classification, but that is not helpful.

Here is the test plot of $\hat{p}$ (from logit) and y=respond for the tabloid data.



While the model works, it is pretty useless to predict a response if $\hat{p} > 0.5$!

Only 112 out of 5,000 actually responded so if we just say no-one responds the MR is 2%.

Deviance loss

Deviance loss measures the quality of the model by looking at $\hat{P}(Y = y \mid X = x)$ produced by the model.

Suppose we have an (x, y) pair where x is a vector of predictors and y is the corresponding outcome. Our model gives us $\hat{P}(Y = y \mid X = x)$.

- If $\hat{P}(Y = y \mid X = x)$ is high, then the observed thing is likely under our model.
- If $\hat{P}(Y = y \mid X = x)$ is small, then the observed thing is un-likely under our model.

If the model says what happened is likely, that is good!

The **deviance** measure of how **bad** things are (for one observation (x, y)) is

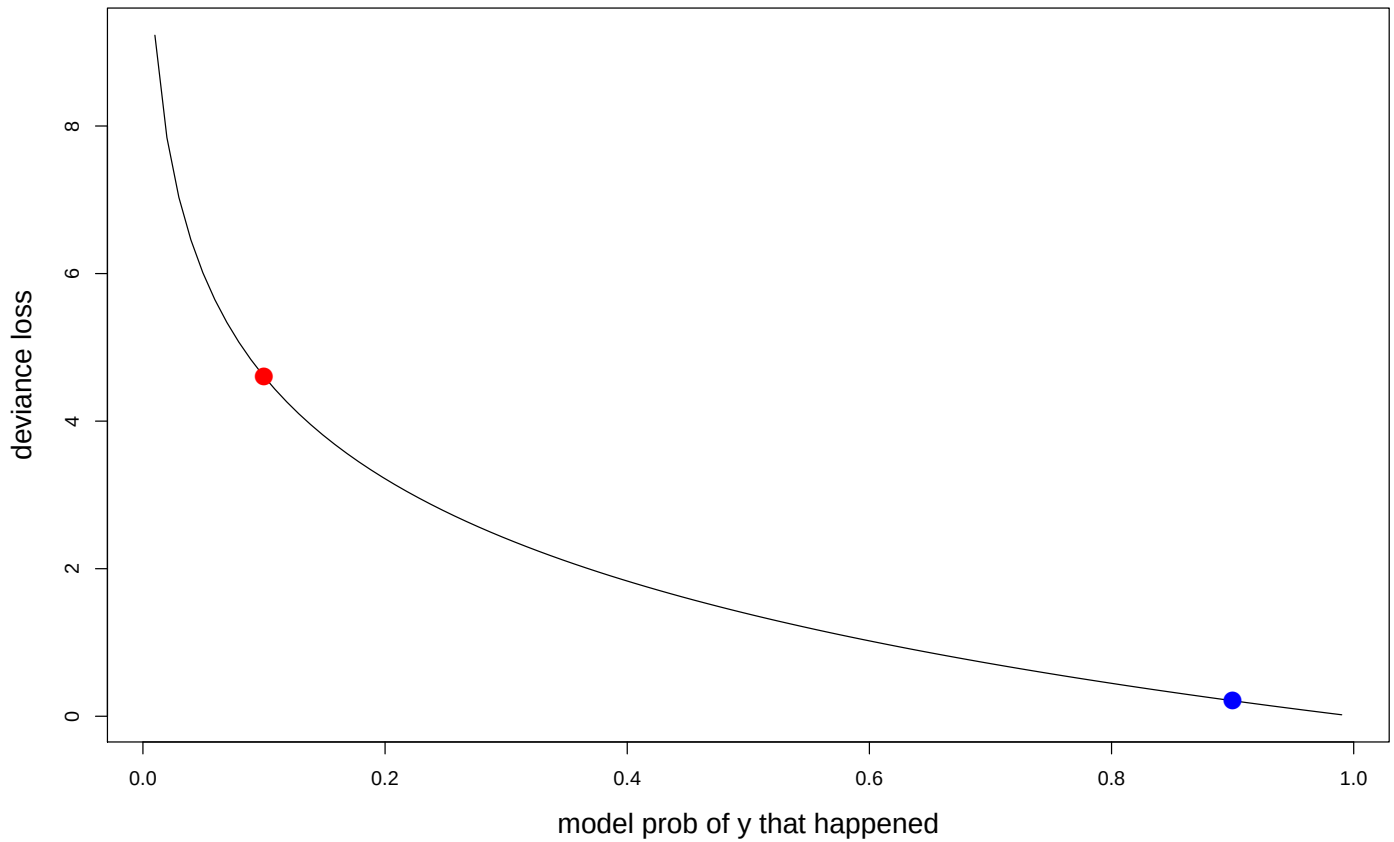
$$L(\hat{P}, x, y) = -2 \log(\hat{P}(Y = y \mid X = x))$$

The total loss for a dataset $\mathcal{D}$ (train or test) is

$$L(\hat{P}) = \sum_{i \in \mathcal{D}} L(\hat{P}, x_i, y_i) = \sum_{i \in \mathcal{D}} -2 \log(P(Y = y_i \mid x_i))$$

The deviance is not terribly interpretable, but it gets used a fair amount.

If you say a certain y outcome is likely and then it happens, your loss is small.



If you say a certain y outcome is very unlikely and then it happens, your loss is big.

The Confusion Matrix

We consider a two-class classification problem.

Confusion matrix separates out the decisions made by the classier, making it explicit how one class is being confused by another.

- Columns are labeled by actual classes.
- Rows are labeled by predicted classes.

	positive	**n**egative
Y	True positives	False negatives
N	False positives	True negatives

How can we obtain miss-classification rate of a classifier?
How can we obtain accuracy of a classifier?

Problems with Unbalanaced Classes

Suppose there are two classifiers whose confusion matrices are:

Confusion matrix of classifier A

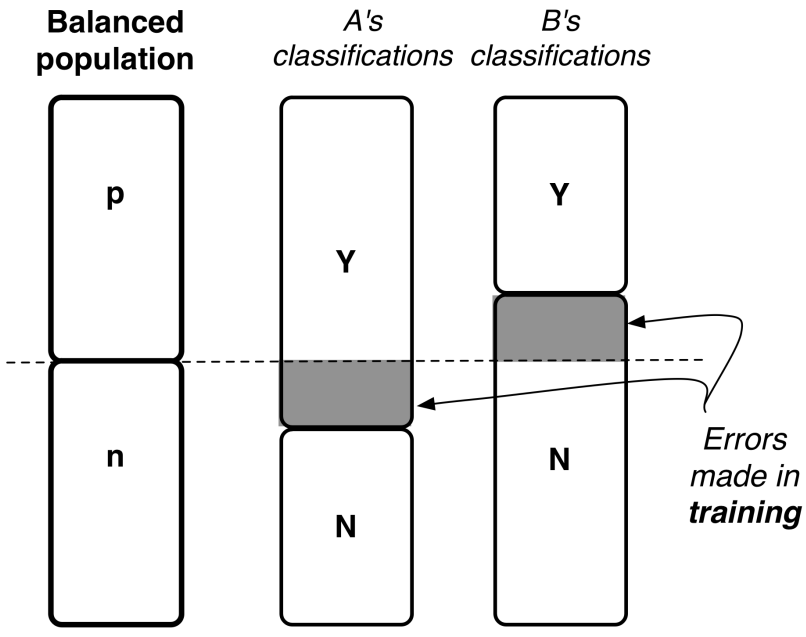
	respond	did not respond
Y	500	200
N	0	300

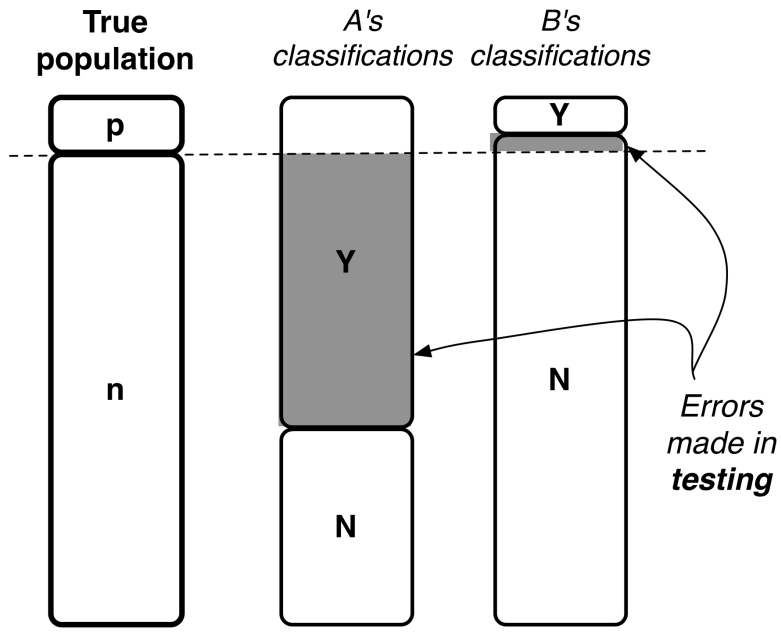
Confusion matrix of classifier B

	respond	did not respond
Y	500	0
N	200	300

They have the same accuracy. Which one would you prefer?

Note: It is common to train a classifier on a balanced training set.


Balanced population
A's classifications
B's classifications
p
n
Y
N
Y
N
Errors made in training


True population
A's classifications
B's classifications
p
n
Y
N
Y
N
Errors made in testing

Problems with Unequal Costs and Benefits

Another problem with miss-classification rate is that it does not make distinction between false positive and false negative errors.

Typically very different kinds of errors with different costs because of consequences of different severity.

Whatever the problem, it is hard to imagine a decision maker being indifferent to types of errors she makes.

- Ideally we should have estimates of costs and benefits of each decision a classifier can make.

Decision Theory and Expected Utility

Expected values are useful in organizing thinking about data-analytic problems.

It allows us to decompose data-analytic thinking into:

- the structure of the problem (possible decision outcomes)
- the elements of the analysis that can be extracted from data (probabilities of outcomes)
- the elements of the analysis that need to be extracted from other sources

Let us go back to the target marketing example.

We would like to assign each consumer a class:

- *likely responder*, or
- *not likely responder*.

We saw that a "common sense" threshold of 50% for deciding whether someone will likely respond would likely lead to targeting very few people.

Still, if, given x, the probability of a response is big, then it should make sense to target the customer.

Alternatively, if our model suggest there is a very low chance they will respond, it may be a waste of money.

How can we use expected values here?

- Let $p_R(x)$ be the estimated probability of response of a consumer described by x as an input.
- Let v_R be the benefit of a customer responding.
- Let v_{NR} be the cost of a customer not-responding.

$$\text{Expected benefit of targeting } = p_R(x) \cdot v_R + [1 - p_R(x)] \cdot v_{NR}$$

If the expected benefit of targeting is positive, you should target.

Suppose that is costs 80 cents to mail the promotion (in our tabloid example).

Suppose that (on average) a customer spends \$40, if they respond.

If they respond, your benefit is \$39.20.

If they do not respond, your cost is \$0.80.

$$p_R(x) \cdot (39.20) + [1 - p_R(x)] \cdot (-0.8) > 0 \quad \Longrightarrow \quad p_R(x) > 0.02$$

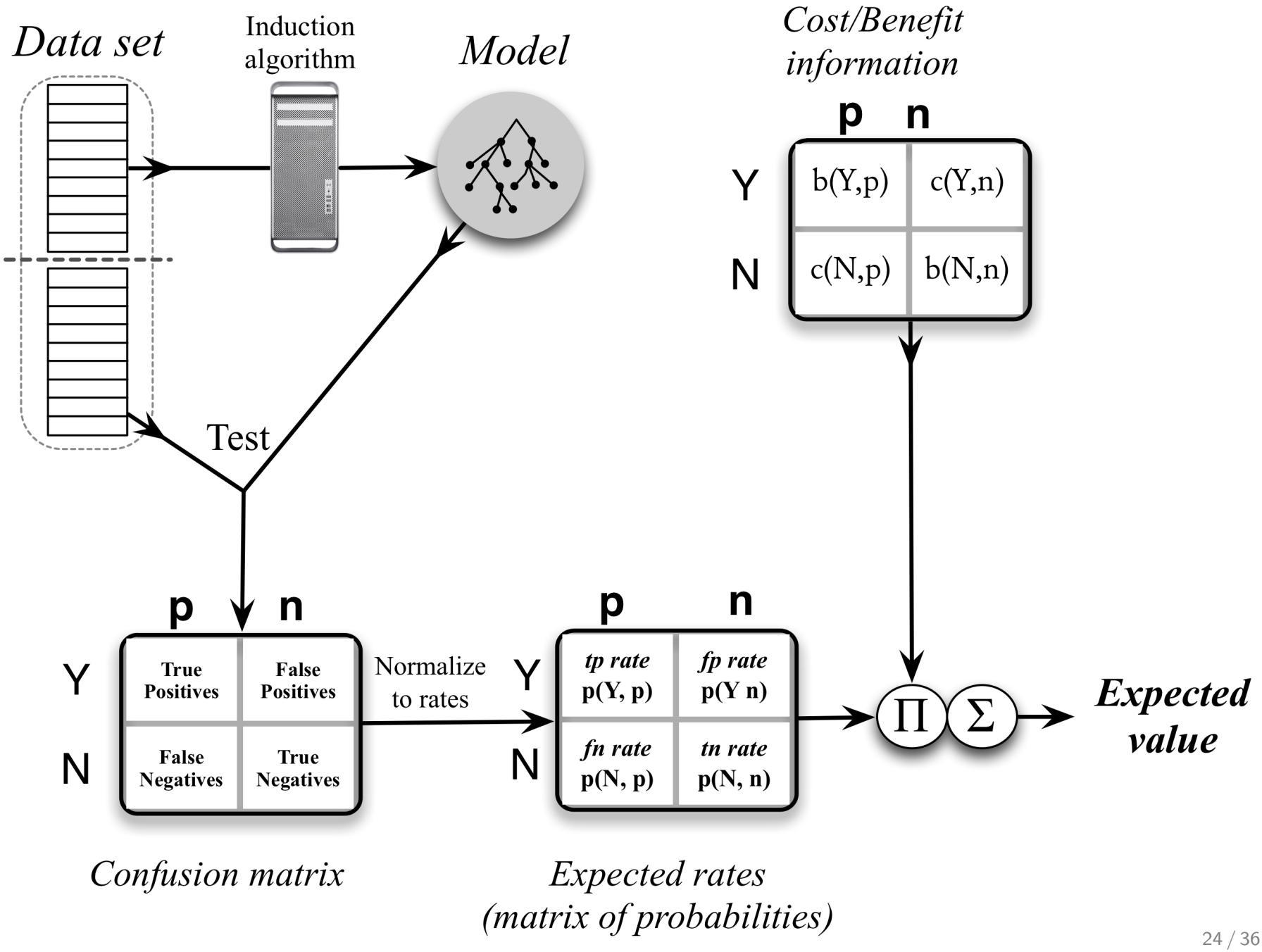
If $p_R(x) > 0.02$, we should do targeting.

Expected Value for Classifier Evaluation

We have just seen how to use expected value framework to evaluate an individual decision.

However, we will use a model to make a set of decisions by applying it to a collection of examples.

We need to look at how to evaluate and compare classifiers when applied to a particular problem.


Data set
Induction algorithm
Model
Cost/Benefit information
p
n
Y
N
b(Y,p)
c(Y,n)
c(N,p)
b(N,n)
Test
p
n
Y
N
True Positives
False Positives
False Negatives
True Negatives
Normalize to rates
p
n
Y
N
tp rate p(Y, p)
fp rate p(Y n)
fn rate p(N, p)
tn rate p(N, n)
Π
Σ
Expected value
Confusion matrix
Expected rates (matrix of probabilities)

Other evaluation metrics

You will encounter a number of evaluation metrics in data science.

- All of them fundamentally summarize the confusion matrix.

	positive	**n**egative
Y	True positives	False negatives
N	False positives	True negatives

True positive rate = TP / (TP + FN) — frequency of being correct
False negative rate = FN / (TP + FN) — frequency of being incorrect

Sensitivity = TN / (TN + FP)
Specificity = TP / (TP + FN) = *True positive rate*

See: `https://en.wikipedia.org/wiki/Sensitivity_and_specificity`

Visualizing Model Performance

The expected profit framework takes a specific set of conditions and generates a single number that represents a profit.

There are a lot of assumptions and details that go into this calculation.

- knowledge of costs and benefits
- accurate estimates of probabilities

Stakeholders outside data science team may have little patience for detail, so we want to provide a higher-level, more intuitive view of model performance.

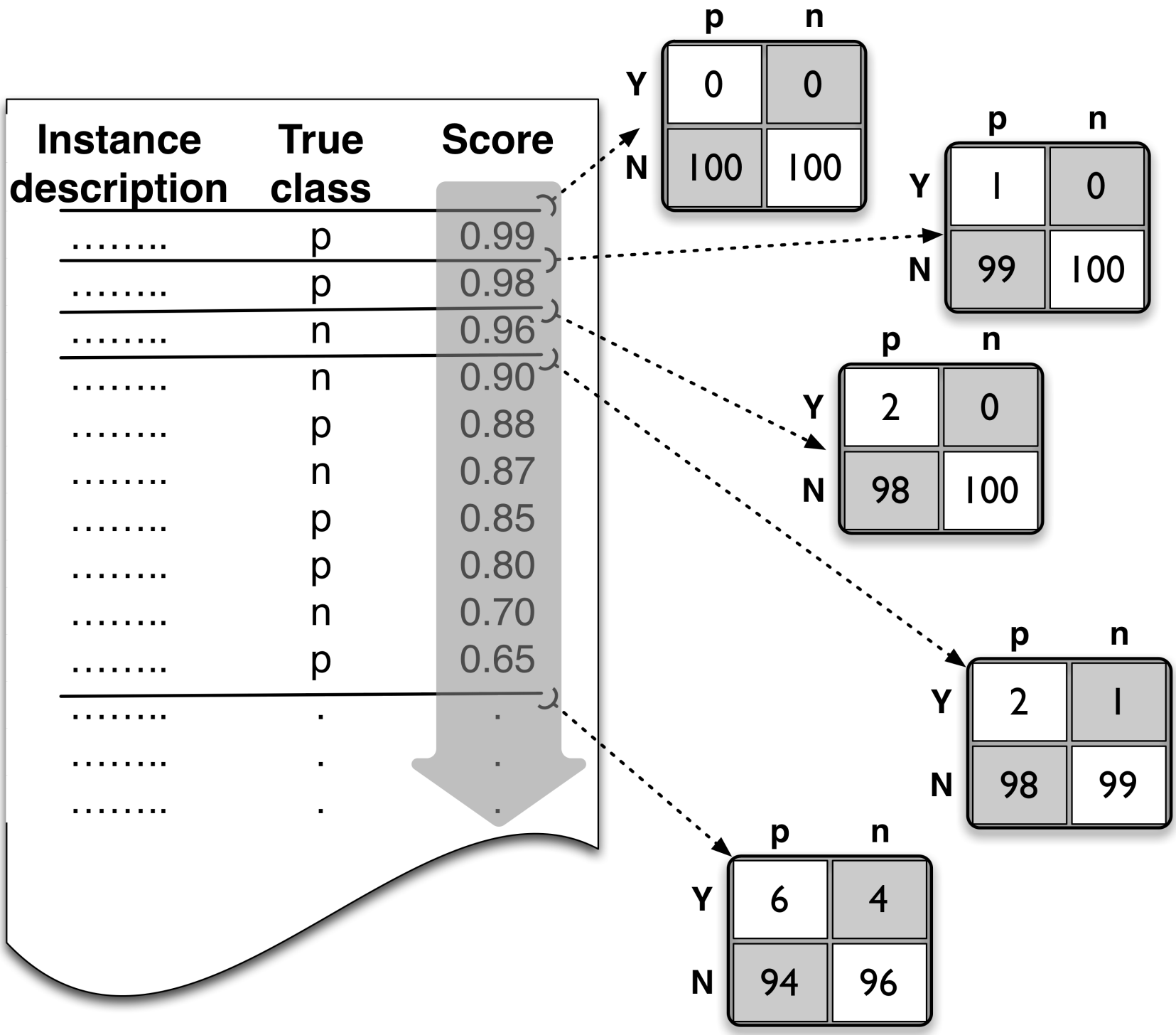
We have seen how the score assigned by a model can be used to compute a decision for each individual case based on its expected value.

A different strategy for making decisions is to rank a set of cases by these scores, and then take actions on the cases at the top of the ranked list.

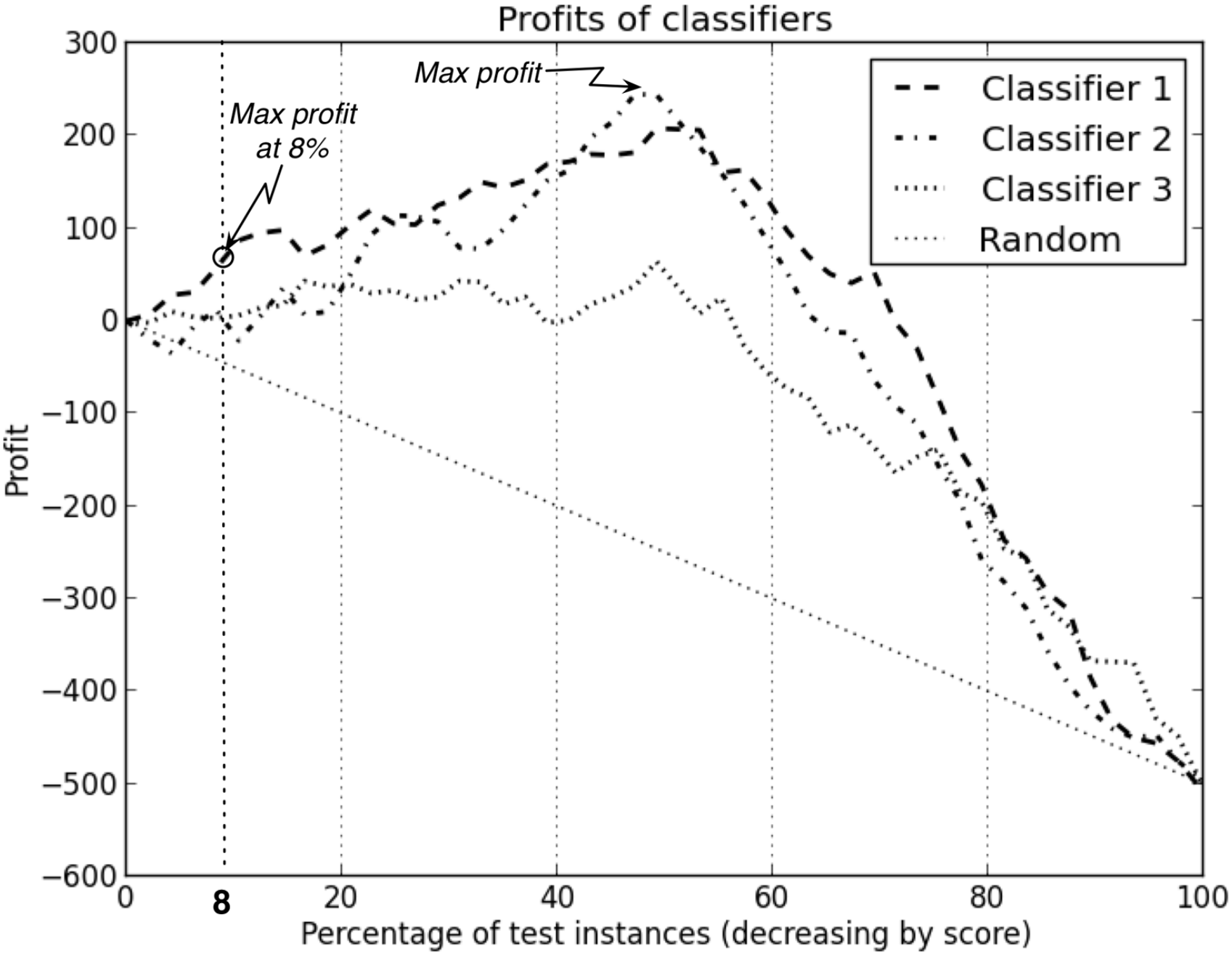
- we may decide to take the top n cases;
- or, equivalently, all cases that score above a given threshold.

This is a good idea/useful when:

- we use classifiers that do not provide true probabilities,
- estimated probabilities are not accurate,
- problems where you have a budget.


Instance description
True class
Score
…….. p 0.99
…….. p 0.98
…….. n 0.96
…….. n 0.90
…….. p 0.88
…….. n 0.87
…….. p 0.85
…….. p 0.80
…….. n 0.70
…….. p 0.65
…….. . .
…….. . .
…….. . .
p n
Y 0 0
N 100 100
p n
Y 1 0
N 99 100
p n
Y 2 0
N 98 100
p n
Y 2 1
N 98 99
p n
Y 6 4
N 94 96

Profit curves



The ROC Curve

Stands for Receiver Operating Characteristic.

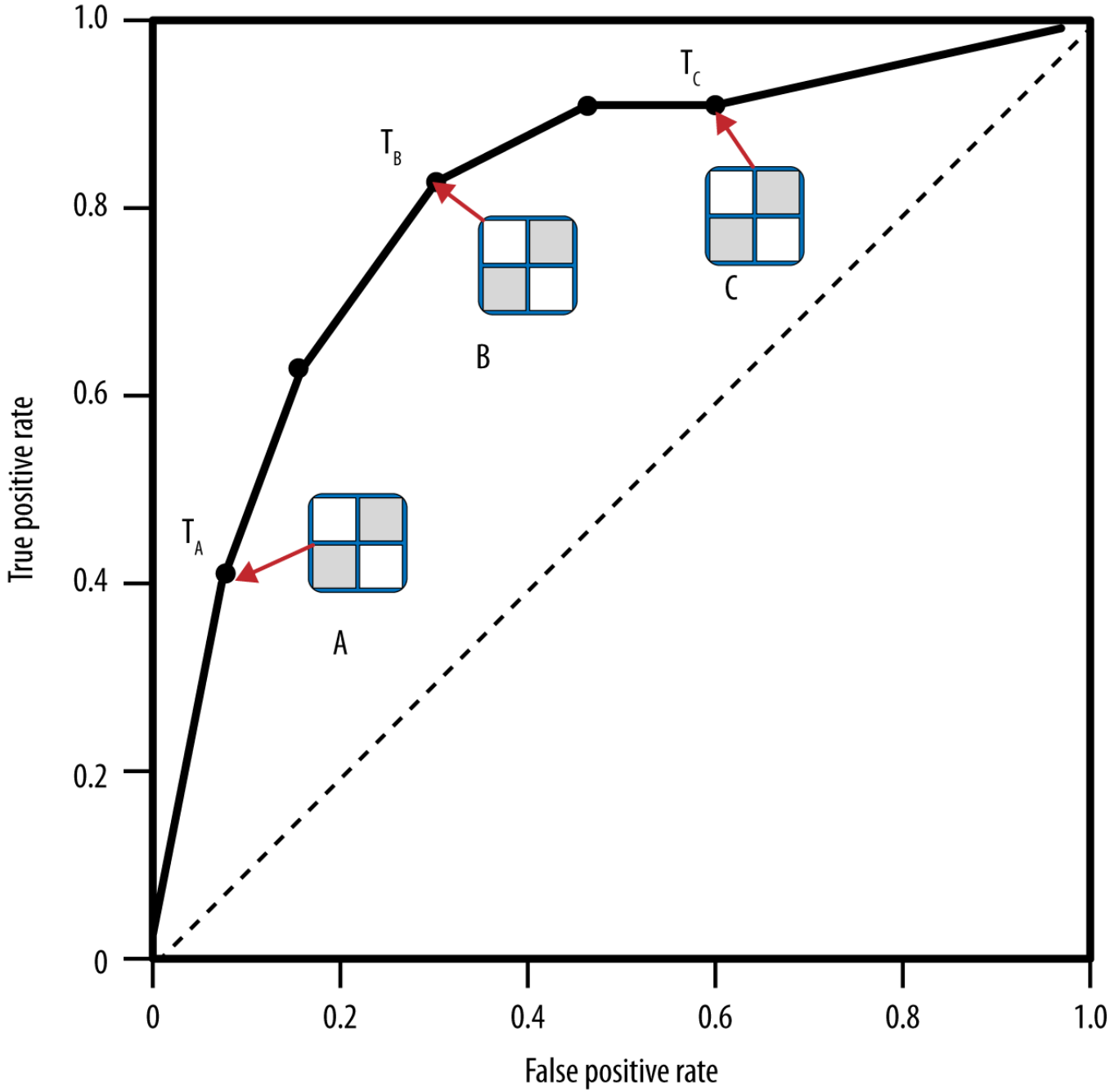
- This used widely in signal processing.

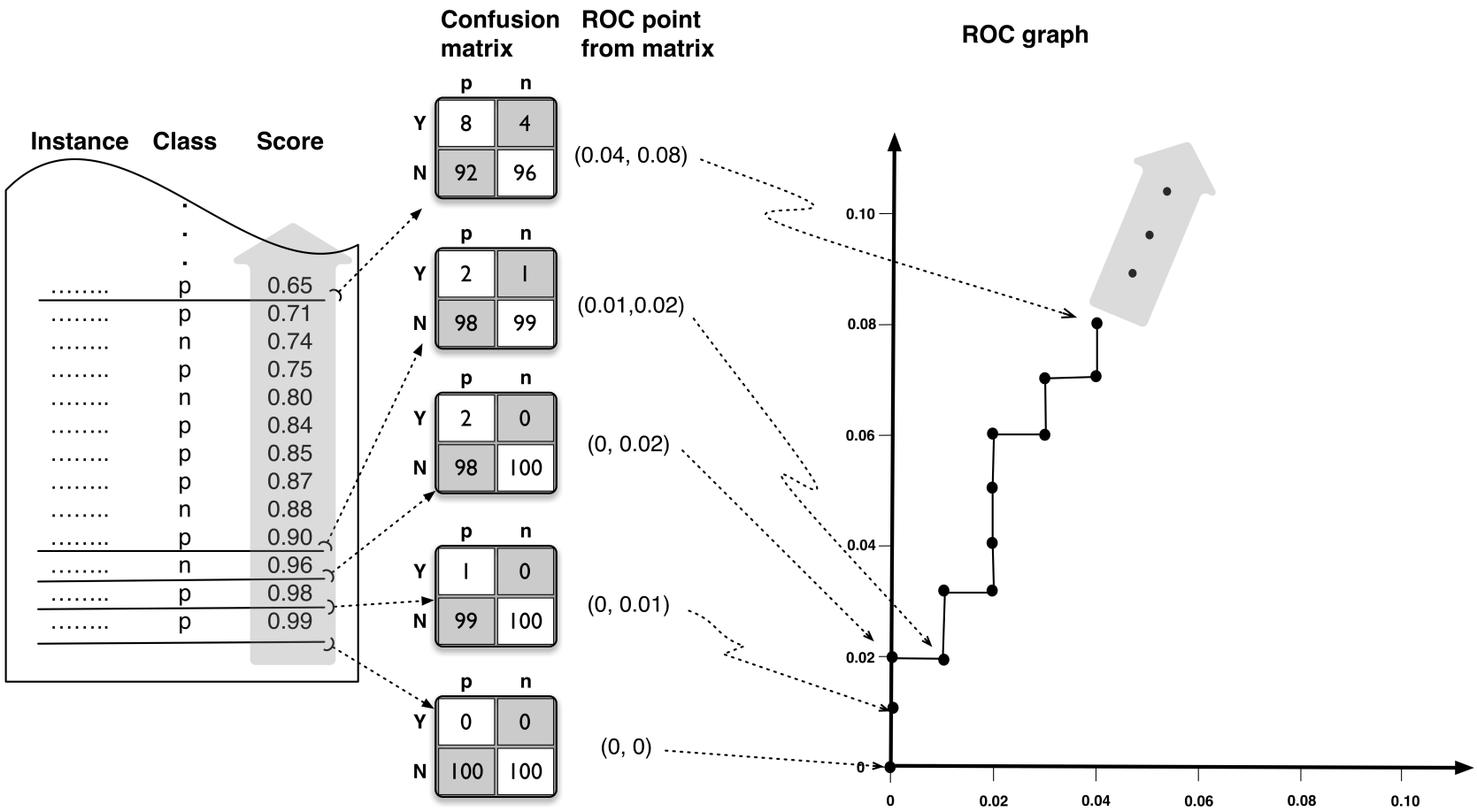
There are two conditions underlying the profit calculation:

- the proportion of positive and negative instances in the target population is known,
- the costs and benefits are known.

If the above are known and are expected to be stable, profit curves may be a good choice for visualization.

The ROC curve shows the entire space of performance possibilities by depicting relative trade-offs that a classifier makes between *benefits* (true positives) and *costs* (false positives).


1.0
0.8
0.6
0.4
0.2
0
True positive rate
0
0.2
0.4
0.6
0.8
1.0
False positive rate
T_A
T_B
T_C
A
B
C

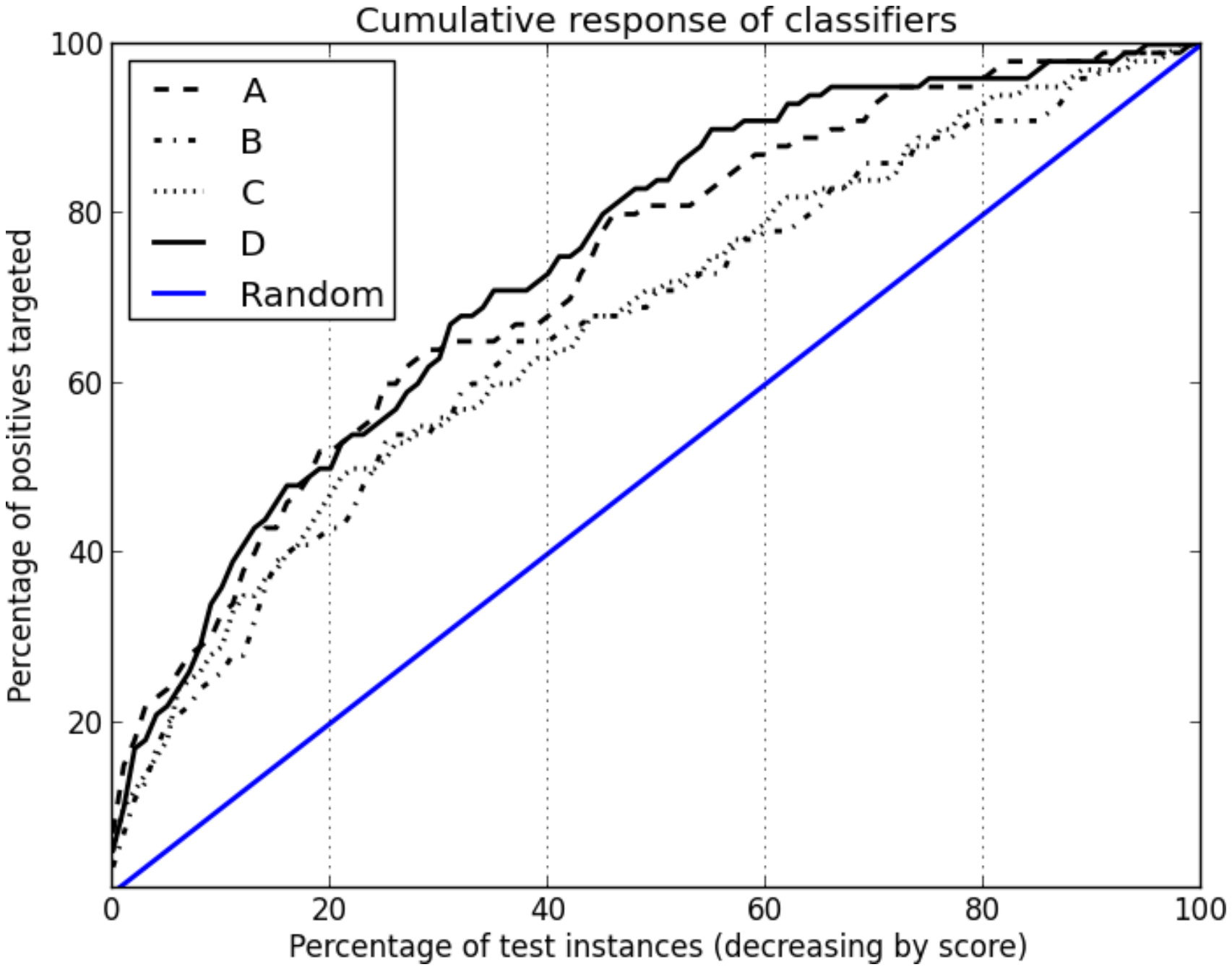

Instance
Class
Score
Confusion matrix
ROC point from matrix
ROC graph
p
n
Y
N
8
4
92
96
(0.04, 0.08)
2
1
98
99
(0.01,0.02)
2
0
98
100
(0, 0.02)
1
0
99
100
(0, 0.01)
0
0
100
100
(0, 0)
0.65
0.71
0.74
0.75
0.80
0.84
0.85
0.87
0.88
0.90
0.96
0.98
0.99
0.10
0.08
0.06
0.04
0.02
0
0
0.02
0.04
0.06
0.08
0.10

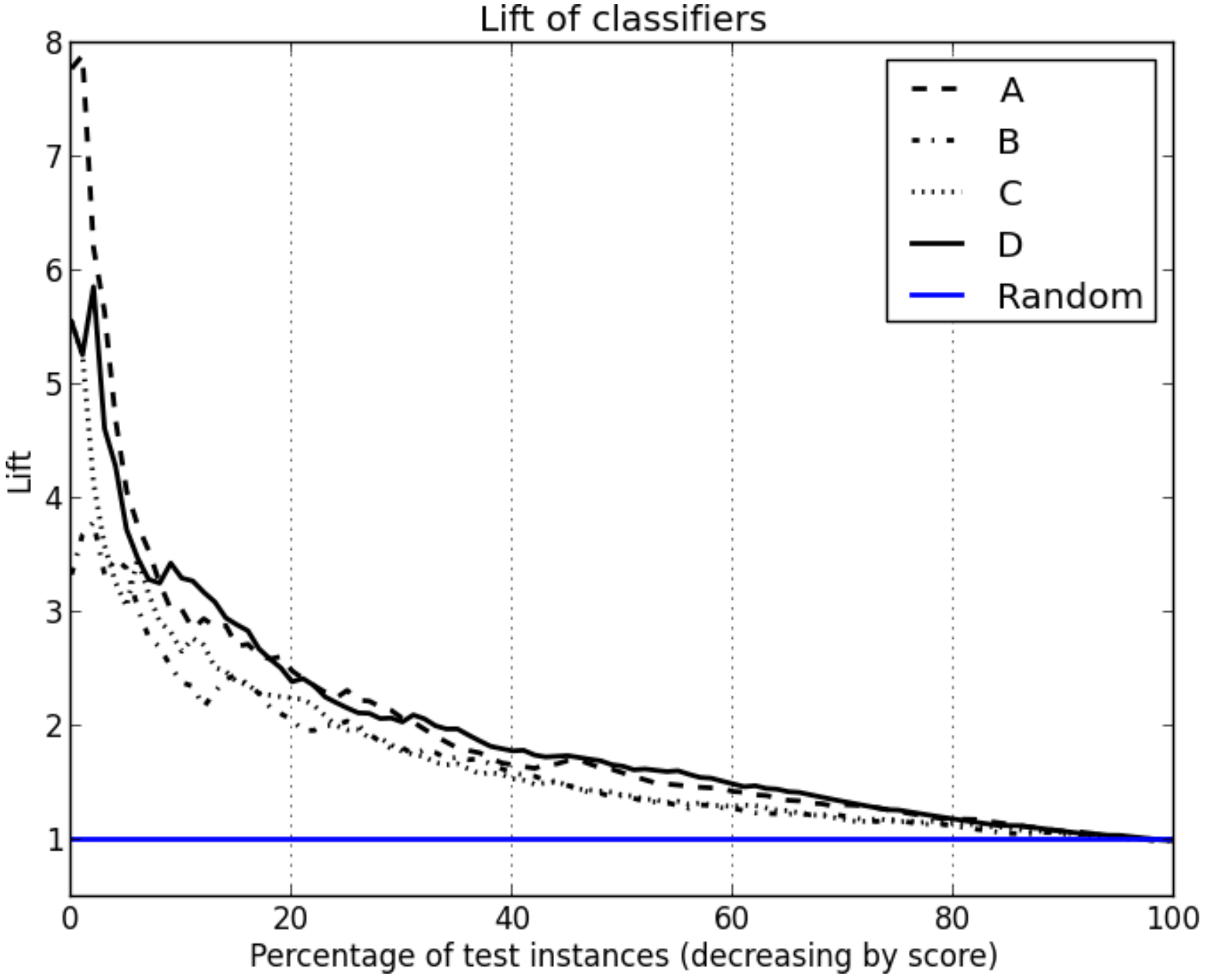
An advantage of ROC graphs is that they decouple classifier performance from the conditions under which the classifiers will be used.

- They are independent of the class proportions as well as the costs and benefits.

An important summary statistic is the *area under the ROC curve* (AUC).

- ROC curve provides more information than its area,
- the AUC is useful when a single number is needed to summarize performance, or
- when nothing is known about the operating conditions.


Cumulative response of classifiers
A
B
C
D
Random
Percentage of positives targeted
Percentage of test instances (decreasing by score)
0
20
40
60
80
100


Lift of classifiers
A
B
C
D
Random
Lift
Percentage of test instances (decreasing by score)
0
20
40
60
80
100
1
2
3
4
5
6
7
8

Summary

Evaluation of classifiers is a critical part of analysis.

Conveying information about evaluation to stakeholders is equally important.

We have seen how different losses are used during training and testing phase.

Visualization techniques can be used to summarize performance of different procedures.