

BCEA :: CHEAT SHEET

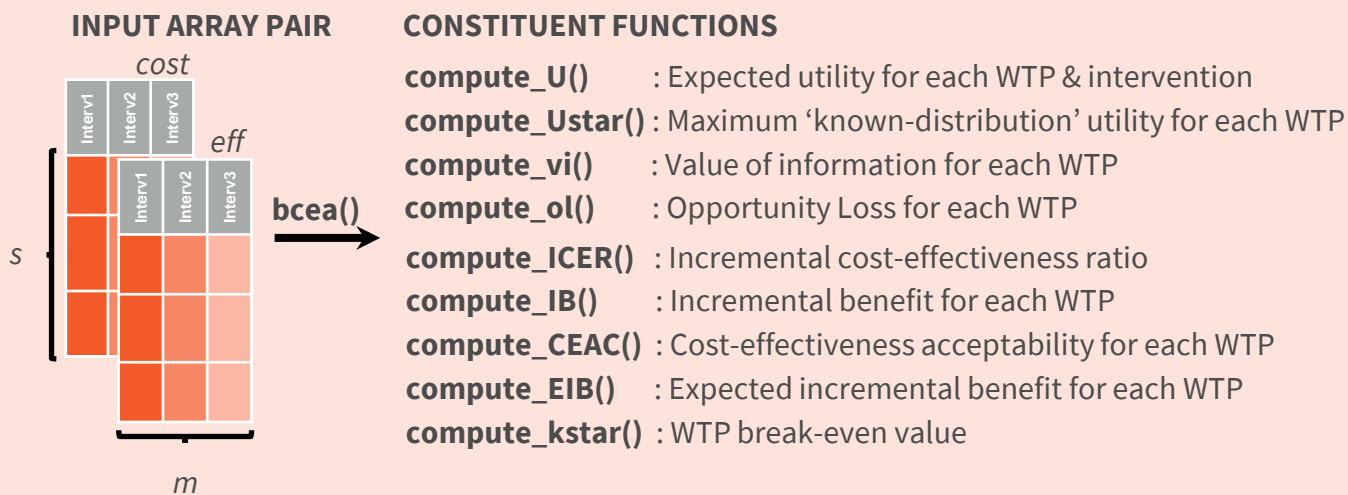


Introduction

Bayesian Cost-Effectiveness Analysis in R

Given a random sample of suitable variables of costs (*cost*) and clinical benefits (*eff*) for two or more interventions produces a health economic evaluation. Inputs may be the results of a Bayesian model (possibly based on MCMC) in the form of simulations from the posterior distributions. For *s* sample points compares one of the *m* interventions (*reference*) to the others (*.comparison*).

`bcea(eff, cost, ref, .comparison, interventions)`



`bcea()` calculates numerous cost-effectiveness analysis statistics. These can be called directly, using the constituent functions, but would require some pre-processing which is already handled by `bcea()`.

Value assignment

There are 3 equivalent ways to assign values to analysis parameters.

- In Constructor*: When first creating a `bcea` object.

```
he <- bcea(eff, cost, ref, .comparison, ...)
```

- Using Setters*: Change directly using replacement functions.

```
setComparison(he) <- comparison
setKmax(he) <- Kmax
setReference(he) <- ref
```

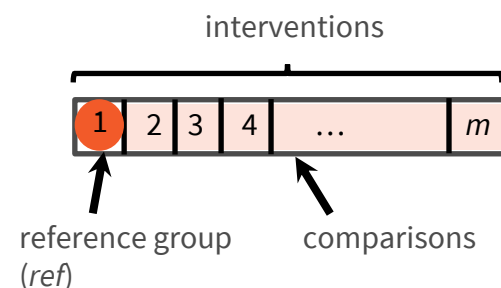
- In plotting call*: At the point of making a plot.

```
eib.plot(he, comparison, ...)
ceac.plot(he, comparison, ...)
ceplane.plot(he, comparison, ...)
```

SELECTING ANALYSIS INTERVENTIONS

Default

The first columns in (*eff, cost*) are the default reference intervention. All other interventions are the comparison interventions unless defined otherwise. E.g. for *m* interventions



Plot

Standard cost-effectiveness analysis output plots. Base R, ggplot2 and plotly versions of plots are available and can be called directly but require extra default parameters.

Expected incremental benefit

`eib.plot(he, comparison = NULL, pos = c(1, 0), size = NULL, plot.cri = NULL, graph = c("base", "ggplot2", "plotly"), ...)`
calls: • `eib_plot_base()`
• `eib_plot_ggplot()`
• `eib_plot_plotly()`

Expected value of information

`evi.plot(he, graph = c("base", "ggplot2", "plotly"), ...)`

Cost-effectiveness planes with contours

`contour[2](he, comparison = 1, scale = 0.5, nlevels = 4, levels = NULL, pos = c(1, 0), xlim = NULL, ylim = NULL, graph = c("base", "ggplot2"), ...)`

Compare optimal scenario to mixed case

`plot.mixedAn(x, y.limits = NULL, pos = c(0, 1), graph = c("base", "ggplot2"), ...)`

Cost-effectiveness acceptability curve

`ceac.plot(he, comparison = NULL, pos = c(1, 0), graph = c("base", "ggplot2", "plotly"), ...)`
calls: • `ceac_plot_base()`
• `ceac_plot_ggplot()`
• `ceac_plot_plotly()`

Cost-effectiveness plane

`ceplane.plot(he, comparison = NULL, pos = c(1, 0), graph = c("base", "ggplot2", "plotly"), ...)`
calls: • `ceplane_plot_base()`
• `ceplane_plot_ggplot()`
• `ceplane_plot_plotly()`

Grid of CE plane, EIB, EVI & CEAC

`plot.bcea(x, comparison = NULL, pos = c(1, 0), graph = c("base", "ggplot2", "plotly"), ...)`

Expected value of perfect partial information

`plot.evppi(x, pos = c(0, 0.8), graph = c("base", "ggplot2"), col = NULL, ...)`

Summarise data

Summary statistics and formatted tables can be used to interrogate a `bcea()` object.

`summary.bcea(he, ...)`

Prints a table with summary results of the health economic evaluation

`summary.mixedAn(he, ...)`

Prints summary table for results of mixed analysis

`sim.table(he, ...)`

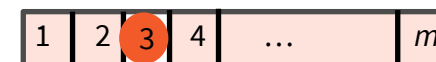
Summary table of simulations from the cost-effectiveness analysis

`make.report(he, ...)`

Constructs the automated report from the output of the BCEA

Set reference

Changing the reference group keeps all the other groups as comparison groups e.g. changing to reference group 3



Set comparisons

Changing the comparison groups keeps the reference group the same e.g. changing to comparison groups 4 to *m*

