TRIAL-BASED COST-EFFECTIVENESS ANALYSIS USING R: A TUTORIAL

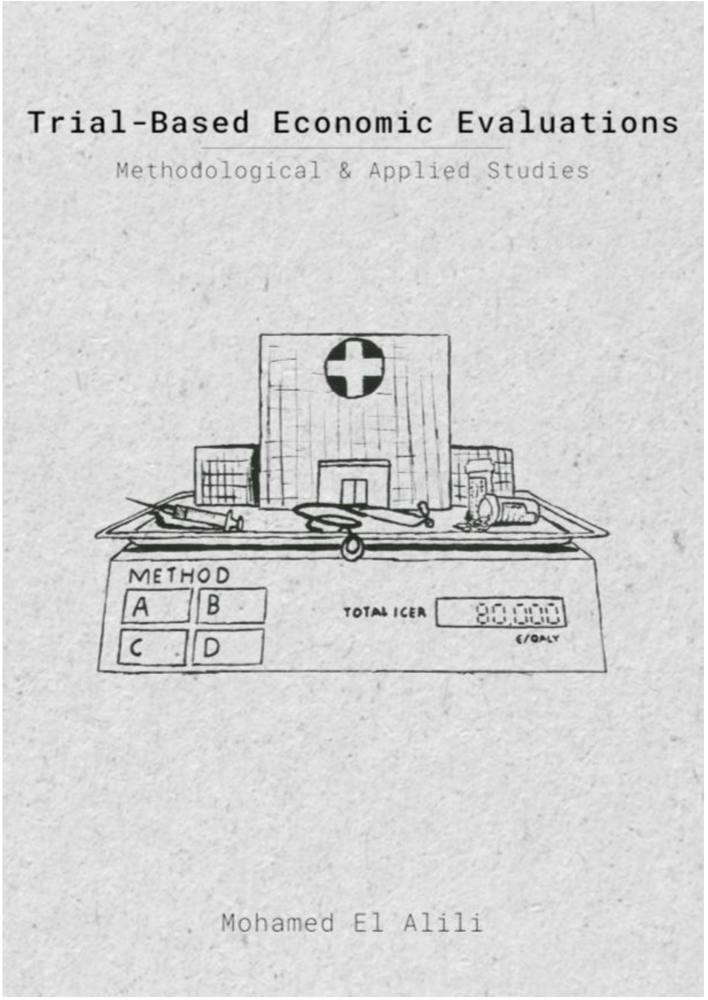
HM Broulikova, A Jornada Ben, M El Alili, JM van Dongen, JE Bosmans

HTA section, Department of Health Sciences
h.m.broulikova@vu.nl

R for HTA workshop
July 2, 2021

BACKGROUND

- Recommendation regarding trial-based CEA are often vague, little guidance how to deal with (Dongen et al. 2020)
  - missing data
  - correlated costs and effects
  - skewed data
  - clustering of the data
  - longitudinal data
  - baseline imbalances



STEPS OF THE TRIAL-BASED CEA

1. Preparing the data
2. Multiple impute data by using chained equations
3. Performing a seemingly unrelated regression model
4. Evaluate uncertainty using bias-corrected and accelerated bootstrapping
5. Presenting the results

PREPARING THE TRIAL DATA

- Treatment group n = 150
- Control group n = 150
- Available covariates age, gender, education, smoking, comorbidity
- follow-up for 1 year, 4 measurements
- total utility and costs at each measurement point

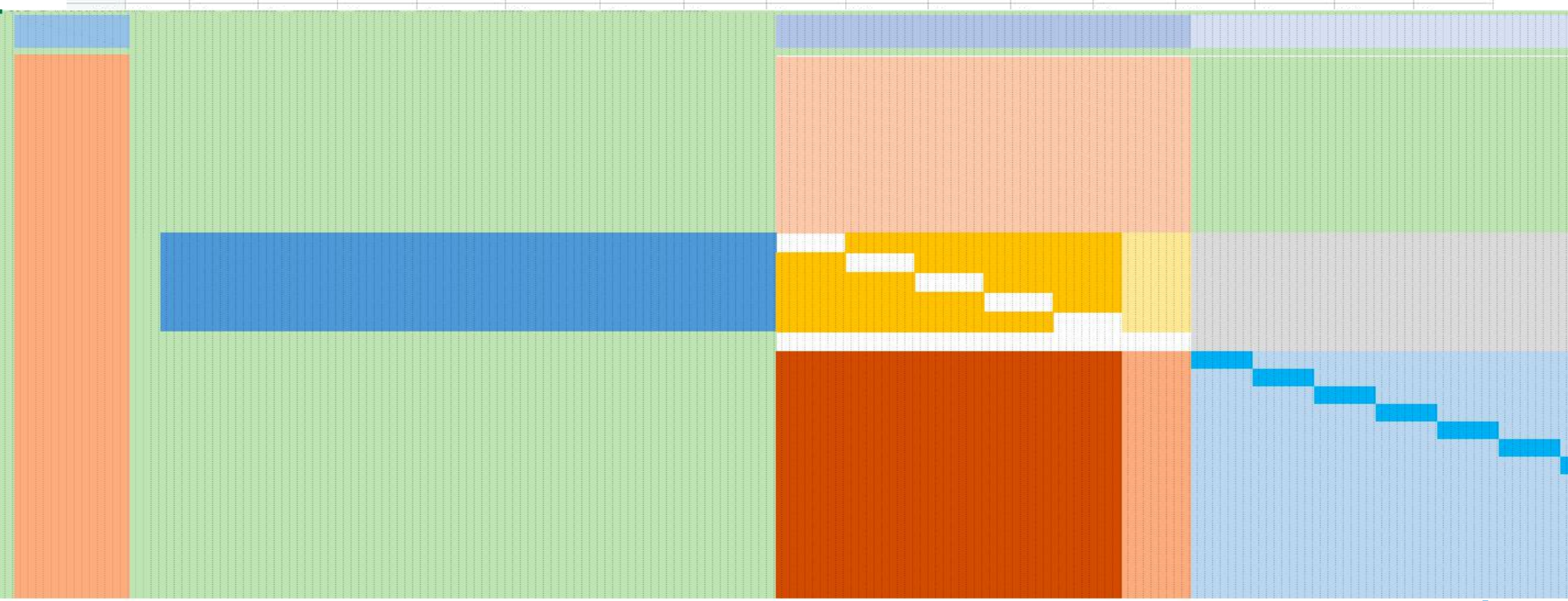
```
> head(dataset)
# A tibble: 6 x 17
     id   trt   age   gen  educ  smok   com   uT0   cT0   uT1   uT2   uT3   uT4     cT1       cT2      cT3    cT4
  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>   <dbl>     <dbl>    <dbl>  <dbl>
1     1     1    65     0     1     0     1 0.862 1199. 0.907 0.858 0.902 0.856 316.    6840.       439.    2611.
2     2     1    95     0     1     0     1 0.776  335. 0.737 NA    0.765 0.734   0.197    0.000506   0.0115   1.08
3     3     0    56     0     3     0     0 0.777  194. 0.722 0.783 NA    0.740  22.2     21.2        10.8     27.0
4     4     0    30     1     1     0     1 0.753   30.5 0.771 0.764 0.781 0.777   0.903    8.85        1.39    20.6
5     5     0    44     1     3     0     0 0.749  612. 0.841 0.831 0.827 0.736  54.7     NA          486.    4341.
6     6     0    44     0     3     0     1 0.781 1252. 0.817 0.813 0.837 0.840 7750.      603.       1476.    NA
```

MULTIPLE IMPUTATION I

- Check missing patterns and define imputation model
- Create customized prediction matrix

```
> p_missing
  id  trt  age  gen educ smok  com  uT0  cT0  uT1  uT2  uT3  uT4  cT1  cT2  cT3  cT4
   0    0    0    0    0    0    0    0    0   31   28   33   26   27   28   30   25
```

	id	trt	age	gen	educ	smok	com	uT0	cT0	uT1	uT2	uT3	uT4	cT1	cT2	cT3	cT4
id	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



MULTIPLE IMPUTATION II

- Change method if needed
- Impute the data per treatment arm – each treatment arm in one list (Faria, 2014)

```
# imputation method (check/set)
mi <- lapply(dataset, mice, predictorMatrix = Prediction_matrix, m = 1, maxit = 0, print = T)
mi[[1]][["method"]]
# needed, prediction method can be changed for individual variables using the code below and then specified
within mice
# Method["var_name"] <- "pmm"
```

```
set.seed(1)
mi <- lapply(dataset, mice, predictorMatrix = Prediction_matrix,  m = 5, maxit = 20, print = T)
#mi <- lapply(mi, mice, predictorMatrix = Prediction_matrix, method = Method, m = 5, maxit = 20, print = T)
# check for problems with imputation
mi[[1]][["loggedEvents"]]
mi[[2]][["loggedEvents"]]
```

SEEMINGLY UNRELATED REGRESSION

```
# perform SUR to get variance
# define regressions to extract variance-covariance matrix
r1 <- Tcosts ~ trt + cT0 + age + gen + educ + smok + com
r2 <- QALY  ~ trt + uT0 + age + gen + educ + smok + com

sur <- lapply(impds, function(x) {
  systemfit(list(costreg = r1, effectreg = r2), "SUR", data=x)})

# extract variance-covariance matrix for the effects (to be used to estimate CI for effects)
varcov <- lapply(sur, function(x) x[["coefCov"]])
var_e <- lapply(varcov, function(x) x[10,10])
var_e <- lapply(var_e, setNames, c("var_effects"))
```

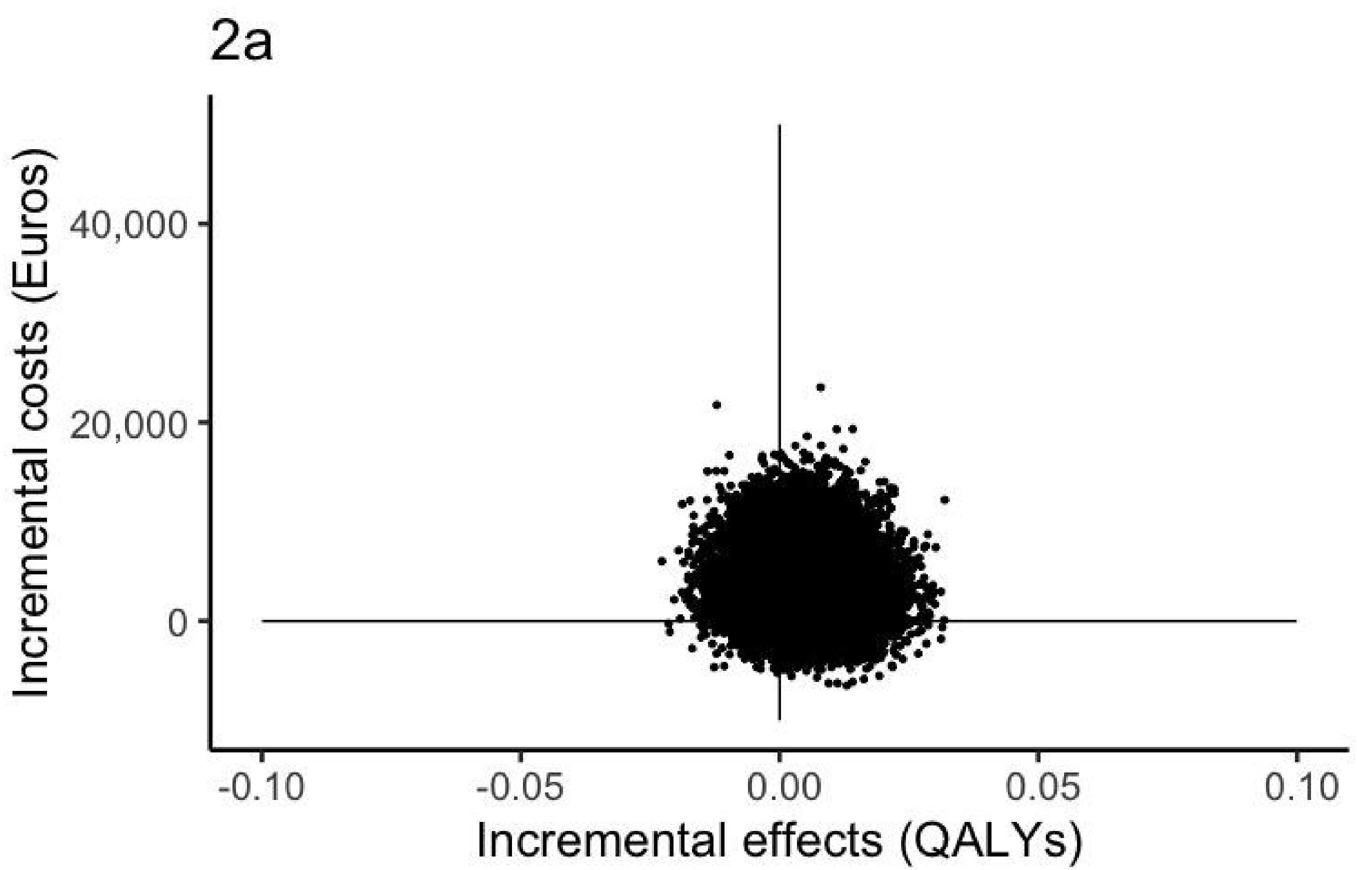
```
# pooling results
# cost and effect differences
effect_diff_pooled <- mean(observed$obseffect_diff)
cost_diff_pooled <- mean(observed$obscost_diff)

# lower and upper level limits for effects using Rubin's rules
Za = 1.95996
W <- mean(observed$var_effects)
B_diff <- (observed$obseffect_diff-effect_diff_pooled)^2
B_sum <- sum(B_diff)
B <- (1/(10-1))*B_sum
T_ <- W + (1+(1/10))*B
seT <- sqrt(T_)
LL_effect_pooled <- effect_diff_pooled - (Za*seT)
UL_effect_pooled <- effect_diff_pooled + (Za*seT)
```

EVALUATE UNCERTAINTY USING BCA BOOTSTRAPPING

```r
# function to execute seemingly unrelated regression for each bootstrapped sample
fsur <- function(impds, i){
  dscc2 <- impds[i,]
  r1 <- Tcosts ~ trt + cT0 + age + gen + educ + smok + com
  r2 <- QALY  ~ trt + uT0 + age + gen + educ + smok + com
  fit <- with(data = impds, exp = systemfit(list(costreg = r1, effectreg = r2), "SUR", data=dscc2))
  betas <- fit$coefficients
  return(c(betas[2], betas[10])) #
}

boot_time <- Sys.time()
set.seed(2)
# bootstrap imputed datasets and get SUR results
bootce <- lapply(impds, function(x) boot(data=x, statistic=fsur, R=5000))
# bootstrap confidence interval calculations for costs (also used to get SUR estimate t0)
bootcic <- lapply(bootce, function(x) boot.ci(boot.out = x, type = c("bca"), index = 1))
# bootstrap confidence interval calculations for effects (also used to get SUR estimate t0)
booteic <- lapply(bootce, function(x) boot.ci(boot.out = x, type = c("bca"), index = 2))
Sys.time() - boot_time


# lower and upper level limits for costs using bca because of the skewness
LL_cost_pooled <- mean(boot_df$LL_cost)
UL_cost_pooled <- mean(boot_df$UL_cost)
```

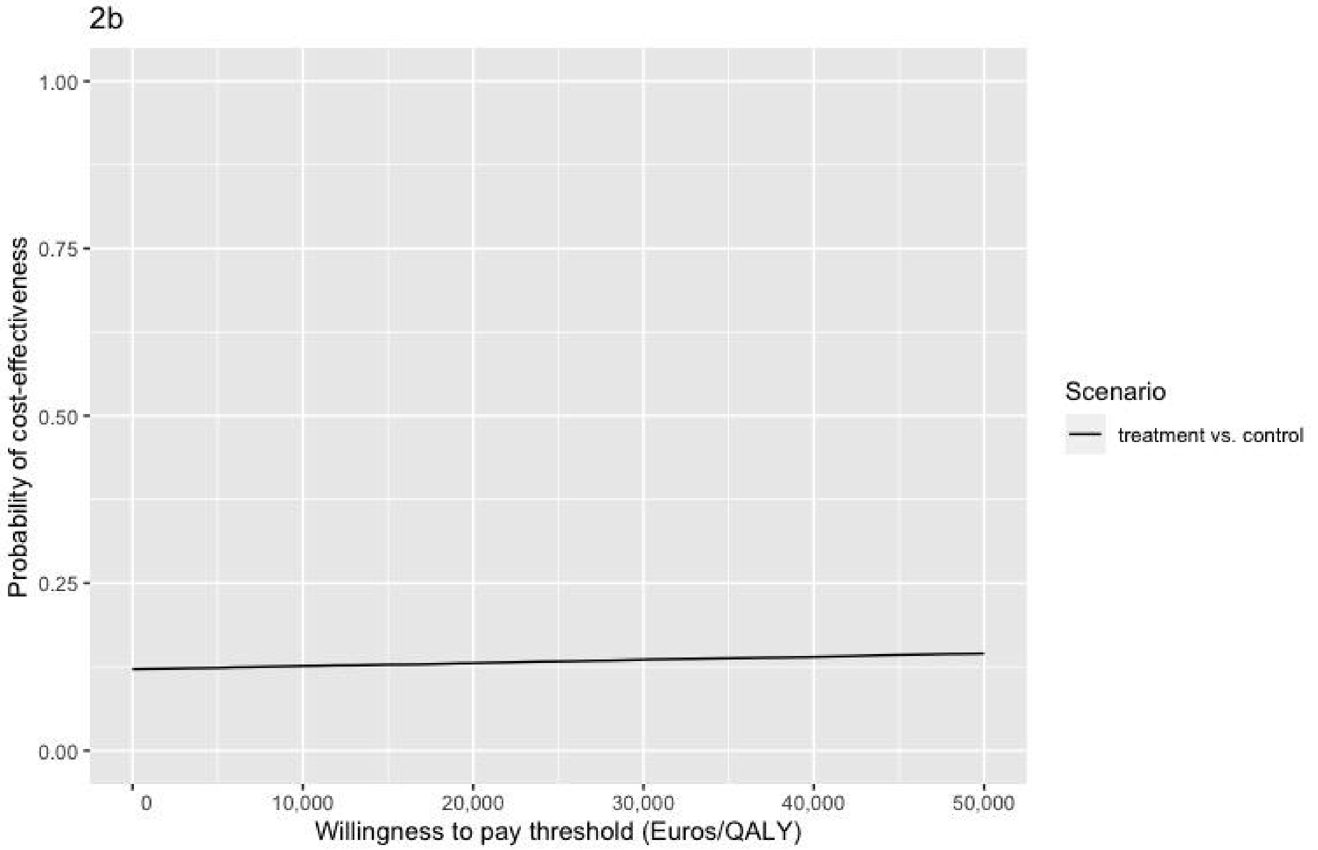
RESULTS I

```
# ICER
ICER <- cost_diff_pooled/effect_diff_pooled

# distribution of the results in the cost-effectiveness plane
boot_df <- muta


NW <- sum(boot_
SW <- sum(boot_
NE <- sum(boot_
SE <- sum(boot_
# check has to 
check <- NW + S
```

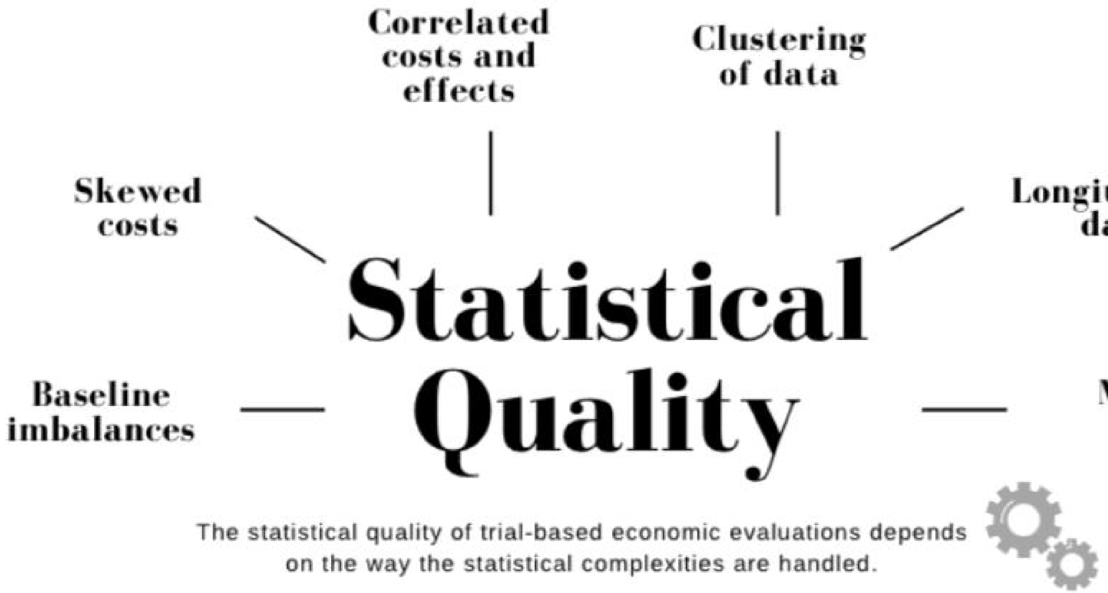


RESULTS II

```
# CEAC
results <- data.table(boot_df)
CEAC <- data.table()
for (wtp in seq(0, 50
  propPos <- results[
  CEAC<- rbind(CEAC,
}

CEAC <- mutate(CEAC,

CEACs_plot <- ggplot(
  geom_line() +
  scale_color_manual(
  labs(title="2b") +
  xlab("Willingness t
  ylab("Probability c
  scale_y_continuous(
  scale_x_continuous(
CEACs_plot
```



THANK YOU FOR YOUR ATTENTION



h.m.broulikova@vu.nl