**The statistical approach in trial-based economic evaluations matters: get your statistics together!**

By:

\*Elizabeth N. Mutubuki,1 \*Mohamed El Alili,2 Judith E. Bosmans,2  Teddy Oosterhuis,2 Frank Snoek,3 Raymond W. J. G. Ostelo,1,4 Maurits W. van Tulder,2,5 Johanna M. van Dongen,1,2

\*Contributed equally

Journal: Expert Review of Pharmacoeconomics and Outcomes Research

1Department of Health Sciences, Faculty of Science, VU Amsterdam, Amsterdam Movement Sciences Research Institute, the Netherlands

2Department of Health Sciences, Faculty of Science, VU Amsterdam, Amsterdam Public Health Research Institute, the Netherlands

3Department of Medical Psychology, Amsterdam UMC, VU Amsterdam, Amsterdam, the Netherlands

4Department of Epidemiology and Biostatistics, Amsterdam UMC, Location VUmc, Amsterdam Movement Sciences Research Institute, the Netherlands

5Department of Physiotherapy & Occupational Therapy, Aarhus University Hospital, Aarhus, Denmark

Corresponding author:

Mohamed El Alili, Department of Health Sciences, Faculty of Science, VU Amsterdam, De Boelelaan 1085, 1081 HV, Amsterdam, the Netherlands. Tel.: +31 20 59 82790; Fax: +31 20 6462457; E-mail: m.elalili@vu.nl

Of note: Please refer to this study when using the below syntax.

\*\*\*\*\*\*\*\*\*\* CEA ANALYSES \*\*\*\*\*\*\*\*\*\*

clear

set more off

cd "XXX"

capture log close

log using "XXXX.smcl", replace

\*\*\*\*\* Fill in n = number of imputations \*\*\*\*\*

local n = XXX

forvalues j=1(1)`n' {

local y = `j'

use "XXXXX`y'", clear

bootstrap bootcost\_diff = \_b[YYYY:XXXX] booteffect\_diff = \_b[YYYY:XXXX], reps(XXXX) seed(XXXX) saving("boots`y'", replace) bca: sureg (YYYY = XXXX) (YYYY = XXXX)

mat betaCE= e(b) /\* extract the matrix of regression coefficients \*/

mat se = e(se) /\* extract standard errors \*/

mat limits = e(ci\_bc) /\* extract confidence limits \*/

mat vari = e(V) /\* extract the variance-covariance matrix \*/

gen cost\_diff = betaCE[1,1] /\* create differential costs \*/

gen effect\_diff = betaCE[1,2] /\* create differential effects \*/

gen N = e(N) /\* extract sample size\*/

gen LL\_effect = limits[1,2]

gen UL\_effect = limits[2,2]

gen LL\_cost = limits[1,1]

gen UL\_cost = limits[2,1]

gen cost\_var = vari[1,1] /\* extract the variance of the mean differential costs from the VC matrix \*/

gen effect\_var = vari[2,2] /\* extract the variance of the mean differential effect from the VC matrix \*/

gen cov = vari[1,2] /\* extract the covariance between mean differential costs and effect \*/

save postboots`y', replace

}

clear

set more off

cd "XXXX"

\*\*\*\*\* Fill in n = number of imputations \*\*\*\*\*

local n = XX

/\* append bootstrap samples in 1 file \*/

use boots1, clear

forvalues k=2(1)`n' {

local z = `k'

append using boots`z'

}

save boots, replace

\*\*\* All information from the extra information from bivariate regression needs to be appended, allowing to pool according to Rubin’s rules \*\*\*

use postboots1, clear

forvalues l=2(1)`n' {

local a = `l'

append using postboots`a'

}

by \_mi\_m, sort: drop if \_n != \_N

save postboots, replace

keep cost\_diff LL\_cost UL\_cost cost\_var effect\_diff LL\_effect UL\_effect effect\_var cov

append using boots

gen Za=1.95996

/\* estimate confidence limits for effects using Rubin's rules \*/

egen effect\_diff\_pooled = mean(effect\_diff)

egen W=mean(effect\_var)

gen \_Bdiff=(effect\_diff-effect\_diff\_pooled)^2

egen \_Bsum=total(\_Bdiff)

gen B=(1/(`n'-1))\*\_Bsum

gen T=W+(1+(1/`n'))\*B

gen seT=sqrt(T)

gen LL\_effect\_pooled=effect\_diff\_pooled -(Za\*seT)

gen UL\_effect\_pooled=effect\_diff\_pooled +(Za\*seT)

/\* estimate bias-corrected and accelerated confidence limits for costs \*/

egen cost\_diff\_pooled = mean(cost\_diff)

egen LL\_cost\_pooled = mean(LL\_cost)

egen UL\_cost\_pooled = mean(UL\_cost)

generate ICER = cost\_diff\_pooled /effect\_diff\_pooled

display ICER

display effect\_diff\_pooled

display LL\_effect\_pooled

display UL\_effect\_pooled

display cost\_diff\_pooled

display LL\_cost\_pooled

display UL\_cost\_pooled

label variable bootcost\_diff "Bootstrapped estimates"

label variable cost\_diff\_pooled "Point estimate"

twoway (scatter bootcost\_diff booteffect\_diff, msize(small)) (scatter cost\_diff\_pooled effect\_diff\_pooled, msize(small)), ///

ytitle(Cost differences (€)) yline(0) xline (0) ///

name(CEplane, replace)

graph save "CEplane.gph", replace

gen quadrantcompl1 = 0

replace quadrantcompl1 = 1 if bootcost\_diff > 0 & booteffect\_diff > 0

replace quadrantcompl1 = 2 if bootcost\_diff < 0 & booteffect\_diff > 0

replace quadrantcompl1 = 3 if bootcost\_diff < 0 & booteffect\_diff < 0

replace quadrantcompl1 = 4 if bootcost\_diff > 0 & booteffect\_diff < 0

label variable quadrantcompl1 "quadrant of CE plane"

label define quadrantcompl1 1 NEQuadrant 2 SEQuadrant 3 SWQuadrant 4 NWQuadrant

sort quadrantcompl1

proportion quadrantcompl1

/\* estimate CEA curve using Rubin's rules \*/

forvalues i= 0 (1000) 80000 { /\* local macro i counts from XXXXXX to XXXXXX in steps of XXXXXX \*/

local x = `i'/ 1000 /\* x is created just for variable names \*/

gen NB`x'=(`i'\*effect\_diff)-cost\_diff /\* NBs are generated for each value of i \*/

gen varNB`x'=`i'^2 \* effect\_var + cost\_var - 2\*`i'\*cov /\* variance of NB is generated \*/

gen seNB`x'=sqrt(varNB`x') /\* standard error of NB is generated \*/

egen meanNB`x'=mean(NB`x')

egen W\_NB`x'=mean(varNB`x')

gen \_Bdiff\_NB`x'=(NB`x'-meanNB`x')^2

egen \_Bsum\_NB`x'=total(\_Bdiff\_NB`x')

gen B\_NB`x'=(1/(`n'-1))\*\_Bsum\_NB`x'

gen T\_NB`x'=W\_NB`x'+(1+(1/`n'))\*B\_NB`x'

gen seT\_NB`x'=sqrt(T\_NB`x')

local z = meanNB`x'/seT\_NB`x'

local prob = normal(`z')

matrix row = (`i',`prob')

matrix ceac = (nullmat(ceac)\row) /\* Matrix containing probability that intervention is cost-effective for each value of i \*/

}

svmat ceac /\* The matrix is converted into variables \*/

matrix drop ceac /\* The unneeded matrix is now dropped \*/

twoway (line ceac2 ceac1), ytitle(Probability intervention cost-effective) yscale(range(0 1)) ylabel(0 (0.2) 1) xtitle(Ceiling ratio: €/ QALY) xscale(range(0 80000)) xlabel(0 (10000) 50000)

graph save "CEAC.gph", replace

save postboots, replace

capture log close