02-B-VoI-Casestudy-pop-vs-pred

Sandro Gsteiger

June 11, 2020

Table of Contents

# Description

Use very simple, stylized desision model to assess the relevance of predictive uncertainty with tumour agnostic therapies.

Compare EVPI and EVPPI for two decision rules:

1. Use the population estimates for decision making.
2. Use the predicted outcomes in a new tissue in the decision rule.

# Stylized decision model

Partitioned survival model.

Weibull model for OS and PFS, shape parameter assumed the same. Shape parameter estimated from Larotrectinib OS data. Scale paramters (with uncertainty) estimated from Larotrectinib 12-months survival data (OS and PFS).

Assessment of impact of predictive uncertainty (new tissue type).

Implementation by adapting functions publically available from DARTH group (<https://github.com/DARTH-git/Partitioned-Survival-Analysis>).

# Base case analysis

## Model inputs

* Baseline survival rates (new treatment): OS and PFS rates and their standard errors roughly taken from Laro data
* Hazard ratios to obtain comparator group: assumptions (set to obtain “workable” model for case study, i.e. new intervention reasonnable likelihood of being cost effective)
* Utilities: assumptions based on from lit
  + Pre-pog: smal diff. between treatments
  + Post-progression:
    - no difference between the two groups
    - this was the preferred assumption by the NICE committee in the Laro dossier
    - the manufacturer and some clinical experts said difference in post-prog. utility were plausible
* Costs:
  + New trt, pre-prog:
    - per pERC report about 20k CAN $ per month; -> 12k GBP
    - per NICE ACM2: list price GBP 15k per month (then confidential discount)
  + Ref trt, pre-prog: per pERC about 2k CAN $, though extreme between tissue variability
  + Post-progression state (indep. of trt); assumption
* Discount rate: 3.5% for costs and outcomes per NHS/NICE recommendation
* Time-horizon: 30 years.

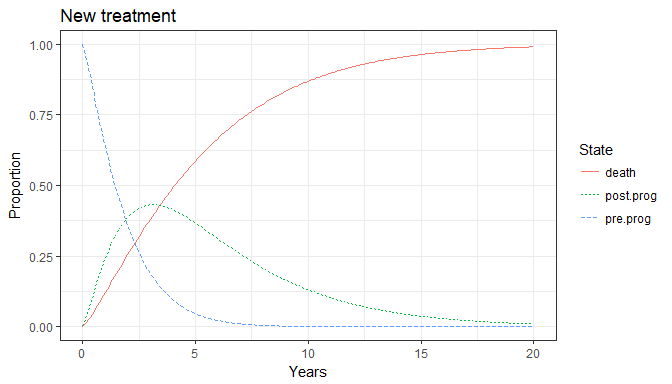
Note that *time scale* is *years*.

model\_pars <- list(  
 shape = 1.2,  
 OS\_scale = 0.128,   
 PFS\_scale = 0.446,  
 HR\_OS = 0.3,  
 HR\_PFS = 0.3,  
 u\_pre\_trt = 0.8, # utility in pre-progrsssion state, new treatment  
 u\_pre\_ref = 0.6, # idem, reference treatment  
 u\_post = 0.45, # utility, post-progression state  
 u\_death = 0,  
 c\_pre\_trt = 2500, # costs in pre-progrsssion state per cycle, new treatment  
 c\_pre\_ref = 1000, # idem, reference treatment  
 c\_post = 300, # costs per cycle, post-progression state  
 c\_death = 0   
)  
  
model\_settings <- list(  
 state\_names = c("Pre-progression", "Post-progression", "Death"),  
 horizon = 30,  
 cycle\_length = 1 / 12, # cycle length 1-month (time scale: years)   
 disc\_rate = 0.035 # rate per year  
)

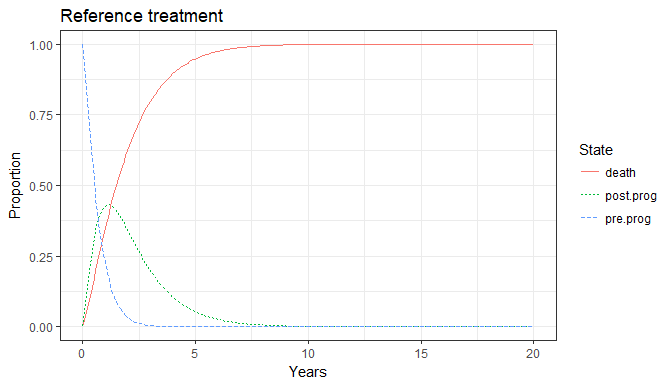
## Economic evaluation

**Figure** Markov traces (base case analysis)

## Warning: Removed 360 rows containing missing values (geom\_path).



## Warning: Removed 360 rows containing missing values (geom\_path).



Calculate total discounted costs and QALYs for the new and the reference treatments.

\*\* Model outputs (base case)\*\*

## new treatment  
ee\_trt

## total.costs total.QALYs   
## 63691.688670 2.679086

## ref treatment  
ee\_ref

## total.costs total.QALYs   
## 1.259134e+04 9.395387e-01

## incremental costs and QALYs  
delta <- ee\_trt - ee\_ref  
delta

## total.costs total.QALYs   
## 51100.349506 1.739548

## ICER  
ICER <- c(costs.per.QALY = delta[[1]] / delta[[2]])  
ICER

## costs.per.QALY   
## 29375.65

# PSA: population model

## Run the PSA and plot the cost-effectiveness plane

Summaries of distributions used for parameter uncertainty.

# OS\_scale parameter, new trt:  
qnorm(mean = 0.128, sd = (0.211 - 0.0513)/6, p = c(0.025, 0.975))

## [1] 0.07583229 0.18016771

# PFS\_scale parameter, new trt:  
qnorm(mean = 0.446, sd = (0.635 - 0.288)/6, p = c(0.025, 0.975))

## [1] 0.3326487 0.5593513

# HR OS, PFS ~ Beta(30, 70) => Expectation: 3/10  
qbeta(30, 70, p = c(0.025, 0.975))

## [1] 0.2146680 0.3929087

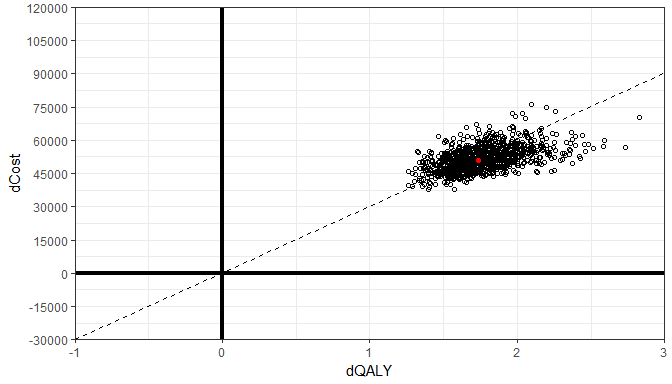
# costs pre-progression state new treatment  
# median: 2500  
qlnorm(meanlog = log(2500), sdlog = 0.05, p = c(0.025, 0.975))

## [1] 2266.626 2757.402

# costs pre-progression state ref treatment  
# median: 1000  
qlnorm(meanlog = log(1000), sdlog = 0.05, p = c(0.025, 0.975))

## [1] 906.6505 1102.9608

## Warning: Removed 1 rows containing missing values (geom\_point).



## Summarize PSA outputs and (re)analyze the PSA outputs with the BCEA package

# PSA outputs  
e <- model\_psa[, c("QALYs.trt", "QALYs.ref")] # ref in bcea terminology is our trt!!  
c <- model\_psa[, c("costs.trt", "costs.ref")] # caution to this !!  
me <- apply(e, MAR = 2, FUN = mean)  
me

## QALYs.trt QALYs.ref   
## 2.7242779 0.9557883

mc <- apply(c, MAR = 2, FUN = mean)  
mc

## costs.trt costs.ref   
## 64667.78 12798.52

diff(mc) / diff(me)

## costs.ref   
## 29329.69

# Compare with basecase analysis  
ee\_trt

## total.costs total.QALYs   
## 63691.688670 2.679086

ee\_ref

## total.costs total.QALYs   
## 1.259134e+04 9.395387e-01

delta

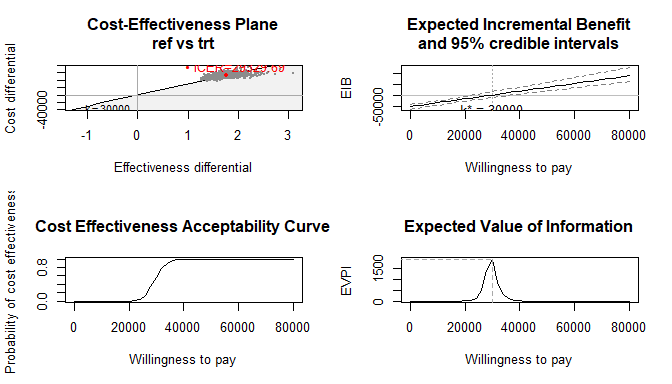
## total.costs total.QALYs   
## 51100.349506 1.739548

ICER

## costs.per.QALY   
## 29375.65

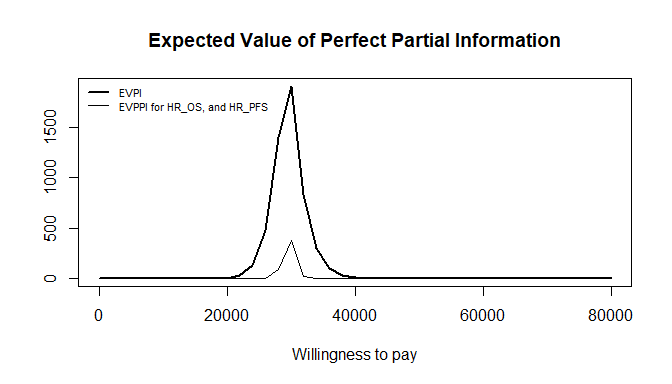
# re-analyze with bcea  
res <- bcea(e, c, interventions = c("ref", "trt"),   
 Kmax = 80000, wtp = seq(0, 80000, 2000))  
  
summary.bcea(res, wtp = 30000)

##   
## Cost-effectiveness analysis summary   
##   
## Reference intervention: ref  
## Comparator intervention: trt  
##   
##   
##   
## Analysis for willingness to pay parameter k = 30000  
##   
## Expected utility  
## ref 17061  
## trt 15875  
##   
## EIB CEAC ICER  
## ref vs trt 1185.4 0.562 29330  
##   
## Optimal intervention (max expected utility) for k=30000: ref  
##   
## EVPI 1912.2

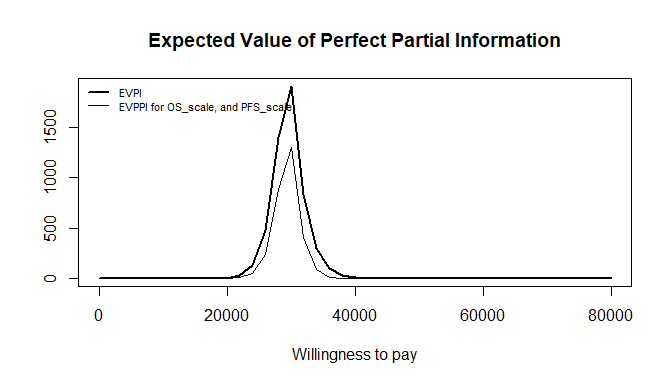


## Calculate EVPPI

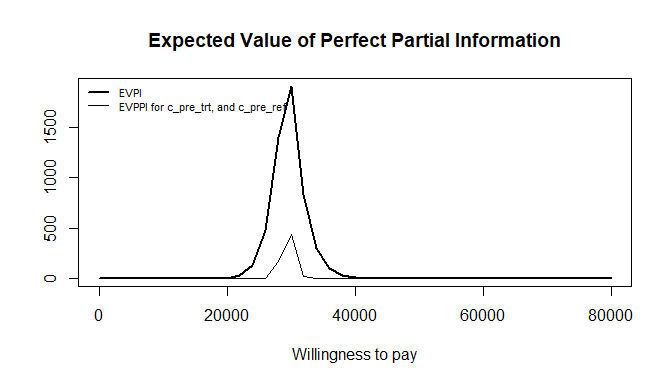
##   
## Calculating fitted values for the GAM regression   
##   
## Calculating fitted values for the GAM regression   
## Calculating EVPPI



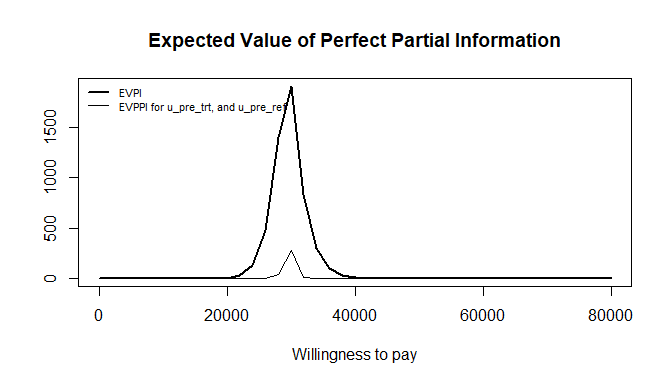
##   
## Calculating fitted values for the GAM regression   
##   
## Calculating fitted values for the GAM regression   
## Calculating EVPPI



##   
## Calculating fitted values for the GAM regression   
##   
## Calculating fitted values for the GAM regression   
## Calculating EVPPI



##   
## Calculating fitted values for the GAM regression   
##   
## Calculating fitted values for the GAM regression   
## Calculating EVPPI



# Redo PSA, but for predictive distribution (new tissue)

Add predictive uncertainty wrt HRs, and costs.

Keep the baseline model (which per se would be expected to vary too!).

## Run the PSA and plot the cost-effectiveness plane

Summaries of distributions used for parameter uncertainty. In particular with respect to reference treatment.

# HR OS, PFS ~ Beta(1.5, 3.5) => Expectation: 3/10, but very uncertain  
qbeta(1.5, 3.5, p = c(0.025, 0.975))

## [1] 0.0284709 0.7162483

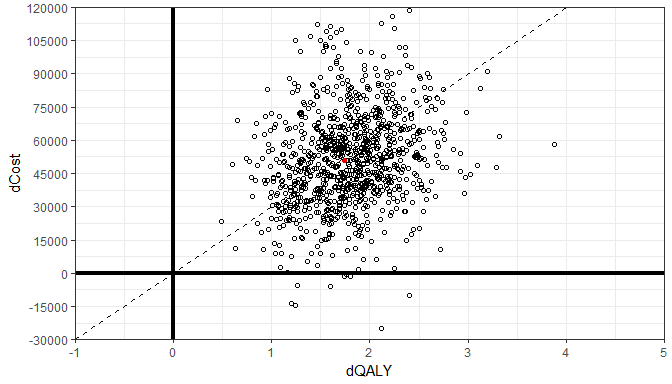
# costs pre-progression state new treatment  
# median: 2500  
qlnorm(meanlog = log(2500), sdlog = 0.3, p = c(0.025, 0.25, 0.75, 0.975))

## [1] 1388.608 2042.029 3060.682 4500.911

# costs pre-progression state ref treatment  
# median: 1000  
qlnorm(meanlog = log(1000), sdlog = 0.75, p = c(0.025, 0.25, 0.75, 0.975))

## [1] 229.9317 602.9824 1658.4233 4349.1177

## Warning: Removed 7 rows containing missing values (geom\_point).



## Summarize PSA outputs and (re)analyze the PSA outputs with the BCEA package

# PSA outputs  
e2 <- model\_psa2[, c("QALYs.trt", "QALYs.ref")] # ref in bcea terminology is our trt!!  
c2 <- model\_psa2[, c("costs.trt", "costs.ref")] # caution to this !!  
me2 <- apply(e2, MAR = 2, FUN = mean)  
me2

## QALYs.trt QALYs.ref   
## 2.7419107 0.9505773

mc2 <- apply(c2, MAR = 2, FUN = mean)  
mc2

## costs.trt costs.ref   
## 66976.90 15621.55

diff(mc2) / diff(me2)

## costs.ref   
## 28668.78

# Compare with basecase analysis  
ee\_trt

## total.costs total.QALYs   
## 63691.688670 2.679086

ee\_ref

## total.costs total.QALYs   
## 12591.3391634 0.9395387

delta

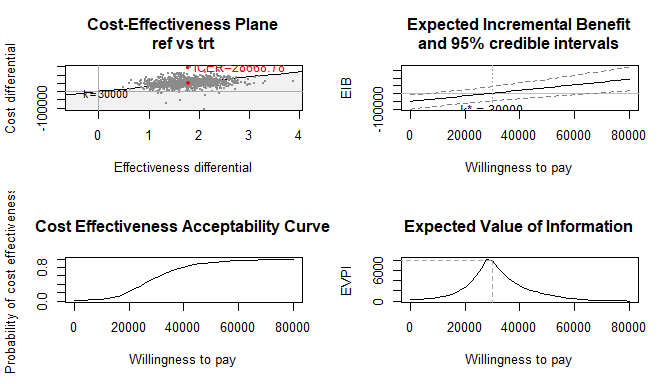
## total.costs total.QALYs   
## 51100.349506 1.739548

ICER

## costs.per.QALY   
## 29375.65

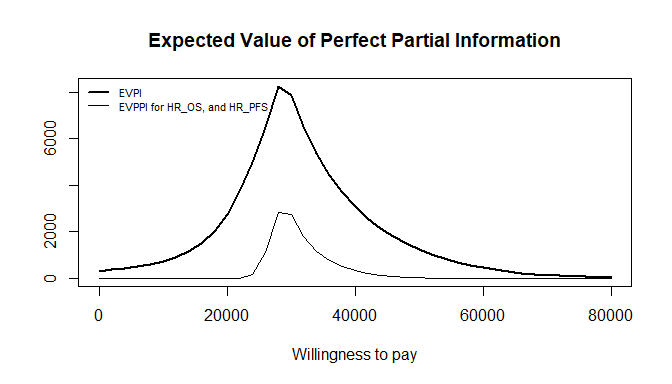
# re-analyze with bcea  
res2 <- bcea(e2, c2, interventions = c("ref", "trt"),   
 Kmax = 80000, wtp = seq(0, 80000, 2000))  
  
summary.bcea(res2, wtp = 30000)

##   
## Cost-effectiveness analysis summary   
##   
## Reference intervention: ref  
## Comparator intervention: trt  
##   
##   
##   
## Analysis for willingness to pay parameter k = 30000  
##   
## Expected utility  
## ref 15280  
## trt 12896  
##   
## EIB CEAC ICER  
## ref vs trt 2384.7 0.557 28669  
##   
## Optimal intervention (max expected utility) for k=30000: ref  
##   
## EVPI 7870

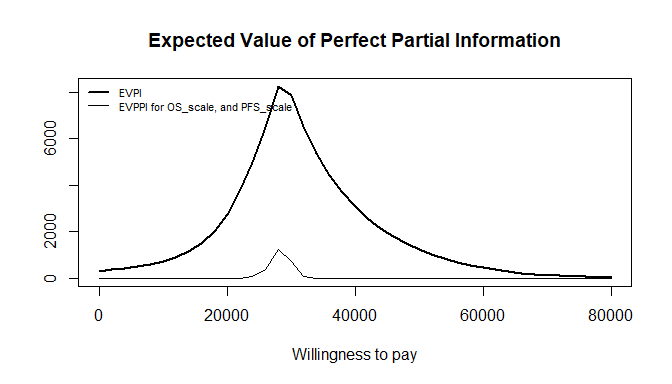


## Calculate EVPPI

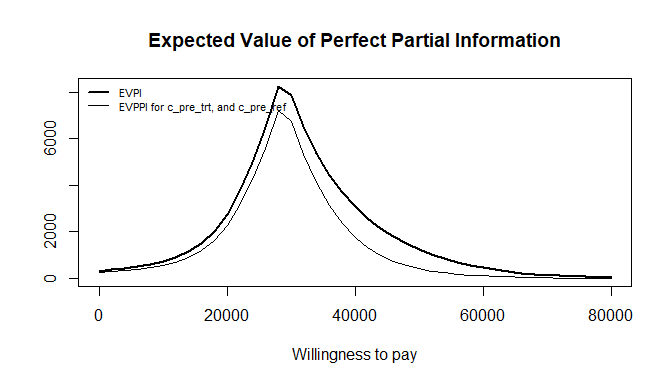
##   
## Calculating fitted values for the GAM regression   
##   
## Calculating fitted values for the GAM regression   
## Calculating EVPPI



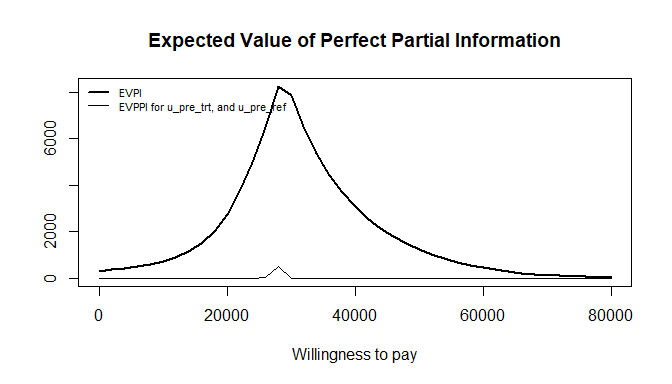
##   
## Calculating fitted values for the GAM regression   
##   
## Calculating fitted values for the GAM regression   
## Calculating EVPPI



##   
## Calculating fitted values for the GAM regression   
##   
## Calculating fitted values for the GAM regression   
## Calculating EVPPI



##   
## Calculating fitted values for the GAM regression   
##   
## Calculating fitted values for the GAM regression   
## Calculating EVPPI



# misc

* Correlation from Michiels et al (2017)
* Heterogeneity: Turner prior
* Heterogeneity: half the Turner prior

# Session info

## [1] "C:/GIT/punta"

## R version 3.4.2 (2017-09-28)  
## Platform: x86\_64-w64-mingw32/x64 (64-bit)  
## Running under: Windows 10 x64 (build 17763)  
##   
## Matrix products: default  
##   
## locale:  
## [1] LC\_COLLATE=English\_United Kingdom.1252   
## [2] LC\_CTYPE=English\_United Kingdom.1252   
## [3] LC\_MONETARY=English\_United Kingdom.1252  
## [4] LC\_NUMERIC=C   
## [5] LC\_TIME=English\_United Kingdom.1252   
##   
## attached base packages:  
## [1] stats graphics grDevices utils datasets methods base   
##   
## other attached packages:  
## [1] BCEA\_2.3-1.1 ggplot2\_2.2.1 tidyr\_0.8.3 dplyr\_0.8.0.1 readxl\_1.3.1   
##   
## loaded via a namespace (and not attached):  
## [1] Rcpp\_1.0.1 knitr\_1.28 magrittr\_1.5 munsell\_0.4.3   
## [5] tidyselect\_0.2.5 lattice\_0.20-35 colorspace\_1.3-2 R6\_2.2.2   
## [9] rlang\_0.3.4 plyr\_1.8.4 stringr\_1.4.0 tools\_3.4.2   
## [13] grid\_3.4.2 nlme\_3.1-131 gtable\_0.2.0 mgcv\_1.8-20   
## [17] xfun\_0.12 htmltools\_0.3.6 lazyeval\_0.2.1 yaml\_2.2.0   
## [21] digest\_0.6.12 assertthat\_0.2.0 tibble\_2.1.1 crayon\_1.3.4   
## [25] Matrix\_1.2-11 purrr\_0.3.2 glue\_1.3.1 evaluate\_0.14   
## [29] rmarkdown\_2.1 labeling\_0.3 stringi\_1.4.3 compiler\_3.4.2   
## [33] pillar\_1.4.3 cellranger\_1.1.0 scales\_0.5.0 pkgconfig\_2.0.3