cvauroc: Command to compute cross-validated area under the curve for ROC analysis after predictive modeling for binary outcomes

Miguel Angel Luque-Fernandez
London School of Hygiene and Tropical Medicine
London, UK
Biomedical Research Institute of Granada (ibs.GRANADA)
University of Granada
Granada, Spain
Andalusian School of Public Health
Granada, Spain
Biomedical Network Research Centers of Epidemiology
and Public Health (CIBERESP-ISCII)
Madrid, Spain
miguel-angel.luque@lshtm.ac.uk

Daniel Redondo-Sánchez
Biomedical Research Institute of Granada (ibs.GRANADA)
University of Granada
Granada, Spain
Andalusian School of Public Health
Granada, Spain
Biomedical Network Research Centers of Epidemiology
and Public Health (CIBERESP-ISCII)
Madrid, Spain
daniel.redondo.easp@juntadeandalucia.es

Camille Maringe
London School of Hygiene and Tropical Medicine
London, UK
camille.maringe@lshtm.ac.uk

Abstract. Receiver operating characteristic (ROC) analysis is used for comparing predictive models in both model selection and model evaluation. ROC analysis is often applied in clinical medicine and social science to assess the tradeoff between model sensitivity and specificity. After fitting a binary logistic or probit regression model with a set of independent variables, the predictive performance of this set of variables can be assessed by the area under the curve (AUC) from an ROC curve. An important aspect of predictive modeling (regardless of model type) is the ability of a model to generalize to new cases. Evaluating the predictive performance (AUC) of a set of independent variables using all cases from the original analysis sample often results in an overly optimistic estimate of predictive

performance. One can use K-fold cross-validation to generate a more realistic estimate of predictive performance in situations with a small number of observations. AUC is estimated iteratively for k samples (the "test" samples) that are independent of the sample used to predict the dependent variable (the "training" sample). `cvauroc` implements k-fold cross-validation for the AUC for a binary outcome after fitting a logit or probit regression model, averaging the AUCs corresponding to each fold, and bootstrapping the cross-validated AUC to obtain statistical inference and 95% confidence intervals. Furthermore, `cvauroc` optionally provides the cross-validated fitted probabilities for the dependent variable or outcome, contained in a new variable named `_fit`; the sensitivity and specificity for each of the levels of the predicted outcome, contained in two new variables named `_sen` and `_spe`; and the plot of the mean cross-validated AUC and k-fold ROC curves.

Keywords: st0569, cvauroc, prediction, area under the curve, receiver operating characteristic curve, classification

1   Introduction

Receiver operating characteristic (ROC) analysis is used for comparing predictive models in both model selection and model evaluation (Pepe 2000). ROC analysis is often applied in clinical medicine and social science to assess the tradeoff between model sensitivity (Se) and specificity (Sp) (Collins et al. 2015). After fitting a binary logistic regression model with a set of independent variables, the predictive performance of this set of variables can be assessed by the area under the curve (AUC) from an ROC curve (Pepe et al. 2008b).

In binary classification, the model prediction is often made based on a continuous random variable X, which is a "score" computed as the linear predictor in a logistic regression model. Given a threshold parameter T, the score is classified as "positive" if $X > T$ and as "negative" otherwise. X follows a probability density $f_1(x)$ if the score actually belongs to class "positive" and a probability density $f_0(x)$ otherwise. Therefore, the true-positive rate (TPR) is given by $\mathrm{TPR}(T) = \int_T^\infty f_1(x)\,dx$, and the false-positive rate (FPR) is given by $\mathrm{FPR}(T) = \int_T^\infty f_0(x)\,dx$. The ROC curve parametrically plots $\mathrm{TPR}(T)$ versus $\mathrm{FPR}(T)$ with T as the varying parameter (Fawcett 2006).

The AUC is a global summary measure of diagnostic accuracy and discrimination (that is, the ability for the logistic model to classify patients as cases and controls). It ranges from 0.5 for accuracy of chance to 1 for perfect accuracy. The closer the curve follows the left-hand border and then the top border of the ROC space, the more area there is under the curve and the more accurate the test. The closer the curve follows the 45-degree diagonal of the ROC space, the less accurate the test. The greater the AUC, the better the test can capture the tradeoff between Se and Sp over a continuous range.

When using normalized units, the AUC is equal to the integral

$$\mathrm{AUC} = \int_{\infty}^{-\infty} \mathrm{TPR}(T)\mathrm{FPR}'(T)\, dT$$
$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} I(T' > T) f_1(T') f_0(T)\, dT'\, dT = P(X_1 > X_0)$$

where X_1 is the score for a positive instance, X_0 is the score for a negative instance, and f_0 and f_1 are probability densities (Fawcett 2006).

An important aspect of predictive modeling (regardless of model type) is the ability of a model to generalize to new cases (Altman et al. 2009). Evaluating the predictive performance (AUC) of a set of independent variables using all cases from the original analysis sample often results in an overly optimistic estimate of predictive performance (LeDell, Petersen, and van der Laan 2015). K-fold cross-validation can generate a more realistic estimate of predictive performance (Pepe, Feng, and Gu 2008a), in contrast with having only one random split of the data into two groups (training and test) when the number of observations is small (LeDell, Petersen, and van der Laan 2015).

Cross-validation is one of the most common resampling techniques for evaluating predictive models. It consists of splitting a sample into several pairs of training and test sets. Usually only one random split into K groups is used, although some prefer to repeat the procedure several times. Cross-validation can be used to estimate TPR and FPR from which ROC curves can be derived and corresponding AUCs calculated. Cross-validation requires that a sample be partitioned into K parts for each randomization. Then K models are generated, each of them built without the cases in the kth partition, which is used for evaluation. That is, each observation x_i is part of one of the K partitions, with (k) returning the partition to which x_i belongs, resulting in X_{ik} observations. A special case when $K = n$ (the number of observations) is called leave-one-out cross-validation or the jackknife procedure. Many applications use $K = 5$ or 10 (James et al. 2013) and perform only one random data split.

Designed for fitting binary logit and probit regression models with a set of independent variables, `cvauroc` provides cross-validated area under the ROC for assessing the predictive performance of that set of variables. The syntax used is similar to any regression model syntax, including logistic models: it requires a dependent (binary) variable and a list of independent variables. A cross-validated AUC value is produced following the analysis of samples of the data (training sets) and is tested on the remaining sample (test set) based on a fold split of the original data. Each fold is analyzed using logistic or probit regression as if it represented the full study sample, and the value of the AUC is calculated from the predictions made on the test set, hence providing the cross-validated fitted probabilities for the dependent variable or outcome. They are contained in a new variable named `_fit` $= \mathrm{Pr}(x_{ik})$, where $\mathrm{Pr}(x_{ik})$ is derived from a logistic or probit regression model as the probability of a positive outcome, as follows:

$$\mathrm{Pr}(x_{ik}) = \frac{e^{(\beta_0 + \sum_{i=1}^{m} \beta_i x_i)}}{\left\{ 1 + e^{(\beta_0 + \sum_{i=1}^{m} \beta_i x_i)} \right\}} \quad \text{for all } x_i \text{ in fold } k$$

We then assume that we applied a diagnostic test classifying each X_{ik} observation as a normal or abnormal subject. Further assume that the higher the outcome value of the diagnostic test, the higher the risk of the subject being abnormal. The points on the nonparametric ROC curve are generated using each possible outcome of the diagnostic test as a classification cutpoint and computing the corresponding Se and $1 - Sp$ for each cutoff in fold k. These points are then connected by straight lines, and the area under the resulting ROC curve is computed using the trapezoidal rule (see the *Appendix*). The default standard error for the area under the ROC curve is computed using the algorithm described by DeLong, DeLong, and Clarke-Pearson (1988).

However, `cvauroc` implements k-fold cross-validation for the AUC for a binary outcome after fitting a logit or probit regression model, averaging the AUCs corresponding to each fold, and bootstrapping the cross-validated AUC to obtain statistical inference and 95% confidence intervals (CI). Furthermore, `cvauroc` provides the cross-validated AUC standard deviation, the fitted probabilities for the dependent variable or outcome, and the Se and Sp with their respective 95% CI. These values are contained in three new variables named _fit, _sen, and _spe, respectively.

`cvauroc` is an open, free program developed in Stata 15.1.

2 The cvauroc command

2.1 Syntax

cvauroc *depvar varlist* [*if*] [*weight*] [, <u>k</u>fold(#) <u>s</u>eed(#) <u>p</u>robit <u>fi</u>t
    <u>d</u>etail graph graphlowess]

depvar represents a binary outcome.

varlist is the list of independent factors whose predictive performance is tested.

pweights are allowed; see [U] **11.1.6 weight**.

2.2 Options

kfold(#) indicates the number of random splits that must be drawn from the original data. # must be an integer greater than 1. The default is kfold(10), but the user can decide the number of folds based on the sample size and number of events.

seed(#) allows the user to specify a seed for the random split of the data in k-fold splits. Analyses and results can therefore be reproducible. The default is seed(7777).

probit allows the user to fit a probit rather than a logit (default) model.

fit allows the user to generate a new variable (_fit) containing the cross-validated probabilities for the dependent variable or outcome.

detail allows the user to tabulate the prevalence of the independent variable or predictor, the Se, the Sp, and false-positive values by each level of the outcome fitted probabilities. Furthermore, it creates two new variables containing the cross-validated Se (_sen) and Sp (_spe) for the independent variable or predictor.

graph allows the user to graph the empirical ROC curves for the respective k folds specified by the user.

graphlowess allows the user to graph a smoothed version of the mean cross-validated ROC curve and the empirical ROC curves for the respective k folds specified by the user.

3 Illustration

We will use an excerpt from Cattaneo (2010) that looks at 4,642 singletons born in Pennsylvania in 1989–1991. We aim to estimate the probability of delivering a low birthweight (lbw) infant. First, we will fit a logistic regression model using maternal marital status (mmarried), the mother's age (mage), the mother's education (medu), the mother's race (mrace), the father's education (fedu), the mother's smoking behavior (mbsmoke), whether the mother had a prenatal doctor's visit in the baby's first trimester (prenatal1), and whether the baby is the mother's first child (fbaby) as independent predictors for lbw. Then, to understand the predictive ability of our chosen model, we will compute the AUC using the classical naïve approach, based on the fitted probabilities of the model, and will compare it with the AUC from the internal validation strategy implemented with the cvauroc statistical package.

We will also show the output generated for the graph and detail options. The graph option displays the ROC and the value of the AUC. The detail option shows the predictor-estimated or independent-variable-estimated cross-validated Se, Sp, and false-positive mean values by the levels of the outcome fitted probabilities. Furthermore, it provides two new variables, _sen and _spe, containing the cross-validated Se and Sp. The fit option provides the fitted probabilities for the dependent variable or outcome contained in the variable _fit.

```
. // Getting the data
. use http://www.stata-press.com/data/r14/cattaneo2.dta
(Excerpt from Cattaneo (2010) Journal of Econometrics 155: 138-154)

. generate lbw = cond(bweight<2500,1,0)

. // Fitting a classical logistic regression model
. logistic lbw mage medu mmarried prenatal1 fedu mbsmoke mrace fbaby, vsquish
```

Logistic regression

		Number of obs	=	4,642	
		LR chi2(8)	=	116.99	
		Prob > chi2	=	0.0000	
Log likelihood = -996.4102		Pseudo R2	=	0.0554	

lbw	Odds Ratio	Std. Err.	z	P>|z|	[95% Conf. Interval]	
mage	1.001829	.0134761	0.14	0.892	.975761	1.028592
medu	.9498238	.0276775	-1.77	0.077	.897097	1.00565
mmarried	.6334254	.1053742	-2.74	0.006	.4571859	.877603
prenatal1	1.053822	.1658733	0.33	0.739	.7740819	1.434656
fedu	1.039733	.0212445	1.91	0.057	.9989178	1.082217
mbsmoke	2.108549	.2991911	5.26	0.000	1.596626	2.78461
mrace	.3905866	.0596674	-6.15	0.000	.2895241	.5269265
fbaby	.9452252	.1313803	-0.41	0.685	.7198196	1.241215
_cons	.1590141	.0640445	-4.57	0.000	.0722114	.350159

Note: _cons estimates baseline odds.

```
. predict fitted, pr

. roctab lbw fitted
```

		ROC		─Asymptotic Normal─	
	Obs	Area	Std. Err.	[95% Conf. Interval]	
	4,642	0.6847	0.0172	0.65095	0.71848

```
. // Internal validation using cvauroc
. cvauroc lbw mage medu mmarried prenatal1 fedu mbsmoke mrace fbaby, seed(3489)
> kfold(10)
1-fold (N=465)........AUC =  0.607
2-fold (N=464)........AUC =  0.700
3-fold (N=464)........AUC =  0.686
4-fold (N=464)........AUC =  0.724
5-fold (N=464)........AUC =  0.669
6-fold (N=465)........AUC =  0.689
7-fold (N=464)........AUC =  0.759
8-fold (N=464)........AUC =  0.653
9-fold (N=464)........AUC =  0.659
10-fold (N=464)........AUC =  0.616

Model:logistic

Seed:3489
```

Cross-validated (cv) mean AUC, SD and Bootstrap Bias Corrected 95%CI

cvMean AUC:	0.6763
Bootstrap bias corrected 95%CI:	0.6431, 0.7172
cvSD AUC:	0.0461

```
. // Using the detail option to display the cross-validated sensitivity and
. // specificity and their respective 95% CI, and using the probit option
. // to fit a probit model instead of a logit model
. cvauroc lbw mage medu mmarried prenatal1 fedu mbsmoke mrace fbaby, seed(3489)
> kfold(10) probit fit detail
  (output omitted)
```

───

Mean cross-validated Sen, Spe and false(+) at lbw predicted values
───

Prevalence of lbw: 6.01%
─────────────────────────────

Summary statistics: mean
  by categories of: _Pp (Predicted Probability)

_Pp	_sen	_spe	_fp
0.02	99.67	0.31	99.69
0.03	89.47	18.94	81.06
0.04	74.24	51.55	48.45
0.05	66.41	63.57	36.43
0.06	61.05	68.88	31.12
(output omitted)			
0.25	0.79	99.73	0.27
0.26	0.09	99.79	0.21
0.27	0.00	99.90	0.10
0.28	0.00	99.99	0.01

```
. // Using the graph option to display the cross-validated ROC curve
. cvauroc lbw mage medu mmarried prenatal1 fedu mbsmoke mrace fbaby, seed(3489)
> kfold(10) probit fit graphlowess
  (output omitted)
```
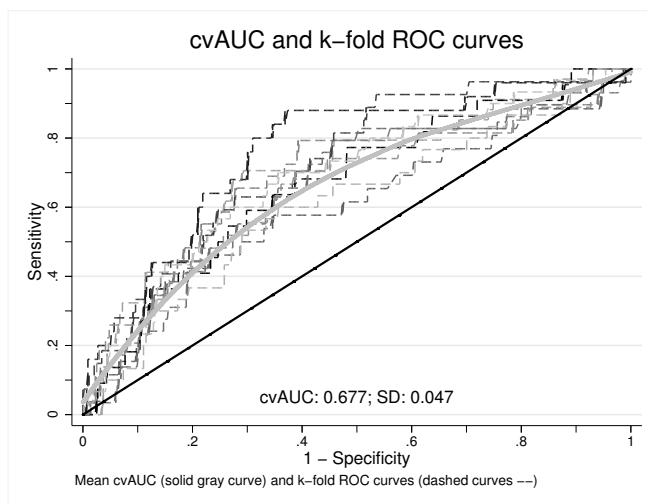

Figure 1. *K*-fold AUC and cross-validated AUC using `cvauroc`

Maternal smoking status and race are the strongest predictors of low birthweights, with babies from smoking mothers having twice the risk of low birthweights. There is further evidence that marital status, father's education, and mother's education are associated with low birthweights. The AUC calculated using `cvauroc` shows 0.0084 points lower accuracy (AUC = 0.6763) than the AUC computed using the classical approach from the predicted probabilities of low birthweights (AUC = 0.6847).

To illustrate `cvauroc`'s sampling weights option, we will randomly generate a censoring indicator variable to then compute the inverse probability of censoring weights (IPCW) and include it as a sampling weight in the model, allowing us to account for censoring.

```
. // Including sampling weights: generating a censoring indicator variable
. generate censor = rbinomial(1,0.4) // Censoring indicator

. generate lbw_cens = lbw if cens == 0 // Uncensored outcome
(1,844 missing values generated)

. tabulate censor
  (output omitted )

. tabulate lbw_cens
  (output omitted )

. // Generating the inverse probability of censoring weights (IPCW)
. logistic censor mage medu mmarried prenatal1 fedu mbsmoke mrace fbaby
  (output omitted )

. predict cw, pr

. generate ipcw = .
(4,642 missing values generated)

. replace ipcw=(cens==1)/cw if cens==1
(1,844 real changes made)

. replace ipcw=(cens==0)/(1-cw) if cens==0
(2,798 real changes made)

. // Including the IPCW as sampling weights in cvauroc
. cvauroc lbw_cens mage medu mmarried prenatal1 fedu mbsmoke mrace fbaby
> [pw=ipcw], kfold(10) seed(3489)
1-fold (N=280).........AUC =  0.768
2-fold (N=280).........AUC =  0.662
3-fold (N=280).........AUC =  0.668
4-fold (N=280).........AUC =  0.695
5-fold (N=279).........AUC =  0.664
6-fold (N=280).........AUC =  0.743
7-fold (N=280).........AUC =  0.612
8-fold (N=280).........AUC =  0.774
9-fold (N=280).........AUC =  0.617
10-fold (N=279).........AUC =  0.735

Model:logistic

Seed:3489
```

```
Cross-validated (cv) mean AUC, SD and Bootstrap Bias Corrected 95%CI
```

cvMean AUC:	0.6938
Bootstrap bias corrected 95%CI:	0.6155, 0.7199
cvSD AUC:	0.0588

Table 1 shows the AUC for the naïve and `cvauroc`-based 10-fold cross-validated methods.

Table 1. AUC from the different methods

Method	AUC
Naïve	0.6847; 95% CI [0.6509, 0.7184]
10-fold cross-validated (IPCW)	0.6938; 95% CI [0.6155, 0.7199]
10-fold cross-validated (uncensored outcome)	0.6763; 95% CI [0.6431, 0.7272]

4   Conclusion

To summarize, we have shown that evaluating the predictive performance of a set of independent variables using all cases from the original analysis sample tends to result in an overly optimistic estimate of predictive performance. However, `cvauroc` is a user-friendly and helpful K-fold internal cross-validation technique that might be considered when reporting the AUC in observational studies.

5   Acknowledgment

6   Programs and supplemental materials

To install the latest version of `cvauroc` from GitHub, you can first install the `github` package (Haghish 2016).

```
. net install github, from("https://haghish.github.io/github/")
```

Then, type the following to install `cvauroc`:

```
. github install migariane/cvauroc
```

To uninstall the package, type

```
. ado uninstall cvauroc
```

Note: The `github` package works only with Stata 13.1 and later. With an earlier version, you can install the package manually by downloading the files from the Github repository and placing them in your `PERSONAL` directory.

To install a snapshot of the corresponding software files as they existed at the time of publication of this article, type

```
. net sj 19-3
. net install st0569      (to install program files, if available)
. net get st0569          (to install ancillary files, if available)
```

7   References

Altman, D. G., Y. Vergouwe, P. Royston, and K. G. M. Moons. 2009. Prognosis and prognostic research: Validating a prognostic model. *British Medical Journal* 338: b605.

Cattaneo, M. D. 2010. Efficient semiparametric estimation of multi-valued treatment effects under ignorability. *Journal of Econometrics* 155: 138–154.

Collins, G. S., J. B. Reitsma, D. G. Altman, and K. G. M. Moons. 2015. Transparent reporting of a multivariable prediction model for individual prognosis or diagnosis (TRIPOD): The TRIPOD statement. *BMC Medicine* 13: 1.

DeLong, E. R., D. M. DeLong, and D. L. Clarke-Pearson. 1988. Comparing the areas under two or more correlated receiver operating characteristic curves: A nonparametric approach. *Biometrics* 44: 837–845.

Fawcett, T. 2006. An introduction to ROC analysis. *Pattern Recognition Letters* 27: 861–874.

Haghish, E. F. 2016. github: A module for building, searching, and installing Stata packages from GitHub. GitHub. https://github.com/haghish/github.

James, G., D. Witten, T. Hastie, and R. Tibshirani. 2013. *An Introduction to Statistical Learning: With Applications in R*. New York: Springer.

LeDell, E., M. Petersen, and M. van der Laan. 2015. Computationally efficient confidence intervals for cross-validated area under the ROC curve estimates. *Electronic Journal of Statistics* 9: 1583–1607.

Pepe, M. S. 2000. An interpretation for the ROC curve and inference using GLM procedures. *Biometrics* 56: 352–359.

Pepe, M. S., Z. Feng, and J. W. Gu. 2008a. Comments on 'Evaluating the added predictive ability of a new marker: From area under the ROC curve to reclassification and beyond' by M. J. Pencina et al., *Statistics in Medicine* (DOI: 10.1002/sim.2929). *Statistics in Medicine* 27: 173–181.

Pepe, M. S., Y. Zheng, Y. Jin, Y. Huang, C. R. Parikh, and W. C. Levy. 2008b. Evaluating the ROC performance of markers for future events. *Lifetime Data Analysis* 14: 86–113.

About the authors

Miguel Angel Luque-Fernandez is an assistant professor of epidemiology (honorary) at the London School of Hygiene and Tropical Medicine and a senior epidemiologist and biostatistician at the Biomedical Research Institute of Granada at the University of Granada. He also holds appointments as a scientific collaborator with the Department of Biostatistics of the Berkeley School of Public Health, the Department of Epidemiology at the Harvard T. H. Chan School of Public Health, and the Spanish Biomedical Network Research Centers of Epidemiology and Public Health (CIBERESP, ISCIII). His research interests lie principally, but not exclusively, in the field of epidemiological methods and comparative effectiveness research targeting the socioeconomic inequalities in cancer outcomes. Currently, Luque-Fernandez is collaborating with colleagues from the Cancer Survival Group at the London School of Hygiene and Tropical Medicine to develop data-adaptive methods for model selection and evaluation based on cross-validation techniques and applying advanced causal inference methods such as targeted maximum likelihood estimation (TMLE) to study cancer outcomes.

Daniel Redondo-Sánchez is a mathematician (University of Granada) specializing in epidemiology (Andalusian School of Public Health and University of Granada). He is currently studying socioeconomic inequalities in the geographic distribution of incidence, mortality, and net survival of cancer in Spain. Redondo-Sánchez holds a research position at the Biomedical Research Institute of Granada and is a research collaborator in the Biomedical Network Research Centers of Epidemiology and Public Health (CIBERESP, ISCIII).

Camille Maringe is a research fellow in the Cancer Survival Group at the London School of Hygiene and Tropical Medicine. She works on developing statistical tools for survival model selection for the prediction of cancer survival. Her primary research interests are in the analyses of linked electronic health records to explain inequalities in cancer outcomes. She is also interested in using observational data to emulate randomized clinical trials.

Appendix

The trapezoidal rule is a technique for approximating the definite integral

$$\int_a^b f(x)\,dx$$

The trapezoidal rule works by approximating the region under the graph of the function $f(x)$ as a trapezoid and calculating its area. It follows that

$$\int_a^b f(x)\,dx \approx \tfrac{\Delta x}{2}\{f(x_0) + 2f(x_1) + 2f(x_2) + 2f(x_3) + 2f(x_4)$$
$$+ \cdots + 2f(x_{n-1}) + f(x_n)\}$$

where $\Delta x = (b-a)/n$ and $x_i = a + i\Delta x$.

The trapezoidal rule may be viewed as the result obtained by averaging the left and right Riemann sums.