The Stata Journal (2009)
9, Number 1, pp. 1–16

Estimation and comparison of receiver operating characteristic curves

Margaret S. Pepe
Fred Hutchinson Cancer Research Center
Seattle, WA
mspepe@u.washington.edu

Gary Longton
Fred Hutchinson Cancer Research Center
Seattle, WA
glongton@fhcrc.org

Holly Janes
Fred Hutchinson Cancer Research Center
Seattle, WA
hjanes@fhcrc.org

Abstract. The receiver operating characteristic (ROC) curve displays the capacity of a marker or diagnostic test to discriminate between two groups of subjects, cases versus controls. We present a comprehensive suite of Stata commands for performing ROC analysis. Nonparametric, semiparametric, and parametric estimators are calculated. Comparisons between curves are based on the area or partial area under the ROC curve. Alternatively, pointwise comparisons between ROC curves or inverse ROC curves can be made. We describe options to adjust these analyses for covariates and to perform ROC regression in a companion article. We use a unified framework by representing the ROC curve as the distribution of the marker in cases where we have standardized it to the control reference distribution.

Keywords: st0154, roccurve, comproc, receiver operating characteristic curve

1 Introduction

1.1 Definition of the ROC curve

The receiver operating characteristic (ROC) curve displays the discriminatory capacity of a marker or test. Suppose $D = 0$ denotes controls and $D = 1$ denotes cases, and assume without loss of generality that larger values of Y are more indicative of a subject being a case. The ROC curve for marker Y is a plot of the true positive rate $\text{TPR}(c) = P(Y \geq c \,|\, D = 1)$ versus the false positive rate $\text{FPR}(c) = P(Y \geq c \,|\, D = 0)$ for the threshold criterion $Y \geq c$, where c varies from $-\infty$ to ∞. It is a monotone increasing function in the unit square tied down at the boundary points $(0, 0)$ and $(1, 1)$. A perfect classifier has an ROC curve that rises steeply along the left axis to the point ($\text{FPR} = 0$, $\text{TPR} = 1$), while an uninformative marker has an ROC curve that is the diagonal 45° line. Key attributes of the ROC curve are that 1) it does not depend on the raw measurement units for Y (it is invariant to monotone increasing transformations of Y); 2) it provides a common scale for comparing performances of different markers; and 3) it displays the range of possible performance levels that can be achieved by varying the threshold.

Figure 1 shows empirical ROC curves for two pancreatic cancer biomarkers (Wieand et al. 1989). The data can be downloaded from the Diagnostic and Biomarkers Statistical Center web site (http://www.fhcrc.org/labs/pepe/dabs), or it can be loaded directly into a Stata session:

```
. use http://labs.fhcrc.org/pepe/book/data/wiedat2b
. roccurve d y1 y2, roc(.2) level(90)
```

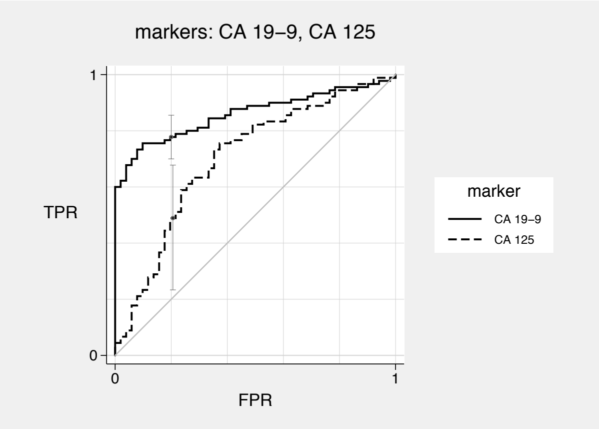

Figure 1. Nonparametric ROC curves for two markers of pancreatic cancer. Ninety percent confidence intervals for ROC(0.2) are displayed.

1.2 Representation in terms of percentile values (PVs)

Let F denote the left-continuous cumulative distribution of Y in the control population, $F(y) = P(Y < y \,|\, D = 0)$. We define a standardization of Y for the ith subject, with a marker value of Y_i.

$$\mathrm{PV}_i = F(Y_i)$$

is the proportion of the control population with values below Y_i. In lay terms, $\mathrm{PV}_i \times 100$ is the percentile of Y_i when the controls are considered the reference population against which to standardize the marker. We next show that the ROC curve can be written as the distribution of these standardized marker measurements (Pepe and Cai 2004; Pepe and Longton 2005). This identity suggests simple algorithms for implementing standard ROC methods and also gives rise to some new methods (Huang and Pepe Forthcoming).

<u>Result 1</u>

The ROC curve is the cumulative distribution of $1 - \text{PV}_D$,

$$\text{ROC}(f) = P(1 - \text{PV}_D \leq f)$$

where PV_D denotes the standardized marker for a case.

<u>Proof</u>

Let y be a marker threshold and note that the corresponding FPR, f, satisfies $F(y) = 1 - f$. Let Y_D denote the marker value from a random case. If the control distribution of Y is continuous, then F is monotone strictly increasing, and we see that

$$\begin{aligned}\text{ROC}(f) &\equiv P(Y_D \geq y)\\ &= P\{F(Y_D) \geq F(y)\}\\ &= P(\text{PV}_D \geq 1 - f) = P(1 - \text{PV}_D \leq f)\end{aligned}$$

If F has discrete mass points, then this proof also holds when y is a mass point. If y is not a mass point but (y^-, y^+) are the closest values, $y^- < y < y^+$, then $f = 1 - F(y^+)$ and $\text{ROC}(f) = P(Y_D > y^-) = P\{F(Y_D) \geq F(y^+)\} = P\{F(Y_D) \geq 1 - f\}$.

2 Estimating the ROC curve

The representation in result 1 suggests that the ROC curve estimation can be accomplished in two steps:

1. Estimate the reference cumulative distribution function (c.d.f.), F, using controls; calculate corresponding standardized marker values for cases.
2. Estimate the cumulative distribution of the standardized marker values for cases.

2.1 The control reference distribution

The empirical estimator of the control reference distribution can be used. Alternatively, a parametric model can be assumed. The `roccurve` command, which we introduce in section 4, allows one to use either the empirical method or a normal parametric distribution model.

Marker values for cases are standardized with the estimator $\widehat{F}$. Write the standardized values as

$$\widehat{\text{PV}}_{Di} = \widehat{F}(Y_{Di}), \quad i = 1, \ldots, n_D$$

where n_D is the number of case observations.

2.2 The c.d.f. of standardized markers

The next step is to estimate the c.d.f. of $1 - \mathrm{PV}_D$, denoted by H. The empirical c.d.f. is a nonparametric option provided by `roccurve`. A parametric model can be used instead. The parametric model has the advantage of providing a smooth ROC curve instead of a step function. The parametric forms allowed by `roccurve` are

$$H(f) = g\{\alpha_0 + \alpha_1 g^{-1}(f)\}$$

where g is a c.d.f. In this form, the domain for H is restricted to $(0, 1)$. As a special case, when $g = \Phi$, the standard normal distribution, the corresponding ROC curve is binormal (Dorfman and Alf 1969):

$$\mathrm{ROC}(f) = H(f) = \Phi\{\alpha_0 + \alpha_1 \Phi^{-1}(f)\}$$

The `roccurve` command also allows the logistic form, $g(\cdot) = \exp(\cdot)/\{1+\exp(\cdot)\}$, which gives rise to a bilogistic ROC curve (Ogilvie and Creelman 1968).

To fit these parametric models, a set of discrete points on the FPR axis is chosen, $\{f_1, \ldots, f_{n_p}\}$. For each case i and for each f_k, a record is created that includes the binary variable $U_{ki} = \mathbf{I}_{(1-\widehat{\mathrm{PV}}_{Di} \leq f_k)}$ and the covariate $g^{-1}(f_k)$. Fitting a binary regression model with the link function, g; the outcome variable, U; and the covariate, $g^{-1}(f)$ yields estimates of (α_0, α_1) (Alonzo and Pepe 2002).

In some applications, one may want to model only the ROC curve over a restricted FPR range, $(a, b) \subset (0, 1)$, in which case the FPR points $\{f_1, \ldots, f_{n_p}\}$ should span the interval (a, b).

In figure 2, we display four different estimators applied to data on the pancreatic cancer biomarker CA-125. The first estimator is the standard empirical ROC curve that results from standardizing with the left-continuous empirical control reference distribution and applying the empirical c.d.f. for H. This is precisely the same as the empirical estimator that is provided by Stata's `roctab` command. The second estimator is the semiparametric binormal estimator that calculates the standardized values with the empirical control distribution for Y but uses a probit link function for g. This rank-invariant semiparametric estimator requires less computation than the binormal estimator provided by Stata's `rocfit` command and appears to have similar efficiency (Alonzo and Pepe 2002). The third estimator assumes that the marker is normally distributed in controls and is not rank invariant. It calculates standardized values as

$$\mathrm{PV}_{Di} = \Phi\{(Y_{Di} - \mathrm{mean})/\mathrm{sd}\}$$

where mean and sd are the sample mean and the standard deviation, respectively, of the control observations. The fourth estimator is fully parametric. In addition to modeling the control reference distribution as normal, it assumes the ROC curve is binormal. The two assumptions taken together are equivalent to assuming that markers for both cases and controls are normally distributed. In practice, the rank-invariant estimators are more popular. Parametric models for the reference distribution have a more prominent role in settings where covariates affect marker distributions, and covariate-specific distributions are difficult to estimate empirically (Janes, Longton, and Pepe 2009).

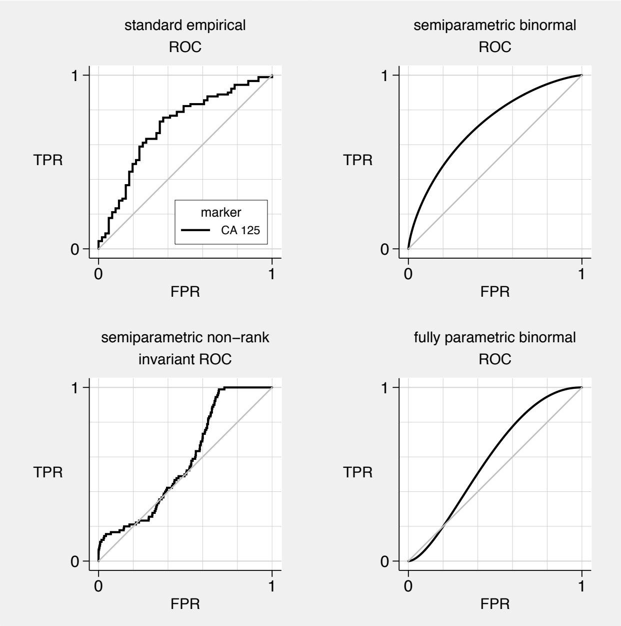

Figure 2. ROC curves for CA-125 as a marker of pancreatic cancer

We created the graphs in figure 2 with the following commands (your graphs will look slightly different because we do not show all the options here):

```
. use http://labs.fhcrc.org/pepe/book/data/wiedat2b
. roccurve d y2, pvcmeth(empirical) rocmeth(nonparametric)
. roccurve d y2, pvcmeth(empirical) rocmeth(parametric) link(probit)
. roccurve d y2, pvcmeth(normal) rocmeth(nonparametric)
. roccurve d y2, pvcmeth(normal) rocmeth(parametric)
```

(*Continued on next page*)

3 Sampling variability

We use bootstrap resampling to calculate pointwise confidence intervals (CIs) for the ROC curve, ROC(f), and for its inverse, ROC$^{-1}(t)$. In particular, if f is the FPR, the $(1-\alpha/2)$ and $\alpha/2$ quantiles of the bootstrap distribution of $\widehat{\text{ROC}}(f)$ are delivered as the $(1-\alpha)$ confidence limits.

The resampling must reflect the study design. If selection to the study was outcome dependent (that is, if a case–control design was used, as is common in early phase studies [Pepe et al. 2001]), then resampling is done separately within case and control strata. On the other hand, if subjects were enrolled without regard to their outcome status, resampling is done accordingly from the entire dataset. In addition, if observations are clustered (for example, if subjects contribute several observations to ROC curve estimation), the `cluster()` option can be used to identify resampling clusters.

4 The roccurve command

4.1 Syntax

The syntax for the `roccurve` command is

`roccurve` *disease_var test_varlist* [*if*] [*in*] [`,` `rocmeth(`*method*`)` `link(`*function*`)` `interval(`*a b* n_p`)` `nograph` `roc(`*f*`)` `rocinv(`*t*`)` `offset(`*#*`)` *connect_options* *twoway_options* `pvcmeth(`*method*`)` `tiecorr` `adjcov(`*varlist*`)` `adjmodel(`*model*`)` `nsamp(`*#*`)` `noccsamp` `nostsamp` `cluster(`*varlist*`)` `level(`*#*`)` `genrocvars` `genpcv` `replace`]

where *disease_var* is the name of the binary-outcome variable, $D = 1$ for a case and $D = 0$ for a control, and *test_varlist* comprises the names of markers or tests for which ROC curves are to be calculated.

4.2 Options

ROC calculation

`rocmeth(`*method*`)` specifies whether `nonparametric` (empirical ROC, the default) or `parametric` ROCs are to be calculated.

`link(`*function*`)` specifies the ROC generalized linear models link function and is valid only if `rocmeth(parametric)` is specified. *function* can be one of the following:

`probit`, the default, corresponds to the binormal ROC model. That is, $\Phi^{-1}\{\text{ROC}(f)\} = \text{intercept} + \text{slope} \times \Phi^{-1}(f)$, where Φ is the standard normal c.d.f.

`logit` corresponds to the bilogistic ROC model. That is, $\text{logit}\{\text{ROC}(f)\} = \text{intercept} + \text{slope} \times \text{logit}(f)$.

`interval(`$a\ b\ n_p$`)` specifies the interval (a, b) and the number of points (n_p) in the interval over which the parametric ROC model is to be fit. The program uses equally spaced points in the interval. The default is `interval(0 1 10)`.

Graph

`nograph` suppresses the ROC plot when only returned numerical results are desired.

`roc(`*f*`)` indicates that bootstrap percentile-based CIs for the ROC at specified $\text{FPR} = f$ are to be included on the plot. The argument must be between 0 and 1. Only one of the `roc(`*f*`)` or `rocinv(`*t*`)` options can be specified.

`rocinv(`*t*`)` indicates that bootstrap percentile-based CIs for the inverse ROC at specified $\text{TPR} = t$ are to be included on the plot. The argument must be between 0 and 1. Only one of the `roc(`*f*`)` or `rocinv(`*t*`)` options can be specified.

`offset(`*#*`)` specifies the x- or y-axis offset from f or t for the placement of second and subsequent CIs for $\text{ROC}(f)$ or $\text{ROC}^{-1}(t)$ to avoid overlap of interval bars for different markers. The argument must be between 0 and 0.02; the default is `offset(0.006)`.

connect_options, the current scheme defaults for ROC connecting line color, pattern, and width, can be overridden with the `lcolor(`*colorstyle*`)`, `lpattern(`*linepatternstyle*`)`, and `lwidth(`*linewidthstyle*`)` options; see [G] ***connect_options***.

twoway_options include various graph options that override default axis options, titles, and overall graph appearance; see [G] ***twoway_options***. Exceptions include *marker_options* and the `by()` option.

The default *region_options* (see [G] ***region_options***), `xsize(`*#*`)` and `ysize(`*#*`)`, are 7 and 5, respectively. These can be overridden. If only one is specified, the other will be calculated to maintain a 7×5 graph-region ratio. Specifying both `xsize(`*#*`)` and `ysize(`*#*`)` at a different ratio is allowed, but this can result in the separation of plot axes from the ROC plot because of the square plot aspect-ratio constraint.

Standardization method

`pvcmeth(`*method*`)` specifies how the percentile values (PVs) are to be calculated. *method* can be one of the following:

`empirical`, the default, uses the empirical distribution of the test measure among controls $(D = 0)$ as the reference distribution for the calculation of case PVs. The PV for the case measure y_i is the proportion of control measures $Y_{\overline{D}} < y_i$.

`normal` models the test measure among controls with a normal distribution. The PV for the case measure y_i is the standard normal c.d.f. of $(y_i - \text{mean})/\text{sd}$, where the mean and the standard deviation are calculated by using the control sample.

`tiecorr` indicates that a correction for ties between case and control values is included in the empirical PV calculation. The correction is important only in calculating summary indices, such as the area under the ROC curve. The tie-corrected PV for a case with the marker value y_i is the proportion of control values $Y_{\overline{D}} < y_i$ plus one half the proportion of control values $Y_{\overline{D}} = y_i$, where $Y_{\overline{D}}$ denotes controls. By default, the PV calculation includes only the first term, i.e., the proportion of control values $Y_{\overline{D}} < y_i$. This option applies only to the empirical PV calculation method.

Covariate adjustment

`adjcov(`*varlist*`)` specifies the variables to be included in the adjustment.

`adjmodel(`*model*`)` specifies how the covariate adjustment is to be done. *model* can be one of the following:

`stratified` PVs are calculated separately for each stratum defined by *varlist* in `adjcov()`. This is the default if `adjmodel()` is not specified and `adjcov()` is. Each case-containing stratum must include at least two controls. Strata that do not include cases are excluded from calculations.

`linear` fits a linear regression of the marker distribution on the adjustment covariates among controls. Standardized residuals based on this fitted linear model are used in place of the marker values for cases and controls.

Sampling variability

These options are relevant only if either the `roc(`*f*`)` or the `rocinv(`*t*`)` option is specified.

`nsamp(`*#*`)` specifies the number of bootstrap samples to be drawn for estimating CIs. The default is `nsamp(1000)`.

`noccsamp` specifies that bootstrap samples be drawn from the combined sample rather than sampling separately from cases and controls; case–control sampling is the default.

`nostsamp` draws bootstrap samples without respect to covariate strata. By default, samples are drawn from within covariate strata when stratified covariate adjustment is requested via the `adjcov()` and `adjmodel()` options.

`cluster(`*varlist*`)` specifies variables identifying bootstrap resampling clusters. See the `cluster()` option of the `bootstrap` command ([R] **bootstrap**).

`level(`*#*`)` specifies the confidence level for CIs as a percentage. The default is `level(95)` or as set by `set level`.

New variable

These are options to create new variables.

`genrocvars` generates new pairs of variables, `fpr#` and `tpr#`, for each marker in *test_varlist*, with ROC coordinates for corresponding marker values. The empirical ROC curve, empirical `rocmeth()`, results from connecting the points as a right-continuous step function. New variable names are numbered (#) according to the marker variable order in *test_varlist*.

`genpcv` generates variables, `pcv#`, to hold PVs for each marker in *test_varlist*. New variable numbers (#) correspond to the marker variable order in *test_varlist*.

`replace` requests that the existing `fpr#`, `tpr#`, or `pcv#` variables be overwritten by `genrocvars` or `genpcv`.

Options to adjust the ROC curve estimates for covariates are described in more detail in another article in this issue (Janes, Longton, and Pepe 2009).

4.3 Saved results

Confidence limits for `roc(`*f*`)` or `rocinv(`*t*`)` and parameters for the ROC generalized linear models parametric curve are saved in `r()` when the corresponding options are specified:

Matrices

`r(ROC_ci)`	$n \times 3$ matrix of `roc(`*f*`)` or `rocinv(`*t*`)` estimates and confidence limits returned when either option is specified. Columns correspond to the point estimate and the lower and upper confidence bounds. Rows correspond to the marker variables included in *test_varlist*.
`r(BNParm)`	$n \times 2$ matrix of binormal or bilogistic curve parameter estimates when `rocmeth(parametric)` is specified. Columns correspond to α_0 and α_1 parameters. Rows correspond to markers.

5 Summary indices

5.1 Area and partial area

Measures derived from the ROC curve are used to summarize discriminatory accuracy. More importantly, they serve as the basis for test statistics with which to compare ROC curves. The most popular index is the area under the ROC curve (AUC), also known as the c-index or probability of correct ordering. AUC $= \text{Prob}(Y_D > Y_N) + 0.5\text{Prob}(Y_D = Y_N)$, where (Y_D, Y_N) are a random pair of case and control marker values. We and others (Pepe 2003, 78; Cook 2007) have argued against using the AUC as a key summary measure because it is not clinically relevant. Subjects do not present clinically as pairs, and typically the clinical problem is not to decide which member of such a pair is the case.

For clinical applications, we prefer to use the ROC (or ROC^{-1}) curve at a specific point. Consider ROC(f). Given that one is willing to accept an FPR equal to f, what proportion of cases will be detected? This is relevant to clinical practice. However, fixing one FPR of interest can be difficult. A compromise is the partial AUC (pAUC) that integrates the ROC curve over a range of FPRs (McClish 1989; Thompson and Zucchini 1989). Because low FPRs are typically of interest, one can calculate the pAUC between 0 and the largest acceptable FPR, denoted by f_0:

$$\text{pAUC}(f_0) = \int_0^{f_0} \text{ROC}(f)df$$

Interestingly, the classic nonparametric estimator of the AUC can be written as the sample mean of the *nonparametric* case PVs (DeLong, DeLong, and Clarke-Pearson 1988; Hanley and Hajian-Tilaki 1997):

$$\widehat{\text{AUC}}_e = \sum_{i=1}^{n_D} \widehat{\text{PV}}_{Di}/n_D$$

When ties between case and control marker values are present, a correction for ties is necessary in calculating the PVs so that $\widehat{\text{AUC}}_e$ corresponds to the trapezoidal empirical AUC:

$$\widehat{\text{PV}}^c_{Di} = \widehat{\text{PV}}_{Di} + \frac{1}{2}\widehat{e}_i$$

where $\widehat{e}_i$ is the proportion of control marker values equal to Y_{Di}. The empirical estimator of the pAUC (Dodd and Pepe 2003) can also be written as a sample mean:

$$\widehat{\text{pAUC}}_e(f_0) = \sum_{i=1}^{n_D} \max\{\widehat{\text{PV}}_{Di} - (1-f_0), 0\}/n_D \tag{1}$$

again with the aforementioned tie correction for cases tied with controls.

By using a parametric model for the control reference distribution, the average of parametric case percentiles yields another estimator of the AUC. Analogously, (1) with parametric case percentiles provides a semiparametric pAUC estimate. Tie corrections are not necessary when the estimated reference distribution is continuous.

In general, the calculation of parametric AUCs and pAUCs requires numerical integration, and these calculations are not output by our programs. The one exception is that the binormal AUC has the closed-form expression $\Phi(\alpha_0/\sqrt{1+\alpha_1^2})$. Stata's `rocfit` command provides this expression after fitting a binormal curve; our programs do not. We provide only estimates that are nonparametric with regard to the shape of the ROC curve. This is also true for point estimates of ROC(f) and ROC$^{-1}(t)$ that are output by the `comproc` command, which we introduce in section 6.

5.2 Comparisons

To compare ROC curves, we calculate a CI for the difference between ROC summary indices. A Wald statistic, dividing the observed difference by its standard error, is compared with the standard normal distribution in order to report a p-value. CIs and standard errors are again derived from the bootstrap distribution of the estimators. The `comproc` command outputs results for one or more of the AUC, ROC(f), ROC$^{-1}(t)$, or pAUC(f), where the fixed FPR $= f$ or fixed TPR $= t$ are specified by the data analyst.

6 The comproc command

6.1 Syntax

The syntax for the `comproc` command is

```
comproc disease_var test_var1 [test_var2] [if] [in] [, auc pauc(f) roc(f)
   rocinv(t) pvcmeth(method) tiecorr adjcov(varlist) adjmodel(model)
   nsamp(#) nobstrap noccsamp nostsamp cluster(varlist) resfile(filename)
   replace level(#)]
```

where *disease_var* is the binary-outcome status variable, and *test_var1* and *test_var2* are the markers. If only one marker is specified, summary indices are output for that marker but no comparisons are made.

6.2 Options

Summary statistics

`auc` compares markers with respect to the AUC. This is the default if no summary statistics are specified.

`pauc(`*f*`)` includes a comparison with respect to the pAUC for FPR < specified *f*. The argument must be between 0 and 1. A tie correction is included in the PV calculation if this option is included among the specified summary statistics and the empirical PV calculation method is used.

`roc(`*f*`)` compares markers with respect to the ROC at the specified FPR = *f*. The argument must be between 0 and 1.

`rocinv(`*t*`)` compares markers with respect to the inverse ROC, ROC$^{-1}(t)$, at the specified TPR = *t*. The argument must be between 0 and 1.

Standardization method

`pvcmeth(`*method*`)` specifies how the PVs are to be calculated. *method* can be one of the following:

`empirical`, the default, uses the empirical distribution of the test measure among controls ($D = 0$) as the reference distribution for the calculation of case PVs. The PV for the case measure y_i is the proportion of control measures $Y_{\overline{D}} < y_i$.

`normal` models the test measure among controls with a normal distribution. The PV for the case measure y_i is the standard normal c.d.f. of $(y_i - \text{mean})/\text{sd}$, where the mean and the standard deviation are calculated by using the control sample.

`tiecorr` indicates that a correction for ties between case and control values is included in the empirical PV calculation. The correction is important only in calculating summary indices, such as the AUC. The tie-corrected PV for a case with the marker value y_i is the proportion of control values $Y_{\overline{D}} < y_i$ plus one half the proportion of control values $Y_{\overline{D}} = y_i$. By default, the PV calculation includes only the first term, i.e., the proportion of control values $Y_{\overline{D}} < y_i$. This option applies only to the empirical PV calculation method.

Covariate adjustment

`adjcov(`*varlist*`)` specifies the variables to be included in the adjustment.

`adjmodel(`*model*`)` specifies how the covariate adjustment is to be done. *model* can be one of the following:

`stratified` PVs are calculated separately for each stratum defined by *varlist* in `adjcov()`. This is the default if `adjmodel()` is not specified and `adjcov()` is. Each case-containing stratum must include at least two controls. Strata that do not include cases are excluded from calculations.

`linear` fits a linear regression of the marker distribution on the adjustment covariates among controls. Standardized residuals based on this fitted linear model are used in place of the marker values for cases and controls.

Sampling variability

`nsamp(`*#*`)` specifies the number of bootstrap samples to be drawn for estimation of standard errors and CIs. The default is `nsamp(1000)`.

`nobstrap` omits bootstrap sampling and estimation of standard errors and CIs. If `nsamp()` is specified, `nobstrap` will override it.

`noccsamp` specifies that bootstrap samples be drawn from the combined sample rather than sampling separately from cases and controls; case–control sampling is the default.

`nostsamp` draws bootstrap samples without respect to covariate strata. By default, samples are drawn from within covariate strata when stratified covariate adjustment is requested via the `adjcov()` and `adjmodel()` options.

`cluster(`*varlist*`)` specifies variables identifying bootstrap resampling clusters. See the `cluster()` option of the `bootstrap` command ([R] **bootstrap**).

`resfile(`*filename*`)` creates a Stata file (a `.dta` file) with the bootstrap results for the included statistics. `bstat` can be run on this file to view the bootstrap results again.

`replace` specifies that if the specified file already exists, then the existing file should be overwritten.

`level(`#`)` specifies the confidence level for CIs as a percentage. The default is `level(95)` or as set by `set level`.

6.3 Saved results

`comproc` saves the following r-class results, where *stat* is one or more of `auc`, `pauc`, `roc`, or `rocinv`, corresponding to the requested summary statistics:

Scalars

`r(`*stat*`1)`	statistic estimate for first marker
`r(`*stat*`2)`	statistic estimate for second marker
`r(`*stat*`delta)`	estimate difference, *stat*2−*stat*1
`r(se_`*stat*`1)`	bootstrap standard-error estimate for first marker statistic
`r(se_`*stat*`2)`	bootstrap standard-error estimate for second marker statistic
`r(se_`*stat*`delta)`	bootstrap standard-error estimate for the difference, *stat*2−*stat*1

In addition, many of the standard e-class bootstrap results left behind by `bstat` are available after running `comproc`.

6.4 Example

The `comproc` command applied to the pancreatic-cancer marker data shown in figure 1 yielded the following results:

```
. set seed 8378923

. comproc d y1 y2, auc roc(0.2)
        Comparison of test measures
                                 test 1: CA 19-9
                                 test 2: CA 125

    percentile value calculation method: empirical
        percentile value tie correction: no

  bootstrap samples drawn
   separately from cases and controls

  # bootstrap samples: 1000
```

(*Continued on next page*)

```
****************
  AUC estimates and difference,
    test 2 - test 1 (aucdelta)
Bootstrap results                               Number of obs      =        141
                                                Replications       =       1000

------------------------------------------------------------------------------
             |    Observed               Bootstrap
             |       Coef.       Bias    Std. Err.  [95% Conf. Interval]
-------------+----------------------------------------------------------------
        auc1 |   .86056644  -.0010577   .03067768    .8004393   .9206936   (N)
             |                                       .7964053   .9174292   (P)
             |                                       .7989107   .9185185  (BC)
        auc2 |   .70413947   .0007451    .0471203    .6117854   .7964936   (N)
             |                                       .6093682   .7955338   (P)
             |                                       .6069717   .7921569  (BC)
    aucdelta |  -.15642697   .0018028   .05788385   -.2698772  -.0429767   (N)
             |                                       -.266122  -.0415033   (P)
             |                                      -.2666667  -.0422658  (BC)
------------------------------------------------------------------------------
(N)    normal confidence interval
(P)    percentile confidence interval
(BC)   bias-corrected confidence interval
test of Ho: auc1 = auc2
  z =    -2.7    p =   .0069
****************
 ROC estimates and difference,
    test 2 - test 1 (rocdelta)
  ROC(f) @ f = .2
Bootstrap results                               Number of obs      =        141
                                                Replications       =       1000

------------------------------------------------------------------------------
             |    Observed               Bootstrap
             |       Coef.       Bias    Std. Err.  [95% Conf. Interval]
-------------+----------------------------------------------------------------
        roc1 |   .77777779   .0011778   .04836552    .6829831   .8725725   (N)
             |                                       .6888889   .8777778   (P)
             |                                             .7   .8888889  (BC)
        roc2 |   .48888889  -.0091667   .13398627    .2262806   .7514971   (N)
             |                                       .2222222         .7   (P)
             |                                       .2333333   .7222222  (BC)
    rocdelta |   -.2888889  -.0103444   .14291224   -.5689918  -.0087861   (N)
             |                                      -.5777777  -.0444444   (P)
             |                                      -.5777777  -.0333334  (BC)
------------------------------------------------------------------------------
(N)    normal confidence interval
(P)    percentile confidence interval
(BC)   bias-corrected confidence interval
test of Ho: roc1 = roc2
  z =      -2    p =    .043
****************
```

The bootstrap result tables are generated by Stata's `estat bootstrap` command.

7 Remarks

Our programs rely on representing the ROC curve as the c.d.f. of the case marker values after they are standardized to the control reference distribution. This representation gives rise to simple algorithms for calculating *standard* nonparametric estimators of the ROC curve, the AUC, and pAUC(f). The representation also provides alternative estimators of the ROC and its summary indices that are semiparametric or fully parametric. In a companion article (Janes, Longton, and Pepe 2009) in this issue of the *Stata Journal*, we describe methods for covariate adjustment and ROC regression. The PV representation is particularly useful in these settings.

Applications to continuous data are our focus. Though the methods can be applied to ordinal markers and diagnostic tests, some standard ROC methods for ordinal data are not included in our routines. In particular, our algorithm for fitting the binormal ROC model does not correspond to the Dorfman and Alf algorithm (Dorfman and Alf 1969) for ordinal data. In addition, the AUC corresponding to a fitted binormal model is not output. Instead, nonparametric AUC estimates are provided. We recommend Stata's `roctab` command for fitting binormal models and calculating corresponding AUCs with ordinal data.

The Diagnostic and Biomarkers Statistical Center web site is a repository of information for the statistical evaluation of diagnostic tests and biomarkers. Included on the web site are datasets that can be used to gain familiarity with the methods and software described in this article.

8 References

Alonzo, T. A., and M. S. Pepe. 2002. Distribution-free ROC analysis using binary regression techniques. *Biostatistics* 3: 421–432.

Cook, N. R. 2007. Use and misuse of the receiver operating characteristic curve in risk prediction. *Circulation* 115: 928–935.

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.

Dodd, L. E., and M. S. Pepe. 2003. Partial AUC estimation and regression. *Biometrics* 59: 614–623.

Dorfman, D. D., and E. Alf Jr. 1969. Maximum-likelihood estimation of parameters of signal-detection theory and determination of confidence intervals—Rating-method data. *Journal of Mathematical Psychology* 6: 487–496.

Hanley, J. A., and K. O. Hajian-Tilaki. 1997. Sampling variability of nonparametric estimates of the areas under receiver operating characteristic curves: An update. *Academic Radiology* 4: 49–58.

Huang, Y., and M. S. Pepe. Forthcoming. Biomarker evaluation and comparison using the controls as a reference population. *Biostatistics*.

Janes, H., G. Longton, and M. S. Pepe. 2009. Accomodating covariates in receiver operating characteristic analysis. *Stata Journal* 9: 17–39.

McClish, D. K. 1989. Analyzing a portion of the ROC curve. *Medical Decision Making* 9: 190–195.

Ogilvie, J. C., and C. D. Creelman. 1968. Maximum-likelihood estimation of receiver operating characteristic curve parameters. *Journal of Mathematical Psychology* 5: 377–391.

Pepe, M. S. 2003. *The Statistical Evaluation of Medical Tests for Classification and Prediction*. Oxford: Oxford University Press.

Pepe, M. S., and T. Cai. 2004. The analysis of placement values for evaluating discriminatory measures. *Biometrics* 60: 528–535.

Pepe, M. S., R. Etzioni, Z. Feng, J. D. Potter, M. L. Thompson, M. Thornquist, M. Winget, and Y. Yasui. 2001. Phases of biomarker development for early detection of cancer. *Journal of the National Cancer Institute* 93: 1054–1061.

Pepe, M. S., and G. Longton. 2005. Standardizing diagnostic markers to evaluate and compare their performances. *Epidemiology* 16: 598–603.

Thompson, M. L., and W. Zucchini. 1989. On the statistical analysis of ROC curves. *Statistics in Medicine* 8: 1277–1290.

Wieand, S., M. H. Gail, B. R. James, and K. L. James. 1989. A family of nonparametric statistics for comparing diagnostic markers with paired or unpaired data. *Biometrika* 76: 585–592.

About the authors

Margaret Pepe is a full member, Gary Longton is a statistical research associate, and Holly Janes is an assistant member in the Public Health Sciences Division of the Fred Hutchinson Cancer Research Center in Seattle. A focus of their research is on the development of new methodology for diagnostic tests and biomarkers, with support provided by the National Cancer Institute (CA 129934 and CA 086368) and the National Institute for General Medical Studies (GM 054438). Pepe, Longton, and Janes also teach courses on statistical methods for evaluating tests and biomarkers.