The Stata Journal (2023)
23, Number 2, pp. 455–490

DOI: 10.1177/1536867X231175314

Visualizing uncertainty in a two-dimensional estimate using confidence and comparison regions

Maren Eckert
Institute of Medical Biometry and Statistics
Division Methods in Clinical Epidemiology
Faculty of Medicine and Medical Center
University of Freiburg
Freiburg, Germany
maren.eckert@imbi.uni-freiburg.de

Werner Vach
Basel Academy for Quality and Research in Medicine
Basel, Switzerland
werner.vach@basel-academy.ch
and Department of Environmental Sciences
University of Basel
Basel, Switzerland
werner.vach@unibas.ch

Abstract. Recently, Eckert and Vach (2020, *Biometrical Journal* 62: 598–609) pointed out that both confidence and comparison regions are useful tools to visualize uncertainty in a two-dimensional estimate. Both types of regions can be based on inverting Wald tests or likelihood-ratio tests. `confcomptwo` enables Stata users to draw both types of regions following one of the two principles for various two-dimensional estimation problems. The use of `confcomptwo` is illustrated by several examples.

Keywords: st0716, confcomptwo, two-dimensional parameter space, confidence region, comparison region, Wald test, profile likelihood, likelihood-ratio test, diagnostic accuracy studies

1 Introduction

1.1 The value of two-dimensional confidence regions

Today, scientists are accustomed to describing the uncertainty of single-parameter estimates with confidence intervals. Even when several parameter estimates are considered simultaneously—for example, when reporting results from multiple regression models—the uncertainty is usually described separately for each single-parameter estimate. This is despite the fact that a methodology for describing uncertainty simultaneously in several parameters is available: confidence regions. The value of using confidence regions

is illustrated in figure 1, depicting the joint uncertainty in two regression coefficient estimates from a multiple regression analysis. First, the confidence region is a gentle reminder that the two covariates—grip strength and age—are positively correlated, and consequently the two regression coefficient estimates are negatively correlated. When interpreting the magnitude of the two estimates, one should consider that an overestimation of one parameter probably implies an underestimation in the other. Second, it is immediately apparent that the point $(0, 0)$ is outside the confidence region, which means that the null hypotheses of no effect of both covariates can be rejected. This can be highly relevant if both regression parameters are not significantly different from 0. In this case, the confidence region is a gentle reminder that this should not be misinterpreted as absence of any association between the outcome and these two covariates.

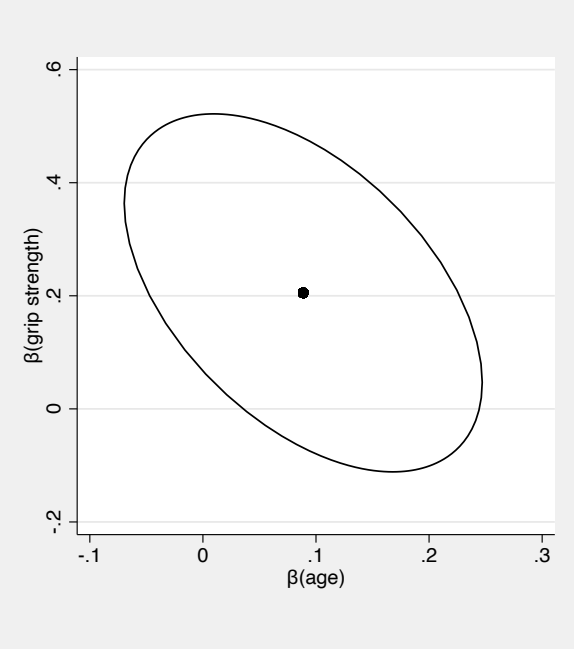

Figure 1. A 95% confidence region for two regression coefficients

Despite this rather obvious value of two-dimensional confidence regions in specific situations, they are rarely used in scientific publications, probably because of the lack of appropriate software for visualization. Indeed, when one uses the Wald test principle to construct two-dimensional confidence regions, visualization requires drawing an ellipsoid. This is usually not supported in standard statistical software packages. When one uses the likelihood-ratio (LR) test principle for construction, a fixed-point problem must be solved numerically for many directions in the two-dimensional space, as recently pointed out by Jaeger (2016). One aim of `confcomptwo` is to close this gap in Stata.

1.2 The need for comparison regions

Formally, confidence regions allow the post hoc testing of null hypotheses that fix the parameter at a certain value. For such a value $\boldsymbol{\theta}_0 \in \mathbb{R}^2$, the null hypothesis $\boldsymbol{\theta} = \boldsymbol{\theta}_0$ can be rejected at the level α if the point $\boldsymbol{\theta}_0$ is not covered by the $(1-\alpha)$ confidence region. This property was used in the previous subsection when we considered the location of the point $(0, 0)$ relative to the confidence region.

In some statistical applications, such hypotheses on single specific point values are of minor interest. Instead, the interest is in demonstrating that the two parameters of interest are within a certain region R, for example, that their average is above a certain threshold. (Concrete examples are given in the following subsection.) Thus, the interest is in testing the null hypothesis $H_0 : \boldsymbol{\theta} \notin R$. In the post hoc setting, this can be approached by checking whether the $(1-\alpha)$ confidence region completely covers the region of interest. If this is the case, the null hypothesis can be rejected at level α.

However, this test approach is very conservative, and the actual level of the test is much lower than α. To address this issue, Eckert and Vach (2020) introduced the concept of a comparison region. A level-α comparison region is a data-dependent region $C \subseteq \mathbb{R}^2$ with the following property:

$$\mathbb{1}(C \subseteq R) \text{ defines a level-}\alpha \text{ test for } H_0 : \boldsymbol{\theta} \notin R \text{ for any convex set } R \subset \mathbb{R}^2$$

Consequently, a comparison region can be used for post hoc testing of the null hypothesis of interest. The logic of the approach is illustrated in figure 2. Comparison regions can be constructed very similarly to confidence regions. Consequently, `confcomptwo` supports drawing of both confidence regions and comparison regions.

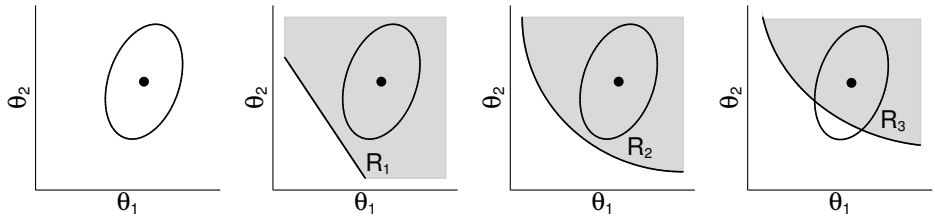

Figure 2. Presentation of a point estimate with a comparison region in a publication and three post hoc comparisons with regions of interest: The null hypotheses $H_0 : \boldsymbol{\theta} \notin R_1$ and $H_0 : \boldsymbol{\theta} \notin R_2$ can be rejected, and the null hypothesis $H_0 : \boldsymbol{\theta} \notin R_3$ cannot be rejected. (This figure was previously published in Eckert and Vach [2020].)

1.3 Motivating examples

1.3.1 Diagnostic accuracy studies

Diagnostic accuracy is a genuine two-dimensional concept. By sharpening the criteria of a diagnostic test, one can reduce the number of false-positive (FP) decisions, but the number of true-positive (TP) decisions decreases, too. Consequently, the standard approach to analyzing the diagnostic accuracy of one test is to consider a pair of parameters such as sensitivity and specificity, positive and negative predictive values, the relative frequency of TP and FP decisions, or the positive and negative LRs. When one analyzes screening tests, it might also be of interest to relate sensitivity to the rate of positive test results because the aim is to achieve a high sensitivity while keeping the number of subjects to be followed up with as low as possible. When one compares two diagnostic tests, the focus is typically on the change in such a pair of parameter values.

There is no universal answer to the question of how to combine the two parameter estimates when a final decision about the usefulness of the test (or the new test in a comparative study) must be made (compare Vach, Gerke, and Hœilund-Carlsen [2012]). When one considers sensitivity and specificity, one is often advised to think about which parameter is more important in the specific clinical context. It is a rather straightforward idea to consider a weighted average of sensitivity and specificity. In practice, however, it seems cumbersome to agree on specific weights. The use of a range of weights has been considered with respect to analyzing sensitivity and specificity (Newcombe 2001) as well as when considering the rate of TP and FP decisions, particularly as part of a decision curve approach (Vickers 2008).

When different stakeholders post hoc perform analyses of a diagnostic accuracy study, each stakeholder may specify different weights and thresholds. Each stakeholder is interested in demonstrating that $w\boldsymbol{\theta}_1+(1-w)\boldsymbol{\theta}_2 \geq t$ for some $w, t \in \mathbb{R}$, that is, that the two parameters are within a certain half-space. If we want to satisfy all stakeholders, we have to reject the null hypothesis that the true parameters are outside the intersection of all of these half-spaces. This intersection is a convex subset of $\mathbb{R}^2$ and represents the common region of interest of all stakeholders.

1.3.2 Balancing between favorable and unfavorable consequences of an intervention

In evaluating a new intervention, one must often judge the balance between favorable and unfavorable consequences. Cost-effectiveness analyses are a classical example. Today, they are standard tools in health technology assessments and play important roles in deciding whether the additional costs can be justified by a gain in effectiveness or whether cost savings are so substantial that they can justify some loss in effectiveness. Similarly, there may be a need to balance the benefit and the risk associated with an intervention, and in benefit–risk assessments, it is common to consider a benefit–risk plane similar to a cost-effectiveness plane (Guo et al. 2010; Mt-Isa et al. 2014). Considering a two-dimensional approach can be particularly helpful to overcome some obstacles with noninferiority trials (Gladstone and Vach 2015).

In these types of analyses, because the parameters of interest are measured on differing scales, it is common to consider the ratio $\boldsymbol{\theta}_2/\boldsymbol{\theta}_1$ for the joint evaluation of two parameter estimates and to define acceptable ratios r. Because $\boldsymbol{\theta}_2/\boldsymbol{\theta}_1 \geq r$ is equivalent to $\boldsymbol{\theta}_2 - r\boldsymbol{\theta}_1 \geq 0$, this still leads to linear hypotheses about the parameters of interest.

1.4 Construction of confidence regions and comparison regions

1.4.1 A general construction principle for comparison regions

Eckert and Vach (2020) describe the following general construction principle for a comparison region:

Lemma 1: Let ϕ_H denote a family of level-α tests for all half spaces $H \subset \Theta$, that is, all subsets of the two-dimensional parameter space Θ of the type $\{(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) | w_1\boldsymbol{\theta}_1 + w_2\boldsymbol{\theta}_2 \geq t\}$ for some $w_1, w_2, t \in \mathbb{R}$. ϕ_H provides a test for the null hypothesis $H_0 : \boldsymbol{\theta} \in H$. Then (under some regularity conditions on the family of tests),

$$C := \bigcap_{\phi_H = 1} \overline{H}$$

defines a level-α comparison region. Here $\overline{H}$ denotes the complement of a set H.

Roughly speaking, C can be interpreted as the intersection of all regions of interest that were "confirmed" by the tests. This general principle can be applied using the family of Wald tests or the family of LR tests. In both cases, the construction of comparison regions turns out to be very similar to the construction of confidence regions.

1.4.2 Explicit formulas for confidence regions and comparison regions

Given an estimate of the covariance matrix $\boldsymbol{\Sigma}$ of $\widehat{\boldsymbol{\theta}}$, applying the Wald test principle results in confidence and comparison regions of the type

$$C_\alpha = \left\{ \boldsymbol{\theta} \in \Theta \mid \left(\boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}\right)' \widehat{\boldsymbol{\Sigma}}^{-1} \left(\boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}\right) < c \right\}$$

The only difference is the choice of the threshold c. $c = \chi^2_{2,1-\alpha}$ defines a $(1-\alpha)$ confidence region. $c = \chi^2_{1,1-2\alpha}$ defines a level-α comparison region.

Using (asymptotic) LR tests as construction principles results in confidence and comparison regions of the type

$$C_\alpha = \left\{ \boldsymbol{\theta} \in \Theta \mid l(\boldsymbol{\theta}) - l^* > -\frac{1}{2}c \right\}$$

with l^* denoting the value of the loglikehood function $l(\theta)$ at the maximum likelihood estimate. Again, the only difference is the choice of the threshold c. $c = \chi^2_{2,1-\alpha}$ defines a $(1-\alpha)$ confidence region. $c = \chi^2_{1,1-2\alpha}$ defines a level-α comparison region for any $\alpha < 0.5$.

Note that, for both approaches, it does hold that 74.2% confidence regions define 5% comparison regions. Consequently, 5% comparison regions are distinctly smaller than 95% confidence regions.

2 The confcomptwo command

2.1 The scope of confcomptwo

The command `confcomptwo` allows comparison and confidence regions to be drawn in two-parameter estimation problems. The approaches based on inverting Wald tests and on inverting LR tests are both supported. With the Wald-test-based approach, `confcomptwo` can be used as a postestimation command or as an immediate command specifying the five necessary input values. With the LR approach, the user must specify a program to compute a log likelihood depending on two parameters.

Eckert and Vach (2020) considered the case of a two-parameter likelihood, but the approach can also be applied to a two-parameter profile likelihood obtained by maximizing a higher-dimensional likelihood over the remaining nuisance parameters. Hence, `confcomptwo` also extends the `pllf` command to compute one-dimensional profile-likelihood-based confidence intervals. This command was provided by Royston (2007).

Actually, `confcomptwo` does not draw a line directly but uses a grid of points, plots the points, and connects the points with straight lines. The drawing is hence based on Stata's `twoway line` command, but `confcomptwo` also supports representing the line with a series of small points. A comparison region and a confidence region can both be drawn simultaneously if both post hoc testing of hypotheses on half spaces and post hoc testing of single-point hypotheses are of interest. Furthermore, `confcomptwo` also allows following the recommendation of Eckert and Vach (2020) to draw comparison regions by a solid line and to draw confidence regions by a dotted line, reminding the user of the intended function—a comparison with a region of interest or with a single point.

2.2 The syntax of confcomptwo

The syntax of `confcomptwo` depends on whether you intend to use the Wald principle after fitting some model, the immediate form, or the LR principle.

If you intend to use `confcomptwo` as a postestimation command and apply the Wald test principle, the syntax is given by

`confcomptwo` *parname1 parname2* [`,` *confcomp_options twoway_options*]

and the program expects to find entries with the column names *parname1* and *parname2* in the vector `e(b)`, including the estimates and corresponding entries for the variances and covariance in the matrix `e(V)`.

The immediate form has the syntax

`confcomptwo` *#1 #2*`,` `se1(`*real*`)` `se2(`*real*`)` [`corr(`*real*`)` *confcomp_options twoway_options*]

with *#1* and *#2* denoting the values of the two parameter estimates. The standard errors are provided using the required `se1()` and `se2()` options. The default correlation is `corr(0)`.

If you intend to use the LR test principle, the syntax is given by

`confcomptwo` *#1 #2*`,` `call(`*expr*`)` [*confcomp_options twoway_options linesearch_options*]

and the *expr* in the required `call()` option describes the call of a user-written program that returns a log-likelihood value in `r(ll)`. It should also return a binary indicator, `r(inside)`, indicating whether the two arguments provided are within the allowed parameter values. The symbols *#1* and *#2* in this expression refer to the two parameter values at which the log likelihood should be evaluated. The parameter set allowed must be equal to $\mathbb{R}^2$ or an open and convex subset of $\mathbb{R}^2$. The log-likelihood function must be strictly concave, and it must tend toward $-\infty$ if the parameter values approach the boundary of the open set or if they tend toward $\pm\infty$.

`confcomptwo` allows the following *confcomp_options*:

`alpha(`*real*`)` defines the level of the comparison region or 1 minus the level of the confidence region. The default is `alpha(0.05)`.

`points(`*numlist*`)` defines the points to be drawn. If *numlist* contains one number, it is interpreted as the number of points required. Otherwise, it defines a grid of directions in which the points are chosen relative to the parameter estimates. `0` and `1` refer to the direction "top of the graph", `0.25` and `0.75` to the directions "right side of the graph" and "left side of the graph", and `0.5` to the direction "bottom of the graph". The default is `points(101)`.

`rescale(`*expr*`)` modifies the choice of the directions. The argument of the expression is noted with a `#` sign. Details are described in section 4.1.

`loptsconf(`*line_options*`)` specifies the look of the line used to draw the confidence region. The default is `loptsconf(lpat(dot))`.

`loptscomp(`*line_options*`)` specifies the look of the line used to draw the comparison region. The default is `loptscomp(lpat(solid))`.

`mopts(`*marker_options*`)` specifies the look of the marker used to draw the point estimate. The default is `mopts(mcol(black) msym(O))`.

`only(comp|conf)` indicates that only confidence regions or comparison regions are to be computed.

`nograph` suppresses the graph.

`savepoints(`*filename*`[ , `*saveoptions*`])` saves the points computed as a Stata dataset with the given *filename*. The dataset includes the variables `est1`, `est2`, `xi1`, `xi2`, `pcomp1`, `pcomp2`, `pconf1`, and `pconf2`. The first two variables represent the input parameter values, and the next two variables represent the direction in which the point was searched for. The last four variables include the coordinates of the points used when drawing the lines. *saveoptions* can be any options allowed by Stata's `save` command.

`reverse` exchanges the two axes in the plot. The default is to use the y axis for the first parameter and the x axis for the second parameter.

`addplot(`*plot* ...`[ || `*plot* ...`[...]][ , below])` adds twoway plots to the graph and should work similarly to Stata's `addplot()` option.

In addition, any *twoway_options* affecting the entire graph can be used except those involving a variable (such as the `by()` option). `confcomptwo` automatically generates a legend, and the `label()` suboption of the `legend()` option can be used to change the text. If the `addplot()` option is used, further entries are generated, and the `order()` suboption must be used to select the entries to be displayed.

If you are using the LR principle, you can also specify some *linesearch_options*, which are explained in section 4.3.

The `dot` line pattern style of Stata produces very small dots that are often hard to see. To address this issue, the options `loptsconf()` and `loptscomp()` also accept specifications of the following type:

`points, n(`*integer*`) [yxratio(`*real*`) `*scatteroptions*`]`

This indicates that the line is drawn by `n()` single points using the `twoway scatter` command. The points are drawn equidistantly, and their appearances can be changed using any option allowed with `twoway scatter`. `yxratio()` must be specified if the span of the y axis is not equal to the span of the x axis. If the `savepoints()` option is used, these additional points are also saved.

2.3 Stored results

`confcomptwo` stores in `r(cmd)` the command that was used to generate the graph. In combination with the `savepoints()` option, this allows the reproduction and further manipulation of the graph, if necessary. The point estimate is stored in the two scalars `r(est1)` and `r(est2)`. If the Wald test principle is used, the variance–covariance matrix $\boldsymbol{\Sigma}$ is stored in the matrix `r(Sigma)`.

2.4 Auxiliary commands to compute a log likelihood

To support the use of the LR principle, we provide the following three commands to compute the (profile) log likelihood for some standard situations that particularly appear in the context of analyzing diagnostic accuracy studies. All commands consider an open subset of $\mathbb{R}^2$ as parameter space.

`llproptwosamples` *#1 #2* [*if*] [*in*] [*weight*], `var(`*varname*`)` `by(`*varname*`)`

computes the log likelihood for a binary outcome variable with success probabilities differing between two subgroups. The log likelihood is evaluated for the two probabilities *#1* and #2, referring to the two subgroups that must be labeled with the values 1 and 2. The following two options are required:

`var(`*varname*`)` defines the binary variable.

`by(`*varname*`)` defines the dichotomous grouping variable.

`fweight`s are allowed; see [U] **11.1.6 weight**. In particular, this likelihood can be used to evaluate sensitivity and specificity together (or positive and negative predictive values) for one diagnostic test. The parameter space is $(0, 1)^2$.

`llproptwocats` *#1 #2* [*if*] [*in*] [*weight*], `var(`*varname*`)` `cats(`*numlist*`)`

computes the profile log likelihood for a categorical variable. The log likelihood is evaluated for the two probabilities *#1* and *#2*, referring to the probabilities of two categories. The probabilities of the other categories remain unspecified. The following two options are required:

`var(`*varname*`)` defines the categorical variable.

`cats(`*numlist*`)` defines the two categories.

`fweight`s are allowed; see [U] **11.1.6 weight**. In particular, this likelihood can be used to evaluate the relative frequency of FP and TP decisions in analyzing one diagnostic test. The two arguments must be greater than 0, and their sum must be less than 1.

`llriskcomptwosamples` *#1 #2* [*if*] [*in*] [*weight*], `var(`*varlist*`)` `by(`*varname*`)`
{`diff`|`logrr`|`logor`|`rr`|`or`} [`noisily` *maximize_options*]

computes the profile log likelihood for two binary variables observed in two subgroups evaluated at two values *#1* and *#2*, referring to the difference in the proportions between the two variables within subgroup 1 and within subgroup 2, respectively. The difference can be expressed as a difference, a log relative-risk, a log odds-ratio, a relative risk, or an odds ratio. The subgroups must be labeled with the values 1 and 2. The following options are required:

`var(`*varlist*`)` defines the two binary variables. The marginal proportion of the second variable is compared with the marginal proportion in the first variable when building differences, risk ratios, or odds ratios.

`by(`*varname*`)` defines the dichotomous grouping variable.

`diff`, `logrr`, `logor`, `rr`, or `or` specifies how the difference in proportions is expressed. Exactly one of these five options must be specified.

`fweights` are allowed; see [U] **11.1.6 weight**. The `noisily` option allows inspection of the output of the `ml max` command, and the *maximize_options* are passed to this command. In particular, this specific likelihood can be used to evaluate changes in sensitivity and specificity in a comparative diagnostic accuracy study with a paired design. The parameter space depends on the choice of the option to express the difference.

NOTE: For relative risks and odds ratios, it typically makes little sense to consider weighted averages. However, confidence and comparison regions still provide a visual impression about the imprecision of the estimates.

All three commands also allow a `force` option that forces the computation of the log likelihood even if parameter values outside the parameter space are specified. This can be useful to plot regions for an estimate on the boundary. An example is given in section 3.5.

3 Examples

3.1 Example 1: Joint confidence region for two regression coefficients

Stata provides `auto.dta`, allowing study of the relation of mileage rating to the weight and the origin (foreign or domestic) of automobiles. The corresponding model is used in the Stata *Base Reference Manual* to illustrate linear regression. `confcomptwo` can supplement this standard analysis by depicting the joint uncertainty of the two regression coefficient estimates:

```
. sysuse auto
(1978 automobile data)
. regress mpg weight foreign
```

Source	SS	df	MS	Number of obs	=	74
				F(2, 71)	=	69.75
Model	1619.2877	2	809.643849	Prob > F	=	0.0000
Residual	824.171761	71	11.608053	R-squared	=	0.6627
				Adj R-squared	=	0.6532
Total	2443.45946	73	33.4720474	Root MSE	=	3.4071

mpg	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
weight	-.0065879	.0006371	-10.34	0.000	-.0078583	-.0053175
foreign	-1.650029	1.075994	-1.53	0.130	-3.7955	.4954422
_cons	41.6797	2.165547	19.25	0.000	37.36172	45.99768

```
. confcomptwo weight foreign, only(conf) rescale(1666)
>   legend(cols(1)) loptsconf(lpattern(solid))
>   ytitle({&beta}(weight)) xtitle({&beta}(foreign))
>   aspect(1) xsize(4) ysize(4.5)
```

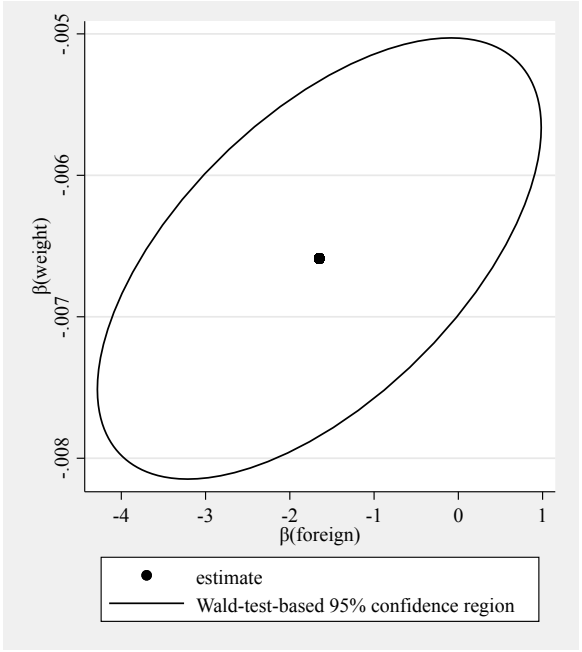

Figure 3. A 95% confidence region for the two regression coefficients considered in example 1.

The resulting figure 3 informs us that the two regression coefficients are positively correlated. This is a simple consequence of the negative correlation between the two covariates: on average, foreign automobiles have lower weights than domestic automobiles.

Note that the parameter rescale was used in this example. This was necessary because the two estimates have very different magnitudes. The value 1,666 reflects the ratio between the span on the x axis (5) and the span on the y axis (0.003). Here we also followed the tradition of drawing confidence regions with solid lines.

3.2 Example 2: Analysis of a single-arm diagnostic accuracy study

Xu et al. (2017) presented a study on the diagnosis of hemodynamically significant coronary stenosis defined by fractional flow reserve ≤ 0.80. One goal was to assess the diagnostic accuracy of angiography-based quantitative flow ratio measurements using the same cutpoint. Using fractional flow reserve as the reference standard, they observed a sensitivity of $106/112 = 94.6\%$ and a specificity of $198/216 = 91.7\%$ based on overall 328 interrogated vessels.

Computing the standard errors of sensitivity and specificity manually, we can use the immediate version of `confcomptwo` to obtain a two-dimensional Wald-test-based comparison region:

```
. confcomptwo 0.946 0.917,
>    se1(`=sqrt((0.946)*(1-0.946)/112)')
>    se2(`=sqrt((0.917)*(1-0.917)/216)')
>    only(comp) xtitle(specificity) ytitle(sensitivity)
>    aspect(1.0) xsize(4) ysize(4.5) xlabel(.85(.05)1) ylabel(.85(.05)1)
>    legend(cols(1))
```

To facilitate the interpretation of the comparison region, we can add a reference line referring to an average of sensitivity and specificity of 0.9 to mimic a specific posttest situation:

```
. confcomptwo 0.946 0.917,
>    se1(`=sqrt((0.946)*(1-0.946)/112)')
>    se2(`=sqrt((0.917)*(1-0.917)/216)')
>    only(comp) xtitle(specificity) ytitle(sensitivity)
>    aspect(1.0) xsize(4) ysize(4.5) xlabel(.85(.05)1) ylabel(.85(.05)1)
>    legend(cols(1))
>    addplot((scatteri .95 .85 .85 .95, connect(l) msymbol(i) lpattern(dash)
>      lcolor(gs7)))
>    legend(order(1 2))
```

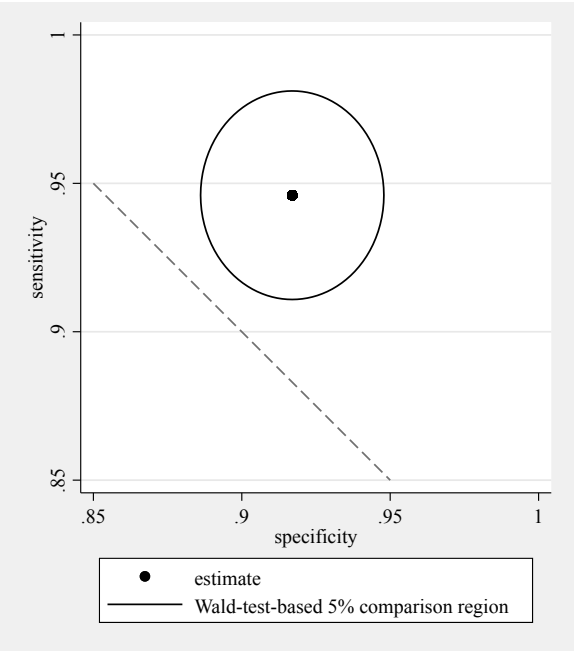

Figure 4. Wald-test-based 5% comparison region for sensitivity and specificity in example 2 with a reference line added

The resulting figure 4 allows the comparison of the reference line with the comparison region. The line is located below the comparison region. This supports the conclusion that the average between sensitivity and specificity is above 0.9.

In the computations carried out above, we have taken advantage of the fact that sensitivity and specificity are based on two separate samples, and we therefore know that they are uncorrelated when conditioning on the observed sample sizes. If manual computation of standard errors is to be avoided, we can use the postestimation version of `confcomptwo`. This requires obtaining the estimates of sensitivity and specificity together from one command and simultaneously an estimate of the variance–covariance matrix. One can approach this by generating a dataset with the study results, translating the index test results into a variable `correct` indicating correct test results, and modeling this variable in a generalized linear model with identity-link and Bernoulli-type variances as a function of the results of the reference test encoded by two indicator variables:

```
. clear
. input indextest reference freq
     indextest  reference       freq
  1. 1 1 106
  2. 0 1   6
  3. 1 0  18
  4. 0 0 198
  5. end
. generate correct = indextest == reference
. generate r1 = reference == 1
. generate r0 = reference == 0
. glm correct r1 r0 [fw=freq], nocons family(bernoulli) link(id)
Iteration 0:  Log likelihood = -85.353345
Iteration 1:  Log likelihood = -85.353345
Generalized linear models                         Number of obs   =        328
Optimization     : ML                             Residual df     =        326
                                                  Scale parameter =          1
Deviance         =  170.7066903                   (1/df) Deviance =   .5236402
Pearson          =          328                   (1/df) Pearson  =   1.006135
Variance function: V(u) = u*(1-u)                 [Bernoulli]
Link function    : g(u) = u                       [Identity]
                                                  AIC             =   .5326423
Log likelihood   = -85.35334513                   BIC             =  -1717.816
------------------------------------------------------------------------------
             |                 OIM
     correct |      Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
          r1 |      .9464286   .0212766    44.48   0.000     .9047273    .9881299
          r0 |      .9166667   .0188056    48.74   0.000     .8798083    .9535251
------------------------------------------------------------------------------
Coefficients are the risk differences.
. confcomptwo r1 r0, only(comp) xtitle(specificity) ytitle(sensitivity)
>   aspect(1.0) xsize(4) ysize(4.5) xlabel(.85(.05)1) ylabel(.85(.05)1)
>   addplot((scatteri .95 .85 .85 .95, connect(l) msymbol(i) lpattern(dash)
>     lcolor(gs7)))
>   legend(order(1 2) cols(1))
```

To obtain LR-test-based regions, we use the `llproptwosamples` command because sensitivity and specificity are proportions from two independent samples. To do so, we must store the values of sensitivity and specificity and recode the variable `reference` because `llproptwosamples` expects the two groups to be labeled with the values 1 and 2:

```
. local sens = _b[r1]
. local spec = _b[r0]
. recode reference (0=2)
(2 changes made to reference)
. confcomptwo `sens' `spec',
>   call(llproptwosamples #1 #2  [fw=freq], var(correct) by(reference))
>   xtitle(specificity) ytitle(sensitivity) only(comp)
>   aspect(1.0) xsize(4) ysize(4.5) xlabel(.85(.05)1) ylabel(.85(.05)1)
>   addplot((scatteri .95 .85 .85 .95, connect(l) msymbol(i) lpattern(dash)
>      lcolor(gs7))
>    (scatteri 1 .85 .85 .925, connect(l) msymbol(i) lpattern(shortdash)
>      lcolor(gs7))
>    (scatteri .925 .85 .85 1, connect(l) msymbol(i) lpattern(longdash)
>      lcolor(gs7)))
>   legend(order(1 2) col(1))
```

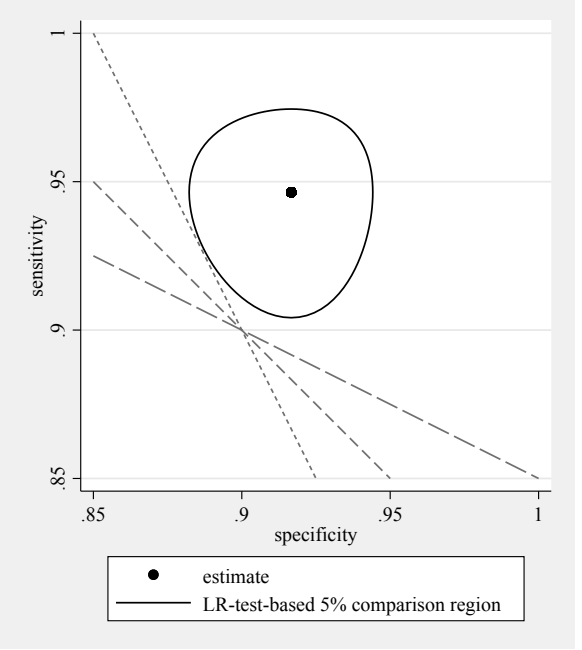

Figure 5. LR-test-based 5% comparison regions for sensitivity and specificity in example 2 with three reference lines added

The resulting figure 5 indicates that the shape of the LR-test-based region is distinctly different from the shape of the Wald-test-based region. This can be explained by the proximities of sensitivity and specificity to 1.0. Two further reference lines were added to the figure, giving the sensitivity twice the weight of specificity and vice versa. For all three choices of weights, one can conclude that the weighted average of sensitivity and specificity is above 0.9.

3.3 Example 3: Analysis of a paired diagnostic accuracy study

Ng et al. (2008) reported a study comparing the diagnostic accuracy of ^{18}F-fluoro-2-deoxyglucose positron emission tomography with extended-field multidetector computed tomography for the detection of distant malignancies in patients with oropharyngeal or hypopharyngeal squamous cell carcinoma. All patients were followed up with for at least 12 months or until death to construct a reference standard. For a suspected malignant lesion, a biopsy of the tissue was taken if possible or close clinical and imaging follow-up was performed. Distant malignancies were found in 26 out of 160 patients enrolled in the study. The two diagnostic tools investigated yielded a sensitivity/specificity of 50.0%/97.8% and of 76.9%/94.0%. The study was conducted in a paired design, so the results can be summarized in a $2 \times 2 \times 2$ contingency table with the frequencies for each combination of results for the two index tests to be compared and the reference standard. This contingency table is available in `studyng.dta`.

```
. use studyng, clear
. list
```

	test1	test2	refere~e	freq
1.	1	1	1	12
2.	1	1	0	2
3.	1	0	1	1
4.	1	0	0	1
5.	0	1	1	8
6.	0	1	0	6
7.	0	0	1	5
8.	0	0	0	125

In a first analysis, the interest is in studying the changes in sensitivity and specificity when replacing test 1 with test 2. Because there are no simple formulas to compute the standard errors of the change, we use the postestimation version of `confcomptwo` to compute Wald-test-based regions. This requires the derivation of the estimates and their covariance matrices with one estimation command. We hence compute estimates of the sensitivities and specificities of both tests—that is, simple relative frequencies—by fitting four intercept-only generalized linear models with identity-link and Bernoulli-type variances, saving the estimates, using Stata's `suest` to obtain the joint covariance matrix, and finally using the `nlcom` command to obtain the two differences of interest and their covariance matrices. In this setting, it might be of interest to also test the null hypotheses of no change in both sensitivity and specificity. Consequently, both comparison and confidence regions are drawn.

```
. generate correct1 = test1 == reference
. generate correct2 = test2 == reference
. quietly glm correct1 if reference == 1 [fw=freq], family(bernoulli) link(id)
. estimates store sens1
. quietly glm correct2 if reference == 1 [fw=freq], family(bernoulli) link(id)
```

```
. estimates store sens2
. quietly glm correct1 if reference == 0 [fw=freq], family(bernoulli) link(id)
. estimates store spec1
. quietly glm correct2 if reference == 0 [fw=freq], family(bernoulli) link(id)
. estimates store spec2
. suest sens1 sens2 spec1 spec2
Simultaneous results for sens1, sens2, spec1, spec2
                                                  Number of obs = 8
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
sens1_corre~1 _cons	.5	.0983659	5.08	0.000	.3072063	.6927937
sens2_corre~2 _cons	.7692308	.0828881	9.28	0.000	.6067731	.9316884
spec1_corre~1 _cons	.9776119	.0128204	76.25	0.000	.9524845	1.002739
spec2_corre~2 _cons	.9402985	.0205322	45.80	0.000	.9000562	.9805408

```
. nlcom (deltasens:[sens2_correct2]_cons-[sens1_correct1]_cons)
>       (deltaspec:[spec2_correct2]_cons-[spec1_correct1]_cons), post
   deltasens: [sens2_correct2]_cons-[sens1_correct1]_cons
   deltaspec: [spec2_correct2]_cons-[spec1_correct1]_cons
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
deltasens	.2692308	.102917	2.62	0.009	.0675171	.4709445
deltaspec	-.0373134	.0195407	-1.91	0.056	-.0756125	.0009856

```
. confcomptwo deltasens deltaspec,
>   loptsconf(points, n(100) yxratio(2) msize(*0.2) mcol(black))
>   xtitle({&Delta} specificity) ytitle({&Delta} sensitivity)
>   addplot(
>     (scatteri 0.15 -0.15 -0.1 0.1, connect(l) msymbol(i) lpattern(dot)
>       lcolor(gs7))
>     (scatteri 0.225 -0.15 -0.1 .066666, connect(l) msymbol(i)
>       lpattern(shortdash) lcolor(gs7))
>     (scatteri .3 -0.15 -.1 0.05, connect(l) msymbol(i) lpattern(longdash)
>       lcolor(gs7)))
>   legend(cols(1) order(1 2 3)) xline(0, lcolor(gs14)) yline(0, lcolor(gs14))
>   xlabel(-.15(.1).15)  ylabel(-.1(.1).5) aspect(2)
>   xsize(2.5) ysize(4)
```

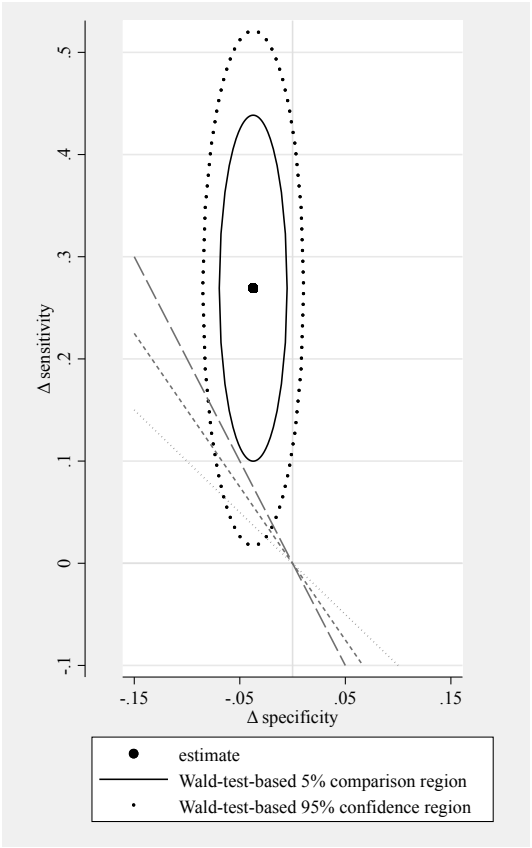

Figure 6. Wald-test-based 5% comparison and 95% confidence regions for the changes in sensitivity and specificity in example 3 with three reference lines added

The resulting figure 6 indicates a distinct improvement in sensitivity and a slight deterioration in specificity. Because of the low prevalence of the disease state of interest, the precision of the estimate of the change in sensitivity is rather low. We included three lines in the graph referring to the situation where some weighted averages of the changes in sensitivity and specificity are equal to zero. The dotted line refers to equal weights, and we can conclude that the increase in sensitivity is larger than the loss in specificity. However, because the sensitivity of the standard test is rather low and the specificity is already rather high, we are actually interested in a substantial improvement in sensitivity. The line with short dashes refers to giving the change in sensitivity a weight 1.5 times higher than the weight given to the change in specificity, and the line with long dashes refers to weighting the change in sensitivity two times higher than the change in specificity. In all three cases, the line does not hit the comparison region. Consequently, we can conclude that the gain in sensitivity is at least twice as great as the loss in specificity.

Note the use of the `aspect()` and `yxratio()` options. Because the precision of the change in sensitivity is much lower than the precision of the change in specificity, the span of the y axis is 0.6, whereas the span of the x axis is only 0.3. The option `aspect(2)` ensures that a change of 0.1 is nevertheless represented by the same distance in y and x direction in the visual inspection of the graph. The option `yxratio(2)` ensures that the points drawn to represent the confidence interval are equidistant.

To obtain LR-test-based regions, we store the two estimates, and then we use the `llriskcomptwosamples` command. The latter requires recoding the `reference` variable:

```
. local deltasens = _b[deltasens]
. local deltaspec = _b[deltaspec]
. generate group = cond(reference == 1,1,2)
. confcomptwo `deltasens' `deltaspec',
>   loptsconf(points, n(100) yxratio(2) msize(*0.2) mcol(black))
>   call(llriskcomptwosamples #1 #2 [fw=freq],
>     var(correct1 correct2) by(group) diff)
>   xtitle({&Delta} specificity) ytitle({&Delta} sensitivity)
>   addplot(
>     (scatteri 0.15 -0.15 -0.1 0.1, connect(l) msymbol(i) lpattern(dot)
>        lcolor(gs7))
>     (scatteri 0.225 -0.15 -0.1 .066666, connect(l) msymbol(i)
>        lpattern(shortdash) lcolor(gs7))
>     (scatteri .3 -0.15 -.1 0.05, connect(l) msymbol(i) lpattern(longdash)
>        lcolor(gs7)))
>   legend(col(1) order(1 2 3)) xline(0, lcolor(gs14)) yline(0, lcolor(gs14))
>   xlabel(-.15(.1).15) ylabel(-.1(.1).5) aspect(2) rescale(2.0)
>   xsize(2.5) ysize(4)
```

The resulting figure is not shown, because it is nearly identical to the one obtained using the Wald test.

Next we present a second analysis considering the change in the relative frequency of TP and FP decisions. Wald-test-based comparison and confidence regions can be approached similarly to above, starting with the empirical estimates of the four relative frequencies expressed as estimates from an intercept-only generalized linear model:

```
. use studyng, clear
. generate TP1 = test1 & reference == 1
. generate TP2 = test2 & reference == 1
. generate FP1 = test1 & reference == 0
. generate FP2 = test2 & reference == 0
. quietly glm TP1 [fw=freq], family(bernoulli) link(id)
. estimates store TP1
. quietly glm TP2 [fw=freq], family(bernoulli) link(id)
. estimates store TP2
. quietly glm FP1 [fw=freq], family(bernoulli) link(id)
. estimates store FP1
. quietly glm FP2  [fw=freq], family(bernoulli) link(id)
. estimates store FP2
```

```
. suest TP1 TP2 FP1 FP2
Simultaneous results for TP1, TP2, FP1, FP2                  Number of obs = 8
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
TP1_TP1						
_cons	.08125	.0216676	3.75	0.000	.0387822	.1237178
TP2_TP2						
_cons	.125	.0262277	4.77	0.000	.0735946	.1764054
FP1_FP1						
_cons	.01875	.010757	1.74	0.081	-.0023334	.0398334
FP2_FP2						
_cons	.05	.0172842	2.89	0.004	.0161237	.0838763

```
. nlcom (deltaTP:[TP2_TP2]_cons-[TP1_TP1]_cons)
>       (deltaFP:[FP2_FP2]_cons-[FP1_FP1]_cons), post
     deltaTP: [TP2_TP2]_cons-[TP1_TP1]_cons
     deltaFP: [FP2_FP2]_cons-[FP1_FP1]_cons
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
deltaTP	.04375	.0184861	2.37	0.018	.0075179	.0799821
deltaFP	.03125	.0164017	1.91	0.057	-.0008967	.0633967

```
. confcomptwo deltaTP deltaFP,
>   xtitle({&Delta} FP) ytitle({&Delta} TP)
>   loptsconf(points, n(150) mcol(black) msize(*0.15))
>   addplot(
>     (scatteri -.01 -.01 .08 .08, connect(line) msymbol(i) lpattern(dot)
>        lcolor(gs7))
>     (scatteri -.01 -.02 .04 .08, connect(line) msymbol(i) lpattern(shortdash)
>        lcolor(gs7))
>     (scatteri -.0067 -.02 .0266 .08, connect(line) msymbol(i)
>        lpattern(longdash) lcolor(gs7)))
>   legend(col(1) order(1 2 3))  xline(0, lcolor(gs14)) yline(0, lcolor(gs14))
>   xlabel(-.02(.02).08) ylabel(0(.02).1)
>   yscale(range(-0.005 0.1)) aspect(1.1)
>   xsize(4) ysize(5)
```

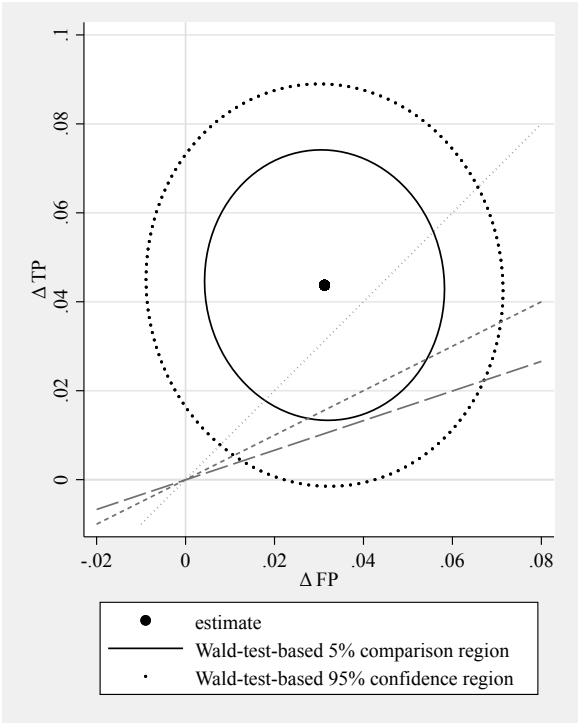

Figure 7. Wald-test-based 5% comparison and 95% confidence regions for the change in the relative frequency of TP and FP decisions in example 3 with three reference lines added

In the resulting figure 7, we observe an increase in the frequency of TP decisions by 4.4%, which is about a factor 1.5 higher than the increase in FP decisions. This looks less impressive than the changes in sensitivity and specificity, which are due to the low prevalence of the disease state of interest in this example. Different stakeholders may have different views on how many additional FP decisions they may accept for one additional TP decision. If they maximally accept only one FP, they are interested in demonstrating that the difference between the TP rate and the FP rate is above 0, that is, in points above the dotted line referring to the weights 1 and -1 for the TP rate and the FP rate, respectively. If they are willing to accept two FP decisions for a gain in one TP decision, they are interested in points above the line with short dashes referring to the weights 1 and -2 for the TP rate and the FP rate, respectively. If they are willing to accept three FP decisions for a gain in one TP decision, they are interested in points above the line with long dashes, which refers to the weights 1 and -3 for the TP rate and the FP rate, respectively. We can observe that they can conclude that the new test implies an advantage only in the last case.

To use the LR approach, we must write a specific program to evaluate the log likelihood. We will present this in section 5 after we have taken a closer look at the formulas used by `llriskcomptwosamples`.

3.4 Example 4: Joint evaluation of benefit and risk in a randomized controlled trial

Saxer et al. (2018) reported the results of a randomized controlled trial comparing two surgical techniques to be used to treat femoral neck fractures in patients over 60 years of age requiring hemiarthroplasty. The aim was to demonstrate that an anterior minimally invasive approach is beneficial to patients in terms of accelerated remobilization compared with a standard lateral Hardinge approach. The primary endpoint was early mobility evaluated at three weeks via the "timed up and go" (TUG) test. This test measures the time needed to get up from a chair, to walk a distance of three meters, and to sit down again. A 20% reduction was regarded as clinically relevant. Various secondary endpoints were considered. We focus here on the endpoint *occurrence of local infections*, reflecting a potential safety concern about the minimally invasive approach.

The study analyzed the primary endpoint by applying a regression model with the covariates treatment arm, age, and functional independence measure at baseline to the log-transformed TUG values. The back-transformed treatment effect was interpreted as an estimate of the median reduction in time required for the TUG. This resulted in an effect estimate corresponding to a 21.5% reduction. The frequency for local infections was 5 out of 96 in the lateral Hardinge arm and 7 out of 79 in the anterior minimally invasive arm, corresponding to an increase of 3.6 infections in 100 patients.

`confcomptwo` can be used to visualize the joint uncertainty in these two estimates in a benefit–risk plane. One can approach this by computing the two estimates using regression commands, combining these results using the `suest` command to determine the correlation, and finally using `confcomptwo` as a postestimation command:

```
. use datarct, clear
. regress logtug arm fim0 age
```

Source	SS	df	MS	Number of obs	=	144
				F(3, 140)	=	12.34
Model	28.384212	3	9.46140399	Prob > F	=	0.0000
Residual	107.312516	140	.76651797	R-squared	=	0.2092
				Adj R-squared	=	0.1922
Total	135.696728	143	.948928166	Root MSE	=	.87551

logtug	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
arm	-.2422539	.1469085	-1.65	0.101	-.5326998	.048192
fim0	-.0140444	.0028972	-4.85	0.000	-.0197723	-.0083166
age	.0276489	.0112462	2.46	0.015	.0054145	.0498833
_cons	3.164169	1.04663	3.02	0.003	1.094925	5.233412

```
. estimates store logtug
```

```
. glm localinf arm, link(id) family(bernoulli)
Iteration 0:  Log likelihood =  -43.28707
Iteration 1:  Log likelihood =  -43.28707

Generalized linear models                         Number of obs   =        175
Optimization     : ML                             Residual df     =        173
                                                  Scale parameter =          1
Deviance         =  86.57414092                   (1/df) Deviance =   .5004286
Pearson          =          175                   (1/df) Pearson  =   1.011561

Variance function: V(u) = u*(1-u/1)               [Binomial]
Link function    : g(u) = u                       [Identity]

                                                  AIC             =   .5175665
Log likelihood   = -43.28707046                   BIC             =  -806.9338
```

localinf	OIM Coefficient	std. err.	z	P>|z|	[95% conf.	interval]
arm	.0365243	.0391983	0.93	0.351	-.0403031	.1133516
_cons	.0520833	.0226777	2.30	0.022	.0076359	.0965308

Coefficients are the risk differences.

```
. estimates store localinf
. suest logtug localinf
Simultaneous results for logtug, localinf                  Number of obs = 175
```

	Robust Coefficient	std. err.	z	P>|z|	[95% conf.	interval]
logtug_mean						
arm	-.2422539	.1433648	-1.69	0.091	-.5232438	.038736
fim0	-.0140444	.0032706	-4.29	0.000	-.0204548	-.0076341
age	.0276489	.009611	2.88	0.004	.0088118	.046486
_cons	3.164169	.9455341	3.35	0.001	1.310956	5.017382
logtug_lnvar						
_cons	-.2658971	.1217107	-2.18	0.029	-.5044457	-.0273486
localinf_lo~f						
arm	.0365243	.0393108	0.93	0.353	-.0405235	.113572
_cons	.0520833	.0227428	2.29	0.022	.0075083	.0966583

```
. nlcom (tug:100*(1-exp(_b[logtug_mean:arm])))
> (localinf:100*_b[localinf_localinf:arm]), post
        tug: 100*(1-exp(_b[logtug_mean:arm]))
   localinf: 100*_b[localinf_localinf:arm]
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
tug	21.51431	11.25209	1.91	0.056	-.5393694	43.56799
localinf	3.652426	3.931081	0.93	0.353	-4.052351	11.3572

```
. confcomptwo tug localinf,
>   ytitle(median gain % TUG)
>   xtitle(increase in local infections per 100 patients)
>   loptsconf(points, n(150) mcol(black) msize(*0.15) yxratio(4))
>   aspect(1) xsize(4) ysize(5)
>   addplot((scatteri -2.5 -5 7.5 15, connect(l) msymbol(i) lpattern(dash)
>     lcolor(gs7)))
>   legend(order(1 2 3) col(1))
```

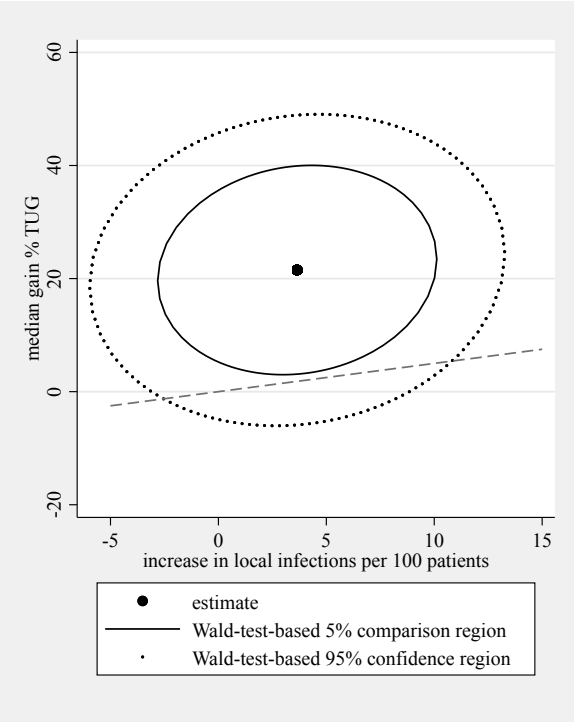

Figure 8. Wald-test-based 5% comparison and 95% confidence regions for the gain in TUG and the increase in number of local infections in example 4

The resulting figure 8 indicates that it is hard to draw firm conclusions because of the substantial imprecision of the estimates. Only if we would regard a very small gain in TUG time as justification for an increase by 1 local infection in 100 patients could we conclude that the true parameter values are above this line. To illustrate this, we show a reference line referring to a ratio between the gain in TUG and the number of additional local infections per 100 patients of 0.5. For 10 additional local infections, this would imply a 5% gain in TUG.

3.5 Example 5: LR-test-based regions for an estimate on the boundary

If the parameter space of a two-dimensional parameter estimation problem is bounded, it is possible for the point estimate to be on the boundary. LR-test-based comparison and confidence regions are then still defined, but drawing is slightly more complicated. However, if we restrict the drawing to directions into the inner part of the parameter space, it is still possible to draw regions. This is illustrated by an artificial example from a single-arm diagnostic study with a sensitivity estimate of 1.0, which results in figure 9:

```
. clear
. input indextest reference freq
     indextest  reference       freq
  1. 1 1 17
  2. 0 0 57
  3. 1 0  7
  4. end
. generate correct = indextest == reference
. recode reference (0=2)
(2 changes made to reference)
. confcomptwo 1.0  `=57/64',
>   call(llproptwosamples #1 #2 [fw=freq], var(correct) by(reference) force)
>   points(0.2501(0.009996)0.7499) only(comp)
>   ytitle(sensitivity) xtitle(specificity)
>   aspect(1.0) xsize(4) ysize(5) xlabel(.8(.05)1) ylabel(.8(.05)1)
>   legend(col(1))
```

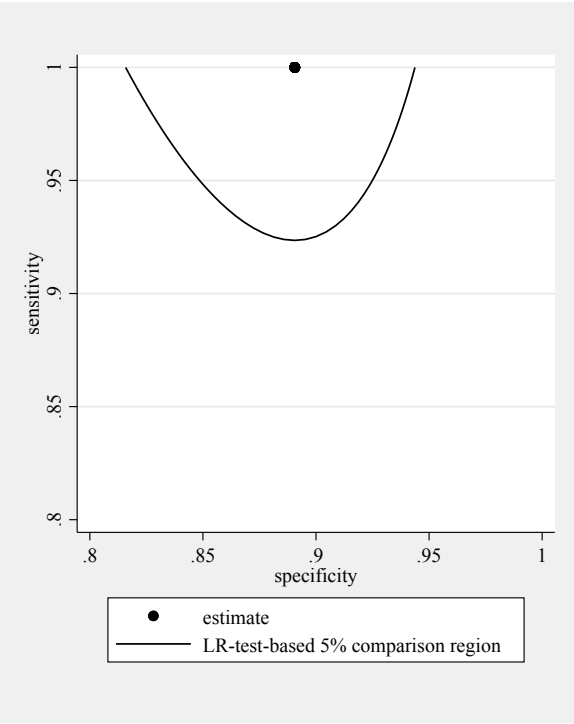

Figure 9. LR-test-based 5% comparison region for sensitivity and specificity in example 5

Note that it was necessary to use the `force` option to ensure that the log likelihood could be evaluated at the point estimate, and note that the `points()` option was used to explicitly specify the range of directions to be considered.

4 Methods and formulas

4.1 Parameterization of directions

The following two subsections describe how to determine the point on the boundary of a comparison or confidence region when looking from the point estimate in a certain direction $\boldsymbol{\xi} \in \mathbb{R}^2 \backslash (0,0)$. A first, straightforward choice for the directions may be $\boldsymbol{\xi}_t = (\cos t, \sin t)'$ with values t from the grid $t_j = \{(j-1)/(J-1)\}2\pi$ for $j = 1, \ldots, J$ with J denoting the overall number of directions considered. This implies that all angles between two neighboring directions are equal, and this is indeed also the default choice in `confcomptwo`. However, the equality of angles holds only with respect to the parameter space with the Euclidean metric. It does not hold with respect to the plane in which we plot the regions in the figure we want to create, because the two dimensions are typically rescaled as part of the drawing process. Moreover, equal angles are optimal only if the shape of the region is close to a circle, and it can be a poor choice in the

case of other shapes. Hence, `confcomptwo` also offers a more general approach: A list of values t_j can be provided (using the `points()` option with a *numlist*), and a function $s(t)$ can be provided in the `rescale()` option. The directions are determined as $\boldsymbol{\xi}_t = \{s(t)\cos t, \sin t\}'$. If $s(t)$ is chosen as a constant, this just implies a rescaling of the directions with respect to the parameter shown on the y axis.

If the units of the axes in the plot plane roughly correspond to the same physical length and the shape of the regions is close to a circle, the default value of $J = 101$ is typically large enough to ensure that the boundary lines of the comparison and confidence regions appear as smooth lines when plotting the calculated points in a graph and connecting them with straight lines.

4.2 Drawing comparison and confidence regions based on the Wald test principle

Given a direction $\boldsymbol{\xi} \in \mathbb{R}^2 \backslash (0,0)$, the point on the boundary of C_α in this direction (starting at $\widehat{\boldsymbol{\theta}}$) is given by

$$\boldsymbol{\theta}(\boldsymbol{\xi}) = \widehat{\boldsymbol{\theta}} + \widehat{\boldsymbol{\Sigma}}\boldsymbol{\xi}\sqrt{\frac{\chi^2_{1,1-2\alpha}}{\boldsymbol{\xi}'\widehat{\boldsymbol{\Sigma}}\boldsymbol{\xi}}}$$

(When $\boldsymbol{\xi}'\widehat{\boldsymbol{\Sigma}}\boldsymbol{\xi}$ equals 0, we set $\boldsymbol{\theta}(\boldsymbol{\xi}) = \widehat{\boldsymbol{\theta}}$.) C_α can easily be drawn by plotting and connecting the points $\boldsymbol{\theta}(\boldsymbol{\xi}_t)$ with t chosen as described in the previous subsection.

When we use the Wald test principle to construct $(1-\alpha)$ confidence regions for $\boldsymbol{\theta}$, the points are given by

$$\boldsymbol{\theta}(\boldsymbol{\xi}) = \widehat{\boldsymbol{\theta}} + \widehat{\boldsymbol{\Sigma}}\boldsymbol{\xi}\sqrt{\frac{\chi^2_{2,1-\alpha}}{\boldsymbol{\xi}'\widehat{\boldsymbol{\Sigma}}\boldsymbol{\xi}}}$$

4.3 Drawing comparison and confidence regions based on the LR test principle

Given a direction $\boldsymbol{\xi} \in \mathbb{R}^2 \backslash (0,0)$, the point on the boundary of C_α in this direction [starting at $\widehat{\boldsymbol{\theta}}_{\mathrm{ML}}$ with the log-likelihood value $l^* = l(\widehat{\boldsymbol{\theta}}_{\mathrm{ML}})$] can be determined by solving the equation

$$l\left(\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} + \gamma\boldsymbol{\xi}\right) = l^* - \frac{1}{2}\chi^2_{1,1-2\alpha}$$

for $\gamma > 0$. Because $\gamma \mapsto l(\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} + \gamma\boldsymbol{\xi})$ is a strictly monotone decreasing function with maximum 0, the solution $\gamma(\boldsymbol{\xi})$ is unique and can easily be determined by a simple line search. We can then plot all points $\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} + \gamma(\boldsymbol{\xi}_t)\boldsymbol{\xi}_t$ with t chosen as described above and connect these points.

The same approach has been recently recommended by Jaeger (2016) for plotting LR-test-based confidence regions. This requires just the replacement of $\chi^2_{1,1-2\alpha}$ with $\chi^2_{2,1-\alpha}$.

The line search is conducted as follows. Let $\widetilde{l}(\gamma) := l(\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} + \gamma\boldsymbol{\xi})$ and $\tau := l^* - (1/2)\chi^2_{1,1-2\alpha}$. When the point $\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} + \gamma\boldsymbol{\xi}_t$ is outside the parameter space Θ of interest (as indicated by the returned value of `r(inside)`), $\widetilde{l}(\gamma)$ is set to $-\infty$. For a starting value $\gamma_0 > 0$, we check whether $\widetilde{l}(\gamma_0)$ is less than, greater than, or equal to τ. In the first case, we decrease γ by the factor $1/f$ until $\widetilde{l}(\gamma)$ is greater or equal to τ. In the second case, we increase γ by the factor f until $\widetilde{l}(\gamma)$ is less than τ. The last two values in this sequence are then used as starting values for a bisectioning search, making use of the regula falsi. The bisectioning stops if the absolute value of the difference between $\widetilde{l}(\gamma)$ and τ is less than 10^{-4}, a value that can be changed by the `lstoler()` option.

For the first grid value t, γ_0 is specified by the `lsstartgamma()` option with a default value of 1.0, and f is specified by the `lsstartfactor()` option with a default value of 10.0. For subsequent grid values, γ_0 is chosen as the final value from the previous search, and f is specified by the `lsfactor()` option with a default value of 1.02 in the case of 100 or more points and is linearly increasing up to 2.0 in the case of only 2 points. These choices usually work properly if the number of points J is large enough. You can use the `lstrace()` option to obtain information about the γ values and corresponding points and log-likelihood values considered in each line search. It is well known that the regula falsi is a suboptimal procedure to determine the root of a concave function. However, using a factor of 1.02 in the line search implies that usually only one bisectioning step is required; hence, there is no need to consider more efficient algorithms. The use of the optimal γ value of the previous step as the starting point for the line search implies that usually only one or two line search steps are required before starting bisectioning.

When the call to the program provided by the user results in an error code different from 0 or the log likelihood evaluates to a missing value, no point is plotted in the corresponding direction, and a warning message is shown.

The validity of the comparison and confidence regions drawn depends heavily on the correct specification of the values of the maximum likelihood (ML) estimate. Hence, `confcomptwo` automatically performs two checks. First, every log-likelihood value computed is compared with the value obtained for the specified ML estimate, and the program stops with an error message if the difference is larger than 10^{-5}. Second, after constructing a point on the boundary, the log likelihood is also evaluated at the parameter value obtained by moving only 0.01% into the direction of the point.

4.4 The llproptwosamples command

The `llproptwosamples` command refers to the following likelihood for a single observation of the binary variable Y and the grouping variable B:

$$L(p_1, p_2) = \begin{cases} p_1^Y(1-p_1)^{1-Y} & \text{if } B = 1 \\ p_2^Y(1-p_2)^{1-Y} & \text{if } B = 2 \end{cases}$$

The command computes the sum of $\log L(p_1, p_2)$ over all observations.

4.5 The llproptwocats command

The `llproptwocats` command refers to the following likelihood for a single observation of the categorical variable Y with prespecified categories c_1 and c_2:

$$L(p_1,p_2)=\begin{cases} p_1 & \text{if } Y=c_1 \\ p_2 & \text{if } Y=c_2 \\ 1-p_1-p_2 & \text{otherwise} \end{cases}$$

The command computes the sum of $\log L(p_1,p_2)$ over all observations.

4.6 The llriskcomptwosamples command

The implementation of the `llriskcomptwosamples` command requires one to find a parameterization for the joint distribution of two binary variables Y_1 and Y_2, which allows one to fix the difference (risk ratio, odds ratio, log risk-ratio, log odds-ratio) between the two marginal probabilities while allowing the other parameters to vary freely without any further restrictions. We use a tetrachoric parameterization (Pearson 1900), which is simple to implement in Stata because of the availability of the `binormal()` function, allowing the computation of the joint cumulative distribution of a bivariate normal distribution with correlation ρ, marginal means 0, and marginal variances 1. This parameterization assumes that Y_1 and Y_2 are derived from two continuous random variables Z_1 and Z_2 following such a bivariate normal distribution by dichotomization at $q_k=\Phi^{-1}(p_k)$ for $k=1,2$, with p_k denoting the prevalence of Y_k. The likelihood of an observation (Y_1,Y_2) is then given by

$$\begin{aligned}
L^{Y_1=1,Y_2=1}(p_1,p_2,\rho) &= P(Z_1<q_1,Z_2<q_2)=\texttt{binormal}(q_1,q_2,\rho) \\
L^{Y_1=1,Y_2=0}(p_1,p_2,\rho) &= P(Z_1<q_1,Z_2>q_2)=P(Z_1<q_1)-P(Z_1<q_1,Z_2<q_2) \\
&= p_1-\texttt{binormal}(q_1,q_2,\rho) \\
L^{Y_1=0,Y_2=1}(p_1,p_2,\rho) &= P(Z_1>q_1,Z_2<q_2)=P(Z_2<q_2)-P(Z_1<q_1,Z_2<q_2) \\
&= p_2-\texttt{binormal}(q_1,q_2,\rho) \\
L^{Y_1=0,Y_2=0}(p_1,p_2,\rho) &= P(Z_1>q_1,Z_2>q_2) \\
&= 1-P(Z_1<q_1)-P(Z_2<q_2)+\texttt{binormal}(q_1,q_2,\rho) \\
&= 1-p_1-p_2+\texttt{binormal}(q_1,q_2,\rho)
\end{aligned}$$

With B denoting the grouping variable, the profile log likelihood is then obtained by maximizing for $b=1,2$

$$\sum_{i,B_i=b}\log L^{y_{1i},y_{2i}}(\rho^b,p_1^b,p_2^b)$$

under the constraints on p_1^b and p_2^b specified when calling the command and then summing up the two values.

The maximization task is approached using Stata's `ml` command. The correlation ρ is expressed as $\tanh\rho'$ with ρ' varying between $-\infty$ and $+\infty$ and p_1 and p_2 are expressed

as functions of the given value ζ of the constraint in the group considered (and the type of the constraint) and a freely varying parameter α as follows:

Type of constraint	Allowed values for ζ		$p_1(\alpha, \zeta)$	$p_2(\alpha, \zeta)$
`diff`	$(-1, 1)$	if $\zeta \geq 0$	$\text{logit}(\alpha)(1-\zeta)$	$\zeta + \text{logit}(\alpha)(1-\zeta)$
$(\zeta = p_2 - p_1)$		if $\zeta < 0$	$-\zeta + \text{logit}(\alpha)(1+\zeta)$	$\text{logit}(\alpha)(1+\zeta)$
`rr`	$(0, \infty)$.	if $\zeta \geq 1$	$\text{logit}(\alpha)/\zeta$	$\text{logit}(\alpha)$
$(\zeta = p_2/p_1)$		if $\zeta < 1$	$\text{logit}(\alpha)$	$\text{logit}(\alpha)\zeta$
`or`	$(0, \infty)$		$\text{logit}(\alpha)$	$\text{logit}(\alpha + \log\zeta)$
$\left(\zeta = \frac{p_2}{1-p_2} / \frac{p_1}{1-p_1}\right)$				
`logrr`	$\mathbb{R}$	if $\zeta \geq 0$	$\text{logit}(\alpha)/\exp(\zeta)$	$\text{logit}(\alpha)$
$\{\zeta = \log(p_2/p_1)\}$		if $\zeta < 0$	$\text{logit}(\alpha)$	$\text{logit}(\alpha)\exp(\zeta)$
`logor`	$\mathbb{R}$		$\text{logit}(\alpha)$	$\text{logit}(\alpha + \zeta)$
$\left\{\zeta = \log\left(\frac{p_2}{1-p_2} / \frac{p_1}{1-p_1}\right)\right\}$				

Consequently, the profile log likelihood is obtained by maximizing

$$\sum_{b=1,2} \sum_{i, B_i = b} \log \widetilde{L}^{y_{1i}, y_{2i}}(\rho'_b, \alpha_b, \zeta_b)$$

with

$$\widetilde{L}^{y_1, y_2}(\rho', \alpha, \zeta) = L^{y_1, y_2}\{\tanh \rho', p_1(\alpha, \zeta), p_2(\alpha, \zeta)\}$$

over ρ'_1, ρ'_2, α_1, and α_2.

Note that the profile likelihood can be interpreted as a conditional likelihood given the frequencies of $B = 1$ and $B = 2$ as well as an unconditional likelihood because, in the latter case, the additional contribution depending on the prevalence π does not affect the maximization task.

5 An example involving a user-defined program to compute the profile log likelihood

To continue with example 3 in section 3.3, we need an appropriate parameterization of the joint distribution of Y_1, Y_2, and D. We use the fact that the change in relative frequency of TP decisions Δ_{TP} can be related to the change in sensitivity Δ_{sens} and the prevalence π, and the change in relative frequency of FP decisions Δ_{FP} can be related to the change in specificity Δ_{spec} and the prevalence. To be precise, we have $\Delta_{\mathrm{sens}} = \Delta_{\mathrm{TP}}/\pi$ and $\Delta_{\mathrm{spec}} = -\Delta_{\mathrm{FP}}/(1-\pi)$. We now use the framework developed for the `llriskcomptwosamples` command in section 4.6 in the case of using the `diff` option. With the notation introduced there, we can express the log likelihood for a trivariate observation (Y_1, Y_2, D) as

$$\begin{array}{ll} \log\widetilde{L}^{Y_1,Y_2}\{\rho_0', \alpha_0, -\Delta_{\mathrm{FP}}/(1-\pi)\} + \log(1-\pi) & \text{if } D = 0 \\ \log\widetilde{L}^{Y_1,Y_2}(\rho_1', \alpha_1, \Delta_{\mathrm{TP}}/\pi) + \log\pi & \text{if } D = 1 \end{array}$$

The profile log likelihood for given values of Δ_{TP} and Δ_{FP} is then obtained by maximization over ρ_0', ρ_1', α_0, α_1, and logit π.

To apply the LR approach, we must provide a program to compute the profile log-likelihood function. We start with defining the program `tetrall` to define the log likelihood corresponding to a tetrachoric parameterization of a bivariate binary distribution. We then define the program `modeldeltaFPTP` to compute the log likelihood suitably for Stata's `ml` command, and we finally define the program `lldeltaFPTP` to compute the profile log likelihood by calling `ml`.

```
. local deltaTP = _b[deltaTP]
. local deltaFP = _b[deltaFP]
. program define tetrall
  1.   syntax varlist(min=2 max=2), cond(string) rho(real) genll(name) p1(real)
> p2(real)
  2.   gettoken y1 y2 : varlist
  3.   tempname q1 q2
  4.   scalar `q1' = invnormal(`p1')
  5.   scalar `q2' = invnormal(`p2')
  6.   quietly generate double `genll'= log(
>     cond(`y1'==1,
>       cond(`y2'==1, binorm(`q1',`q2',`rho'), `p1'- binorm(`q1',`q2',`rho')),
>       cond(`y2'==1, `p2'- binorm(`q1',`q2',`rho'),
>                     1 - `p1' - `p2' + binorm(`q1',`q2',`rho'))
>     ))
  7. end
```

```
. program define modeldeltaFPTP
  1.   args lnf alpha0 atanhrho0 alpha1 atanhrho1 logitpi
  2.   local testvars : char _dta[testvars]
  3.   local refvar : char _dta[refvar]
  4.   local deltaTP: char _dta[deltaTP]
  5.   local deltaFP: char _dta[deltaFP]
  6.   local pi = invlogit(`logitpi')
  7.   local rho0 = tanh(`atanhrho0')
  8.   local rho1 = tanh(`atanhrho1')
  9.   local deltasens = `deltaTP'/`pi'
 10.   local deltaspec = -`deltaFP'/(1-`pi')
 11.   if `deltasens'>=0 {
 12.     local p1alpze1 = invlogit(`alpha1')*(1-`deltasens')
 13.         local p2alpze1 = `deltasens' + invlogit(`alpha1')*(1-`deltasens')
 14.   }
 15.   else {
 16.     local p1alpze1 = -`deltasens' + invlogit(`alpha1')*(1+`deltasens')
 17.     local p2alpze1 = invlogit(`alpha1')*(1+`deltasens')
 18.   }
 19.   if `deltaspec'>=0 {
 20.     local p1alpze0 = invlogit(`alpha0')*(1-`deltaspec')
 21.     local p2alpze0 = `deltaspec' + invlogit(`alpha0')*(1-`deltaspec')
 22.   }
 23.   else {
 24.         local p1alpze0 = -`deltaspec' + invlogit(`alpha0')*(1+`deltaspec')
 25.     local p2alpze0 = invlogit(`alpha0')*(1+`deltaspec')
 26.   }
 27.   tempvar aux1 aux0
 28.   tetrall `testvars',
>     rho(`rho1') p1(`p1alpze1') p2(`p2alpze1') genll(`aux1') cond(`refvar'==1)
 29.   tetrall `testvars',
>     rho(`rho0') p1(`p1alpze0') p2(`p2alpze0') genll(`aux0') cond(`refvar'==0)
 30.   quietly replace `lnf' = cond(`refvar'==1,`aux1',`aux0') +
>                       log(cond(`refvar'==1,`pi',1-`pi'))
 31. end
. program define lldeltaFPTP, rclass
  1.   syntax anything [if] [in] [fw], testvars(varlist) refvar(varname) *
  2.   preserve
  3.   marksample touse
  4.   quietly keep if `touse'
  5.   numlist "`anything'", min(2) max(2)
  6.   gettoken deltaFP deltaTP : anything
  7.   local inside = -1<min(`deltaFP',`deltaTP') & max(`deltaFP',`deltaTP') <1
  8.   return local inside `inside'
  9.   if `inside' {
 10.     char _dta[testvars] `testvars'
 11.     char _dta[refvar] `refvar'
 12.     char _dta[deltaFP] `deltaFP'
 13.     char _dta[deltaTP] `deltaTP'
 14.     ml model lf modeldeltaFPTP
>          (alpha0:) (atanhro0:) (alpha1:) (atanrho1:) (logitpi:) [`weight'`exp']
 15.     ml max, `options'
 16.     return scalar ll=e(ll)
 17.   }
 18.   else return scalar ll=.
 19.   restore
 20. end
. generate correct1 = test1 == reference
```

```
. generate correct2 = test2 == reference
. confcomptwo `deltaFP' `deltaTP',
>   call(lldeltaFPTP #1 #2 [fw=freq], testvars(correct1 correct2)
>    refvar(reference))
>   xtitle({&Delta} FP) ytitle({&Delta} TP) reverse
>   addplot(
>     (scatteri -0.01 -0.01 .08 .08, connect(l) msymbol(i) lpattern(dot)
>        lcolor(gs7))
>     (scatteri -.01 -.02 .04 .08, connect(line) msymbol(i) lpattern(shortdash)
>        lcolor(gs7))
>     (scatteri -.0067 -.02 .0266 .08, connect(line) msymbol(i)
>        lpattern(longdash) lcolor(gs7)))
>   legend(col(1) order(1 2 3))  xline(0, lcolor(gs14)) yline(0, lcolor(gs14))
>   xlabel(-.02(.02).08) ylabel(0(.02).1)
>   yscale(range(-0.005 0.1)) aspect(1.1)
>   xsize(4) ysize(5)
```

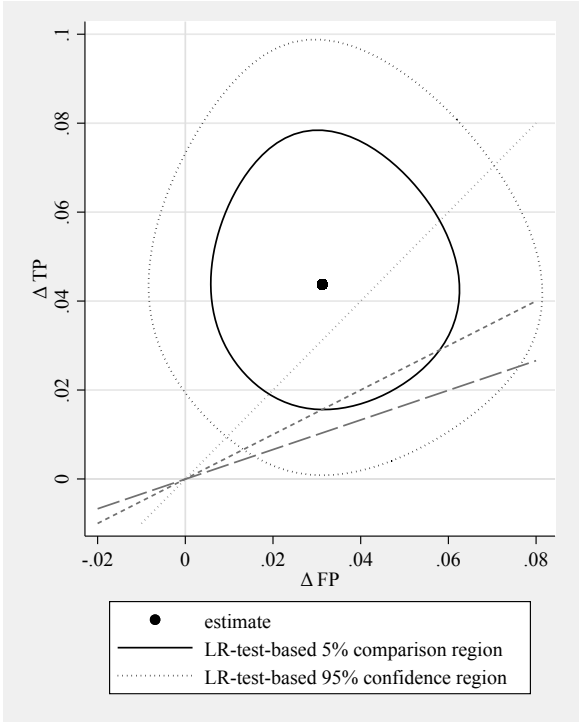

Figure 10. LR-test-based 5% comparison and 95% confidence regions for the change in the relative frequency of FP and TP decisions in example 3

In the resulting figure 10, we can observe that the shape of the regions based on the LR test principle deviates distinctly from the elliptic shape of the Wald-test-based regions. However, the conclusions remain the same.

6 Conclusions

In some areas of statistical applications (such as diagnostic accuracy studies), it is essential to present a joint evaluation of two parameter estimates. Two-dimensional confidence and comparison regions are useful tools supporting this goal. Drawing such regions is quite challenging and not easily accomplished within the existing graphical tools in Stata. Therefore, the `confcomptwo` command presented in this article can be seen as a useful and necessary addition that may stimulate an improved presentation of estimation results in specific settings.

The examples presented in this article indicate that Wald-test-based and LR-test-based comparison regions may substantially differ in their shape. First investigations of finite-sample properties (Eckert and Vach 2020) suggest that LR-test-based comparison regions are closer to their nominal level than Wald-test-based comparison regions. Therefore, we recommend using LR-test-based comparison regions. However, LR-test-based comparison regions also rely on inverting tests that are only asymptotically valid. In the long run, the use of exact methods might be desirable.

The optimal choice of directions for drawing the boundary points of the regions is a nontrivial problem. `confcomptwo` gives the user a flexible framework for the choice of directions, but in principle this choice can be optimized automatically. This holds at least in the case of Wald-test-based regions, for which the shape is determined by the variance–covariance matrix.

It can be seen as a disadvantage of `confcomptwo` that the computation of the ML estimates is decoupled from the drawing of the LR-test-based confidence regions. However, it is not trivial to derive profile likelihood functions from ordinary likelihood functions when nonstandard transformations of the parameters (such as those considered in our examples) are involved.

7 Acknowledgment

This work was supported by the German Research Foundation (DFG) [VA 88/5-1].

8 Programs and supplemental materials

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

```
. net sj 23-2
. net install st0716     (to install program files, if available)
. net get st0716         (to install ancillary files, if available)
```

9 References

Eckert, M., and W. Vach. 2020. On the use of comparison regions in visualizing stochastic uncertainty in some two-parameter estimation problems. *Biometrical Journal* 62: 598–609. https://doi.org/10.1002/bimj.201800232.

Gladstone, B. P., and W. Vach. 2015. Analyzing noninferiority trials: It is time for advantage deficit assessment—An observational study of published noninferiority trials. *Open Access Journal of Clinical Trials* 7: 11–21. https://doi.org/10.2147/OAJCT.S74821.

Guo, J. J., S. Pandey, J. Doyle, B. Bian, Y. Lis, and D. W. Raisch. 2010. A review of quantitative risk-benefit methodologies for assessing drug safety and efficacy-report of the ISPOR risk-benefit management working group. *Value in Health* 13: 657–666. https://doi.org/10.1111/j.1524-4733.2010.00725.x.

Jaeger, A. 2016. Computation of two- and three-dimensional confidence regions with the likelihood ratio. *American Statistician* 70: 395–398. https://doi.org/10.1080/00031305.2016.1182946.

Mt-Isa, S., C. E. Hallgreen, N. Wang, T. Callréus, G. Genov, I. Hirsch, et al. 2014. Balancing benefit and risk of medicines: A systematic review and classification of available methodologies. *Pharmacoepidemiology and Drug Safety* 23: 667–678. https://doi.org/10.1002/pds.3636.

Newcombe, R. G. 2001. Simultaneous comparison of sensitivity and specificity of two tests in the paired design: A straightforward graphical approach. *Statistics in Medicine* 20: 907–915. https://doi.org/10.1002/sim.906.

Ng, S.-H., S.-C. Chan, C.-T. Liao, J. T.-C. Chang, S.-F. Ko, H.-M. Wang, S.-C. Chin, C.-Y. Lin, S.-F. Huang, and T.-C. Yen. 2008. Distant metastases and synchronous second primary tumors in patients with newly diagnosed oropharyngeal and hypopharyngeal carcinomas: Evaluation of ^{18}F-FDG PET and extended-field multi-detector row CT. *Neuroradiology* 50: 969–979. https://doi.org/10.1007/s00234-008-0426-2.

Pearson, K. 1900. Mathematical contributions to the theory of evolution. VII: on the correlation of characters not quantitatively measurable. *Philosophical Transactions of the Royal Society of London, Series A* 195: 1–47. https://doi.org/10.1098/rsta.1900.0022.

Royston, P. 2007. Profile likelihood for estimation and confidence intervals. *Stata Journal* 7: 376–387. https://doi.org/10.1177/1536867X0700700305.

Saxer, F., P. Studer, M. Jakob, N. Suhm, R. Rosenthal, S. Dell-Kuster, W. Vach, and N. Bless. 2018. Minimally invasive anterior muscle-sparing versus a transgluteal approach for hemiarthroplasty in femoral neck fractures—A prospective randomised controlled trial including 190 elderly patients. *BMC Geriatrics* 18: 222. https://doi.org/10.1186/s12877-018-0898-9.

Vach, W., O. Gerke, and P. F. Hœilund-Carlsen. 2012. Three principles to define the success of a diagnostic study could be identified. *Journal of Clinical Epidemiology* 65: 293–300. https://doi.org/10.1016/j.jclinepi.2011.07.004.

Vickers, A. J. 2008. Decision analysis for the evaluation of diagnostic tests, prediction models, and molecular markers. *American Statistician* 62: 314–320. https://doi.org/10.1198/000313008x370302.

Xu, B., S. Tu, S. Qiao, X. Qu, Y. Chen, J. Yang, L. Guo, et al. 2017. Diagnostic accuracy of angiography-based quantitative flow ratio measurements for online assessment of coronary stenosis. *Journal of the American College of Cardiology* 70: 3077–3087. https://doi.org/10.1016/j.jacc.2017.10.035.

About the authors

Maren Eckert is a research associate at the Institute of Medical Biometry and Statistics. The work presented here is part of her PhD thesis.

Werner Vach is a senior researcher in applied methodology at the Basel Academy for Quality and Research in Medicine and has supervised the thesis.