PH 724 Data Analysis & Scientific Communication Incorporating Use of Generative AI

Part 12: Assessing the Performance of Prediction Models

Zhiyang Zhou (zhou67@uwm.edu, zhiyanggeezhou.github.io)

2026/11/22 20:04:12

Central framework of [P#8] Steyerberg et al. (2010)

Question	Performance dimension	Main measures
How close are predicted probabilities to outcomes?	Overall performance	Brier, scaled Brier, Nagelkerke R^2
Can the model rank cases above noncases?	Discrimination	AUC/c-statistic/c-index, discrimination slope
Are predicted risks numerically trustworthy?	Calibration	Calibration-in-the-large intercept, calibration slope, HL test
Does a new marker improve predictions?	Reclassification	Reclassification table, NRI, IDI
Would using predictions improve decisions?	Clinical usefulness	Net benefit, decision curves

---

Reproducible teaching dataset

```r
set.seed(724)
n <- 800
x1 <- rnorm(n)
x2 <- rbinom(n, 1, 0.45)
marker <- rnorm(n)
prob_true <- plogis(-0.7 + 1.0*x1 + 0.7*x2 + 0.6*marker)
y <- rbinom(n, 1, prob_true)
dat <- data.frame(y, x1, x2, marker)
# Training/testing split
idx <- sample(seq_len(n), size=floor(0.65*n))
train <- dat[idx, ]
test <- dat[-idx, ]
# 2 models
m0 <- glm(y ~ x1 + x2, family=binomial(), data=train)
m1 <- glm(y ~ x1 + x2 + marker, family=binomial(), data=train)
# Predictions issued by the 2 models
p0 <- predict(m0, newdata=test, type="response")
```

```r
p1 <- predict(m1, newdata=test, type="response")


yv <- test$y
clip <- function(p) pmin(pmax(p, 1e-8), 1-1e-8)
head(data.frame(y=yv, baseline=round(p0,3), extended=round(p1,3)))
```

Overall performance: How close are predictedprobabilities to outcomes?

Brier score

Brier score is the mean squared error of predicted probabilities:

$$\text{Brier} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - p_i)^2.$$

- **Lower is better**: 0 means perfect predictions.
- For binary outcomes the theoretical range is 0 to 1; a constant prevalence prediction has expected Brier score $\pi(1-\pi)$.
- Combines information about calibration and discrimination; depends on event prevalence.

```r
brier <- function(y, p) mean((y-p)^2)
c(m0=brier(test$y,p0), m1=brier(test$y,p1))
```

Scaled Brier score Compares against the no-information prediction $\hat{p}_{\text{null}} = \bar{Y}$:

$$\text{Brier}_{\text{scaled}} = 1 - \frac{\text{Brier}}{\text{Brier}_{\text{null}}} \quad \text{with} \quad \text{Brier}_{\text{null}} = \hat{p}_{\text{null}}(1-\hat{p}_{\text{null}})$$

- **Higher is better**: 1: perfect predictions; 0: equal to predicting prevalence for everyone.
- Negative values are possible for a poorly performing model.

```r
scaled_brier <- function(y,p) 1 - brier(y,p)/(mean(y)*(1-mean(y)))
c(m0=scaled_brier(test$y,p0), m1=scaled_brier(test$y,p1))
```

Nagelkerke's pseudo-R^2

Denote the binary log likelihood by

$$\ell = \sum_i \{Y_i \log p_i + (1-Y_i)\log(1-p_i)\}.$$

Nagelkerke's pseudo-R^2 rescales the Cox–Snell index:

$$R^2_N = \frac{1-\exp\{2(\ell_0 - \ell_m)/n\}}{1-\exp(2\ell_0/n)}.$$

- ℓ_0: intercept-only model log likelihood
- ℓ_m: fitted model log likelihood.
- A likelihood-based model-fit measure, not ordinary variance explained.
- Most commonly computed on training data.

```r
nagelkerke <- function(model) {
  ll <- as.numeric(logLik(model))
  ll0 <- as.numeric(logLik(update(model, . ~ 1)))
  n <- nobs(model)
  (1-exp(2*(ll0-ll)/n))/(1-exp(2*ll0/n))
}
c(m0=nagelkerke(m0), m1=nagelkerke(m1))
```

Discrimination: Can the model rank cases abovenoncases?

Receiver operating characteristic (ROC) curve and the area under ROC curve (AUROC or AUC)

Given threshold $t \in (0, 1)$,

$$\text{Sensitivity}(t) = \Pr(\hat{p}_{Y=1} \geq t), \quad \text{Specificity}(t) = \Pr(\hat{p}_{Y=0} < t).$$

- $\hat{p}_{Y=1}$: The predicted probability of the outcome for a randomly selected individual who actually experienced the outcome (i.e., $Y = 1$).
- $\hat{p}_{Y=0}$: The predicted probability of the outcome for a randomly selected individual who did NOT experience the outcome (i.e., $Y = 0$).

Receiver operating characteristic curve plots sensitivity against $(1 -$ specificity$)$ across all the values of thresholds.

The area under ROC curve is:

$$\text{AUC} = \Pr(\hat{p}_{Y=1} > \hat{p}_{Y=0}) + \tfrac{1}{2}\Pr(\hat{p}_{Y=1} = \hat{p}_{Y=0}).$$

- Also called the c-statistic or c-index for binary outcomes.
- AUC assesses **ranking**, not whether risks are numerically correct.
  - AUC $= 0.5$: random guess; 1: perfect ranking.

```r
roc0 <- pROC::roc(test$y, p0, quiet=TRUE)
roc1 <- pROC::roc(test$y, p1, quiet=TRUE)
plot(roc0, col="gray35", legacy.axes=TRUE, main="Validation ROC curves")
plot(roc1, col="steelblue", add=TRUE)
legend("bottomright", legend=c("m0", "m1"),
       col=c("gray35","steelblue"), lty=1, bty="n")
c(m0=as.numeric(pROC::auc(roc0)), m1=as.numeric(pROC::auc(roc1)))
```

Discrimination slope

Definition: Difference in average predicted event risk between cases and controls:

$$\bar{\hat{p}}_{Y=1} - \bar{\hat{p}}_{Y=0}.$$

- An intuitive separation measure; larger is better.
- Unlike AUC, sensitive to the magnitudes of predicted probabilities.

```r
disc_slope <- function(y,p) mean(p[y==1])-mean(p[y==0])
c(m0=disc_slope(test$y,p0), m1=disc_slope(test$y,p1))
```

Calibration: Are predicted risks numericallytrustworthy?

Calibration-in-the-large intercept

The calibration-in-the-large intercept assesses whether the **average predicted risk agrees with the observed event rate**.

Fit a logistic model:

$$\operatorname{logit} \Pr(Y_i = 1) = \alpha + \operatorname{logit}(\hat{p}_i),$$

- $\hat{p}_i$ is the predicted event probability
- **Ideal:** $\alpha = 0$.
- $\alpha > 0$: the model tends to **underpredict** event risk.
- $\alpha < 0$: the model tends to **overpredict** event risk.

```
cal_intercept <- function(y, p) {
  fit <- glm(
    y ~ 1,
    offset = qlogis(clip(p)),
    family = binomial()
  )
  unname(coef(fit)[1])
}
c(m1 = cal_intercept(test$y, p0), m0 = cal_intercept(test$y, p1))
```

Calibration slope

The calibration slope assesses whether the **predicted risks are too extreme or not extreme enough**.

Fit a logistic model:

$$\operatorname{logit} \Pr(Y_i = 1) = \alpha + \beta \operatorname{logit}(\hat{p}_i),$$

where both α and β are estimated.

- **Ideal:** $\alpha = 0$ and $\beta = 1$.
- Interpret β first
  - $\beta = 1$: Predicted risks have the correct spread.
  - $\beta < 1$: Predicted risks tend to be **too extreme**, often indicating overfitting.
  - $\beta > 1$: Predicted risks tend to be **not extreme enough**.
- Interpret α conditional on β. When $\beta = 1$:
  - $\alpha > 0$: The model systematically underpredicts risk.
  - $\alpha < 0$: The model systematically overpredicts risk.

```
cal_params <- function(y, p) {
  fit <- glm(
    y ~ qlogis(clip(p)),
    family = binomial()
  )
  setNames(
    unname(coef(fit)),
    c("intercept", "slope")
  )
}
cal_params(test$y, p0)
cal_params(test$y, p1)
```

Hosmer-Lemeshow (HL) test

Split predictions into G quantile groups and then compare observed O_g with expected $E_g = \sum_{i \in g} p_i$.

Test statistic:

$$\chi^2_{\mathrm{HL}} = \sum_{g=1}^{G} \frac{(O_g - E_g)^2}{E_g(1 - E_g/n_g)}.$$

- Approximate χ^2_{G-2} reference under usual assumptions.
- Sensitive to sample size and arbitrary group definitions.
- If the Hosmer-Lemeshow test is statistically significant, it suggests that the predictive model is poorly calibrated.
- **A nonsignificant test is not evidence of good calibration.**

```
ResourceSelection::hoslem.test(test$y, p1, g=10)
```

Reclassification: Does a biomarker add value?

Reclassification table

Compare risk categories for the **same patients** under two models.

- Rows: old-model risk category
- Columns: new-model category.
- Upward movement: An individual is assigned to a higher predicted-risk category by the new model than by the baseline model.
- Downward movement: An individual is assigned to a lower predicted-risk category by the new model than by the baseline model.
- Off-diagonal counts represent movements; **movement alone is not improvement**.
- For cases, upward movement is favorable; for noncases, downward movement is favorable.
- Risk thresholds should be **clinically meaningful and prespecified**.

```
cutoff <- 0.20
old <- ifelse(p0 >= cutoff, "High", "Low")
new <- ifelse(p1 >= cutoff, "High", "Low")
movement <- ifelse(
  old == "Low" & new == "High", "Up",
  ifelse(
    old == "High" & new == "Low", "Down", "Same"
  )
)
# Individual-level reclassification
reclassification <- data.frame(
  Y = yv,
  old = old,
  new = new,
  movement = movement
)
head(reclassification)
# Reclassification table
table(old = old, new = new)
# Movement separately for cases and controls
table(Y = test$y, movement = movement)
```

Category-based net reclassification improvement (NRI)

$$\text{NRI} = \underbrace{\Pr(\uparrow|\ Y=1) - \Pr(\downarrow|\ Y=1)}_{\text{NRI}_{event}} + \underbrace{\Pr(\downarrow|\ Y=0) - \Pr(\uparrow|\ Y=0)}_{\text{NRI}_{nonevent}}.$$

- Positive NRI: net movement in the correct direction.
- Can be unstable, threshold-dependent, and clinically misleading if categories are arbitrary.
- Report **event and nonevent components** alongside total NRI.

```
nri_cat <- function(y, oldp, newp, cut=0.20) {
  movement <- as.integer(newp>=cut)-as.integer(oldp>=cut)
  event <- mean(movement[y==1]==1)-mean(movement[y==1]==-1)
  nonevent <- mean(movement[y==0]==-1)-mean(movement[y==0]==1)
  c(event=event, nonevent=nonevent, NRI=event+nonevent)
}
nri_cat(test$y,p0,p1,cutoff)
```

Integrated discrimination improvement (IDI)

IDI is the **change in discrimination slope**:

$$\text{IDI} = D_{\text{model 1}} - D_{\text{model 0}}.$$

- $D_{\text{model 0}}$: Discrimination slope for model 0.
- $D_{\text{model 1}}$: Discrimination slope for model 1.

```
idi <- function(y,oldp,newp) disc_slope(y,newp)-disc_slope(y,oldp)
idi(test$y,p0,p1)
```

Clinical usefulness: Would using predictions improve decisions?

Net benefit (NB)

Given a (clinically relevant) threshold p_t, suppose actions will be taken for those patients with predicted event risk $\hat{p}_i \geq p_t$. Then the net benefit is:

$$\text{NB}(p_t) = \underbrace{\Pr(Y=1, \hat{p} \geq p_t)}_{\text{Benefit of correctly taking action}} - \underbrace{\Pr(Y=0, \hat{p} \geq p_t)\frac{p_t}{1-p_t}}_{\text{Weighted harm of unnecessarily taking action}}.$$

- NB measures the benefit of correctly taking action minus the weighted harm of unnecessarily taking action; larger is better. - NB > 0: The benefits of taking action exceed its weighted harms. - NB $= 0$: The benefits and weighted harms balance. - NB < 0: The weighted harms exceed the benefits.

```
net_benefit <- function(y,p,pt) {
  pos <- p>=pt
  mean(pos & y==1) - mean(pos & y==0)*pt/(1-pt)
}
pt <- 0.20
c(m0=net_benefit(test$y,p0,pt),
  m1=net_benefit(test$y,p1,pt),
  treat_all=mean(test$y)-(1-mean(test$y))*pt/(1-pt),
  treat_none=0)
```

Decision curve analysis (DCA)

Plot net benefit across a **plausible range** of thresholds; not necessarily 0–100%.

```
thresholds <- seq(0.05,0.75,by=0.01)
nb0 <- sapply(thresholds,function(t) net_benefit(test$y,p0,t))
nb1 <- sapply(thresholds,function(t) net_benefit(test$y,p1,t))
nb_all <- mean(test$y)-(1-mean(test$y))*thresholds/(1-thresholds)
plot(thresholds,nb0,type="l",lwd=2,col="gray35",
     xlab="Threshold",ylab="Net benefit",
     ylim=range(c(0,nb0,nb1,nb_all)))
lines(thresholds,nb1,col="steelblue",lwd=2)
lines(thresholds,nb_all,col="firebrick",lty=2)
abline(h=0,lty=3)
legend("topright",c("m0","m1","Treat-all","Treat-none"),
       col=c("gray35","steelblue","firebrick","black"),
       lty=c(1,1,2,3),bty="n")
```

Key takeaways

- Discrimination asks who is higher-risk; calibration asks how accurate the risk numbers are.
- No single metric establishes predictive usefulness.
- NRI and IDI describe changes when adding predictors, but not necessarily decision value.
- Net benefit explicitly incorporates threshold-dependent consequences.
- The paper recommends **always reporting discrimination and calibration**, adding decision-analytic measures when the model guides decisions.

References

1. Steyerberg EW, Vickers AJ, Cook NR, et al. (2010) Assessing the performance of prediction models: A framework for some traditional and novel measures. *Epidemiology*. 21:128–138. https://pmc.ncbi.nlm.nih.gov/articles/PMC3575184/