Diagnostic Accuracy in R

From the 2 × 2 table to ROC curves, uncertainty and clinical interpretation

Rverse Analytics

Lock the orientation before calculating. Rows = index test, columns = reference standard; positive disease is the event. Predictive values describe the tested population and therefore change with prevalence.

The 2 × 2 core

	Disease +	Disease −
Test +	TP	FP
Test −	FN	TN

Metric	Formula	Meaning
Sensitivity	TP / (TP + FN)	P(test +
Specificity	TN / (TN + FP)	P(test −
PPV	TP / (TP + FP)	P(disease +
NPV	TN / (TN + FN)	P(disease −
LR+	sensitivity / (1 − specificity)	multiply pretest odds after + result
LR−	(1 − sensitivity) / specificity	multiply pretest odds after − result
Diagnostic OR	(TP × TN) / (FP × FN)	LR+ / LR−; less clinically direct

```
tp <- 90; fp <- 10; fn <- 20; tn <- 180

sens <- tp / (tp + fn)
spec <- tn / (tn + fp)
ppv  <- tp / (tp + fp)
npv  <- tn / (tn + fn)
lr_p <- sens / (1 - spec)
lr_n <- (1 - sens) / spec

# Exact binomial confidence intervals
binom.test(tp, tp + fn)$conf.int  # sensitivity CI
binom.test(tn, tn + fp)$conf.int  # specificity CI
```

From pretest to post-test probability

```
odds <- function(p) p / (1 - p)
prob <- function(o) o / (1 + o)

pretest <- 0.20
post_positive <- prob(odds(pretest) * lr_p)
post_negative <- prob(odds(pretest) * lr_n)

# Predictive values at a target prevalence
predictive_values <- function(sens, spec, prevalence) {
  c(
    PPV = sens * prevalence /
      (sens * prevalence + (1 - spec) * (1 - prevalence)),
    NPV = spec * (1 - prevalence) /
      (spec * (1 - prevalence) + (1 - sens) * prevalence)
  )
}
```

ROC analysis for a continuous marker

```
library(pROC)

# State control/case order and direction explicitly
r <- roc(response = d$disease, predictor = d$marker,
         levels = c("No", "Yes"), direction = "<", ci = TRUE)

auc(r); ci.auc(r)
coords(r, x = "best", best.method = "youden",
```

```
       ret = c("threshold", "sensitivity", "specificity"),
       transpose = FALSE)

# Prespecified operating point is preferable for confirmatory work
coords(r, x = 10, input = "threshold",
       ret = c("sensitivity", "specificity", "ppv", "npv"),
       transpose = FALSE)

# Compare paired AUCs measured in the same patients
roc.test(r_test_a, r_test_b, paired = TRUE, method = "delong")
plot(r, legacy.axes = TRUE, print.auc = TRUE)
```

AUC is ranking performance: it does not show calibration, clinical value, or performance at the intended threshold. A data-optimized "best" threshold is optimistic unless validated independently.

Design traps and the right analysis

Trap	Bias / problem	Better practice
Case-control sampling	PPV/NPV are not population estimates	transport sens/spec with target prevalence
Only positives verified	verification bias	verify a representative sample; model missing reference
Imperfect reference standard	misclassification	sensitivity analysis / latent-class approach
Multiple observations per patient	CI too narrow	patient-level bootstrap or clustered model
Threshold chosen and evaluated on same data	optimistic accuracy	nested resampling or external validation
Missing marker values deleted silently	spectrum changes	report flow and missingness; justify method

Paired tests and agreement

```
# Compare paired binary tests (discordant pairs drive McNemar's test)
mcnemar.test(table(d$test_a, d$test_b), correct = TRUE)

# Cohen's kappa measures agreement beyond chance, not accuracy
irr::kappa2(d[c("reader_a", "reader_b")], weight = "unweighted")
```

Report checklist

- Describe participant spectrum, setting, sampling, blinding and reference standard.
- Give the full 2 × 2 counts plus sensitivity, specificity, predictive values and 95% CIs.
- State threshold selection before performance estimates; give units and direction.
- Report prevalence with PPV/NPV and justify transport to the target clinical setting.
- Distinguish discrimination, calibration, agreement and clinical utility.

Rverse Analytics · rverseanalytics.com · Original reference sheet