Empirical Likelihood Test of AUC and pAUC

Yumin Zhao*, Xue Ding†, Mai Zhou‡

Abstract

Using a real data example (data set `aSAH` from the package `pROC`), we illustrate here (1) a novel empirical likelihood approach to test hypothesis and construct confidence intervals for AUC; (2) We also show how to test and construct confidence intervals, by empirical likelihood, for the partial AUC (pAUC) using a nuisance parameter/profile trick.

The empirical likelihood ratio test under our setup yields an asymptotic chi square distribution under null hypothesis.

Keywords and Phrases: Chi square distribution; Two sample empirical likelihood ratio; Partial AUC; ROC curve; Wilks confidence intervals, Nuisance parameter.

1 Definition and Estimations

Let X and Y, with respective distribution functions F and G, be the results of a continuous-scale test for a healthy and a disease subject, respectively.

The theoretical AUC of the test results X and Y of the above healthy and disease subjects can be represented by (Hanley and McNeil 1982) [4]:

$$AUC = \int_{-\infty}^{\infty} (1 - G(s)) dF(s) = Pr(Y > X) \ . \tag{1}$$

Given a random sample $X_1, \cdots, X_m$ of test results from healthy population and independently another random sample $Y_1, \cdots, Y_n$ of test results from the disease population, a non-parametric estimate of AUC is

$$\widehat{AUC} = \frac{1}{nm} \sum_{i=1}^{m} \sum_{j=1}^{n} I[Y_j > X_i] + 0.5 I[Y_j = X_i] \ . \tag{2}$$

See for example: [3] or from many books. Good test has a larger AUC value.

To evaluate two diagnostic tests on a portion of ROC curves, the partial AUC is proposed (McClish 1989) [7]). The theoretical value of a partial AUC, (pAUC$(0,p)$), where $0 < p < 1$, can be written as

$$pAUC(0,p) = \int_{\tau}^{\infty} (1 - G(s)) dF(s) = Pr(Y > X > \tau) \ , \tag{3}$$

where $\tau = F^{-1}(1-p)$.

Dodd and Pepe (2003) [3] proposed a non-parametric estimator for the pAUC$(0,p)$:

$$\widehat{pAUC}(0,p) = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \{ I[Y_j > X_i] + 0.5 I[Y_j = X_i] \} \, I[X_i > \tau] \tag{4}$$

*Eli Lilly and Company, Indianapolis, IN 46225, U.S.A. Email: zyyztwo@gmail.com

†Dr. Bing Zhang Department of Statistics, University of Kentucky, Lexington, KY 40536, U.S.A.

‡Dr. Bing Zhang Department of Statistics, University of Kentucky, Lexington, KY 40536, U.S.A. Email: maizhou@gmail.com

where τ is the $(1-p)th$ quantile of X, or equivalently, $\tau = F^{-1}(1-p)$. If the quantile τ in (4) is not known, as is usually the case, Dodd and Pepe (2003) [3] suggested that an empirical quantile estimate be substituted: $\hat{\tau} = \hat{F}^{-1}(1-p) = \inf\{s : \hat{F}(s) \geq 1-p\}$; where $\hat{F}$ is an empirical distribution based on $X_1, \cdots, X_m$. Dodd and Pepe (2003) [3] also used some linear interpolation technique to improve the empirical quantile estimate. This is equivalent to smoothing, which we shall discuss next.

2 Smoothing

In the previous section, the estimators of AUC, (2), and estimator of pAUC, (4), were defined using the indicator function, and we treated the $[Y_j > X_i]$ and $[Y_j = X_i]$ cases separately. If we replace the indicator function by a smoothed version, we can handle these $[Y_j > X_i]$ and $[Y_j = X_i]$ cases with one function.

In addition, the estimator of pAUC, (4), involves the estimated quantile $\tau = F^{-1}(1-p)$. As mentioned earlier, Dodd and Pepe (2003) [3] have used some smoothing when estimate the quantile before estimating the pAUC. Indeed, whenever a sample quantile is involved, smoothing is a must.

We hereby specify a typical smoothed indicator function. Since this function is going to replace the indicator function $I[y > x]$, we shall call it $I_\epsilon(y, x)$ where the bandwidth parameter $\epsilon > 0$ controls the degree of smoothing. When $\epsilon \to 0$, the function $I_\epsilon(y, x)$ becomes the original indicator function (except when $x = y$).

$$I_\epsilon(y, x) = \begin{cases} 1\,, & \text{if } (x-y) < -\epsilon\ ; \\ 0.5 - \frac{3(x-y)}{4\epsilon} + \frac{(x-y)^3}{4\epsilon^3}\,, & \text{if } -\epsilon \leq (x-y) \leq \epsilon\ ; \\ 0\,; & \text{if } (x-y) > \epsilon\ . \end{cases} \tag{5}$$

This function is implemented in the package as `smooth3` or `smooth3vec`. Other smoothing functions are also possible and should lead to similar results. We have a function `smooth5vec` in the R package which is based on 5th order polynomial. Notice when $y = x$ we have $I_\epsilon(y, x) = 0.5$; and when $|x - y| > \epsilon$, we have $I_\epsilon(y, x) = I[y > x]$.

Also, using empirical likelihood for testing hypothesis involving quantiles was investigated by Chen and Hall (1993) [1]. One take away message from Chen and Hall paper is that the sample quantile function needs to be smoothed. The smoothing makes the empirical likelihood ratio converge faster to the limiting chi square distribution (thus the empirical likelihood ratio test is more accurate).

The defining equation for quantile τ can be written as $F(\tau) = 1 - p$ which can also be written as

$$\mathbf{E} I[X \leq \tau] = 1 - p\,. \tag{6}$$

We recall $F(\cdot)$ is the unknown distribution function for X_i. We shall also smooth the indicator function in (6) by $I_\xi(\cdot, \cdot)$ and replace (6) by

$$\mathbf{E} I_\xi(\tau, X) = 1 - p\,. \tag{7}$$

Here we chose the bandwidth ξ following Chen and Hall's recommendation. The bandwidth ϵ in (8) and (9) may follow other guidelines.

To summarize: we shall use the smoothed estimator of AUC

$$\widehat{AUC}_\epsilon = \frac{1}{nm} \sum_{i=1}^{m} \sum_{j=1}^{n} I_\epsilon(Y_j, X_i)\,. \tag{8}$$

A smoothed estimator of pAUC can be similarly defined to (4):

$$p\widehat{AUC}_{\epsilon,\xi}(0, p) = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} I_\epsilon(Y_j, X_i)\, I_\xi(X_i, \hat{\tau}) \tag{9}$$

where the quantile τ is now defined by (7), the sample version is defined by

$$\hat{\tau} \quad \text{is the solution to} \quad \frac{1}{m}\sum_{i=1}^{m} I_{\xi}(X_i, \tau) = p\,. \tag{10}$$

3 Empirical Likelihood

Owen (1988) [9] was first to coin the term "empirical likelihood" and made many contributions to the theory and practice of empirical likelihood method. The first book on this fascinating topic is also by Owen in 2001 [10]. One of the empirical likelihood theorems contained in this book is the *two-sample empirical likelihood theorem* (section 11.4). We formulate below a version specific for the inference of AUC (h function in (i)) and the joint inference of pAUC and τ ($\boldsymbol{h}$ function in (ii)) below.

Theorem 1 (Two Sample Empirical Likelihood Theorem) Suppose $X_1, \cdots, X_m$ are iid random variables with distribution $F(t)$. We further suppose, independent of the X's, that $Y_1, \cdots, Y_n$ are iid random variables with distribution $G(t)$. Let the true parameter $\boldsymbol{\theta} \in R^r$ be defined by the equation $\mathbf{E}\boldsymbol{h}(X, Y, \boldsymbol{\theta}) = \mathbf{0}$, where $\boldsymbol{h}$ is a function with values in R^r specified either in (i) or (ii) below.

(i) *For the inference of AUC*: here we have $r = 1$ and $\theta = AUC$; and $h(X, Y, \theta) = \{I[Y > X] + 0.5I[Y = X]\} - \theta$. We further assume $Pr(Y \geq X) \neq 1$.

(ii) *For the joint inference of pAUC(0, p) and the* $(1-p)$*th quantile,* τ*, of* X: here we have $r = 2$ and $\boldsymbol{\theta} = (\theta_1, \theta_2) = (pAUC(0, p), \tau)$; and $\boldsymbol{h} = (h_1, h_2)$ where

$$\begin{aligned} h_1(X, Y, \boldsymbol{\theta}) &= (I[Y > X] + 0.5I[Y = X])I[X > \theta_2] - \theta_1 \\ h_2(X, Y, \boldsymbol{\theta}) &= h_2(X, \theta_2) = I[X \leq \theta_2] - (1 - p)\ . \end{aligned} \tag{11}$$

We further assume $0 < p < 1$; $F'(\tau) > 0$ and $Pr(Y \geq X | X > \tau) \neq 1$.

Define the two-sample empirical likelihood ratio

$$\begin{aligned} R(\boldsymbol{\theta}) = \sup_{u_i, v_j} \Big\{ \prod_{i=1}^{m} m u_i \prod_{j=1}^{n} n v_j\ ;\ \text{s.t. } u_i > 0;\ v_j > 0;\ \sum_{i=1}^{m} u_i = 1;\ \sum_{j=1}^{n} v_j = 1; \\ \sum_{i=1}^{m}\sum_{j=1}^{n} \boldsymbol{h}(X_i, Y_j, \boldsymbol{\theta}) u_i v_j = \mathbf{0} \Big\}\ . \end{aligned} \tag{12}$$

As sample size $\min(m, n)$ go to infinite, if $\boldsymbol{\theta}$ is the true parameter value, we have

$$-2\log R(\boldsymbol{\theta}) \xrightarrow{\mathcal{D}} \chi^2_{(r)}$$

where $\chi^2_{(r)}$ denotes a chi squared distribution with r degrees of freedom.

Proof: The empirical likelihood book of Owen (2001) section 11.4 contains a proof of this theorem for the case $r = 1$. The conditions Owen imposed on the h functions are easy to check with our h in (i) or (ii) above. For the case $r = 2$ the proof is similar. ■

Remark: We have stated the conditions (i) and (ii) in Theorem 1 without smoothing due to its clear connections to AUC/pAUC. If we apply the smoothing as detailed in previous section, the h function in (i) or (ii) of Theorem 1 needs to be modified as follows: all the indicator functions there shall be replaced by either $I_{\epsilon}(\cdot, \cdot)$ or $I_{\xi}(\cdot, \cdot)$. Theorem 1 is still valid after this smoothing modification: the chi square limit still hold for the -2 log empirical likelihood ratio. However, the convergence to chi square will be faster with smoothing. When profiling the empirical likelihood (next theorem), smoothing is not only a good idea but a must.

The empirical likelihood theorem above with an h specified in (i) immediately gives us a test of AUC when sample sizes are reasonably large: for testing $H_0 : AUC = \theta^*$ vs. $H_A : AUC \neq \theta^*$ the p-value can be computed as $Pr(\chi^2_{(1)} > -2\log R(\theta^*))$, where $\chi^2_{(1)}$ denotes a chi-square random variable with degree of freedom 1. The 95% confidence interval for AUC is

$$\{\theta^* | \text{ s.t. } -2\log R(\theta^*) < 3.84146 = \chi^2_{(1)}(0.95)\} \ .$$

For the test of the pAUC$(0, p)$, however, some more work is needed. The above empirical likelihood theorem, with $\boldsymbol{h}$ specified in (ii), only gives us a test of pAUC and τ jointly. We, however, are most likely interested only in testing pAUC alone. This calls for a profiling of the empirical likelihood ratio.

Profile empirical likelihood ratio is studied by Qin and Lawless (1994) [12]. They demonstrated that under reasonable smoothness conditions (so that certain derivatives exist) the profiling of empirical likelihood just behave the same as in the (well known) parametric likelihood case. Owen (2001) [10] Chapter 3 also discussed this topic.

Theorem 2 (Profile Empirical Likelihood) Assume the same conditions specified in Theorem 1 also hold here. We take the $\boldsymbol{h}$ function as specified in (ii) there (for pAUC and τ) but with smoothed indicator functions as discussed in the Remark following Theorem 1. Recall in this case $r = 2$ and $\boldsymbol{\theta} = (\theta_1, \theta_2) = (pAUC, \tau)$.

Define the profile log empirical likelihood ratio

$$W(\theta_1) = \inf_{\theta_2} -2\log R(\boldsymbol{\theta}) \,, \tag{13}$$

where $R(\boldsymbol{\theta})$ is given in Theorem 1.

As sample size $\min(m, n) \to \infty$, we have $W(\theta_1) \xrightarrow{\mathcal{D}} \chi^2_{(1)}$ if θ_1 is the true pAUC value.

Proof: See a proof in Qin and Lawless (1994) [12] Corollary 5 or Owen (2001) [10] Chapter 3. The proof was based on a two-term Taylor expansion of the likelihood ratio. The required smoothness conditions can be easily checked since we used a smoothed indicator function. ■

One consequence of Theorem 2 is that we can use W defined in (13) to test hypothesis and construct confidence intervals for pAUC similar to those procedures we discussed after Theorem 1, using $-2\log R(\theta^*)$ for AUC.

We shall discuss the computational methods for the empirical likelihood ratio defined in Theorems 1 and 2 later.

4 Example

Here we show the use of package `emplikAUC` with an illustrative example:

Example Background: As a real data example we analyze the performance of the biomarker s100b in the blood of patients at hospital admission after aneurysmal subarachnoid haemorrhage (aSAH) as a predictor of their 6-month outcome. The data is from Robin et al. (2011) [13] and more information can be found in Turck et al. (2010) [14]. It contains 113 patients, among which 41 are classified as poor outcome (disease) after 6-month. The data values are recorded with precision 0.01. In the below analysis, we used either a smoothing window $\epsilon = 0.05$ (default value) or 0.005. Notice the smallest difference within recorded data value here is 0.01 and therefore when $\epsilon = 0.005$, the only smoothing is to set the indicator $I[y > x]$ equal to 0.5 when x and y are identical, otherwise no smoothing is applied.

The quantile smoothing bandwidth ξ is taken to be $m^{-0.75}$ when analyzing pAUC.

R code for the example. The data set `aSAH` is from package `pROC`. We assume, of course, the packages `pROC`, `emplikAUC` version 0.3, `rootSolve` and `emplik2` version 1.32 are installed.

4.1 R codes for testing AUC and confidence interval

```
###### Get the data set aSAH for marker s100b ######
library(pROC)
data(aSAH)
Xis <- aSAH$s100b[aSAH$outcome == "Good"]
Yis <- aSAH$s100b[aSAH$outcome == "Poor"]

library(emplikAUC)
############ Compute the estimator of AUC ############
sum(smooth3(x=Xis, y=Yis))/(length(Xis)*length(Yis))
## 0.7321436      #### estimate of AUC, with default smoothing: eps=0.05 #####
#### If we use a smaller (=0.005) window width for smooth3, we get
sum(smooth3(x=Xis, y=Yis, eps=0.005))/(length(Xis)*length(Yis))
## 0.7313686      ##### estimate of AUC with smoothing: eps=0.005 #####
```

From above, we see the estimated AUC of the ROC curve for s100b is 0.7321436 or 0.7313686, depending on the smoothing parameter.

Next we test the hypothesis (compute p-value for) $H_0 : AUC = 0.73$; or 0.821502; or 0.623016. We use either the default smoothing parameter, or `eps=0.005`.

```
eltest4aucONE(theta=0.73, x=Xis, y=Yis, ind=smooth3, tol.u=1e-6, tol.v=1e-6, tol.H0=1e-6)
## $lambda
## [1] 0.8470956
##
## $u
##            [,1]       [,2]       [,3]       [,4]       [,5]       [,6]       [,7]       [,8]
## [1,] 0.01389148 0.01389553 0.01387525 0.01384887 0.01395442 0.01390413 0.01387525 0.01387525
## ......
## ......
## ......
## $v
##           [,1]       [,2]      [,3]       [,4]       [,5]       [,6]       [,7]       [,8]
## [1,] 0.0244692 0.02455493 0.0244083 0.02449509 0.02430056 0.02425494 0.02427034 0.02465355
## ......
## ......
## ......
## $`-2LLR`
## [1] 0.001819299
##
## $Pval
## [1] 0.9659779       #### p-value for testing the H0: AUC = 0.73
##
## $iterNum
## [1] 2

#### More tests. To save space, we only show the "-2LLR" output. This time, we also want to use
#### a different smoothing eps=0.005.
Mysmoo <- function(x,y) {smooth3(x,y,eps=0.005)}
eltest4aucONE(theta=0.821502, x=Xis, y=Yis, ind=Mysmoo, tol.u=1e-6, tol.v=1e-6, tol.H0=1e-6)$"-2LLR"
## [1] 3.841464         #### this chi square df1 value gives a p-value of 5 percent.

eltest4aucONE(theta=0.623016, x=Xis, y=Yis, ind=Mysmoo, tol.u=1e-6, tol.v=1e-6, tol.H0=1e-6)$"-2LLR"
## [1] 3.841491         #### this leads to a 5 percent p-value.
```

Next, we compute the 95% confidence interval for AUC using the empirical likelihood method. First with smoothing eps=0.005 then again with eps=0.05. The confidence interval for the eps=0.005 case can actually be read from the above test results.

```
findULNEW(step=0.03, fun=eltest4aucONE, MLE=0.73, x=Xis, y=Yis, ind=Mysmoo,
                                         tol.u=1e-6, tol.v=1e-6, tol.H0=1e-6)
## $Low
## [1] 0.6230165
##
## $Up
## [1] 0.8215019
##
## $FstepL
## [1] 7.450581e-09
##
## $FstepU
## [1] 7.450581e-09
##
## $Lvalue
## [1] 3.841459
##
## $Uvalue
## [1] 3.841459

findULNEW(step=0.03, fun=eltest4aucONE, MLE=0.73, x=Xis, y=Yis, ind=smooth3,
                                           tol.u=1e-6, tol.v=1e-6, tol.H0=1e-6)
## $Low
## [1] 0.6262756
##
## $Up
## [1] 0.8210439
##
## $FstepL
## [1] 7.450581e-09
##
## $FstepU
## [1] 9.679001e-09
##
## $Lvalue
## [1] 3.841459
##
## $Uvalue
## [1] 3.841459
```

We see the 95% confidence interval for AUC, using smoothing eps=0.005, is [0.6230165, 0.8215019]. We point out that our confidence interval is non-symmetric, i.e. not centered at the estimator: $0.7313686 \neq (0.6230165 + 0.8215019)/2$ which is a great feature of the Wilks type confidence intervals. Similarly, with eps=0.05, we get 95% confidence interval [0.6262756, 0.8210439], which is not center at the estimator (with same eps), $0.7321436 \neq (0.6262756 + 0.8210439)/2$.

The same computation of test and confidence interval can also be achieved by another function in the package emplikAUC, namely el2test4auc( ) which uses a different computing algorithm (EM) and calling code from the R package emplik2, as shown below.

```
temp <- el2test4auc(theta=0.73, x=Xis, y=Yis, ind=smooth3)
temp$"-2LLR"
## $`-2LLR`
## [1] 0.001819299
```

```
temp$Pval
## $Pval
## [1] 0.9659779
##### we see that the output "-2LLR" and "Pval" are the same as before. This function
##### also has other outputs.
Mysmoo <- function(x,y) {smooth3(x,y,eps=0.005)}
el2test4auc(theta=0.821502, x=Xis, y=Yis, ind=Mysmoo)$"-2LLR"
## [1] 3.841464      #### this chi square df1 value gives a p-value of 5 percent.

el2test4auc(theta=0.623016, x=Xis, y=Yis, ind=Mysmoo)$"-2LLR"
## [1] 3.841491      #### this leads to a 5 percent p-value.

findULNEW(step=0.03, fun=el2test4auc, MLE=0.73, x=Xis, y=Yis, ind=Mysmoo)
## $Low
## [1] 0.6230165
##
## $Up
## [1] 0.8215019
##
## $FstepL
## [1] 7.450581e-09
##
## $FstepU
## [1] 7.450581e-09
##
## $Lvalue
## [1] 3.841459
##
## $Uvalue
## [1] 3.841459

findULNEW(step=0.03, fun=el2test4auc, MLE=0.73, x=Xis, y=Yis, ind=smooth3)
## $Low
## [1] 0.6262756
##
## $Up
## [1] 0.8210439
##
## $FstepL
## [1] 7.450581e-09
##
## $FstepU
## [1] 9.678999e-09
##
## $Lvalue
## [1] 3.841459
##
## $Uvalue
## [1] 3.841459
```

This function `el2test4auc( )` could be slower compared to `eltest4aucONE( )`.

As a comparison, the confidence interval for AUC obtained by `pROC` package `ci.auc` function is [0.63012, 0.83262] using the 'DeLong' method, and using the bootstrap method we got [0.6265, 0.8276]. When computing the bootstrap confidence interval (here and two more below) we `set.seed(123)` and used 250,000 bootstrap repetitions. The 'Delong' confidence interval is symmetric about the estimator.

4.2 R code for pAUC

Patients with poor post-aSAH outcome require specific health care management, therefore the clinical test must be highly specific. A pAUC with specificity in the 80% to 100% range maybe of interest. This corresponds to pAUC(0, 0.2). The estimator of pAUC(0, 0.2) here for biomarker s100b is 0.08061155. The 95% empirical likelihood confidence interval for the pAUC$(0, 0.2)$ is [0.04981071, 0.114224]. As a comparison, the bootstrap 95% confidence interval is [0.05068, 0.1158] using `pROC` package.

Here comes the code for testing and confidence interval of pAUC$(0, 0.2)$. We first get an estimator of τ and pAUC$(0, 0.2)$. The quantile estimator τ is by our smoothing definition as in section 3.

```
##### Use our smooth definition to get an estimators of 80th quantile of F and
##### further, the pAUC(0,0.2) estimator. #####
myEstPaucT(x=Xis, y=Yis, partial=0.2, eps=0.005, epsT=(72)^(-0.75))
## $`tau(1-partial)`
## [1] 0.2083062
##
## $`Pauc(0,partial)`
## [1] 0.08061155
#### The output value can change a little bit for other smoothing values of eps/epsT.
```

Now jointly testing $F^{-1}(0.8) = tau = 0.2083062$ and $pAUC(0, 0.2) = 0.08061155$. These are NPMLE values, therefore the returned p-value should be 1 and $-2\log$ likelihood ratio should be zero. Here, the null distribution is (approximately) chi square with 2DF.

```
eltest4paucT(tau=0.2083062, true=0.08061155, ind=smooth3, p=0.2, x=Xis, y=Yis, epsxy=0.005,
                epsT=(72)^(-0.75), tol.u=1e-6, tol.v=1e-6, tol.H0=1e-6)

## $lambda
## [1] 0 0
##
## $u
##            [,1]       [,2]       [,3]       [,4]       [,5]       [,6]       [,7]
## [1,] 0.01388889 0.01388889 0.01388889 0.01388889 0.01388889 0.01388889 0.01388889
## ......
## ......
## $v
##            [,1]       [,2]       [,3]       [,4]       [,5]       [,6]       [,7]
## [1,] 0.02439024 0.02439024 0.02439024 0.02439024 0.02439024 0.02439024 0.02439024
## ......
## ......
## $`-2LLR`
## [1] 0
##
## $IterNum
## [1] 1

###### If we use the alternative function, (using EM algorithm):
el2testPaucT(tau=0.2083062, pauc=0.08061155, ind=smooth3, partial=0.2, x=Xis, y=Yis,
                   epsxy=0.005, epsT=(72)^(-0.75))
## [1] 2.273737e-13
###### This function only returns the "-2LLR" value. (compare to '-2LLR' [1] 0 above)
```

Let us test a few more pairs of $\tau, pAUC(0, 0.2)$ values. We shall only compare the $-2\log$ likelihood ratio values.

```
eltest4paucT(tau=0.2, true=0.09, ind=smooth3, p=0.2, x=Xis, y=Yis, epsxy=0.005,
                  epsT=(72)^(-0.75), tol.u=1e-6, tol.v=1e-6, tol.H0=1e-6)$"-2LLR"
## [1] 0.3163623

el2testPaucT(tau=0.2, pauc=0.09, ind=smooth3, partial=0.2, x=Xis, y=Yis,
                  epsxy=0.005, epsT=(72)^(-0.75))
## [1] 0.3163862


eltest4paucT(tau=0.2, true=0.1, ind=smooth3, p=0.2, x=Xis, y=Yis, epsxy=0.005,
                 epsT=(72)^(-0.75), tol.u=1e-6, tol.v=1e-6, tol.H0=1e-6)$"-2LLR"
## [1] 1.534045
el2testPaucT(tau=0.2, pauc=0.1, ind=smooth3, partial=0.2, x=Xis, y=Yis,
                 epsxy=0.005, epsT=(72)^(-0.75))
## [1] 1.534021
###### The returned values are basically the same from two functions. When the
###### parameter values to be tested are too far from the NPMLE, the computation will fail.

eltest4paucT(tau=0.2, true=0.14, ind=smooth3, p=0.2, x=Xis, y=Yis, epsxy=0.005,
                 epsT=(72)^(-0.75), tol.u=1e-6, tol.v=1e-6, tol.H0=1e-6)
## Error in stode(y, times, func, parms = parms, ...) :
## Model function must return a list of values, of which first element has length =length of y

el2testPaucT(tau=0.2, pauc=0.14, ind=smooth3, partial=0.2, x=Xis, y=Yis, epsxy=0.005,
                 epsT=(72)^(-0.75))
## Error in solve.default(constraintp, -constraint) :
## system is computationally singular: reciprocal condition number = 7.82754e-17
```

Next, testing H_0 : pAUC(0, 0.2)=0.08 alone. Here the null distribution of the -2LLR should be approximately chi square with 1DF. In the required inputs are the parameters `nuilow` and `nuiup`. This is the interval we shall search for the nuisance parameter when profiling the empirical likelihood. For pAUC(0, 0.2) values near NPMLE, this interval can reasonably set centered at NPMLE of tau. When we are testing the pAUC(0, 0.2) values above the NPMLE, this interval should move down; and when testing the pAUC(0, 0.2) values below NPMLE, this interval should move up.

The output `Nupar` is the final value of nuisance parameter when search to minimize the -2 log likelihood ratio. The interval `nuilow` and `nuiup` should contain `Nupar` comfortably.

```
eltest4paucONE(theta=0.08, x=Xis, y=Yis, ind=smooth3, nuilow=0.15, nuiup=0.25, partial=0.2,
                epsxy=0.005, epsT=(72)^(-0.75))
## $`-2LLR`
## [1] 0.001175356
##
## $Nupar
## [1] 0.2097241
##
## $Pval
## [1] 0.9726511

el2testPauc(theta=0.08, x=Xis, y=Yis, ind=smooth3, nuilow=0.15, nuiup=0.25, partial=0.2,
                epsxy=0.005, epsT=(72)^(-0.75))
## $`-2LLR`
## [1] 0.001167056
##
## $Nupar
## [1] 0.2097152
```

```
##
## $Pval
## [1] 0.9727478
```

Notice the function `el2testPauc( )` is slower.

Next, find 95% confidence interval of pAUC(0,0.2) by trial and error. We omit some (un-success trials) computation.

```
el2testPauc(theta=0.049810707, x=Xis, y=Yis, ind=smooth3, nuilow=0.2, nuiup=0.39,
            partial=0.2, epsxy=0.005, epsT=(72)^(-0.75))
## $`-2LLR`
## [1] 3.841459
##
## $Nupar
## [1] 0.3049045
##
## $Pval
## [1] 0.05

el2testPauc(theta=0.114224, x=Xis, y=Yis, ind=smooth3, nuilow=0.1, nuiup=0.25,
            partial=0.2,epsxy=0.005, epsT=(72)^(-0.75))
## $`-2LLR`
## [1] 3.841459
##
## $Nupar
## [1] 0.1777102
##
## $Pval
## [1] 0.05
##

##### Thus we verified the 95% confidence interval for pAUC(0,0.2): is [0.049810, 0.114224],
##### since testing the upper/lower bound, we get -2log W(.) = chisq(0.95, df=1) ####

###### Same calculetion by using another function. Some small difference, likely due to
###### convergence control.

eltest4paucONE(theta=0.049810707, x=Xis, y=Yis, ind=smooth3, nuilow=0.2, nuiup=0.39, partial=0.2,
                 epsxy=0.005, epsT=(72)^(-0.75))
## $`-2LLR`
## [1] 3.841443
##
## $Nupar
## [1] 0.3049043
##
## $Pval
## [1] 0.05000047

eltest4paucONE(theta=0.114224, x=Xis, y=Yis, ind=smooth3, nuilow=0.11, nuiup=0.22, partial=0.2,
                 epsxy=0.005, epsT=(72)^(-0.75))
## $`-2LLR`
## [1] 3.841376
##
## $Nupar
## [1] 0.1777324
##
```

```
## $Pval
## [1] 0.05000247
```

Finally, let us try to use the function `findUnew( )` and `findLnew( )` to find the upper and lower confidence bound automatically.

```
findUnew(fun=el2testPauc, MLE=0.0806, x=Xis, y=Yis, ind=smooth3, nuilow=0.1, nuiup=0.3,
                     partial=0.2,epsxy=0.005, epsT=(72)^(-0.75))
## $Up
## [1] 0.114224      #### Upper bound of 95 percent confidence interval.
##
## $FstepU
## [1] 7.450581e-09
##
## $Uvalue
## [1] 3.841459

findLnew(fun=el2testPauc, MLE=0.0806, x=Xis, y=Yis, ind=smooth3, nuilow=0.2, nuiup=0.4,
                      partial=0.2,epsxy=0.005, epsT=(72)^(-0.75))
## $Low
## [1] 0.04981071    #### Lower bound of 95 percent confidence interval.
##
## $FstepL
## [1] 7.450581e-09
##
## $Lvalue
## [1] 3.841459

##### We may also use findULNEW( ) to find the upper/lower confidence interval bounds together

findULNEW(fun=el2testPauc, MLE=0.0806, x=Xis, y=Yis, ind=smooth3, nuilow=0.1, nuiup=0.4,
                  partial=0.2,epsxy=0.005, epsT=(72)^(-0.75))
$Low
[1] 0.04981071

$Up
[1] 0.114224

$FstepL
[1] 7.450581e-09

$FstepU
[1] 7.450581e-09

$Lvalue
[1] 3.841459

$Uvalue
[1] 3.841459
##### When try to find the upper and lower bound of CI together, we must supply the
##### parameters nuilow/nuiup that are wide enough to suit both search. It may not be
##### easy (that is why we try to do upper and lower bound separately), but here it worked.


#####  Same computation but using a different function. However this function is more demanding
#####  to achieve convergence, and thus needs a narrower interval, [nuilow, nuiup] for
```

```
#####  nuisance parameter search. But this function is much faster when convergent.


findUnew(fun=eltest4paucONE, MLE=0.0806, x=Xis, y=Yis, ind=smooth3, nuilow=0.13, nuiup=0.24,
                partial=0.2, epsxy=0.005, epsT=(72)^(-0.75))
## $Up
## [1] 0.1142243
##
## $FstepU
## [1] 7.450581e-09
##
## $Uvalue
## [1] 3.841459

findLnew(fun=eltest4paucONE, MLE=0.0806, x=Xis, y=Yis, ind=smooth3, nuilow=0.20, nuiup=0.40,
                   partial=0.2, epsxy=0.005, epsT=(72)^(-0.75))
## $Low
## [1] 0.04981065
##
## $FstepL
## [1] 7.450581e-09
##
## $Lvalue
## [1] 3.841459
##### Using this function we fail to find the Upper and Lower bound together by using findULNEW( ).
```

5 Computation Algorithm For Inference of AUC

One way to compute the empirical likelihood ratio in Theorem 1 is to call the functions in the R package `emplik2`. However, the computation used there is based on the EM algorithm and can be slow at times. We developed another (iterative) algorithm for computing the empirical likelihood ratio, which we shall discuss next.

Using Lagrange multiplier method for the constrained optimization problem of Theorem 1, the Lagrangian is

$$L(u_i, v_j, \gamma, \eta, \lambda) = \sum_{i=1}^{m} \log m u_i + \sum_{j=1}^{n} \log n v_j - \gamma \left(\sum u_i - 1 \right) - \eta \left(\sum v_j - 1 \right) \\ - \lambda \sum_i \sum_j u_i v_j h(X_i, Y_j, \theta) \, . \tag{14}$$

Taking derivatives and set them to zero leads to

$$u_i(\lambda, v_1, \cdots, v_n) = \frac{1}{m + \lambda \sum_j h(X_i, Y_j, \theta) v_j} \, , \qquad i = 1, \cdots, m \tag{15}$$

$$v_j(\lambda, u_1, \cdots, u_m) = \frac{1}{n + \lambda \sum_i h(X_i, Y_j, \theta) u_i} \, , \qquad j = 1, \cdots, n. \tag{16}$$

The above two sets of equations plus the following (constraint requirement)

$$\sum_i \sum_j \frac{h(X_i, Y_j, \theta)}{[m + \lambda \sum_k h(X_i, Y_k, \theta) v_k] \, [n + \lambda \sum_k h(X_k, Y_j, \theta) u_k]} = 0 \tag{17}$$

are the system of equations we need to solve to obtain maximized $\log R(\theta)$. Notice there are $m+n+1$ equations for $m+n+1$ variables (u_i, v_j, λ). Compared to the situation of Owen (1988) [9], we have $m+n$ more equations here: namely (15) and (16). A direct solution seems elusive. But the following iterative method works in our investigations.

Initialize $u_i^{(0)} = 1/m$ and $v_j^{(0)} = 1/n$,

1. Plug $u_i^{(s)}$ and $v_j^{(s)}$ into equation (17), solve for λ, call the solution $\lambda^{(s+1)}$.
2. Using $u_i^{(s)}$ and $v_j^{(s)}$ and $\lambda^{(s+1)}$ obtained above, plug into the right hand side of equations (15) and (20). This yields $u_i^{(s+1)}, v_j^{(s+1)}$.
3. With $u_i^{(s+1)}$ and $v_j^{(s+1)}$, repeat steps 1–2 to obtain $\lambda^{(s+2)}$ and $u_i^{(s+2)}, v_j^{(s+2)}$.
4. Iterate until λ converges.

This is the algorithm we used when computing examples and carry out simulations.

An implementation of above is in our package `emplikAUC`.

6 Computation Algorithm For Inference of pAUC

The first step here is to compute the log empirical likelihood ratio $\log R(\boldsymbol{\theta})$ where $\boldsymbol{\theta} = (\theta_1, \theta_2)$. Here $\boldsymbol{\theta}$ and $\boldsymbol{h}$ are as defined in Theorem 1 equation (11) but with smoothing. This is similar to the calculation detailed in the above subsection for AUC except one constrain equation, (17), becomes two constrain equations, (18) – (19) now, and $\boldsymbol{\lambda} = (\lambda_1, \lambda_2)$, $\boldsymbol{h}$ and $\boldsymbol{\theta}$ are now vectors of length two.

$$\sum_{i=1}^{m}\sum_{j=1}^{n} \frac{h_1(X_i, Y_j, \boldsymbol{\theta})}{[m + \boldsymbol{\lambda}^\top \sum_k \boldsymbol{h}(X_i, Y_k, \boldsymbol{\theta}) v_k]\,[n + \boldsymbol{\lambda}^\top \sum_k \boldsymbol{h}(X_k, Y_j, \boldsymbol{\theta}) u_k]} = 0 \tag{18}$$

$$\sum_{i=1}^{m} \frac{h_2(X_i, \theta_2)}{m + \boldsymbol{\lambda}^\top \sum_k \boldsymbol{h}(X_i, Y_k, \boldsymbol{\theta}) v_k} = 0 \tag{19}$$

The above two constrain equations plus (15) – (16), (with the obvious modification of $\lambda \to \boldsymbol{\lambda}$, $h \to \boldsymbol{h}$ and $\theta \to \boldsymbol{\theta}$), are the system of $n+m+2$ equations for $n+m+2$ unknowns $(u_i, v_j, \lambda_1, \lambda_2)$ we need to solve. The iterative algorithm detailed in the previous subsection also works well here.

Now assume we have obtained $\log R(\boldsymbol{\theta})$. The second step is the minimization (profiling) of $\log R(\boldsymbol{\theta})$ over θ_2. This is an unconstrained minimization problem over one variable and an obvious starting point for θ_2 is the sample quantile $\hat{\tau}$. In the simulation and examples, we have used R function `optimize` to accomplish this.

The profile will yield $\log W(\theta_1)$ which can be used to test and construct confidence intervals for pAUC.

7 Remarks

To deal with the inference of pAUC, we view the estimation in the larger framework of a two-parameter problem, by explicitly include the nuisance parameter τ. This naturally lead to the profile likelihood technique when statistical inference for only one of the two parameters is needed.

When sample sizes goes to infinite, the Wilks and the Wald type confidence intervals (assume both available) are equivalent. However, for smaller samples, it is a generally accepted fact that the Wilks confidence intervals have several advantages over the Wald confidence intervals, see section 3 of Meeker and Escobar [8] and additional references there. The disadvantage they mentioned for the Wilks method is the computational difficulty. But with ever faster computers and publicly available software like R, this is much less of a problem nowadays.

We list here briefly the advantages of Wilks confidence interval:

1). The Wilks confidence intervals are not necessary symmetric about the MLE, rather, it tries to reflect the skewness in the given data.

2). The Wilks confidence intervals are always within the parameter space, while a Wald confidence interval of a probability can include negative values, for example.

3). Once we obtained the Wilks confidence interval for a parameter θ, $[a, b]$ (say), the Wilks confidence interval for $g(\theta)$ is just $[g(a), g(b)]$ (assuming g is increasing).

4). When using Wilks, there is no need to figuring out the variance of the MLE and estimate it.

5). The actual error rates for Wilks intervals are often closer to the nominal than the Wald.

6). Bootstrap re-sampling based procedures rely on the random number generator and the bootstrap repetitions used. A different seed or different number of repetitions will lead to slightly different confidence intervals. Our empirical likelihood confidence interval do not have this problem.

7). The Wilks confidence interval is based on likelihoods and there is well developed theory to handle nuisance parameter in the likelihood analysis. We use this feature to handle nuisance parameter in the inference of pAUCs.

References

[1] Chen, SX. and Hall, P. (1993). Smoothed Empirical Likelihood Confidence Intervals for Quantiles. *The Annals of Statistics* **21,** 1166-1181.

[2] DeLong, ER, DeLong, DM, Clarke-Pearson, DL. (1988). Comparing the area under two or more correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics* 44: 837–845.

[3] Dodd, LE, and Pepe, MS. (2003). Partial AUC Estimation and Regression. *Biometrics* **59**, 614-623.

[4] Hanley, JA. and McNeil, BJ. (1982). The Meaning and Use of the Area under a Receiver Operating Characteristic (ROC) Curve. *Radiology* **143,** 29-36.

[5] He, Y, and Escobar, M. (2008). Nonparametric statistical inference method for partial areas under receiver operating characteristic curves, with application to genomic studies. *Statist. Medicine* **27,** 5291-5308.

[6] Krzanowski, W, and Hand, D. (2009). *ROC Curves for Continuous Data,* Chapman & Hall/CRC, Boca Raton.

[7] McClish, DK. (1989). Analyzing a portion of the ROC curve. *Medical Decision Making* **9(3),** 190-195.

[8] Meeker, W. and Escobar, L. (1995). Teaching about Approximate Confidence Regions Based on Maximum Likelihood Estimation. *The American Statistician*, Vol. 49, No. 1, pp. 48–53.

[9] Owen, AB. (1988). Empirical Likelihood Ratio Confidence Intervals for a Single Functional. *Biometrika* **75,** 237 - 249.

[10] Owen, AB. (2001). *Empirical Likelihood*. Chapman & Hall/CRC: Boca Raton, London, New York, Washington, D.C.

[11] Pepe, MS. (2003). *The Statistical Evaluation of Medical Tests for Classification and Prediction,* Oxford University Press Inc., New York.

[12] Qin, J. and Lawless, J. (1994). Empirical likelihood and general estimating equations. *Ann. Statist.* **22,** 300-325.

[13] Robin, X. Turck, N. Hainard, A. Tiberti, N. Lisacek, F. Sanchez, JC. Müller, M. (2011). pROC: an open-source package for R and S+ to analyze and compare ROC curves. *BMC Bioinformatics*, **12**, 77. DOI: 10.1186/1471-2105-12-77.

[14] Turck N, Vutskits L, Sanchez-Pena P, Robin X, Hainard A, Gex-Fabry M, Fouda C, Bassem H, Mueller M, Lisacek F, Puybasset L, and Sanchez J. (2010). A multiparameter panel method for outcome prediction following aneurysmal subarachnoid hemorrhage. *Intensive Care Med* **36** 107–115.

[15] Wieand, HS, Gail, M, and James, B. (1989). A family of nonparametric statistics for comparing diagnostic markers with paired or unpaired data. *Biometrika* 76: 585–592.

[16] Yang, H., Lu, K. and Zhao, Y. (2017). A nonparametric approach for partial areas under roc curves and ordinal dominance curves. *Statistica Sinica* **27**, 357-371.

[17] Yu, J., Yang, L., Vexler, A. and Hutson, AD. (2016). Easy and accurate variance estimation of the nonparametric estimator of the partial area under the ROC curve and its application. *Statist. Medicine* **35**, Issue 13, 2251-2282.

[18] Zhao, YM. (2016). Statistical Inference on Trimmed Means, Lorenz Curves, and Partial Area Under ROC Curves by Empirical Likelihood Method. University of Kentucky Theses and Dissertations–Statistics. 24. https://uknowledge.uky.edu/statistics_etds/24

[19] Zhou, M. (2016). *Empirical Likelihood Method in Survival Analysis.* CRC Press: Boca Raton.

[20] Zhou, XH, Obuchowski, N. and McClish, D. (2011). *Statistical Methods in Diagnostic Medicine, 2nd Ed.* John Wiley & Sons Inc., Hoboken, New Jersey.

[21] Zhang, DD, Zhou, XH, Daniel, H. Freeman, J, Freeman, JL. (2002). A non-parametric method for the comparison of partial areas under roc curves and its application to large health care data sets. *Statist. Medicine* **21,** 701-715.

[22] Zou, KH., Liu, A., Bandos, AI., Ohno-Machado, L. and Rockette, H. (2012). *Statistical Evaluation of Diagnostic Performance: Topics in ROC Analysis,* CRC Press, Boca Raton.

[23] Zou, K. Hall, WJ. Shapiro, D. (1997). Smooth non-parametric receiver operating characteristic (ROC) curves for continuous diagnostic tests. *Statist. Medicine* **16**, 2143-2156.