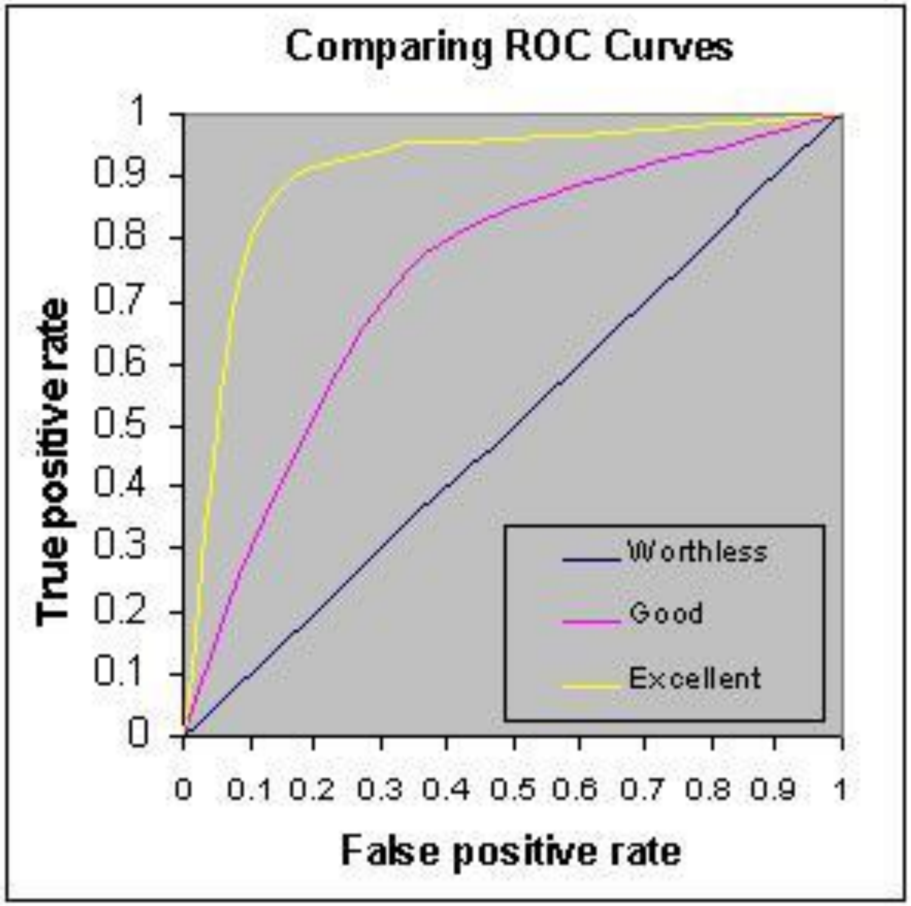
Classifier Performance and Model Selection

Overview

- Evaluating the performance of a classifier
  - Confusion Matrix
  - **ROC** curve
- Selecting a model for the problem
  - Bias / Variance tradeoff
  - Cross Validation

Receiver Operating Characteristic Methodology



Introduction to ROC curves

- ***ROC*** = ***R**eceiver **O**perating **C**haracteristic*
- Started in electronic signal detection theory (1940s - 1950s)
- Has become very popular in biomedical applications, particularly radiology and imaging
- Also used in **machine learning applications** to assess classifiers
- Can be used to compare tests/procedures

ROC curves: simplest case

- Consider diagnostic **test** for a disease
- Test has 2 possible outcomes:
  - ‘**positive’** = suggesting presence of disease
  - ‘**negative**’
- An individual can test either positive or negative for the disease

Hypothesis testing refresher

- 2 ‘competing theories’ regarding a population parameter:
  - *NULL* hypothesis *H* (‘straw man’)
  - ALTERNATIVE hypothesis *A* (‘claim’, or theory you wish to test)
- *H:* NO DIFFERENCE
  - any observed deviation from what we expect to see is due to *chance variability*
- *A:* THE DIFFERENCE IS *REAL*

Test statistic

- Measure how far the observed data are from what is expected *assuming the NULL H* by computing the value of a *test statistic* (TS) from the data
- The particular TS computed depends on the parameter
- For example, to test the population mean μ, the TS is the sample mean (or standardized sample mean)
- The NULL is rejected if the TS falls in a user-specified ‘rejection region’

Types of errors

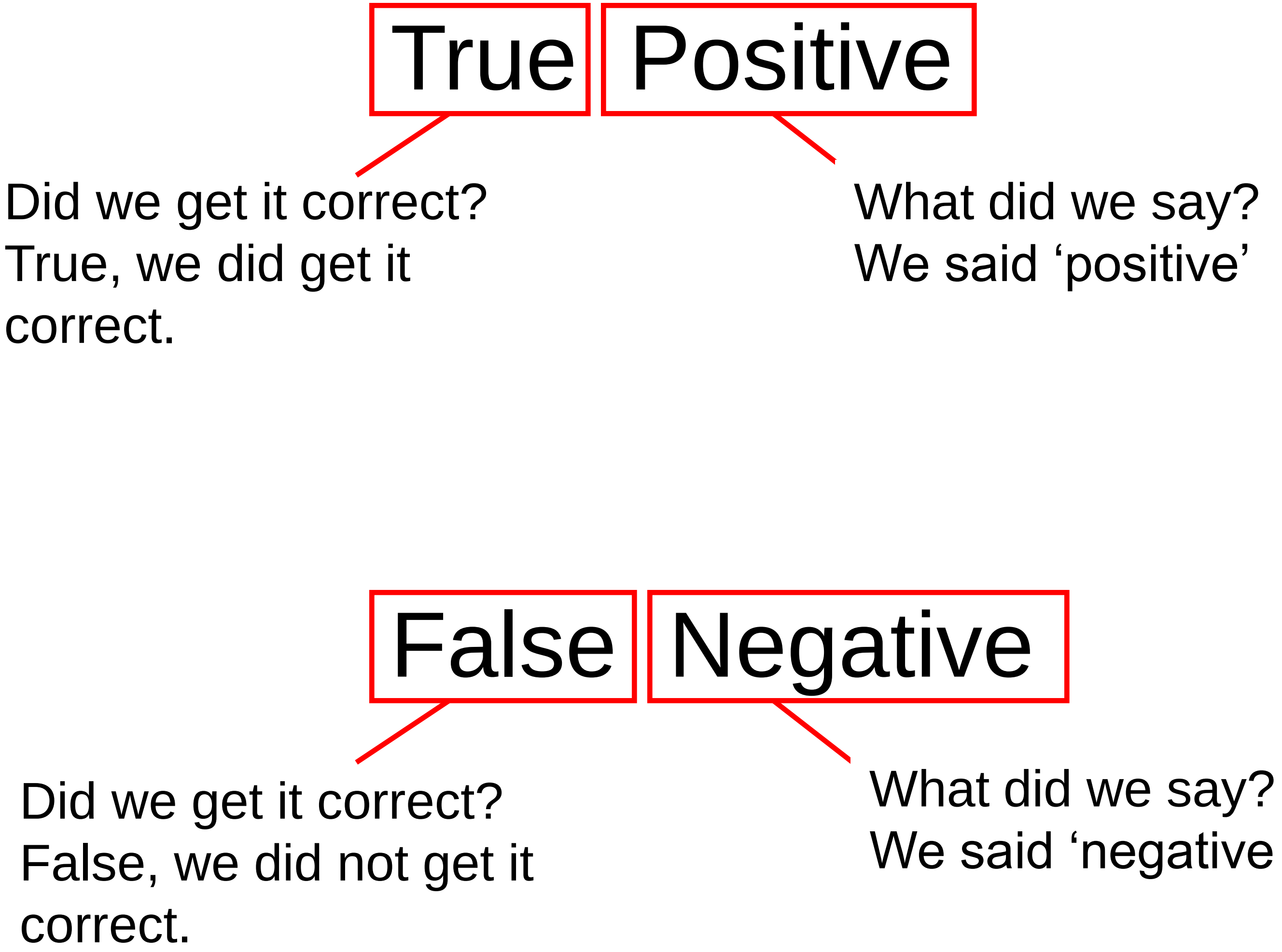
		True class	
		p	n
Hypothesized class	Y	True Positives	False Positives
	N	False Negatives	True Negatives
Column totals:		P	N

$$\text{FP rate} = \frac{\text{FP}}{\text{N}} \qquad \text{TP rate} = \frac{\text{TP}}{\text{P}} = \text{Recall}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \qquad \text{F-score} = \text{Precision} \times \text{Recall}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{P} + \text{N}}$$

Two parts to each: whether you got it correct or not, and what you guessed.



True Positive

Did we get it correct? True, we did get it correct.

What did we say? We said ‘positive’

False Negative

Did we get it correct? False, we did not get it correct.

What did we say? We said ‘negative

Sensitivity and Specificity

Count up the total number of each label (TP, FP, TN, FN) over a large dataset. In ROC analysis, we use two statistics:

$$\textbf{Sensitivity} = \frac{TP}{TP+FN}$$

Can be thought of as the likelihood of spotting a positive case when presented with one.

$$\textbf{Specificity} = \frac{TN}{TN+FP}$$

Can be thought of as the likelihood of spotting a negative case when presented with one.

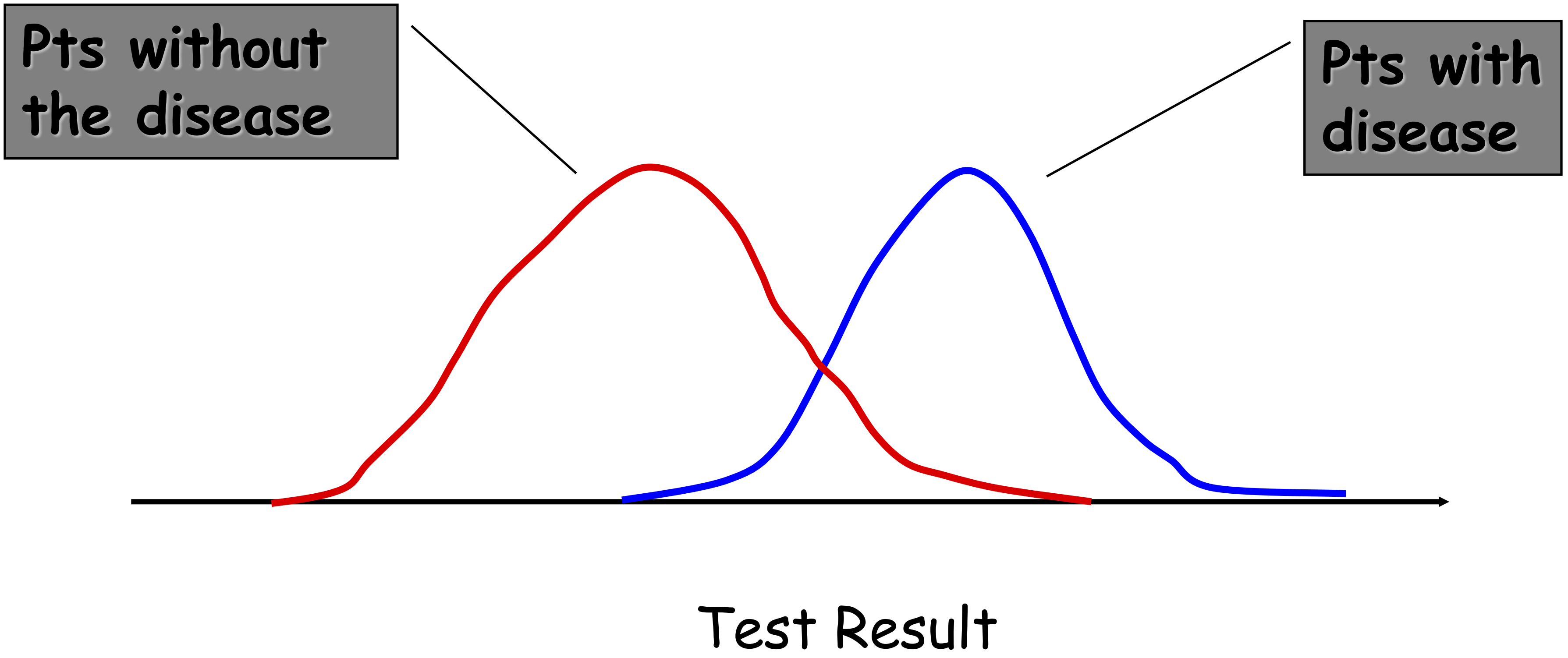
		Ground Truth	
		1	0
Prediction	1	TRUE POS 60	FALSE POS 80
	0	FALSE NEG. 30	TRUE NEG. 20

60+30 = 90 cases in the dataset were class 1

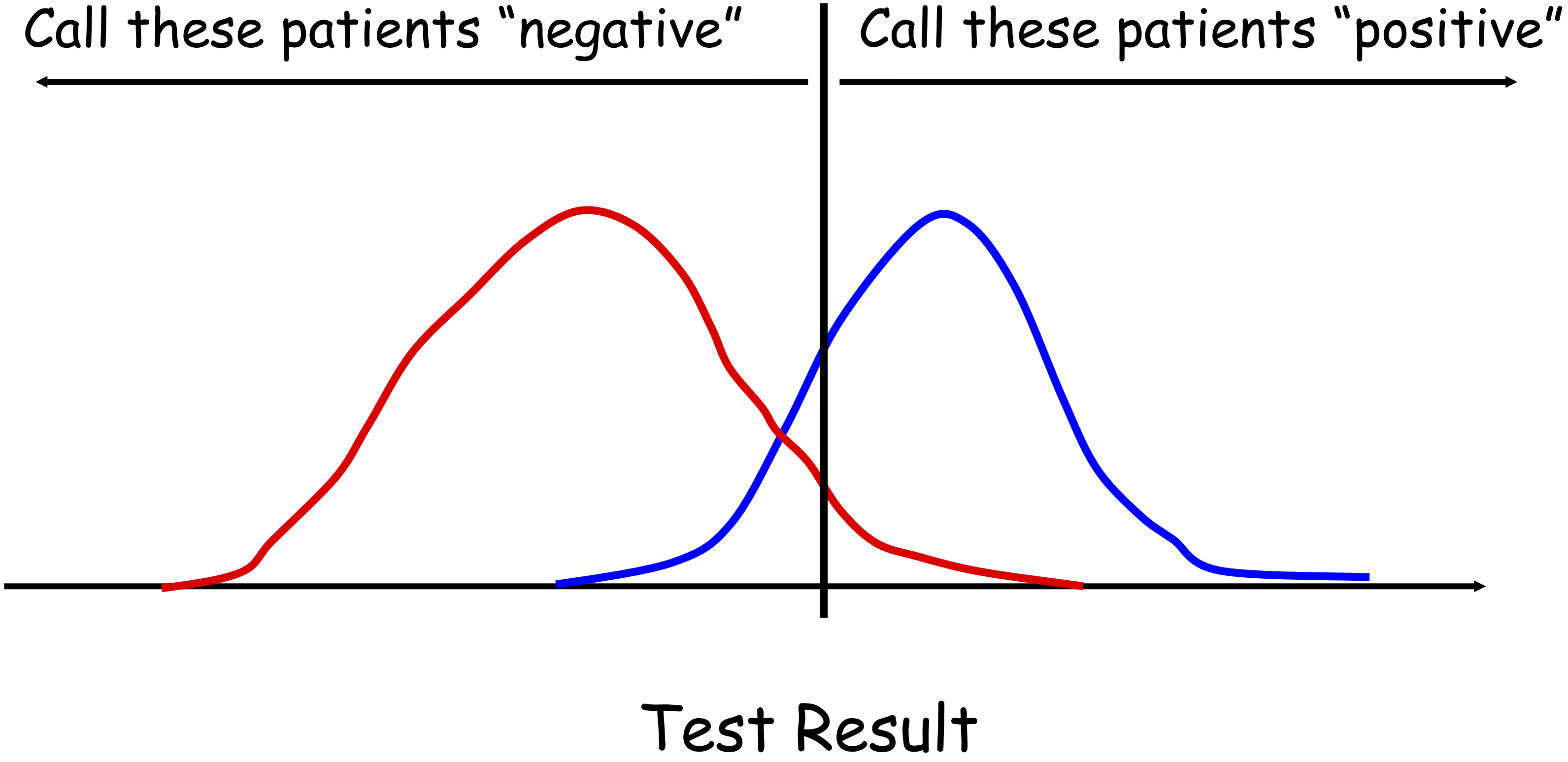
80+20 = 100 cases in the dataset were class 0

90+100 = 190 examples in the data overall

Specific Example



Threshold

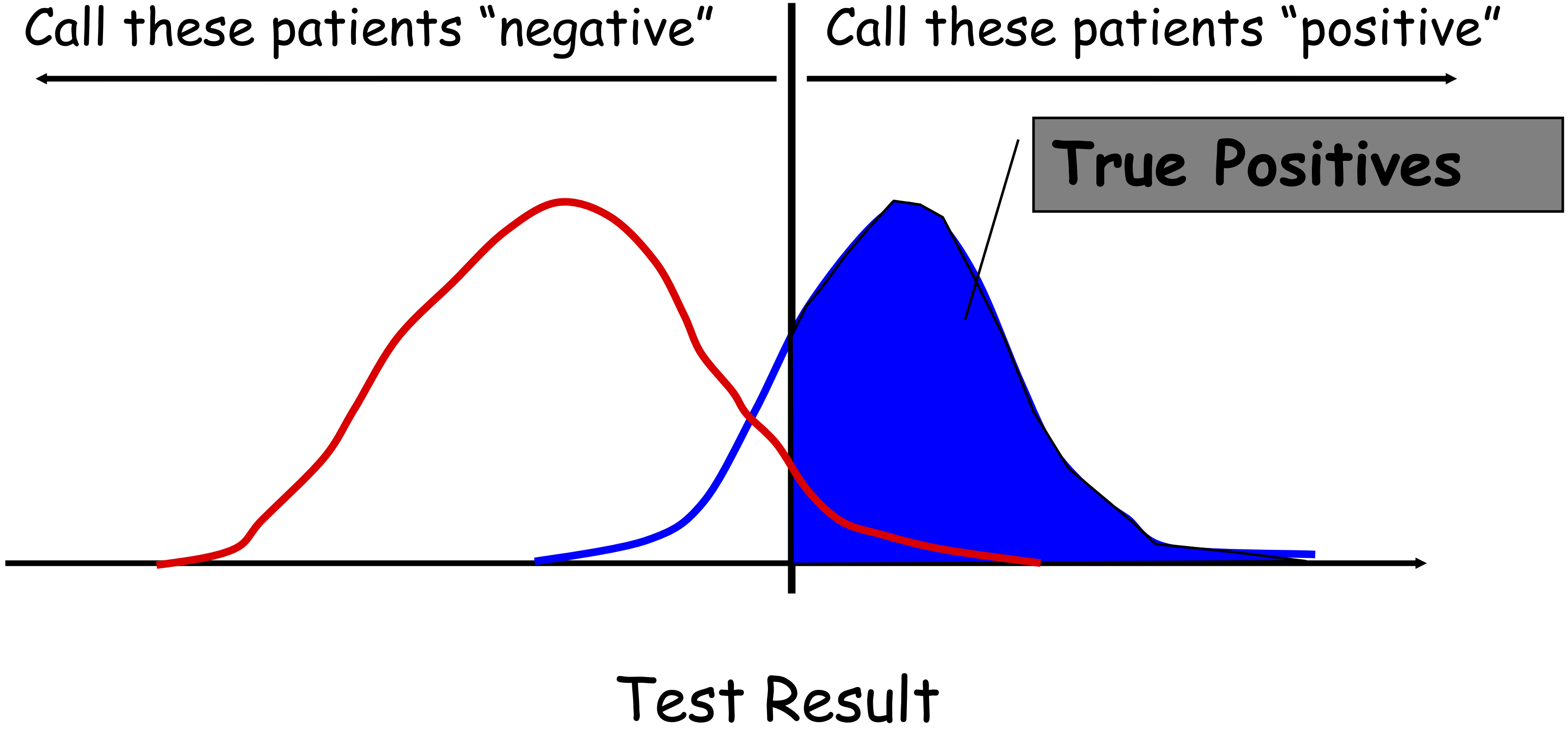


Call these patients “negative”

Call these patients “positive”

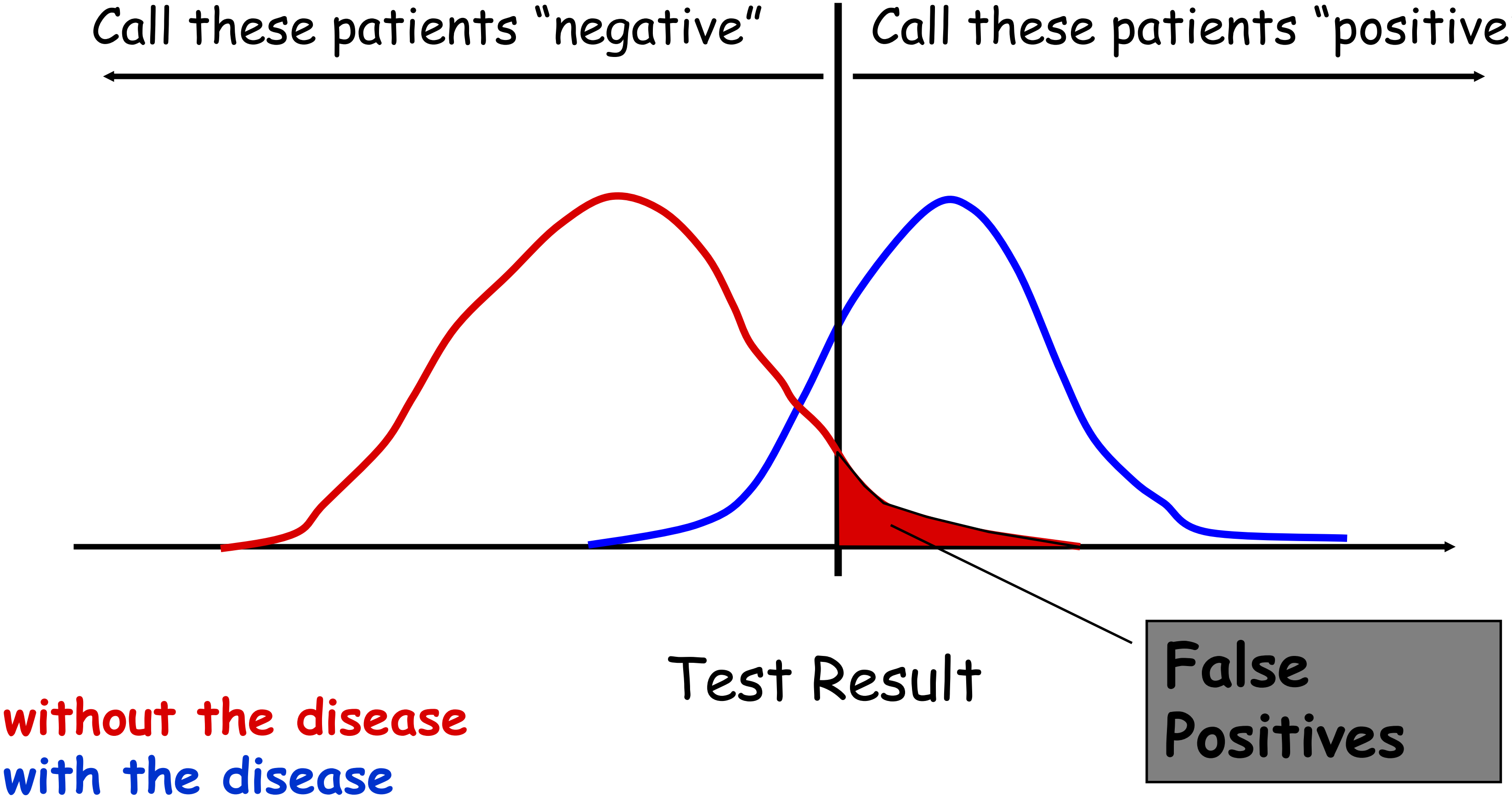
Test Result

Some definitions ...



without the disease

with the disease



Call these patients “negative”

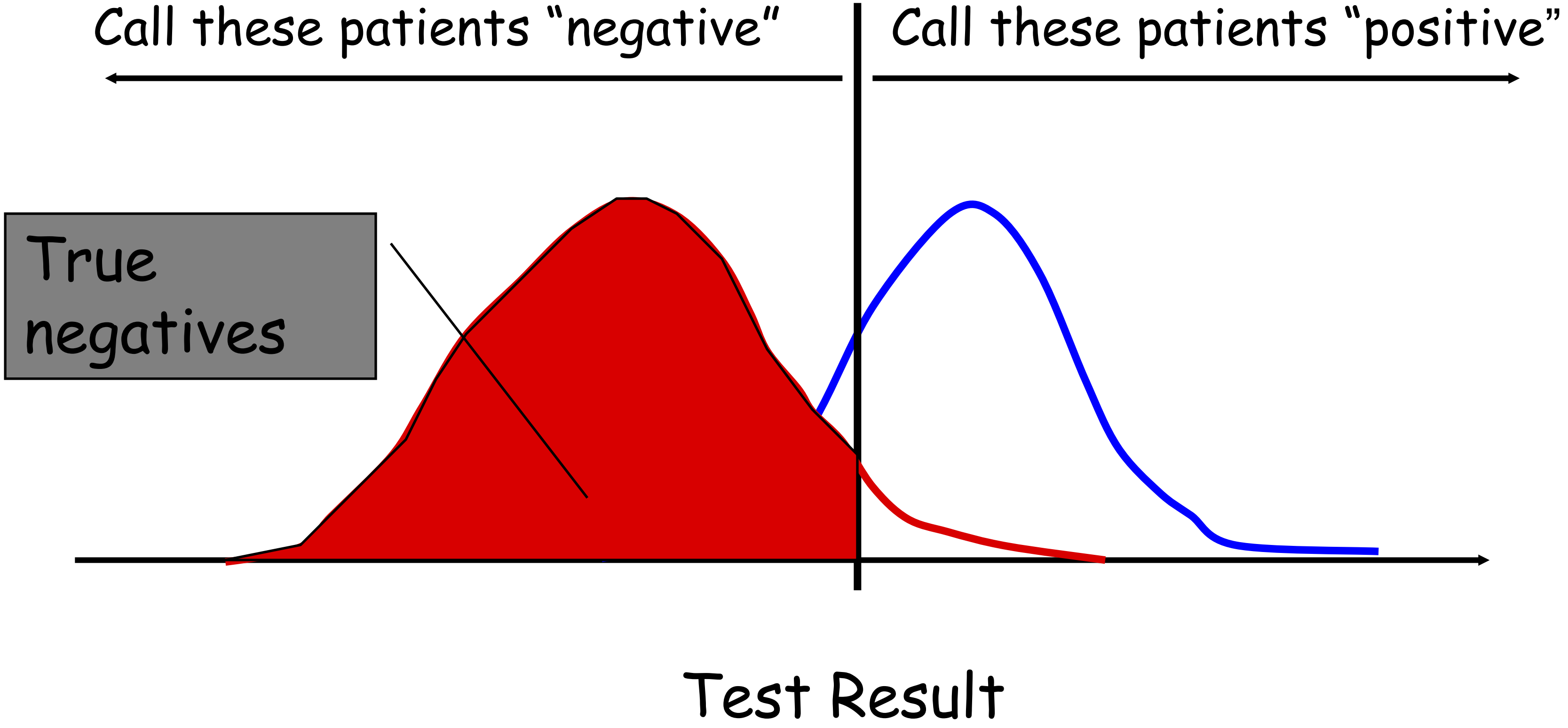
Call these patients “positive”

Test Result

False Positives

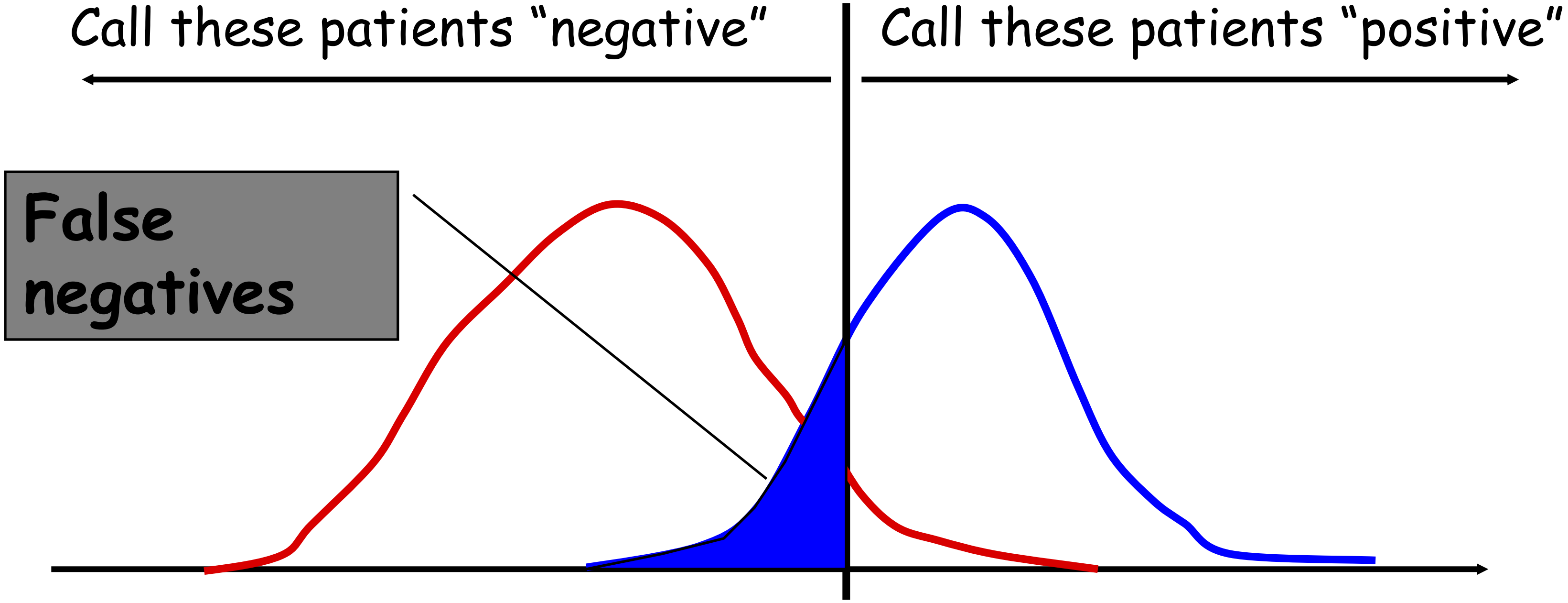
without the disease

with the disease



without the disease

with the disease



Call these patients "negative"

Call these patients "positive"

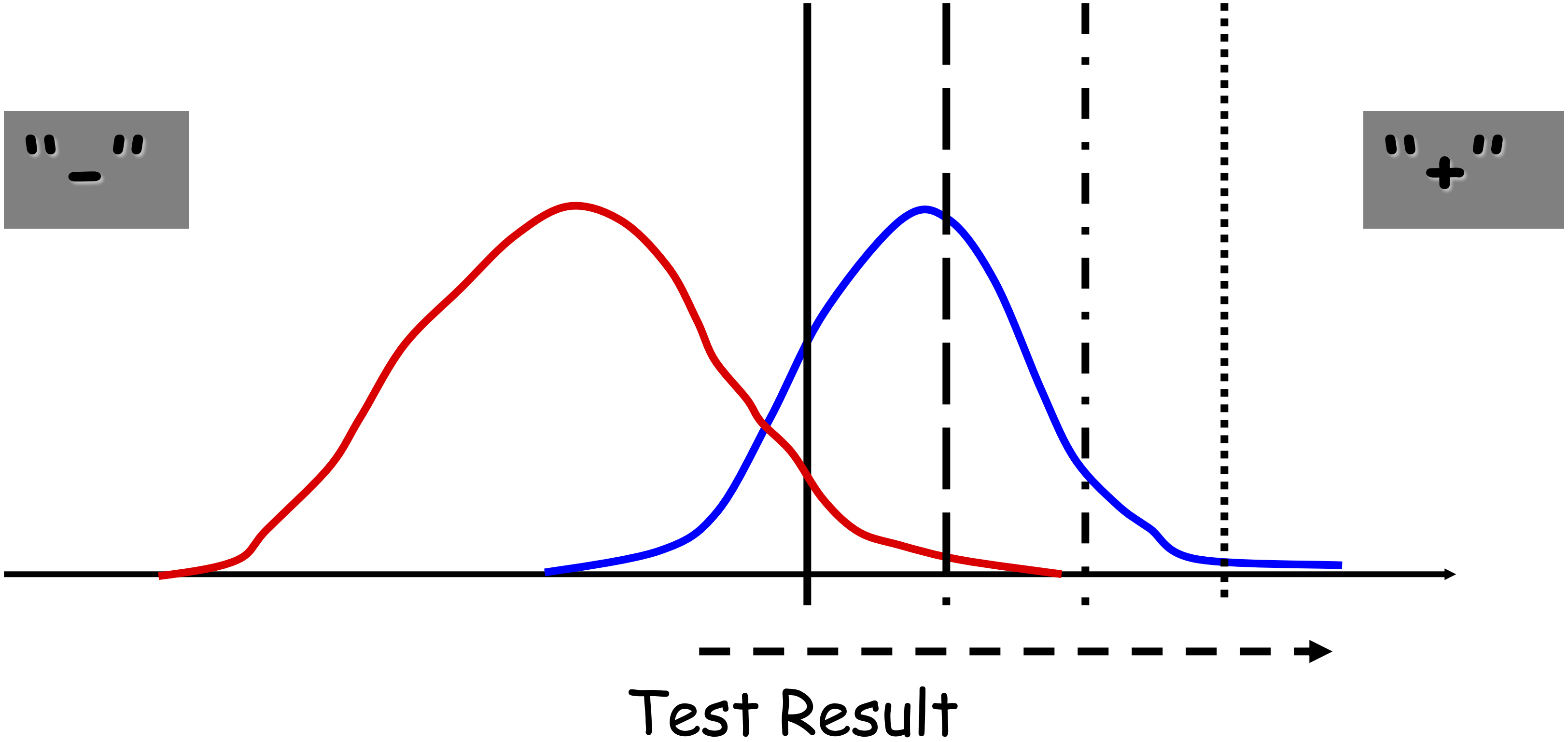
False negatives

Test Result

without the disease

with the disease

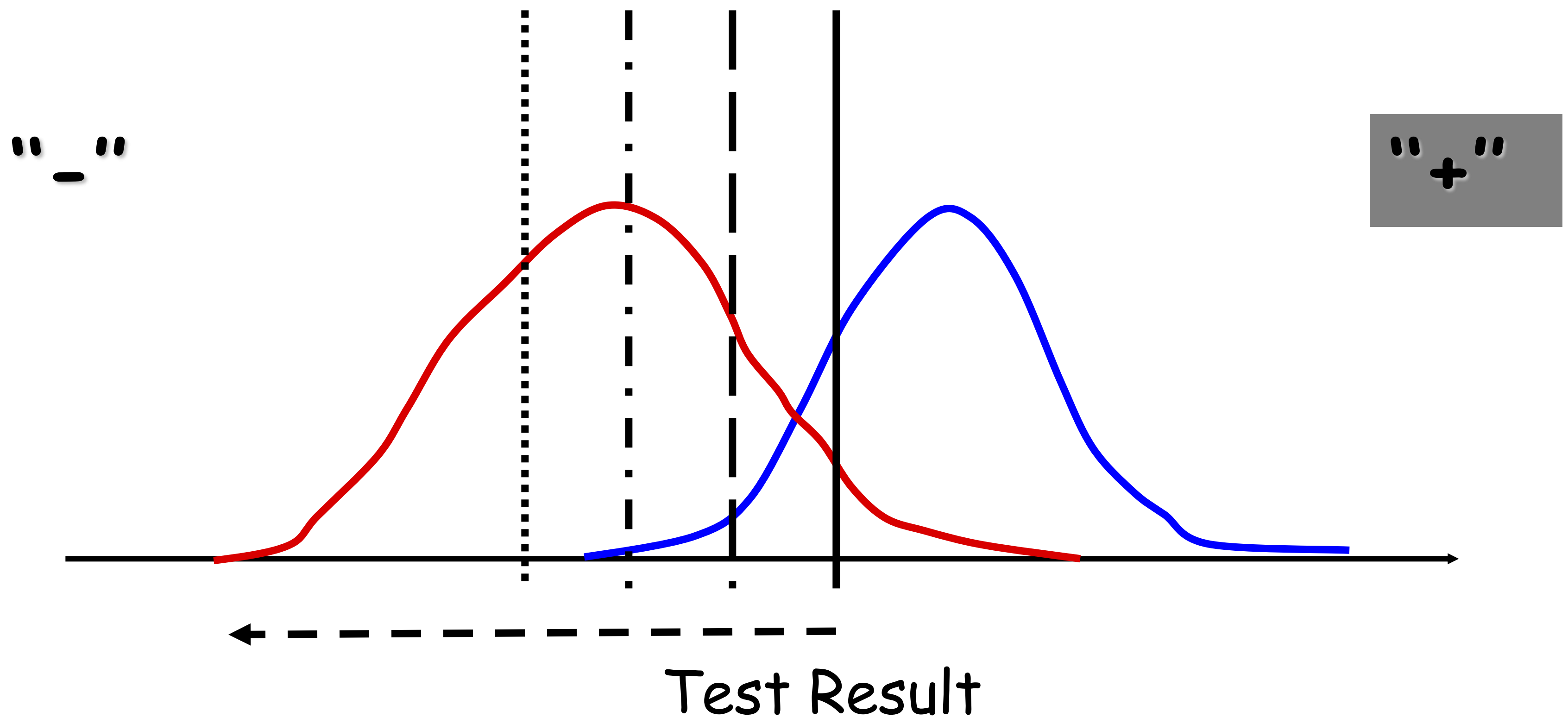
Moving the Threshold: right



without the disease

with the disease

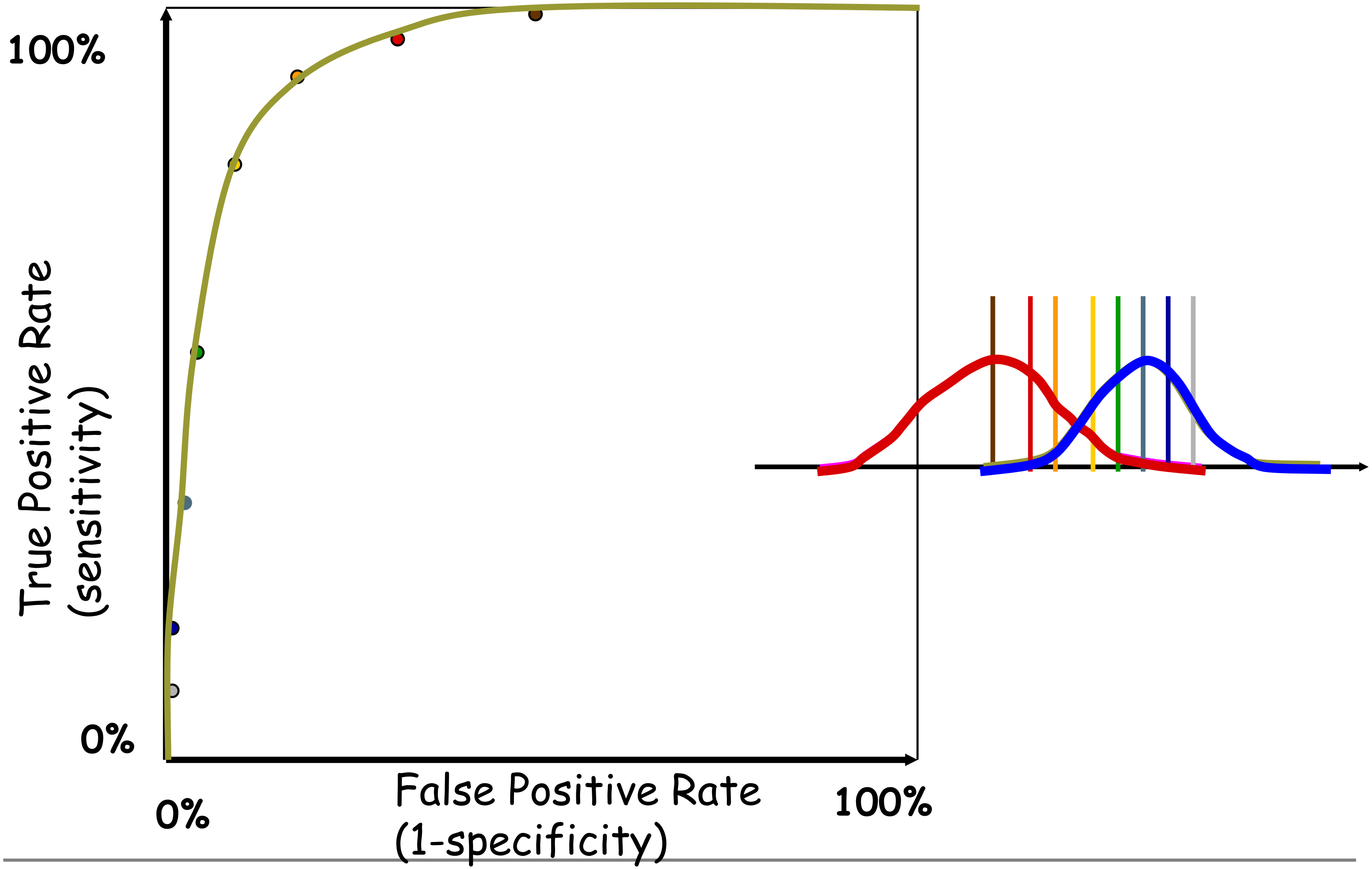
Moving the Threshold: left



without the disease

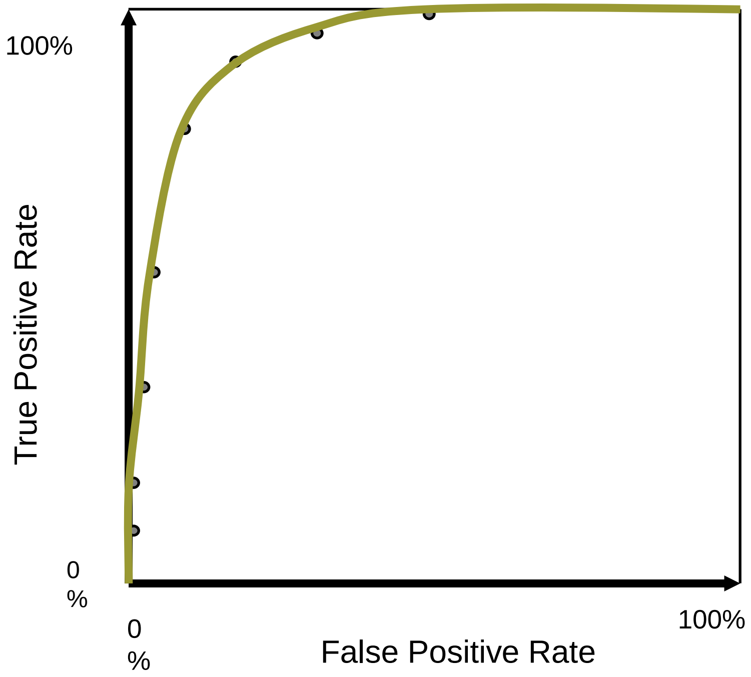
with the disease

ROC curve

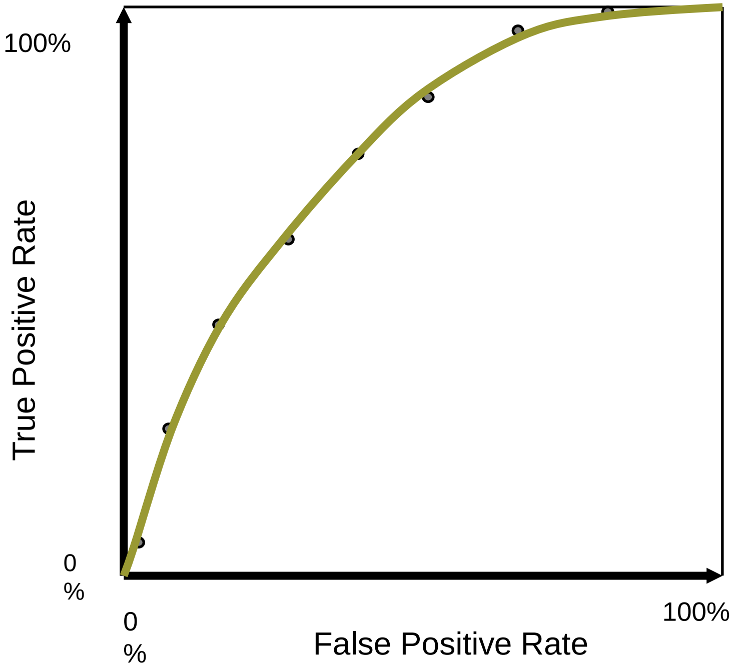

100%
True Positive Rate
(sensitivity)
0%
0%
False Positive Rate
(1-specificity)
100%

ROC curve comparison

A good test:

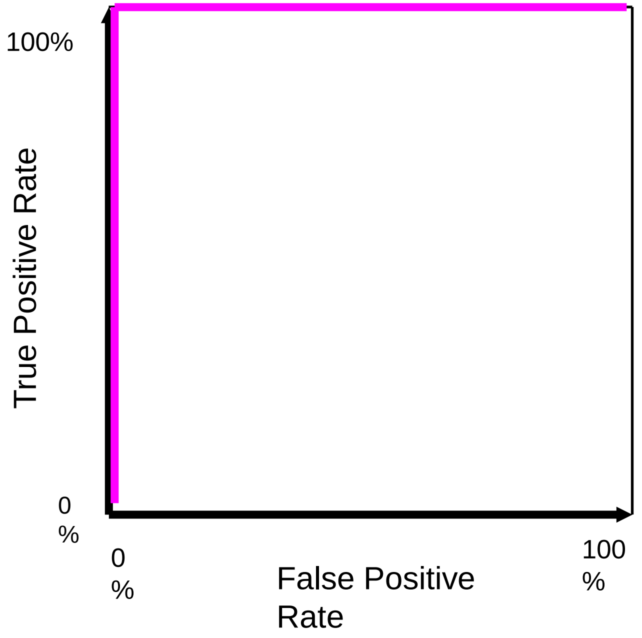


A poor test:



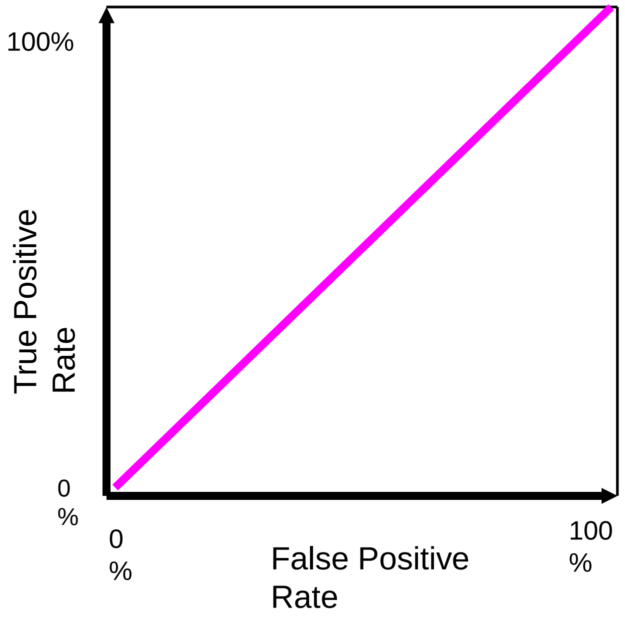
ROC curve extremes

Best Test:



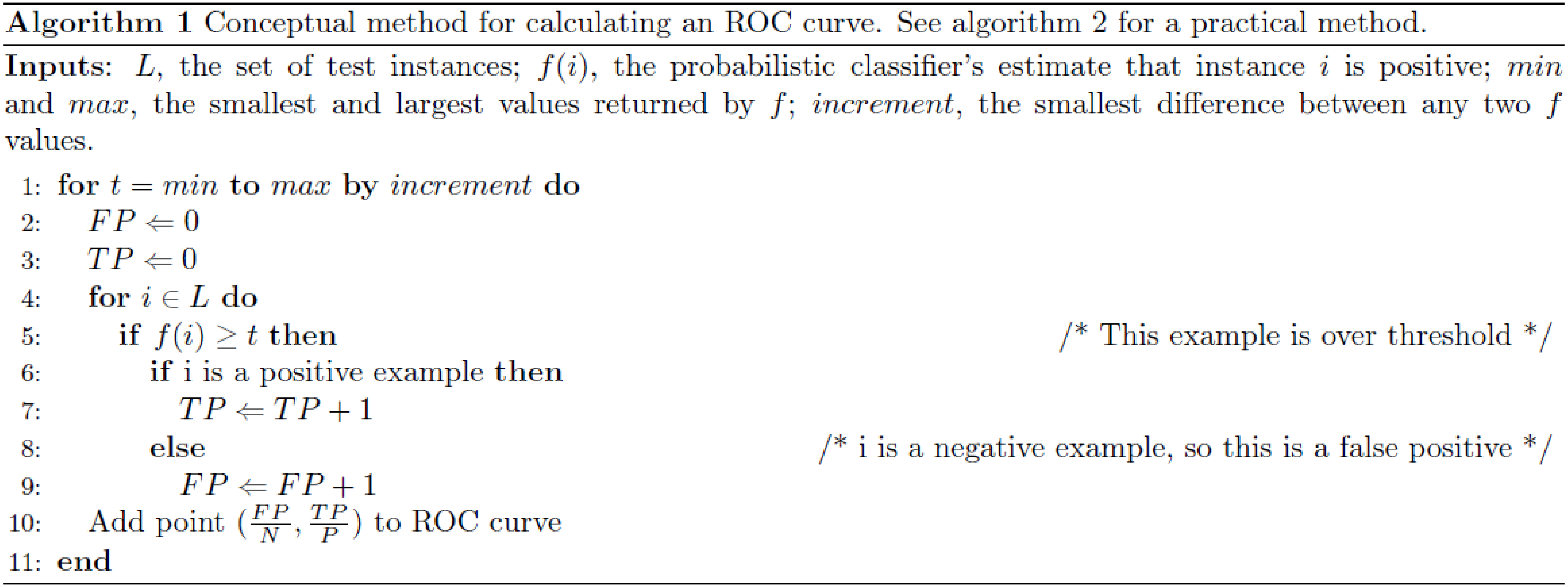
The distributions don't overlap at all

Worst test:



The distributions overlap completely

ROC Algorithm Conceptually



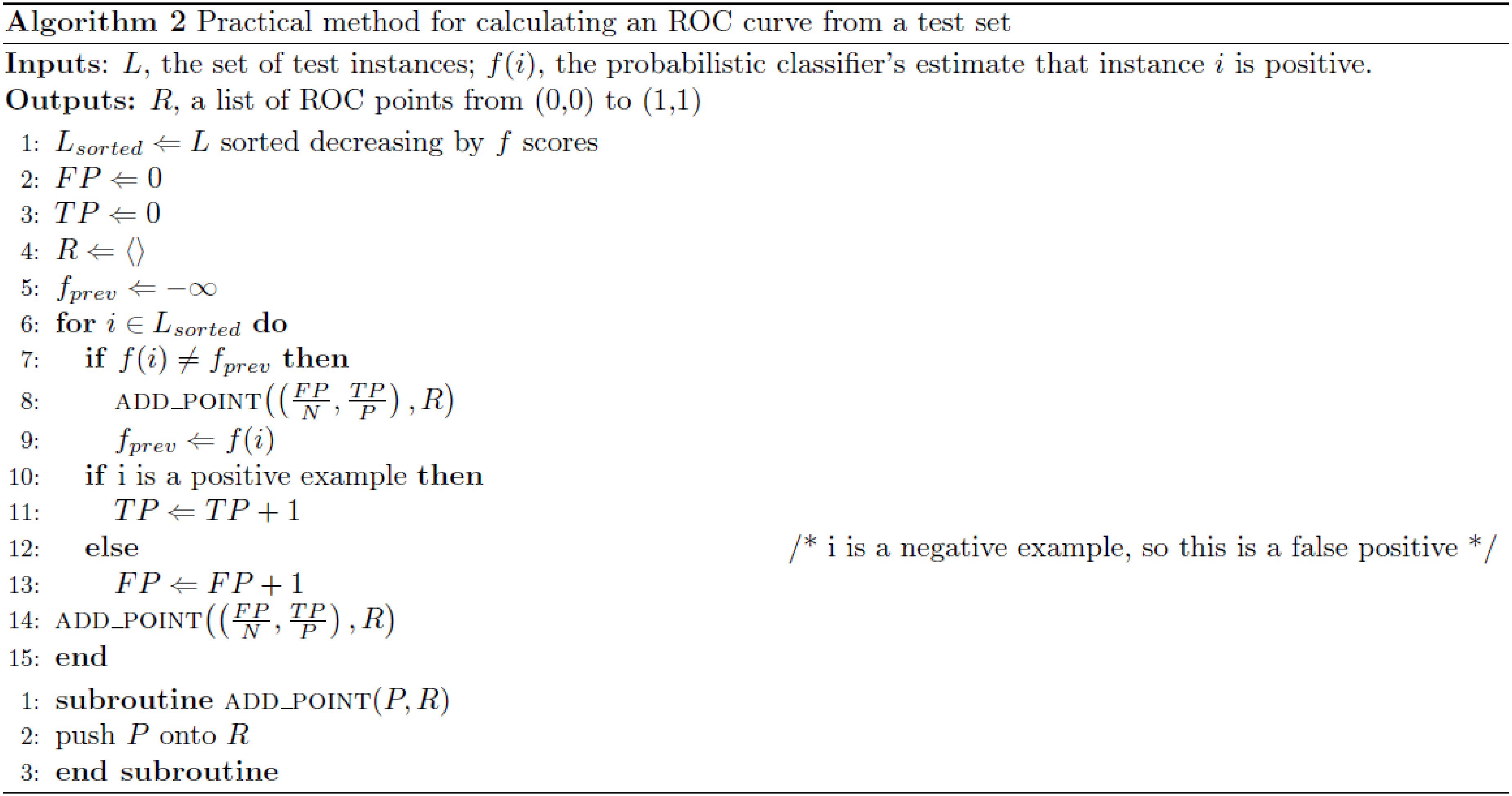
Algorithm 1 Conceptual method for calculating an ROC curve. See algorithm 2 for a practical method.

Inputs: L, the set of test instances; $f(i)$, the probabilistic classifier's estimate that instance i is positive; min and max, the smallest and largest values returned by f; $increment$, the smallest difference between any two f values.

```
for $t = min$ to $max$ by $increment$ do
  $FP \Leftarrow 0$
  $TP \Leftarrow 0$
  for $i \in L$ do
    if $f(i) \geq t$ then                /* This example is over threshold */
      if i is a positive example then
        $TP \Leftarrow TP + 1$
      else                              /* i is a negative example, so this is a false positive */
        $FP \Leftarrow FP + 1$
  Add point $(\frac{FP}{N}, \frac{TP}{P})$ to ROC curve
end
```

- What is the **computational complexity** of this algorithm?

Practical ROC Algorithm



Algorithm 2 Practical method for calculating an ROC curve from a test set

Inputs: L, the set of test instances; $f(i)$, the probabilistic classifier's estimate that instance i is positive.

Outputs: R, a list of ROC points from (0,0) to (1,1)

```
$L_{sorted} \Leftarrow L$ sorted decreasing by $f$ scores
$FP \Leftarrow 0$
$TP \Leftarrow 0$
$R \Leftarrow \langle\rangle$
$f_{prev} \Leftarrow -\infty$
for $i \in L_{sorted}$ do
    if $f(i) \neq f_{prev}$ then
        ADD_POINT$\left(\left(\frac{FP}{N}, \frac{TP}{P}\right), R\right)$
        $f_{prev} \Leftarrow f(i)$
    if i is a positive example then
        $TP \Leftarrow TP + 1$
    else                                        /* i is a negative example, so this is a false positive */
        $FP \Leftarrow FP + 1$
ADD_POINT$\left(\left(\frac{FP}{N}, \frac{TP}{P}\right), R\right)$
end

subroutine ADD_POINT$(P, R)$
push $P$ onto $R$
end subroutine
```

- What is the **computational complexity** of this algorithm?

Model Assessment and Selection

Introduction

- ***Generalization*** performance of a learning method:
  - Measure of prediction capability on independent test data
  - Guide model selection
  - Give quantitative assessment of chosen model
- Chapter shows key methods and how to apply them to model selection

Bias, Variance, and Model Complexity

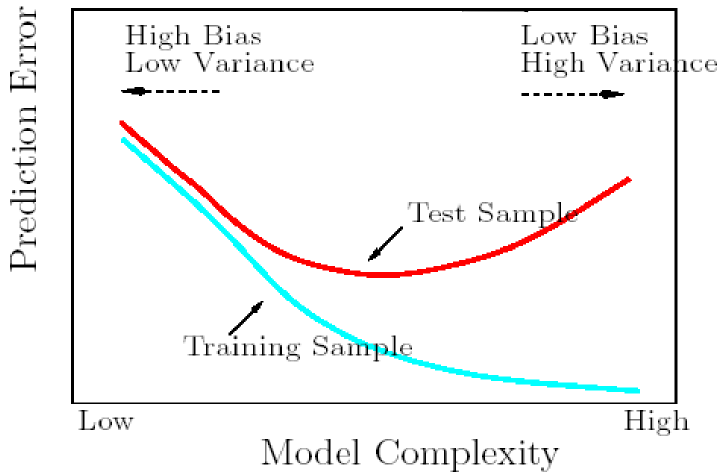


Figure 7.1: *Behavior of test sample and training sample error as the model complexity is varied.*

Generalization:
test sample vs. training sample performance

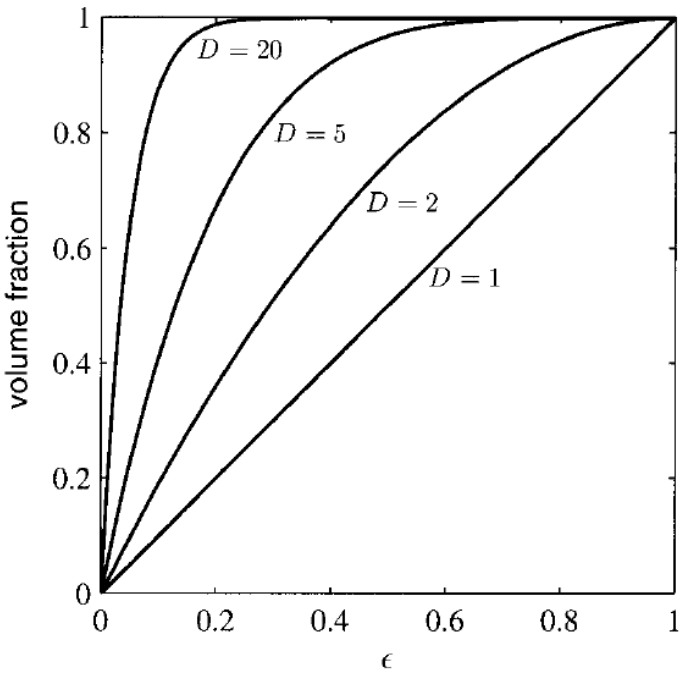
- Training data usually monotonically increasing performance with model complexity

- Bias and Variance are a **tradeoff**
- **Bias** in this context refers to the difference between the model prediction and the actual data
- **Variance** in this context refers to the sensitivity of the model's predictions to realizations of the model

The Curse of Dimensionality

- As the dimensionality of the problem increases, most of the volume (and therefore data) lies within a thin shell at the surface of a hypersphere

Plot of the fraction of the volume of a sphere lying in the range $r = 1-\epsilon$ to $r = 1$ for various values of the dimensionality D.



- This implies that data is **very sparse** throughout most of the volume!
- Therefore it is hard to find a **representative data sample** for a high-dimensional space.

Measuring Performance

- Target variable Y
- Vector of inputs X
- Prediction model $\hat{f}(X)$

- Typical Choices of Loss function

$$L\left(Y, \hat{f}(X)\right) = \begin{cases} \left(Y - \hat{f}(X)\right)^2 & \textit{squared error} \\ \left|Y - \hat{f}(X)\right| & \textit{absolute error} \end{cases}$$

Generalization Error

- **Test error** aka. Generalization error

$$\boxed{Err = E\left[L\left(Y, \hat{f}(X)\right)\right]}$$

- Note: This **expectation** averages anything that is random, including the randomness in the training sample that it produced
- **Training error**

$$\boxed{\overline{err} = \frac{1}{N}\sum_{i=1}^{n} L\left(y_i, \hat{f}(x_i)\right)}$$

  - **average loss** over training sample
  - not a good estimate of **test error** (next slide)

Training Error

- Choosing a model by examining **training error** leads to **overfitting**
  - not a good estimate of test error
  - consistently decreases with model complexity
  - drops to zero with high enough complexity

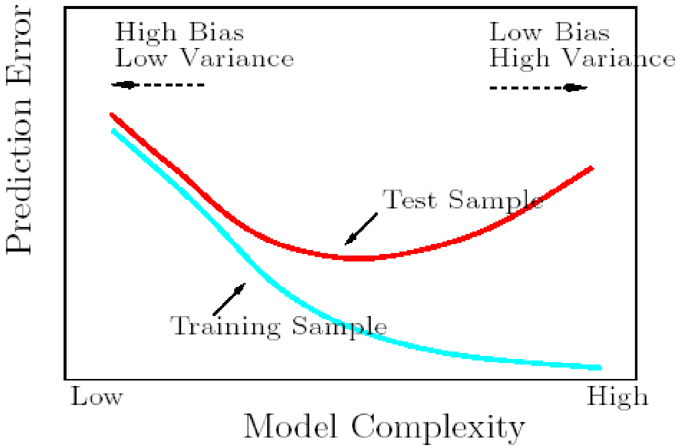


Figure 7.1: *Behavior of test sample and training sample error as the model complexity is varied.*

- **Overfitted model**:

the model describes **noise** in the data rather than the underlying relationship

Two separate goals

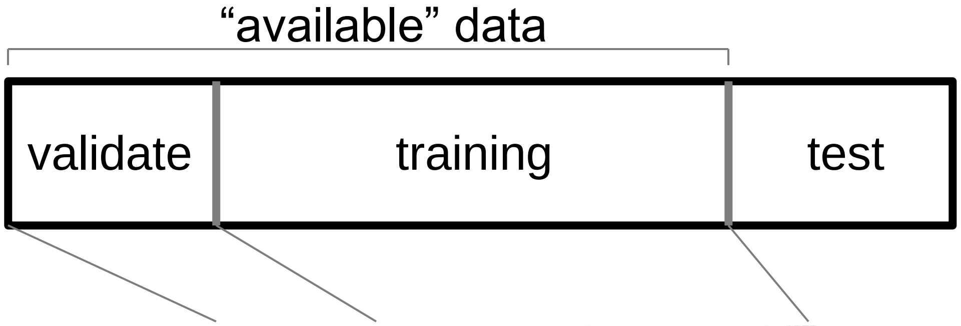
- **Model selection**:
  - Estimating the performance of different models in order to choose the (approximate) best one
- **Model assessment**:
  - Having chosen a final model, estimating its prediction error (generalization error) on new data

- Ideal situation: Data Rich:
  - split data into the 3 parts for *training*, *validation (est. prediction error+select model)*, and *testing (assess model)*
  - Typical split*: 50% / 25% / 25%*

- Data-Poor situation
  - *Approximation of validation* step either analytically (AIC, BIC, MDL, SRM)
  - AIC/BIC - http://www.youtube.com/watch?v=YkD7ydzp9_E
  - or by efficient sample reuse (cross-validation, bootstrap)

Cross Validation

- Used for determining the values of parameters of a model
- Also used for selecting a model
- The idea is simple: split all available data into 3 pieces – training, validation and testing

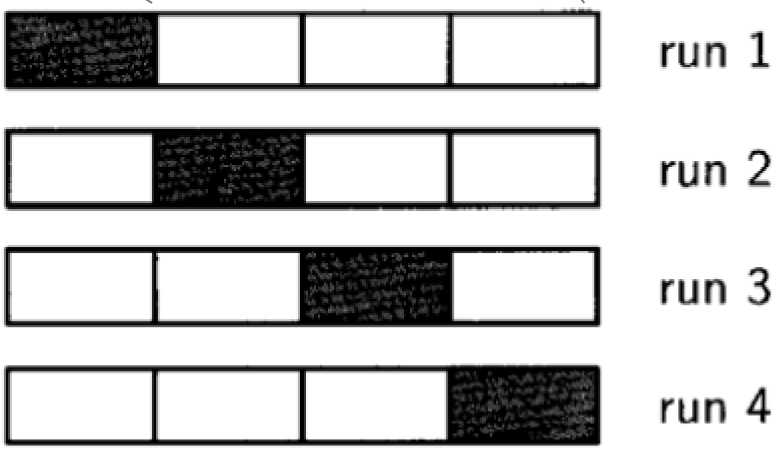


“available” data

validate

training

test



The technique of S-fold cross-validation, illustrated here for the case of $S = 4$, involves taking the available data and partitioning it into S groups (in the simplest case these are of equal size). Then $S - 1$ of the groups are used to train a set of models that are then evaluated on the remaining group. This procedure is then repeated for all S possible choices for the held-out group, indicated here by the red blocks, and the performance scores from the S runs are then averaged.