0.1 Learning Objectives

By the end of this lecture, you will be able to:

- Understand binary classification problems
- Interpret confusion matrices and compute TP, TN, FP, FN
- Compute sensitivity, specificity, precision, and recall
- Understand the tradeoff between sensitivity and specificity
- Connect predictive values to Bayes' theorem
- Interpret ROC curves and AUC

1. Binary Classification

1.1 Recall: Radar Detection

Recall: In B&T Example 1.9, a radar system detects aircraft with $P(\text{detect} \mid \text{aircraft}) = 0.99$ but also has false alarms: $P(\text{detect} \mid \text{no aircraft}) = 0.10$. Using Bayes' rule, we found $P(\text{aircraft} \mid \text{detect}) = 0.3426$ — a positive detection only meant a 34% chance the aircraft was actually there!

Today we formalize this kind of problem.

1.2 What is Binary Classification?

Definition: Binary Classification

Classifying the elements of a set into **two groups** on the basis of a classification rule.

A **classification rule** is a function that takes as input an element and outputs a group membership.

1.3 Binary Classification Examples

- **Medical testing**: determining if a patient has a certain disease or not
- **Quality control** in industry: deciding whether a specification has been met
- **Information retrieval**: deciding whether a page should be in search results
- **Network security**: deciding whether a packet is malicious or not
- **Machine learning**: deciding whether an input belongs to a category
- **Digital communications**: determining whether a received bit is a 1 or a 0

1.4 The Classification Setup

We measure something about a sample (e.g., concentration of an enzyme in a blood sample, voltage level in a transistor).

We specify a **threshold** value for the classifier:

- Samples with values **above** the threshold $\rightarrow$ class 1
- Samples with values **below** the threshold $\rightarrow$ class 2

Key question: How do we assess the performance of the classifier?

2. Definitions and Terminology

2.1 Random Variables for Classification

Recall our disease testing example. We define two random variables:

$$D = \begin{cases} 1 \text{ if sample is infected} \\ 0 \text{ if sample is not infected} \end{cases}$$

$$T = \begin{cases} 1 \text{ if test is positive} \\ 0 \text{ if test is negative} \end{cases}$$

2.2 Four Possible Outcomes

The test is not completely accurate, so there are 4 outcomes:

	Disease Present ($D = 1$)	**Disease Absent** ($D = 0$)
Test Positive ($T = 1$)	True Positive (TP)	False Positive (FP)
Test Negative ($T = 0$)	False Negative (FN)	True Negative (TN)

2.3 Type I and Type II Errors

- **False Positive (Type I Error)**: $P(T = 1 \mid D = 0)$
  - The test says positive, but the sample is actually negative
- **False Negative (Type II Error)**: $P(T = 0 \mid D = 1)$
  - The test says negative, but the sample is actually positive

2.4 The Confusion Matrix

Definition: Confusion Matrix

A 2×2 matrix storing the **counts** observed in a data set for each outcome.

The entries are counts (the number of times each outcome was observed).

$$\text{TP} = |T = 1 \cap D = 1|$$
$$\text{TN} = |T = 0 \cap D = 0|$$
$$\text{FP} = |T = 1 \cap D = 0|$$
$$\text{FN} = |T = 0 \cap D = 1|$$

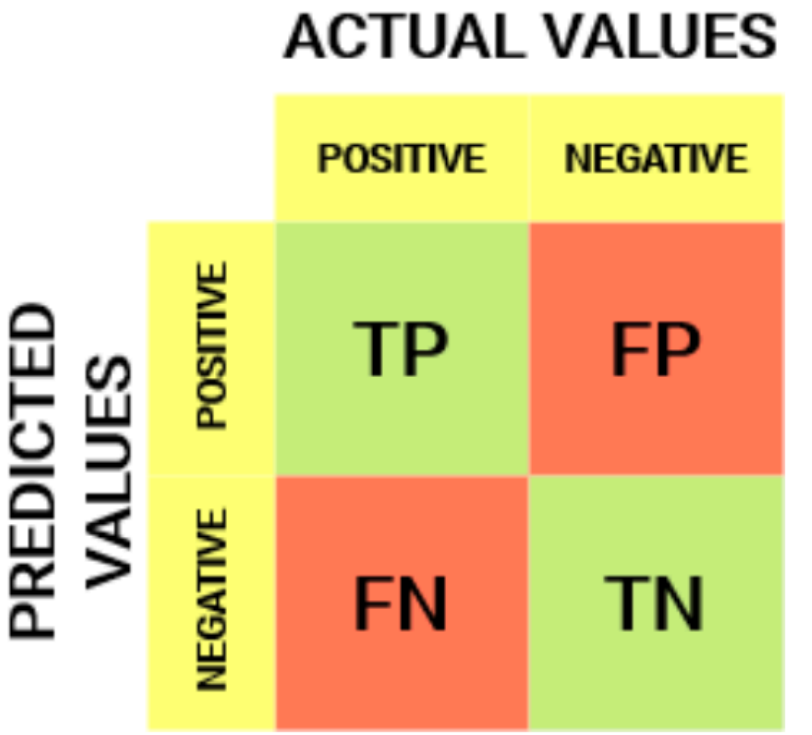


Image from Analytics Vidhya

3. Key Metrics

3.1 Prevalence

Definition: Prevalence

$$\pi = P(D = 1)$$

How common is the disease (or condition) in the population?

3.2 Sensitivity

Definition: Sensitivity

$$\eta = P(T = 1 \mid D = 1) \quad \text{True Positive Rate (TPR)}$$

Of all the actually positive cases, how many did we correctly identify?

Also called **recall**.

3.3 Specificity

Definition: Specificity

$$\theta = P(T = 0 \mid D = 0) \quad \text{True Negative Rate (TNR)}$$

Of all the actually negative cases, how many did we correctly identify?

Note: the **False Positive Rate** (FPR) $= 1 - \theta$.

The terms "sensitivity" and "specificity" were introduced by biostatistician Jacob Yerushalmy in 1947.

3.4 Sensitivity from Counts

We can also express sensitivity using the confusion matrix counts:

$$\eta = \mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$

In words: the fraction of all actual positive cases that the classifier correctly identified.

3.5 Equivalence: Counts and Probabilities

We can show the count-based definition equals the conditional probability:

$$\begin{aligned}\frac{\text{TP}}{\text{TP}+\text{FN}} &= \frac{|T=1 \cap D=1|}{|T=1 \cap D=1| + |T=0 \cap D=1|} \\ &= \frac{P(T=1 \cap D=1)}{P(D=1)} \\ &= P(T=1 \mid D=1)\end{aligned}$$

Similarly: $\theta = \frac{\text{TN}}{\text{TN}+\text{FP}} = P(T=0 \mid D=0)$

3.6 Precision

Definition: Precision

Of the cases we *predicted* positive, how many were actually positive?

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Note: precision $\neq$ recall (sensitivity).

- **Recall** asks: did we find all the positives?
- **Precision** asks: were our positive predictions correct?

3.7 Summary of Metrics

Metric	Formula	Question it answers
Accuracy	$\frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$	Overall, how often is the classifier correct?
Precision	$\frac{\text{TP}}{\text{TP} + \text{FP}}$	When it predicts positive, is it right?
Recall	$\frac{\text{TP}}{\text{TP} + \text{FN}}$	Does it find all the positives?
Specificity	$\frac{\text{TN}}{\text{TN} + \text{FP}}$	Does it correctly identify negatives?
F1 Score	$\frac{2\times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$	Balance between precision and recall

4. Tradeoff Between Sensitivity and Specificity

4.1 The Fundamental Tradeoff

Sensitivity and specificity are **inversely related**.

If you change the threshold to increase one, you necessarily decrease the other.

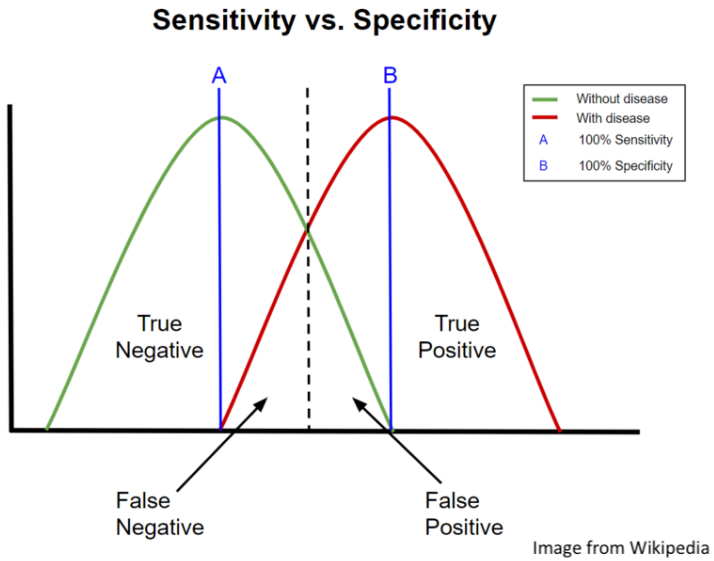


Image from Wikipedia

4.2 Example Setup

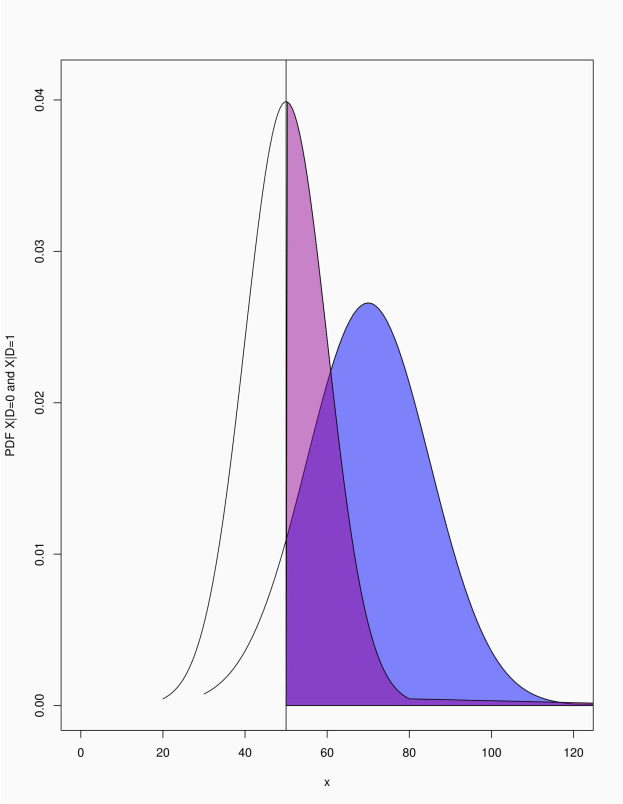
Suppose we have a classifier that computes a score X for each sample.

- For infected samples: $X \mid (D = 1) \sim \text{Norm}(70, 15)$
- For healthy samples: $X \mid (D = 0) \sim \text{Norm}(50, 10)$

We choose a threshold x^*:

- If $X > x^*$: test is positive ($T = 1$)
- If $X \leq x^*$: test is negative ($T = 0$)

4.3 Low Threshold ($x^* \approx 52$)

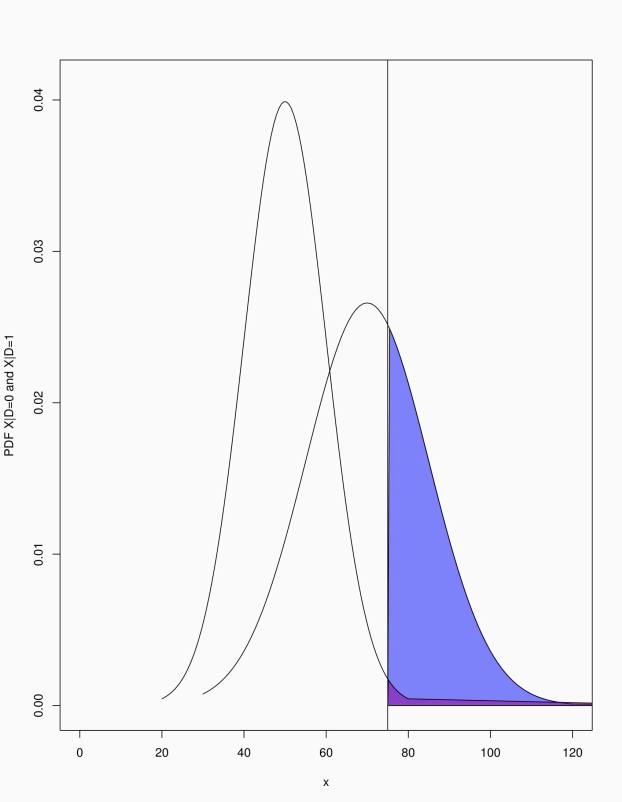


Sensitivity is high: almost all infected samples score above the threshold (large blue area).

Specificity is low: many healthy samples also score above the threshold (purple area leaks right).

We catch almost every infected sample, but have a huge false positive rate!

4.4 High Threshold ($x^* = 75$)



Sensitivity is low: many infected samples score below the threshold (small blue area).

Specificity is high: very few healthy samples score above the threshold.

We have very few false positives, but miss most infected samples!

4.5 Sensitivity and Specificity as Areas

Sensitivity $\eta = P(X > x^* \mid D = 1)$

$\rightarrow$ area under the $X \mid D = 1$ curve to the *right* of x^* (blue)

Specificity $\theta = P(X \leq x^* \mid D = 0)$

$\rightarrow$ area under the $X \mid D = 0$ curve to the *left* of x^* (purple)

As x^* increases: sensitivity $\downarrow$ and specificity $\uparrow$

As x^* decreases: sensitivity $\uparrow$ and specificity $\downarrow$

4.6 Computing Sensitivity and Specificity in R

```
# Sensitivity: P(X > x* | D=1)
xstar <- 58
eta <- 1 - pnorm(xstar, mean=70, sd=15)
eta  # 0.7881

# Specificity: P(X <= x* | D=0)
theta <- pnorm(xstar, mean=50, sd=10)
theta  # 0.7881
```

4.7 Computing in Python

```
from scipy.stats import norm
xstar = 58
eta = 1 - norm.cdf(xstar, loc=70, scale=15)    # 0.7881
theta = norm.cdf(xstar, loc=50, scale=10)       # 0.7881
```

Try it yourself

Talk to your neighbor and try to solve this problem.

Pick a value of x^* and compute η and θ.

5. Predictive Values and Bayes' Theorem

5.1 How Do We Assess Classifier Quality?

Recall our notation:

- η is sensitivity, $\eta = P(T = 1 \mid D = 1)$
- θ is specificity, $\theta = P(T = 0 \mid D = 0)$
- π is prevalence, $\pi = P(D = 1)$

We now look at three important measures of quality:

1. The **predictive value of a positive test** (γ)
2. The **predictive value of a negative test** (δ)
3. The **ROC curve**

5.2 Predictive Value of a Positive Test

What we really want to know: if someone tests positive, what is the probability they actually have the disease?

Definition: Predictive Value of a Positive Test

$$\gamma = P(D = 1 \mid T = 1)$$

5.3 Deriving γ with Bayes' Theorem

$$\gamma = \frac{P(D = 1 \cap T = 1)}{P(T = 1)}$$

$$= \frac{P(T = 1 \mid D = 1) \cdot P(D = 1)}{P(T = 1)}$$

$$= \frac{P(T = 1 \mid D = 1) \cdot P(D = 1)}{\underbrace{P(T = 1|D = 1)P(D = 1) + P(T = 1|D = 0)P(D = 0)}_{\text{Law of Total Probability}}}$$

$$= \frac{\eta\pi}{\eta\pi + (1 - \theta)(1 - \pi)}$$

5.4 Predictive Value of a Negative Test

Definition: Predictive Value of a Negative Test

$$\delta = P(D = 0 \mid T = 0)$$

Similarly:

$$\delta = \frac{P(T = 0 \mid D = 0) \cdot P(D = 0)}{P(T = 0)}$$

$$= \frac{\theta(1 - \pi)}{\theta(1 - \pi) + (1 - \eta)\pi}$$

5.5 Example: Rare Disease Testing

Suppose the prevalence $\pi = 0.005$ (0.5%), sensitivity $\eta = 0.98$, and specificity $\theta = 0.96$.

If a random person tests positive, what is the probability they have the disease?

5.6 Example: Rare Disease Solution

$$\begin{aligned}\gamma &= \frac{\pi\eta}{\pi\eta + (1-\pi)(1-\theta)} \\ &= \frac{0.005 \times 0.98}{0.005 \times 0.98 + 0.995 \times 0.04} \\ &= \frac{0.0049}{0.0049 + 0.0398} \\ &= 0.1096\end{aligned}$$

Note: Only about **11%** of positive tests are true positives! The low prevalence means false positives dominate.

5.7 Example: Malware Detection

A detection test for malware has sensitivity $\eta = 0.98$ and specificity $\theta = 0.96$. The prevalence of malware is $\pi = 0.005$.

1. What proportion of codes that test positive actually have malware?
2. What percentage of all codes will be subjected to further testing?
3. What fraction of codes falls into each of the four categories?

5.8 Malware Detection: Part 1

What proportion of codes that test positive actually have malware?

This is $P(D = 1 \mid T = 1) = \gamma = 0.1096$

5.8 Malware Detection: Part 1

same as the disease example.

Only about **11%** of flagged executables actually contain malware!

5.9 Malware Detection: Part 2

What percentage of all codes will be subjected to further testing?

We need $P(T = 1)$, which is the denominator from our Bayes' calculation:

$$\begin{aligned} P(T = 1) &= P(T = 1|D = 1)P(D = 1) + P(T = 1|D = 0)P(D = 0) \\ &= 0.98 \times 0.005 + 0.04 \times 0.995 \\ &= 0.0049 + 0.0398 \\ &= 0.0447 \end{aligned}$$

About **4.5%** of all codes will need further testing.

5.10 Malware Detection: Part 3

What fraction of codes falls into each category?

$$\text{TP} = P(T = 1 \cap D = 1) = \eta \cdot \pi = 0.98 \times 0.005 = 0.0049$$
$$\text{TN} = P(T = 0 \cap D = 0) = \theta \cdot (1 - \pi) = 0.96 \times 0.995 = 0.9552$$
$$\text{FN} = P(T = 0 \cap D = 1) = (1 - \eta) \cdot \pi = 0.02 \times 0.005 = 0.0001$$
$$\text{FP} = P(T = 1 \cap D = 0) = (1 - \theta) \cdot (1 - \pi) = 0.04 \times 0.995 = 0.0398$$

The vast majority of "positives" are **false positives** (0.0398 vs 0.0049).

6. ROC Curves

6.1 ROC Curve

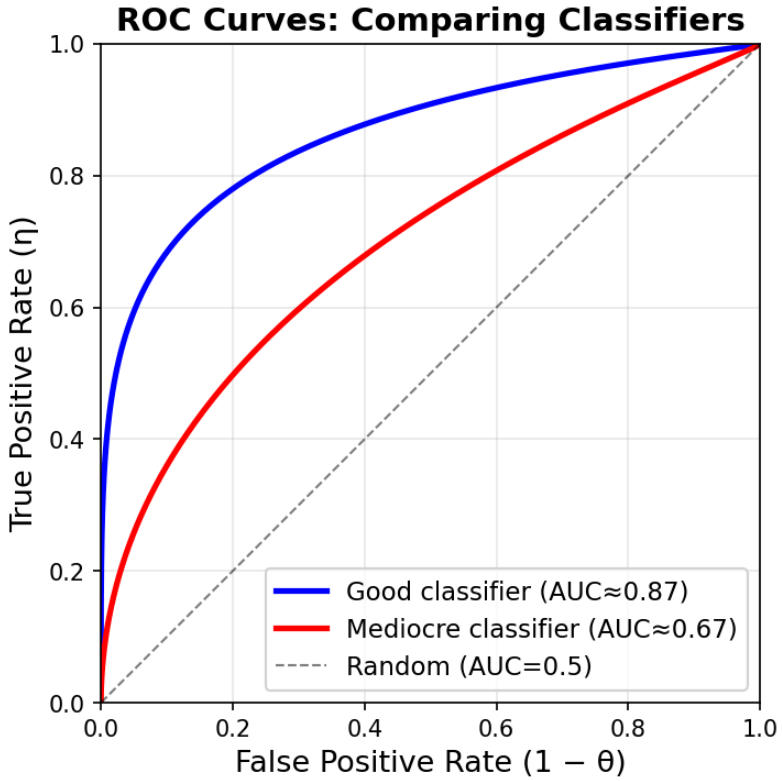
Definition: ROC Curve

The **Receiver Operating Characteristic** curve plots:

- **True Positive Rate** (η) on the y-axis
- **False Positive Rate** ($1 - \theta$) on the x-axis

for different threshold values x^*.

6.2 Good vs Bad Classifiers



- A **strong classifier**: curve hugs the top-left corner (AUC close to 1.0)
- **Random guessing**: curve follows the diagonal (AUC = 0.5)
- The further the curve bows toward the top-left, the better

6.3 Building the ROC Curve

For each threshold x^*, we compute one point on the ROC curve:

- FPR $= 1 - \theta = P(X > x^* \mid D = 0)$
- TPR $= \eta = P(X > x^* \mid D = 1)$

As x^* varies from $-\infty$ to $+\infty$:

- $x^* = -\infty$: TPR $= 1$, FPR $= 1$ (top-right corner)
- $x^* = +\infty$: TPR $= 0$, FPR $= 0$ (bottom-left corner)

6.4 Area Under the Curve (AUC)

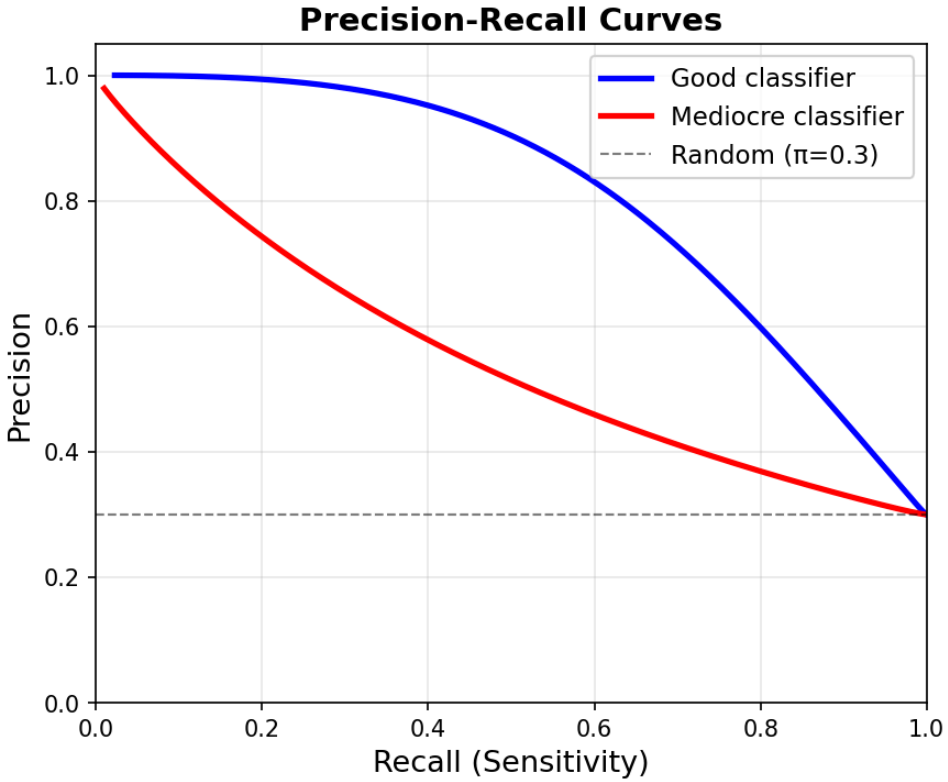
Definition: AUC

The **Area Under the ROC Curve** summarizes classifier performance in a single number.

- AUC $= 1.0$: perfect classifier
- AUC $= 0.5$: no better than random guessing
- AUC < 0.5: worse than random (flip the predictions!)

If you have several competing classifiers, you can use the AUC to select the best one.

6.5 Precision-Recall Curve



An alternative to the ROC curve that plots **Precision** vs **Recall**.

- Random baseline is a horizontal line at $y = \pi$ (prevalence)
- A strong classifier hugs the top-right corner
- More informative than ROC when classes are **imbalanced**

6.6 ROC vs Precision-Recall

- **ROC curves** are useful when classes are roughly balanced
- **PR curves** are better when the positive class is rare (e.g., fraud, malware)
  - ROC can look overly optimistic with imbalanced classes because TNs dominate

Both use AUC as a summary statistic.

6.7 Computing the ROC in R

```r
# X|D=0 ~ Norm(50, 10),  X|D=1 ~ Norm(70, 15)
x <- seq(30, 90, by=1)
eta <- 1 - pnorm(x, 70, 15)       # sensitivity (TPR)
theta <- pnorm(x, 50, 10)         # specificity
fpr <- 1 - theta                   # false positive rate

# Plot the ROC curve
plot(fpr, eta, type='l', xlim=c(0,1), ylim=c(0,1),
     xlab="FPR", ylab="TPR", main="ROC Curve")
abline(0, 1, lty=2, col="grey")  # random baseline
```

HW5 will ask you to do this!

6.8 Recap

Today we covered:

- Binary classification: assign elements to one of two groups using a threshold
- Confusion matrix: counts of TP, TN, FP, FN
- Sensitivity (recall/TPR) and specificity (TNR) trade off against each other
- Predictive values use Bayes' theorem: low prevalence means many false positives
- ROC curve plots TPR vs FPR for different thresholds; AUC measures overall quality
- Next: Probability bounds (Markov and Chebyshev inequalities)