Simulating Bayesian Operating Characteristics of Bayesian Sequential RCT

Frank Harrell
Department of Biostatistics
Vanderbilt University School of Medicine

2021-02-20

Contents

Introduction **1**
Technical Note 2

Simulation Parameters **2**

Simulation to Estimate Probabilities of Stopping for Different Reasons **3**

Simulation to Check Reliability of Evidence at Moment of Stopping **8**

Futility Analysis **11**

Operating Characteristics With Less Frequent Looks **15**

Summary **25**

More Information **26**

Computing Environment **26**

Introduction

There are many types of operating characteristics to consider for a statistical assessment of evidence for treatment effects, for example

1. Bayesian or frequentest power or frequentist type I assertion probability if the study goes to its planned end and there is only one look at the data at that time. This is a pre-study tendency.
2. Pre-study tendencies for various successes or failures to happen sometime during the course of the study if the data are analyzed sequentially. In the frequentist setting, in this case is the probability of *ever* rejecting H0 when the treatment is ignorable (no benefit and no harm).
3. Reliability of post-study evidence quantification when there is a single data look
4. Reliability of the evidence quantification at the decision point in a sequential study. I.e., at the first data look at which an evidence threshold is exceeded what is the reliability of a frequentist null hypothesis test at that look or of a Bayesian posterior probability at that look.
5. Properties of a futility assessment that is done in mid course, e.g., of a frequentist conditional power calculation or a Bayesian posterior predictive probability of ultimate study success

Once the study has begun and the first analysis is done, purely pre-study operating characteristics such as 1. and 2. are no longer relevant. In this report we are concerned with 2. and 4.

Consider a two-treatment arm therapeutic randomized controlled trial (RCT) for assessing treatment, with 1:1 randomization ratio and a univariate ordinal outcome. Though primary analyses should always be covariate-adjusted we will not consider covariate adjustment in these simulations. The trial design considered here is a fully sequential one with unlimited data looks so as to minimize the expected sample size. We will estimate the probability that the posterior probability, under various priors, will reach given targets for efficacy and harm/inefficacy, as a function of the sequential look number. These operating characteristics are useful for study planning, especially with regard to avoiding a futile trial from the outset. Once data are available, however, the simulations in the last part of this document are the most important. This second set of simulations addresses the reliability of evidence at the decision point, i.e., at the moment of stopping for efficacy, inefficacy, or harm.

For simplicity we align the looks with completion of follow-up of patients in the trial, e.g., a look after the 50th patient is followed will assume that patients 1-49 have also completed follow-up.

The simulations presented here assume that a normal distribution is an adequate approximation to the distribution of the treatment effect estimate, here a treatment B : treatment A log odds ratio. The data model is a proportional odds ordinal logistic model analyzed with the R `rms` package `lrm` function. Simulations are done by the `Hmisc` package `estSeqSim` and `gbayesSeqSim` functions. As implemented in the R `Hmisc` package gbayes function, the Bayesian posterior distribution used to approximate (thus avoiding another simulation loop for MCMC posterior draws) the real posterior distribution is normal, making for very quick approximate posterior probability calculations when the prior distribution used for the log OR is also Gaussian as used here. Three priors are used:

- a skeptical prior for assessing evidence for efficacy
- a flat prior for assessing evidence for efficacy
- a flat prior for studying posterior probabilities of inefficacy/harm
- an optimistic prior for inefficacy/harm

See here for more simulations and graphical presentations of them.

Technical Note

Our simulations involve simulating ordinal data for the planned maximum sample size and then progressively revealing more and more of this sample as sequential looks progress. As with a real study, the data in the sequential looks overlap and this needs to be taken into account for certain analyses of posterior probability paths.

Unlike comparing means with a two-sample t-test, the formula for the variance of a log odds ratio is not a simple function of the group sample sizes. So one cannot compute the variance of a log odds ratio computed on one patient from treatment A and one from treatment B then divide that variance by the number of A-B patient pairs to get the variance of a log odds ratio for an arbitrary number of patients. Were that not the case, our simulations would have been even faster because one could simulate from a normal distribution instead of actually fitting the logistic model for each sample and look.

Simulation Parameters

Unlike here, we take treatment benefit to mean that the odds ratio (OR) < 1. Since a large number of true ORs are used in the simulation, we simulate only 500 clinical trials per OR and borrow information across ORs to compute Bayesian operating characteristics. ORs will vary from benefit to harm over this sequence: 0.4, 0.45, ..., 1.25. The maximum sample size will be 1000 patients, with the first look taken at the 25th patient. Look every patient until 100 patients, then for faster simulations look only every 5 patients until a maximum sample size of 1000. Posterior probabilities will be computed for the following assertion and prior distribution combinations:

- Efficacy: P(OR < 1) with a skeptical prior P(OR < 0.5) $= 0.025$
- Efficacy: P(OR < 1) with a flat prior (log(OR) has mean 0 and SD 100)

- Inefficacy/harm: P(OR > 1) with a flat prior (log(OR) has mean 0 and SD 100)
- Inefficacy/harm: P(OR > 1) with an optimistic prior (log(OR) has mean log(0.85) = -0.1625 and SD of 0.5)

For computing the probabilities of hitting various targets, the following posterior probability thresholds are used:

- Efficacy: P(OR < 1) > 0.95
- Inefficacy/harm: P(OR > 1) > 0.9

Simulation to Estimate Probabilities of Stopping for Different Reasons

First define functions needed by the `Hmisc` package `estSeqSim` and `gbayesSeqSim` functions.

```
# Use a low-level logistic regression call to speed up simulations

lfit <- function(x, y) {
    # lrm.fit.bare will be in version 6.1-0 of rms; until then use lrm.fit
  f <- rms::lrm.fit.bare(x, y)
  cof <- f$coefficients
  k <- length(cof)
  c(cof[k], f$var[k, k])
}

# Data generation function
gdat <- function(beta, n1, n2) {
  # Cell probabilities for a 7-category ordinal outcome for the control group
  p <- rev(c(2, 1, 2, 7, 8, 38, 42) / 100)

  # Compute cell probabilities for the treated group
  p2 <- pomodm(p=p, odds.ratio=exp(beta))
  y1 <- sample(1 : 7, n1, p,  replace=TRUE)
  y2 <- sample(1 : 7, n2, p2, replace=TRUE)
  list(y1=y1, y2=y2)
}
```

Now run the simulations. The result is saved so that the model fits are not run each time this report is compiled.

```
ors   <- seq(0.4, 1.25, by=0.05)
looks <- c(25 : 100, seq(105, 1000, by=5))

if(! file.exists('seqsim.rds')) {
  set.seed(3)
  # estSeqSim took 38m for 2.3M model fits; plan on 0.001s per OR/simulation/look
  print(system.time(est <- estSeqSim(log(ors), looks, gdat, lfit, nsim=500, progress=TRUE)))
  saveRDS(est, 'seqsim.rds', compress='xz')
} else est <- readRDS('seqsim.rds')

# Define assertions and priors to be used for them
# Assertion 1: log(OR) < 0 under prior with prior mean 0 and sigma: P(OR>2)=0.025
# Assertion 2: log(OR) < 0 under flat prior
# Assertion 3: log(OR) > 0 under flat prior (sigma=100)
# Assertion 4: log(OR) > 0 under optimistm prior with mean log(0.85), sigma=0.5
```

```
asserts <- list(list('Efficacy',                   '<', 0, cutprior=log(2), tailprob=0.025),
                list('Efficacy flat',              '<', 0, mu=0,         sigma=100),
                list('Inefficacy/harm flat',       '>', 0, mu=0,         sigma=100),
                list('Inefficacy/harm optimistic', '>', 0, mu=log(0.85), sigma=0.5))

s <- gbayesSeqSim(est, asserts=asserts)

head(s)
```

```
  sim  parameter look        est      vest        p1       mean1       sd1
1   1 -0.9162907   25 -0.5608048 0.6702457 0.6070526 -0.08819149 0.3246568
2   1 -0.9162907   26 -0.7216567 0.6351371 0.6432983 -0.11872801 0.3232548
3   1 -0.9162907   27 -0.5308043 0.6012535 0.6118197 -0.09140265 0.3217666
4   1 -0.9162907   28 -0.6291448 0.5922355 0.6335884 -0.10969854 0.3213456
5   1 -0.9162907   29 -0.7187434 0.5847377 0.6534119 -0.12664488 0.3209870
6   1 -0.9162907   30 -0.8010207 0.5784064 0.6715144 -0.14241269 0.3206779
         p2      mean2       sd2        p3      mean3       sd3        p4
1 0.7533229 -0.5607672 0.8186579 0.2466771 -0.5607672 0.8186579 0.2628995
2 0.8173968 -0.7216109 0.7969296 0.1826032 -0.7216109 0.7969296 0.2246521
3 0.7531798 -0.5307724 0.7753821 0.2468202 -0.5307724 0.7753821 0.2597402
4 0.7931801 -0.6291075 0.7695456 0.2068199 -0.6291075 0.7695456 0.2363885
5 0.8263650 -0.7187014 0.7646591 0.1736350 -0.7187014 0.7646591 0.2158080
6 0.8538774 -0.8009744 0.7605084 0.1461226 -0.8009744 0.7605084 0.1976082
       mean4       sd4
1 -0.2707199 0.4267123
2 -0.3204430 0.4235439
3 -0.2706787 0.4202129
4 -0.3010270 0.4192764
5 -0.3291055 0.4184808
6 -0.3552087 0.4177965
```

```
attr(s, 'asserts')
```

```
                     label  cutprior tailprob         mu      sigma assertion
1                  Efficacy 0.6931472    0.025  0.0000000   0.353653       < 0
2              Efficacy flat        NA       NA  0.0000000 100.000000       < 0
3       Inefficacy/harm flat        NA       NA  0.0000000 100.000000       > 0
4 Inefficacy/harm optimistic        NA       NA -0.1625189   0.500000       > 0
```

```
alabels <- attr(s, 'alabels')  # named vector to map p1 p2 p3 p4 to labels
```

First let's examine the effect of the priors by making two pairwise comparisons: differences in posterior probabilities of efficacy under skeptical vs. flat prior, and differences in posterior probabilities of inefficacy under flat and optimistic priors.

```
w <- data.table(s)
u <- w[, .(p12max=max(abs(p1 - p2)), p12mean=mean(abs(p1 - p2)),
           p34max=max(abs(p3 - p4)), p34mean=mean(abs(p3 - p4))), by=.(look)]
z  <- melt(u, measure.vars=c('p12max', 'p12mean', 'p34max', 'p34mean'),
           variable.name='which', value.name='diff')
k <- c(p12max='Efficacy max',   p12mean='Efficacy mean',
       p34max='Inefficacy max', p34mean='Inefficacy mean')
z[, w := k[which]]
ggp(ggplot(z, aes(x=look, y=diff, color=w)) + geom_line() +
    guides(color=guide_legend(title='Comparison')) +
    xlab('Look') + ylab('Absolute Difference in Posterior Probabilities'))
```

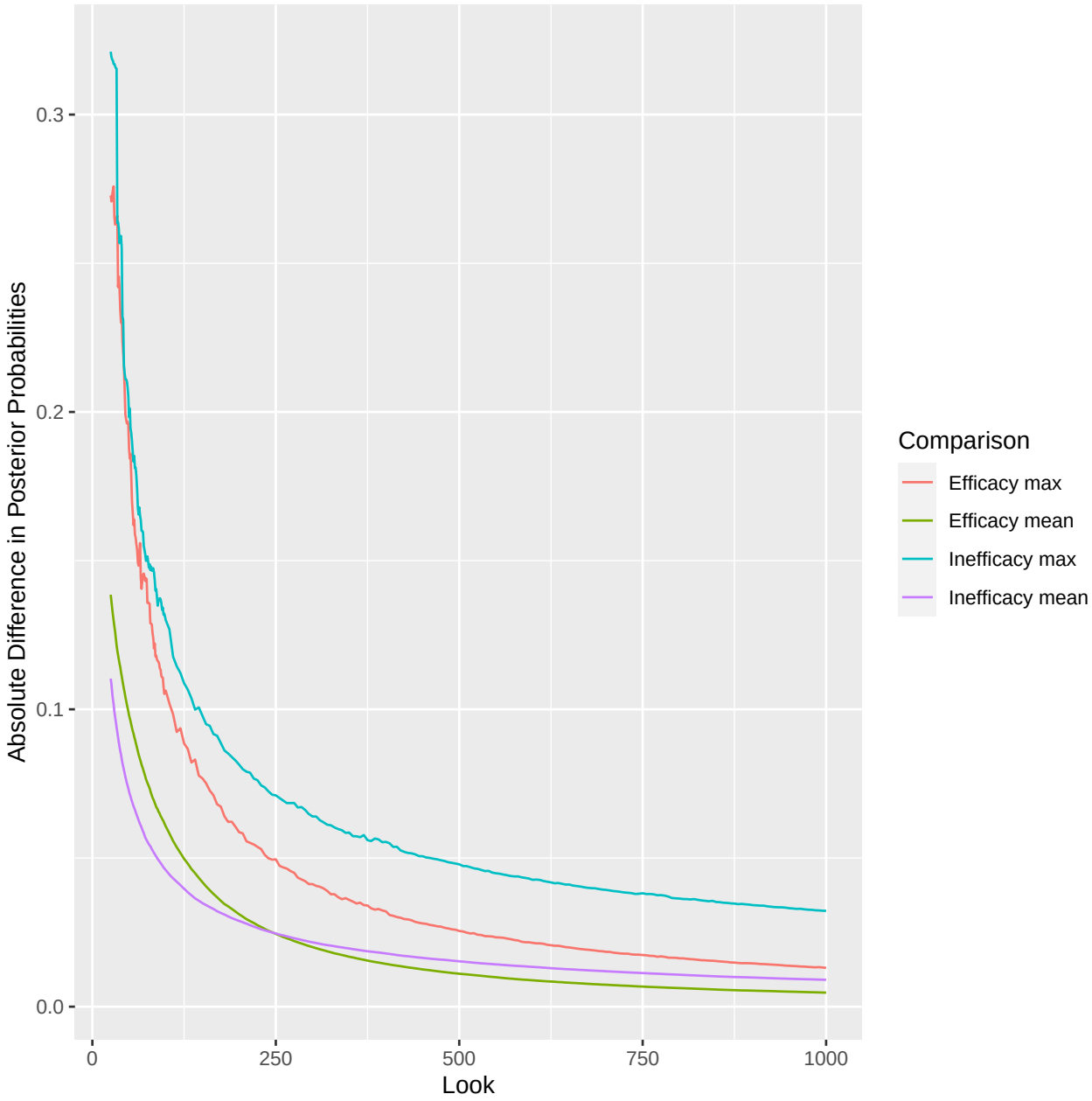

Figure 1: Mean and maximum absolute difference in posterior probabilities for efficacy and inefficacy to gauge the effect of using a skeptical prior instead of a flat prior

Compute the cumulative probability of hitting assertion-specific targets as data looks advance.

```
# Reshape results into taller and thinner data table so can plot over 3 assertions

ps <- names(alabels)
w  <- melt(w,
           measure.vars=list(ps, paste0('mean', 1:4), paste0('sd', 1:4)),
           variable.name='assert', value.name=c('p', 'mean', 'sd'))
w[, assert := alabels[assert]]
```

```
head(w)
```

```
   sim  parameter look        est      vest   assert         p        mean
1:   1 -0.9162907   25 -0.5608048 0.6702457 Efficacy 0.6070526 -0.08819149
2:   1 -0.9162907   26 -0.7216567 0.6351371 Efficacy 0.6432983 -0.11872801
3:   1 -0.9162907   27 -0.5308043 0.6012535 Efficacy 0.6118197 -0.09140265
4:   1 -0.9162907   28 -0.6291448 0.5922355 Efficacy 0.6335884 -0.10969854
5:   1 -0.9162907   29 -0.7187434 0.5847377 Efficacy 0.6534119 -0.12664488
6:   1 -0.9162907   30 -0.8010207 0.5784064 Efficacy 0.6715144 -0.14241269
          sd
1: 0.3246568
2: 0.3232548
3: 0.3217666
4: 0.3213456
5: 0.3209870
6: 0.3206779
```

```
# Define targets
target <- c(Efficacy                    = 0.95,
            'Efficacy flat'             = 0.95,
            'Inefficacy/harm flat'      = 0.9,
            'Inefficacy/harm optimistic' = 0.9)

w[, target := target[assert]]     # spreads targets to all rows
# hit = 0/1 indicator if hitting target at or before a certain look equals
# 1 if cumulative number of target hits > 0 at that look
u <- w[, .(hit = 1*(cumsum(p > target) > 0), look=look), by=.(sim, parameter, assert)]

# To compute for each assertion/prior/parameter combination the
# first look at which the target was hit run the following
# (first is set to infinity if never hit)
# f <- w[, .(first=min(look[p > target])), by=.(sim, parameter, assert)]
# with(subset(f, assert == 'Efficacy' & parameter == 0), mean(is.finite(first)))
```

Next estimate the probability of hitting the posterior probability targets as a function of the true parameter generating the data, the look number, and the assertion. We use simple stratified proportions for this purpose because of an adequate number of replications per condition. (With fewer replications we would have used logistic regression to interpolate probability estimates for more precision.) Estimates are made for a subset of the true odds ratios.

```
# Estimate the probability at OR 0.4 0.5 ... using simple proportions
ors  <- round(exp(u$parameter), 2)
us   <- u[ors == round(ors, 1), ]   # subset data table with OR incremented by 0.1
prop <- us[, .(p=mean(hit)), by=.(parameter, look, assert)]
prop <- prop[, txt := paste0(assert, '<br>OR:', exp(parameter), '<br>n:', look, '<br>', round(p, 3))]
ggp(ggplot(prop, aes(x=look, y=p, color=factor(exp(parameter)), label=txt)) + geom_line() +
  facet_wrap(~ assert) +
  xlab('Look') + ylab('Proportion Stopping') +
  guides(color=guide_legend(title='True OR')) +
  theme(legend.position = 'bottom'), tooltip='label')
prp <- prop[assert == 'Efficacy', ]
lookup <- function(n=c(500, 1000), param=log(c(1, 0.7))) {
    r <- numeric(length(n) * length(param))
    i <- 0
    for(N in n) {
```

```
        for(pa in param) {
            i <- i + 1
            r[i] <- prp[look == N & abs(parameter - pa) < 1e-5, p]
        }
    }
    r
}
Sm <- rbind(Sm,
                        data.frame(topic='Power to detect efficacy',
                                               cond1  = 'almost unlimited looks',
                                               cond2  = 'no futility stopping',
                                               cond3  = c(rep('n=500', 2), rep('n=1000', 2)),
                                               cond4  = rep(c('OR=1', 'OR=0.7'), 2),
                                               amount = lookup(n=c(500, 1000))))
```

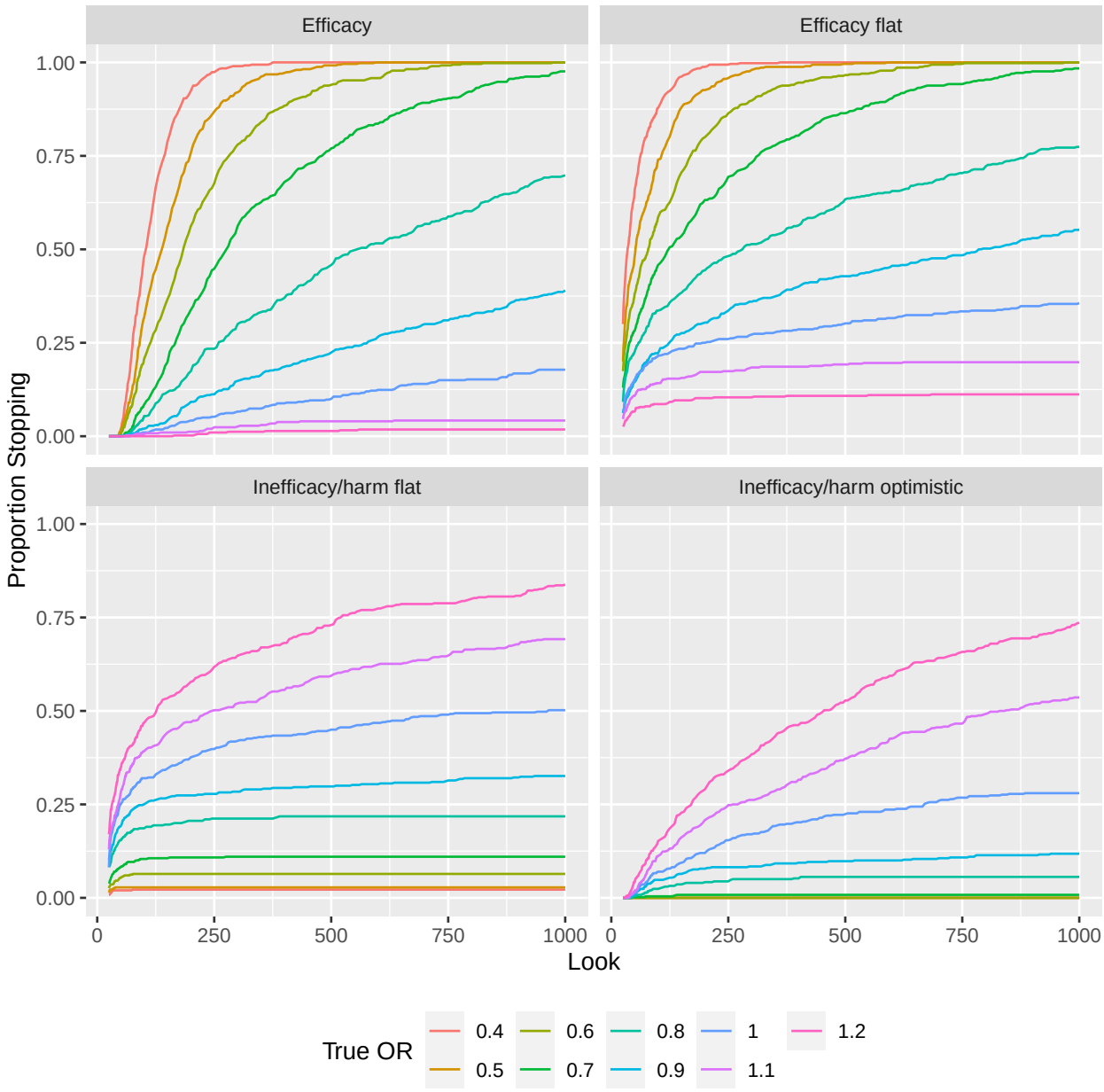

Figure 2: Bayesian power: Probability of stopping early with sufficient evidence for an effect, or the probability of obtaining that evidence at the last look where n=1000, as a function of the true unknown effect (OR)

Simulation to Check Reliability of Evidence at Moment of Stopping

The simulations above addressed the question of what might have happened regarding trial stoppage under specific true unknown treatment effects. Now we address the more important question: at the moment of stopping how reliable is the current evidence that was used to make the decision to stop? To do this, as was done here we simulate clinical trials more in alignment with how the real world works. We take a sample of true effects and run one clinical trial for each true effect, i.e., one parameter value that is taken to define the data generating process (in addition to the base cell probabilities for the 7-level ordinal outcome). Were the effects drawn from the same prior distribution (log normal distribution for the OR) as the one used during data analysis, the posterior probability at the moment of stopping would mathematically have to be perfectly calibrated. Here we simulate from a slightly different distribution that is less skeptical.

The goal of Bayesian analysis is to uncover the unknown parameters that generated our data. So a simulation that is in the Bayesian spirit is one in which we run a randomized trial for each drawn value of the treatment effect, and we run one trial (`nsim=1`) for that value. We sample 3000 log odds ratios from a normal distribution so are running 3000 clinical trials.

```
# Log odds ratios drawn from a normal distribution with mean 0 and SD 0.5
# Analysis prior for efficacy used 0 and SD 0.354

set.seed(4)
nor <- 3000
lors <- rnorm(nor, 0, 0.5)
looks <- c(25 : 100, seq(105, 1000, by=5))

if(! file.exists('seqsim2.rds')) {
  set.seed(5)
  # estSeqSim took 10.5m for 768,000 model fits
  print(system.time(est2 <- estSeqSim(lors, looks, gdat, lfit, nsim=1)))
  saveRDS(est2, 'seqsim2.rds', compress='xz')
} else est2 <- readRDS('seqsim2.rds')
s <- gbayesSeqSim(est2, asserts=asserts)
head(s)
```

```
  sim parameter look         est      vest        p1        mean1       sd1
1   1 0.1083774   25  0.09012818 0.5629420 0.4795766  0.016383966 0.3198976
2   1 0.1083774   26 -0.10355546 0.5392905 0.5243936 -0.019495018 0.3186300
3   1 0.1083774   27  0.01954542 0.5275928 0.4953008  0.003745507 0.3179671
4   1 0.1083774   28 -0.14519033 0.5095408 0.5359743 -0.028614400 0.3168932
5   1 0.1083774   29 -0.18859189 0.4992482 0.5475463 -0.037780822 0.3162514
6   1 0.1083774   30 -0.33614215 0.4845743 0.5865646 -0.068960571 0.3152965
         p2       mean2       sd2        p3       mean3       sd3        p4
1 0.4521939  0.09012310 0.7502735 0.5478061  0.09012310 0.7502735 0.4192291
2 0.5560690 -0.10354988 0.7343442 0.4439310 -0.10354988 0.7343442 0.3639062
3 0.4892665  0.01954439 0.7263367 0.5107335  0.01954439 0.7263367 0.4003356
4 0.5805863 -0.14518293 0.7138031 0.4194137 -0.14518293 0.7138031 0.3508907
5 0.6052282 -0.18858247 0.7065574 0.3947718 -0.18858247 0.7065574 0.3374241
6 0.6854068 -0.33612586 0.6960969 0.3145932 -0.33612586 0.6960969 0.2926359
        mean4       sd4
1 -0.08482362 0.4160754
2 -0.14384283 0.4132977
3 -0.10398433 0.4118540
4 -0.15681529 0.4095283
5 -0.17121864 0.4081459
6 -0.22160867 0.4060994
```

```
attr(s, 'asserts')
```

```
                      label  cutprior tailprob         mu      sigma assertion
1                   Efficacy 0.6931472    0.025  0.0000000   0.353653       < 0
2              Efficacy flat        NA       NA  0.0000000 100.000000       < 0
3       Inefficacy/harm flat        NA       NA  0.0000000 100.000000       > 0
4 Inefficacy/harm optimistic        NA       NA -0.1625189   0.500000       > 0
```

For each assertion/evidence target subset to studies that ever met the target. Save the posterior probability at the moment at which the target was hit. Compare the average of such probabilities with the proportion of such conclusive studies for which the true unknown parameter value generating the study's data met the criterion. For example, when considering evidence for efficacy as $\text{P(OR} < 1) = \text{P}(\log(\text{OR}) < 0) > 0.95$, find

the look at which P first crossed 0.95 and save its posterior probability (which will overshoot the 0.95 a bit) and compute the proportion of true effects (parameter values) for these "early stopping for efficacy" studies that are negative (to estimate the probability of a true treatment benefit).

```
# Function to compute the first value of x that exceeds a threshold
g <- function(x, threshold) {
    i <- 1 : length(x)
    first <- min(which(x > threshold))
    if(! is.finite(first)) return(Inf)
    x[first]
}
w <- data.table(s)
ppstop  <- w[, .(pp=g(p1, 0.95), first=min(look[p1 > 0.95])), by=.(parameter)]
```

```
med     <- ppstop[is.finite(first), median(first)]
pavg    <- ppstop[is.finite(pp),    mean(pp)]
pactual <- ppstop[is.finite(pp),    mean(parameter < 0)]
rnd <- function(x) round(x, 3)
rnd(c(pavg, pactual))
```

```
[1] 0.955 0.977
```

We see that the average posterior probability of efficacy at the moment of stopping for this probability exceeding 0.95 was 0.955 and the estimated true probability of efficacy of 0.977 was even higher than that. So even though we stopped with impressive evidence for any efficacy, the evidence was actually understated under this simulation model. Had the data generating prior been as skeptical as the analysis prior, the two quantities would be identical.

The median look at the time of stopping for efficacy for those studies every reaching P > 0.95 was 205. Examine the estimated true probability of efficacy for those stopped later than this vs. those stopped earlier.

```
ppstop[is.finite(pp), .(proportionTrueEfficacy=mean(parameter < 0)),
       by=.(belowMedian=first <= med)]
```

```
   belowMedian proportionTrueEfficacy
1:        TRUE              0.9868421
2:       FALSE              0.9673540
```

The estimate of the true probability of efficacy is larger when stopping at or before observation 205. For stopping after 205, the effect of prior distribution used in the analysis starts to wear off.

Now evaluate calibration of posterior probabilities used for stopping early for inefficacy/harm.

```
# Note that each study has a unique parameter since sampled from a continuous dist.
ppstop   <- w[, .(pp=g(p3, 0.9), first=min(look[p3 > 0.9])), by=.(parameter)]
```

```
pavg3    <- ppstop[is.finite(pp), mean(pp)]
pactual3 <- ppstop[is.finite(pp), mean(parameter > 0)]
rnd(c(pavg3, pactual3))
```

```
[1] 0.924 0.846
```

```
ppstop  <- w[, .(pp=g(p4, 0.9), first=min(look[p4 > 0.9])), by=.(parameter)]
```

```
pavg4    <- ppstop[is.finite(pp), mean(pp)]
pactual4 <- ppstop[is.finite(pp), mean(parameter > 0)]
rnd(c(pavg4, pactual4))
```

```
[1] 0.910 0.932
```

Recall that we used different priors for assessing evidence for inefficacy/harm, to be careful about stopping early for inefficacy. Especially for the last situation, the analysis priors are more optimistic about the true treatment effect than the distribution used for the population of efficacies we simulated from.

We see that the average posterior probability of inefficacy at the moment of stopping for this probability exceeding 0.9 was 0.924, and the estimated true probability of inefficacy was 0.846 when using a flat prior for analysis. When using an optimistic prior (negative mean on the log odds ratio scale), the average probability of inefficacy at the moment of stopping for inefficacy was 0.91 and the estimated true probability of inefficacy was 0.932.

Futility Analysis

Typical reasons for stopping a clinical trial early are

1. sufficient evidence for efficacy
2. sufficient evidence for inefficacy/harm
3. futility

Futility refers to it being unlikely that even if the study were to proceed to its maximum planned sample size, the probability is low that 1. or 2. would obtain. Futility can be assessed in a formal way using Bayesian posterior predictive distributions, which take into account the limitations of evidence at the time that futility is assessed as well as uncertainty about the future data that have yet to be observed. Futility analysis is the only setting where the planned sample size is considered, when using a Bayesian sequential design.

Since we are simulating many clinical trials and are progressively revealing the data up to a planned maximum sample size of 1000 patients, we can easily simulate futility guidance at a certain point in time. For simplicity we consider only 1. above, i.e., we assess futility in such a way as to only consider evidence for efficacy a success and ignore the possibility that we may wish for the trial to provide definitive evidence that a treatment is ineffective or harmful. Suppose that we want to assess futility after 400 patients have completed follow-up, and the ultimate sample size cannot go above 1000. We start with the subset of studies that have not been stopped early for either efficacy or inefficacy. For those studies, record the posterior probability of efficacy at the 400 patient mark. Relate that posterior probability to the probability that a later posterior probability will exceed 0.95, estimating that probability as a function of the 400-patient probability of efficacy using logistic regression on the binary outcome of posterior probability ever exceeding 0.95.

We continue to use the second set of simulated trials where the true unknown efficacy odds ratios come from a log normal distribution.

```
# For each trial compute the lowest sample size at which the efficacy or inefficacy target
# was hit.  Then keep those trials for which it was not hit by 400 patients
u   <- w[, .(first = min(look[p1 > 0.95 | p3 > 0.9]), look=look, p1=p1, p3=p3),
         by=.(parameter)]
```

```
v <- u[look >= 400 & first > 400, ]
length(v[, unique(parameter)])    # number of RCTs remaining
```

```
[1] 667
```

```
# For each simulated study find the first look at which efficacy target hit
# R sets the computed sample size to infinity if the target was never hit
ef <- v[, .(efirst = min(look[p1 > 0.95]), pcurrent=p1[look == 400]), by=.(parameter)]
```

```
dd <- datadist(ef); options(datadist='dd')
f <- lrm(is.finite(efirst) ~ rcs(qlogis(pcurrent), 4), data=ef)
f
```

Logistic Regression Model

```
lrm(formula = is.finite(efirst) ~ rcs(qlogis(pcurrent), 4), data = ef)
```

{

		Model Likelihood Ratio Test		Discrimination Indexes		Rank Discrim. Indexes	
Obs	667	LR χ^2	297.80	R^2	0.505	C	0.879
FALSE	455	d.f.	3	g	2.354	D_{xy}	0.758
TRUE	212	$\Pr(>\chi^2)$	<0.0001	g_r	10.527	γ	0.758
$\max \lvert \frac{\partial \log L}{\partial \beta} \rvert$	4×10^{-5}			g_p	0.329	τ_a	0.329
				Brier	0.129		

}

%latex.default(U, file = ”“, first.hline.double = FALSE, table = FALSE, longtable = TRUE, lines.page = lines.page, col.just = rep(”r“, ncol(U)), rowlabel =””, already.math.col.names = TRUE, append = TRUE)%

	$\hat{\beta}$	S.E.	Wald Z	$\Pr(>\lvert Z \rvert)$
Intercept	-2.4648	0.5814	-4.24	<0.0001
pcurrent	1.6124	0.9980	1.62	0.1062
pcurrent’	0.7740	2.2207	0.35	0.7274
pcurrent”	-3.2691	6.4875	-0.50	0.6143

```
P <- Predict(f, pcurrent, fun=plogis)
xl <- 'P(OR < 1) at 400th Patient'
yl <- 'P(P(OR < 1) > 0.95) by 1000th Patient'
switch(outfmt,
             pdf  = ggplot(P, xlab=xl, ylab=yl),
             html = plotp(P, xlab=xl, ylab=yl))
```

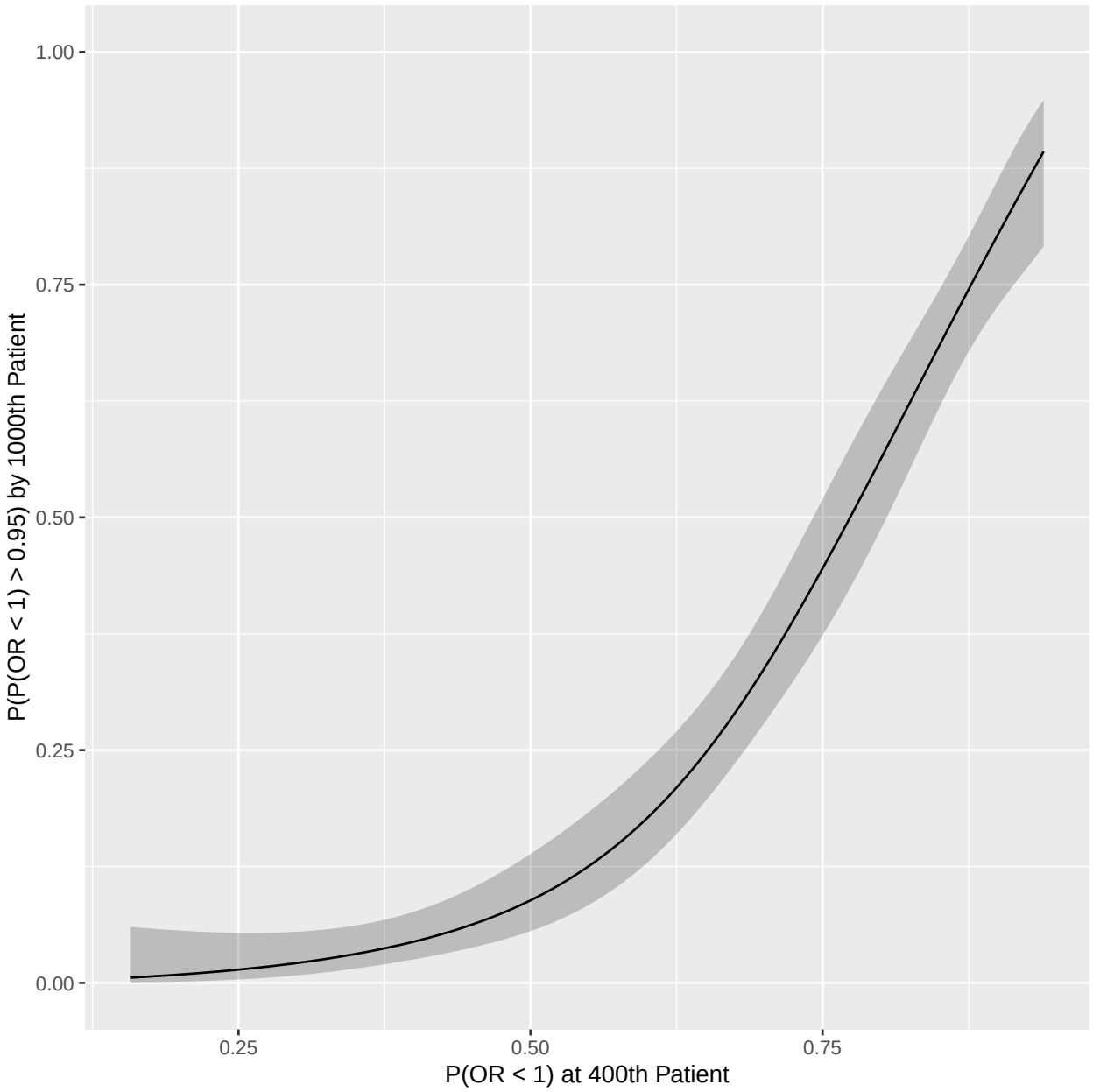

Figure 3: Logistic model estimates of the probability having sufficient evidence for efficacy by the planned study end, given the posterior probability of efficacy at n=400

For there to be a 0.5 chance of hitting efficacy evidence target by the planned study end, the posterior probability of efficacy after 400 patients would have to exceed 0.75. For at least a 0.125 chance of finding efficacy, the posterior probability at 400 would have to exceed 0.5, which is almost the same as saying that the treatment effect needs to be pointing in the right direction. To have at least a 0.25 chance of finding efficacy, the current posterior probability would have to exceed 0.625.

Now repeat the calculations from the standpoint of the futility assessment being made after 700 patients have completed follow-up.

```
# For each trial compute the lowest sample size at which the efficacy or inefficacy target
# was hit.  Then keep those trials for which it was not hit by 700 patients
```

```
v <- u[look >= 700 & first > 700, ]
length(v[, unique(parameter)])    # number of RCTs remaining
```

```
[1] 420
```

```
# For each simulated study find the first look at which efficacy target hit
ef <- v[, .(efirst = min(look[p1 > 0.95]), pcurrent=p1[look == 700]), by=.(parameter)]
```

```
dd <- datadist(ef); options(datadist='dd')
f <- lrm(is.finite(efirst) ~ rcs(qlogis(pcurrent), 4), data=ef)
f
```

Logistic Regression Model

```
lrm(formula = is.finite(efirst) ~ rcs(qlogis(pcurrent), 4), data = ef)
```

{

		Model Likelihood Ratio Test		Discrimination Indexes		Rank Discrim. Indexes	
Obs	420	LR χ^2	116.51	R^2	0.452	C	0.899
FALSE	366	d.f.	3	g	7.478	D_{xy}	0.797
TRUE	54	$\Pr(>\chi^2)$	<0.0001	g_r	1767.951	γ	0.797
$\max \lvert \frac{\partial \log L}{\partial \beta} \rvert$	4×10^{-6}			g_p	0.179	τ_a	0.179
				Brier	0.080		

}

%latex.default(U, file = ”“, first.hline.double = FALSE, table = FALSE, longtable = TRUE, lines.page = lines.page, col.just = rep(”r“, ncol(U)), rowlabel =””, already.math.col.names = TRUE, append = TRUE)%

	$\widehat{\beta}$	S.E.	Wald Z	$\Pr(>\lvert Z\rvert)$
Intercept	-2.3434	8.4137	-0.28	0.7806
pcurrent	13.3734	27.7510	0.48	0.6299
pcurrent’	-12.5370	36.1693	-0.35	0.7289
pcurrent”	28.7637	98.8864	0.29	0.7711

```
P <- Predict(f, pcurrent, fun=plogis, conf.int=FALSE)
xl <- 'P(OR < 1) at 700th Patient'
yl <- 'P(P(OR < 1) > 0.95) by 1000th Patient'
switch(outfmt,
             pdf  = ggplot(P, xlab=xl, ylab=yl),
             html = plotp(P, xlab=xl, ylab=yl))
```

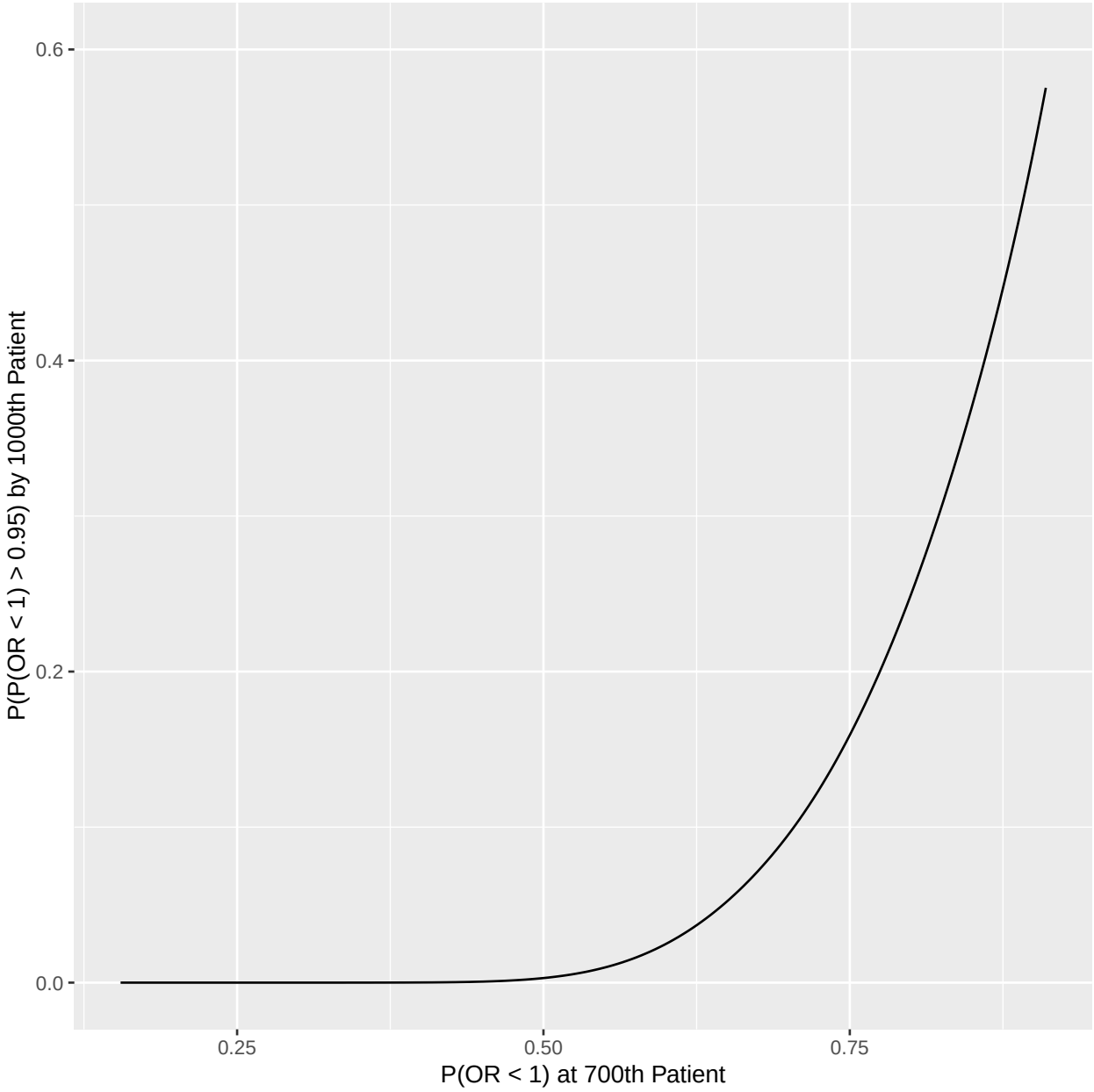

Figure 4: Logistic model estimates of the probability having sufficient evidence for efficacy by the planned study end, given the posterior probability of efficacy at n=700

There are only 54 RCTs that later hit the efficacy target when neither it nor the inefficacy target were hit by 700 patients, so more simulations may be needed to get reliable estimates. But it can be seen that the current (at 700 patients) posterior probability needs to be larger to have a reasonable chance of reaching an efficacy conclusion than it had to be after only 400 patients.

Operating Characteristics With Less Frequent Looks

Instead of looking almost continuously, let's determine the operating characteristics under a sequential design in which the maximum sample size is 1000 patients, the first look as after 100 patients have been followed, and a look is taken every 100 patients after that. Unlike earlier simulations we add a twist: only count

simulated trials that did not stop previously for futility.

```
ors   <- seq(0.4, 1.25, by=0.05)
looks <- seq(100, 1000, by=100)
if(! file.exists('seqsimless.rds')) {
  set.seed(5)
    est <- estSeqSim(log(ors), looks, gdat, lfit, nsim=500)
    saveRDS(est, 'seqsimless.rds', compress='xz')
    } else est <- readRDS('seqsimless.rds')
s <- gbayesSeqSim(est, asserts=asserts)
head(s)
```

```
  sim  parameter look        est       vest        p1      mean1       sd1
1   1 -0.9162907  100 -0.8845577 0.15391137 0.9344354 -0.3965564 0.2626786
2   1 -0.9162907  200 -1.2672169 0.08225853 0.9997001 -0.7644440 0.2227603
3   1 -0.9162907  300 -1.1338281 0.05437178 0.9999754 -0.7902733 0.1946712
4   1 -0.9162907  400 -1.0343128 0.03981947 0.9999968 -0.7845353 0.1737913
5   1 -0.9162907  500 -1.0479410 0.03198230 0.9999999 -0.8345378 0.1595914
6   1 -0.9162907  600 -1.0454613 0.02625587 1.0000000 -0.8640686 0.1473104
         p2      mean2       sd2           p3      mean3       sd3           p4
1 0.9879237 -0.8845441 0.3923124 1.207629e-02 -0.8845441 0.3923124 2.416259e-02
2 0.9999950 -1.2672064 0.2868063 4.973196e-06 -1.2672064 0.2868063 3.244024e-05
3 0.9999994 -1.1338219 0.2331769 5.795730e-07 -1.1338219 0.2331769 2.756842e-06
4 0.9999999 -1.0343087 0.1995478 1.090184e-07 -1.0343087 0.1995478 4.017237e-07
5 1.0000000 -1.0479376 0.1788357 2.317407e-09 -1.0479376 0.1788357 9.171218e-09
6 1.0000000 -1.0454585 0.1620364 5.519270e-11 -1.0454585 0.1620364 2.216671e-10
       mean4       sd4
1 -0.6094232 0.3086472
2 -0.9937225 0.2487840
3 -0.9603173 0.2113268
4 -0.9145335 0.1853335
5 -0.9475168 0.1683891
6 -0.9615448 0.1541443
```

```
w <- data.table(s)
w  <- melt(w,
           measure.vars=list(ps, paste0('mean', 1:4), paste0('sd', 1:4)),
           variable.name='assert', value.name=c('p', 'mean', 'sd'))
w[, assert := alabels[assert]]
head(w)
```

```
   sim  parameter look        est       vest   assert         p       mean
1:   1 -0.9162907  100 -0.8845577 0.15391137 Efficacy 0.9344354 -0.3965564
2:   1 -0.9162907  200 -1.2672169 0.08225853 Efficacy 0.9997001 -0.7644440
3:   1 -0.9162907  300 -1.1338281 0.05437178 Efficacy 0.9999754 -0.7902733
4:   1 -0.9162907  400 -1.0343128 0.03981947 Efficacy 0.9999968 -0.7845353
5:   1 -0.9162907  500 -1.0479410 0.03198230 Efficacy 0.9999999 -0.8345378
6:   1 -0.9162907  600 -1.0454613 0.02625587 Efficacy 1.0000000 -0.8640686
          sd
1: 0.2626786
2: 0.2227603
3: 0.1946712
4: 0.1737913
5: 0.1595914
6: 0.1473104
```

```
w[, target := target[assert]]     # spreads targets to all rows
u <- w[, .(hit = 1*(cumsum(p > target) > 0), look=look),
       by=.(sim, parameter, assert)]
ors  <- round(exp(u$parameter), 2)
us   <- u[ors == round(ors, 1), ]   # subset data table with OR incremented by 0.1
prop <- us[, .(p=mean(hit)), by=.(parameter, look, assert)]
prop[, txt := paste0(assert, '<br>OR:', exp(parameter), '<br>n:', look, '<br>', round(p, 3))]
ggp(ggplot(prop, aes(x=look, y=p, color=factor(exp(parameter)), label=txt)) + geom_line() +
  facet_wrap(~ assert) +
  xlab('Look') + ylab('Proportion Stopping') +
  guides(color=guide_legend(title='True OR')) +
  theme(legend.position = 'bottom'), tooltip='label')
prp <- prop[assert == 'Efficacy', ]
Sm <- rbind(Sm,
                        data.frame(topic='Power to detect efficacy',
                                             cond1  = 'look every 100 patients',
                                             cond2  = 'no futility stopping',
                                             cond3  = c(rep('n=500', 2), rep('n=1000', 2)),
                                             cond4  = rep(c('OR=1', 'OR=0.7'), 2),
                                             amount = lookup(n=c(500, 1000))))
```

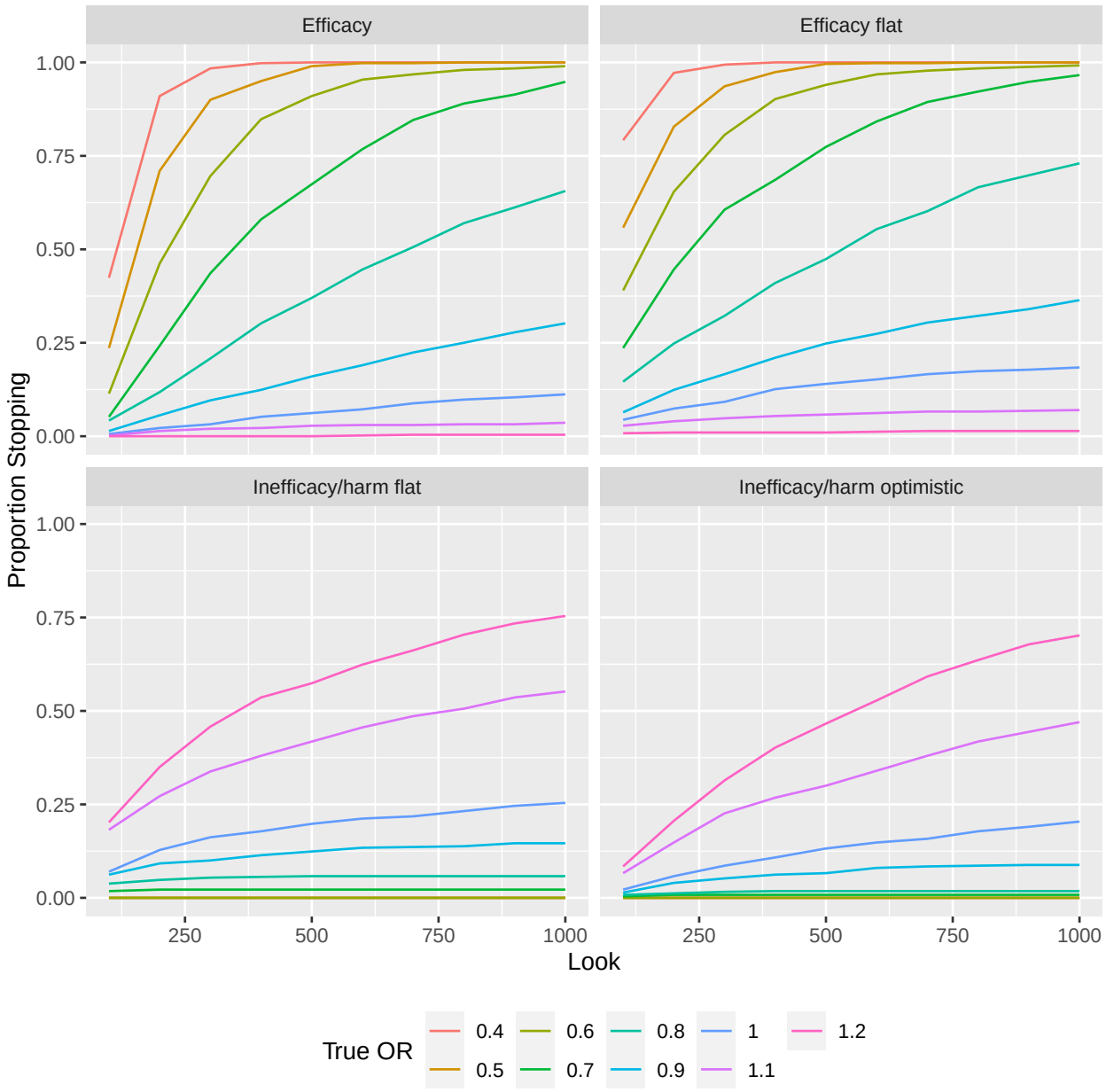

Figure 5: Bayesian power when looks are made only after every 100 patients and when no early stopping for futility is allowed

Let's develop a futility rule so that the operating characteristics can be re-run excluding simulated trials that stopped for futility at an earlier look. We will use a posterior probability of efficacy of 0.95 as the efficacy evidence target for at the end or for stopping early for efficacy. Futility is taken to mean a final posterior probability less than 0.95 at the planned maximum sample size of 1000 patients. Instead of computing the proportion of simulated trials hitting the 0.95 threshold at $n = 1000$ we use a binary logistic model with a tensor spline in the current posterior probability and sample sizes to estimate the probability if hitting 0.95. Futility probabilities are estimated in the absence of knowledge of the data generating OR.

First fetch the final posterior probability and merge it with earlier looks. Then show probabilities of hitting the target estimated by computing proportions within intervals containing 450 observations, and add loess estimates to the same plot. Then the binary logistic model is fitted and plotted.

```
u <- w[assert == 'Efficacy', .(sim, parameter, look, p)]
setkey(u, parameter, sim, look)
u[, final := p[look == 1000], by=.(parameter, sim)]
dd <- datadist(u); options(datadist='dd')
with(u[look < 1000,], plsmo(p, final >= 0.95, group=look, method='intervals', mobs=450))
with(u[look < 1000,], plsmo(p, final >= 0.95, group=look, add=TRUE))
```

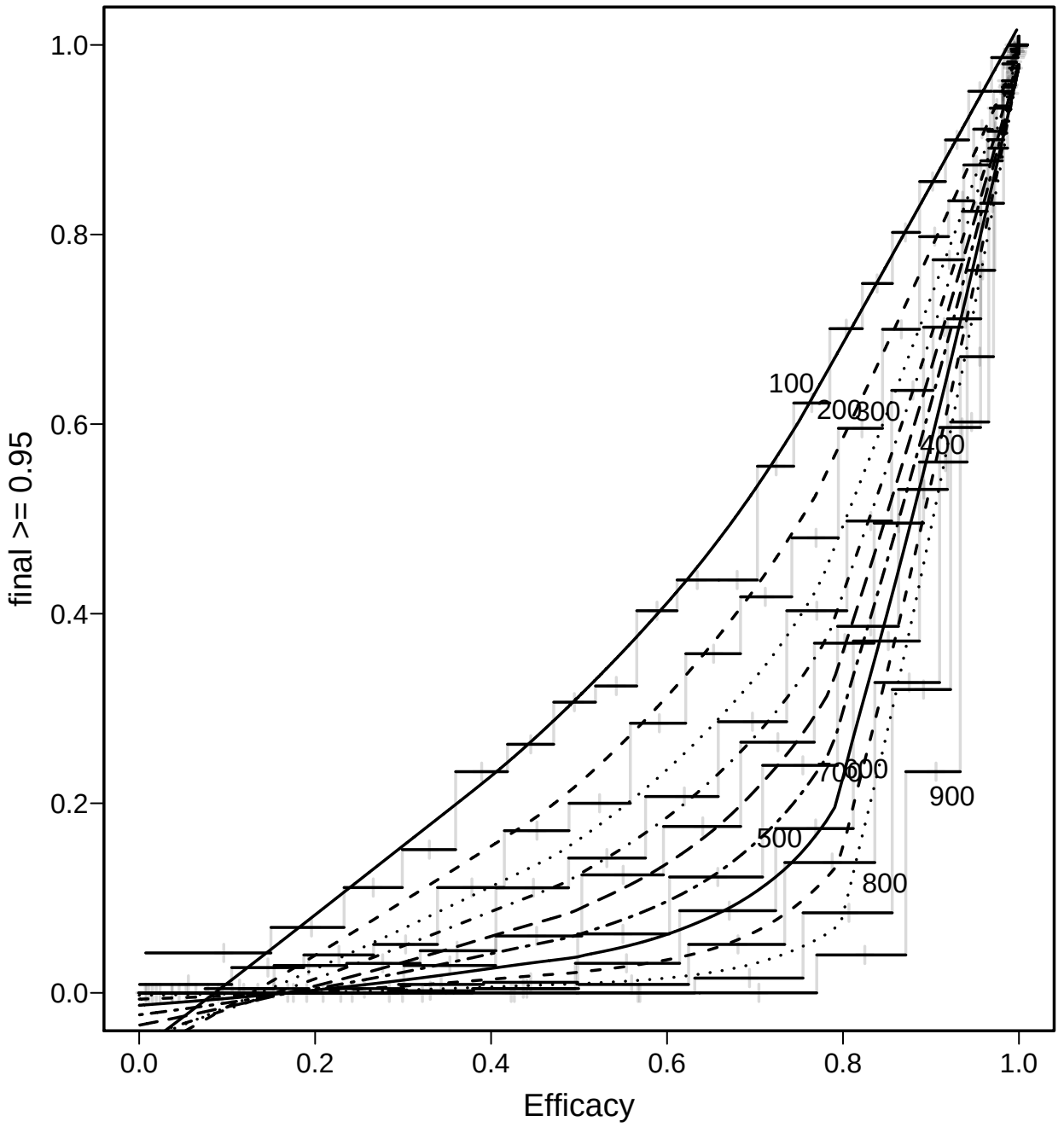

Figure 6: Simple proportons and loess smoothed estimates of the relationship between the current posterior probability of efficacy, the sample size at which the current probability was calculated, and the probability that the final posterior probability of efficacy exceeds 0.95. Proportions are computed after dividing current probabilities into intervals each containing 450 values.

```
f <- lrm(final >= 0.95 ~ rcs(look, 5) * rcs(p, 5), data=u[look < 1000, ], scale=TRUE, maxit=35)
P <- Predict(f, p, look=seq(100, 900, by=200), fun=plogis, conf.int=FALSE)
xl <- 'Current Posterior Probability'
yl <- 'P(P > 0.95 at 1000)'
# plotp treated look as a continuous variable and created a color image plot
# had to use regular ggplot
# switch(outfmt,
#            pdf  = ggplot(P, xlab=xl, ylab=yl),
#            html = plotp(P,  xlab=xl, ylab=yl))
ggplot(P, xlab=xl, ylab=yl)
```

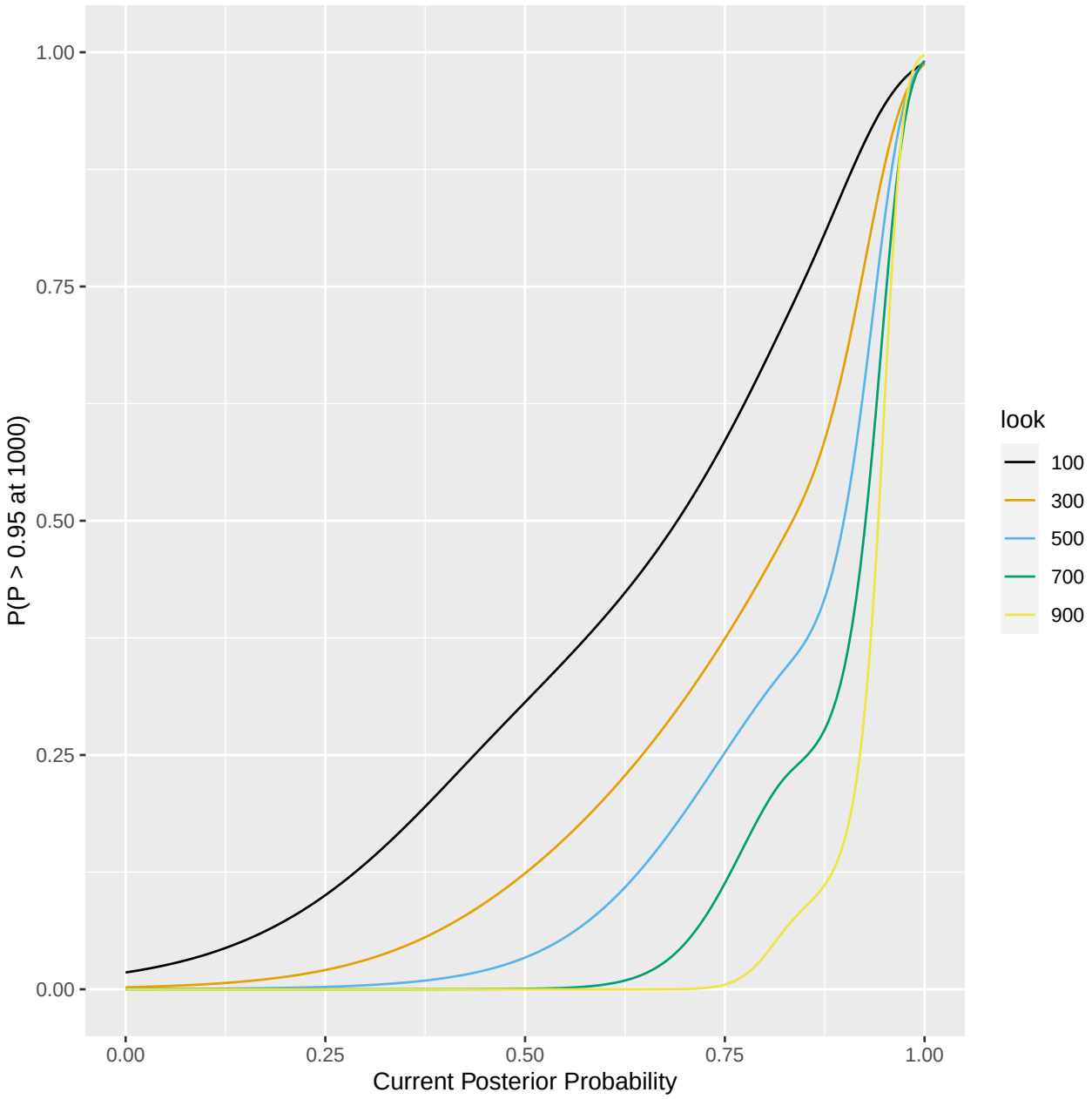

Figure 7: Logistic model estimates of the relationship between the current posterior probability of efficacy, the sample size at which the current probability was calculated, and the probability that the final posterior probability of efficacy exceeds 0.95. A tensor spline interaction surface is modeled between sample size and current probability.

To speed up calculations we solve for the minimum current posterior probability that for a given sample size results in a probability of 0.1 of hitting the target at $n = 1000$.

```
minp <- numeric(9)
ps   <- seq(0, 1, length=5000)
i <- 0
for(lo in seq(100, 900, by=100)) {
    i <- i + 1
```

```
    ptarget <- predict(f, data.frame(look=lo, p=ps), type='fitted')
    minp[i]    <- approx(ptarget, ps, xout=0.1)$y
}
round(minp, 3)
```

```
[1] 0.249 0.360 0.463 0.550 0.615 0.672 0.741 0.802 0.864
```

```
# Check
round(predict(f, data.frame(look=seq(100, 900, by=100), p=minp), type='fitted'), 7)
```

```
        1         2         3         4         5         6         7         8
0.1000000 0.1000000 0.1000000 0.1000000 0.1000000 0.1000000 0.1000000 0.1000000
        9
0.0999999
```

```
fut <- function(p, look)
    ifelse(look == 1000, FALSE, p < minp[look / 100])
```

Compute the proportion of trials that are stopped early for futility, by true OR.

```
z <- u[, .(futile = 1L * (cumsum(fut(p, look)) > 0),
                     eff    = 1L * (cumsum(p >= 0.95) > 0),
                     look   = look,
                     p      = p),
                     by=.(parameter, sim)]
pstop <- z[, .(p = mean(futile)), by=.(parameter, look)]
if(outfmt == 'html') {
  pstop[, txt := paste0('OR:', exp(parameter), '<br>n:', look, '<br>', round(p, 3))]
  ggplotly(ggplot(pstop, aes(x=look, y=p, color=factor(exp(parameter)),
                                     label=txt)) + geom_line() +
      guides(color=guide_legend(title='True OR')) +
      xlab('Sample Size') + ylab('Probability of Stopping for Futility'), tooltip='label')
} else {
  ggplot(pstop, aes(x=look, y=p, color=factor(exp(parameter)),
                                      linetype = parameter == 0)) + geom_line() +
      guides(color=guide_legend(title='True OR'), linetype=FALSE) +
      xlab('Sample Size') + ylab('Probability of Stopping for Futility')
}
prp <- pstop
Sm <- rbind(Sm,
                        data.frame(topic='Probability of stopping early for futility',
                                             cond1  = 'look every 100 patients',
                                             cond2  = '',
                                             cond3  = c(rep('n=300', 2), rep('n=600', 2)),
                                             cond4  = rep(c('OR=1', 'OR=0.7'), 2),
                                             amount = lookup(n=c(300, 600))))
```

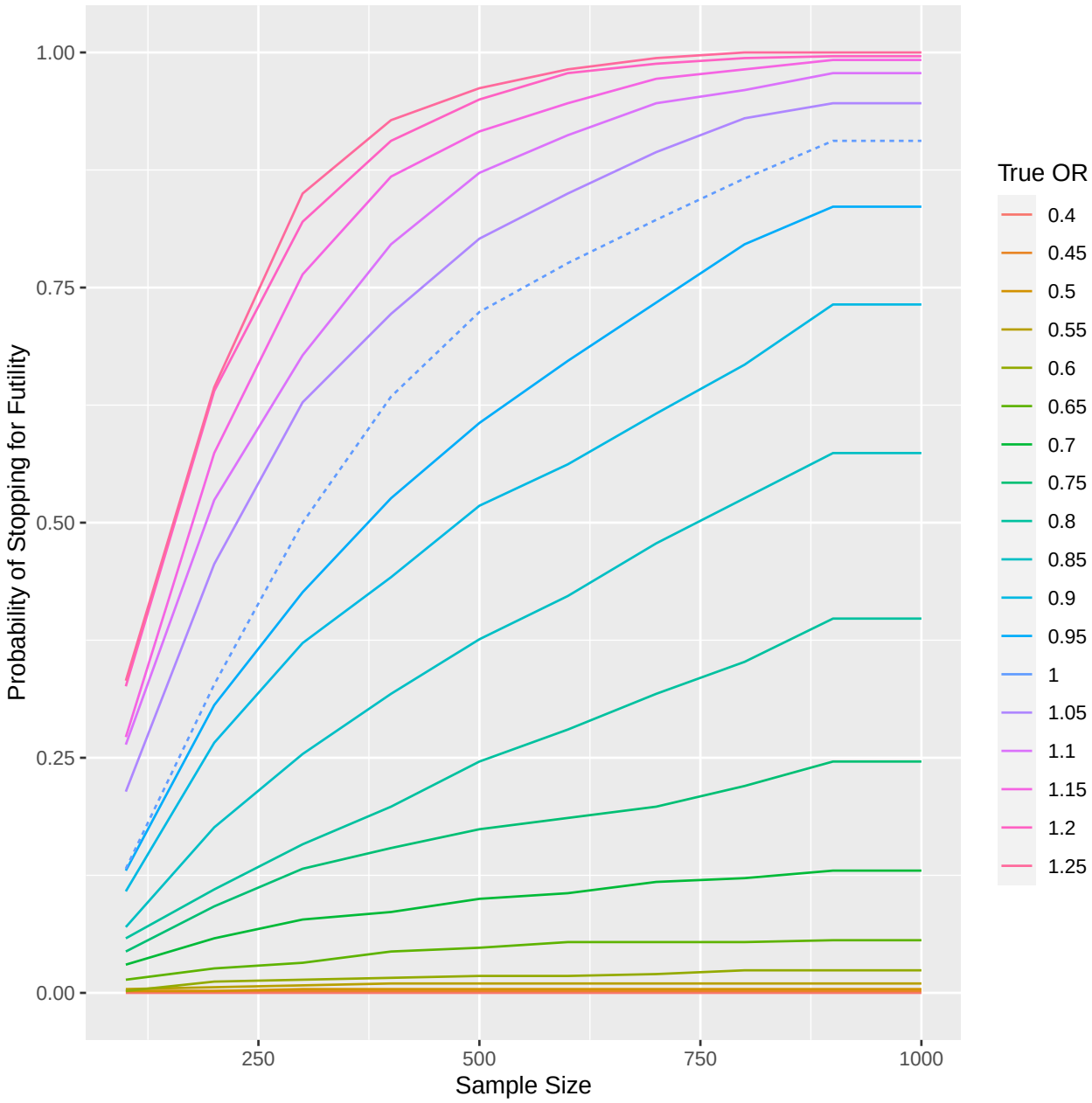

Figure 8: Probability of stopping for futility, as a function of the unknown true odds ratio and the current sample size, when looks are taken every 100 patients

Now compute the operating characteristics, where declaring earlier for futility is treated as not declaring efficacy later.

```
ph <- z[, .(p = mean(eff * (1 - futile))), by=.(parameter, look)]
if(outfmt == 'html') {
    ph[, txt := paste0('OR:', exp(parameter), '<br>n=', look, '<br>', round(p, 3))]
    ggplotly(ggplot(ph, aes(x=look, y=p, color=factor(exp(parameter)),
                            label = txt)) + geom_line() +
      guides(color=guide_legend(title='True OR')) +
      xlab('Sample Size') + ylab('Probability of P(efficacy) > 0.95'), tooltip='label')
```

```
} else {
ggplot(ph, aes(x=look, y=p, color=factor(exp(parameter)),
                            linetype=parameter == 0)) + geom_line() +
    guides(color=guide_legend(title='True OR'), linetype=FALSE) +
    xlab('Sample Size') + ylab('Probability of P(efficacy) > 0.95')
}
ph[parameter == 0, .(look, p)]
```

```
    look     p
 1:  100 0.006
 2:  200 0.022
 3:  300 0.032
 4:  400 0.052
 5:  500 0.058
 6:  600 0.066
 7:  700 0.074
 8:  800 0.074
 9:  900 0.062
10: 1000 0.068
```

```
prp <- ph
Sm <- rbind(Sm,
                        data.frame(topic='Power to detect efficacy',
                                                cond1  = 'look every 100 patients',
                                                cond2  = 'with futility stopping',
                                                cond3  = c(rep('n=500', 2), rep('n=1000', 2)),
                                                cond4  = rep(c('OR=1', 'OR=0.7'), 2),
                                                amount = lookup(n=c(500, 1000))))
```

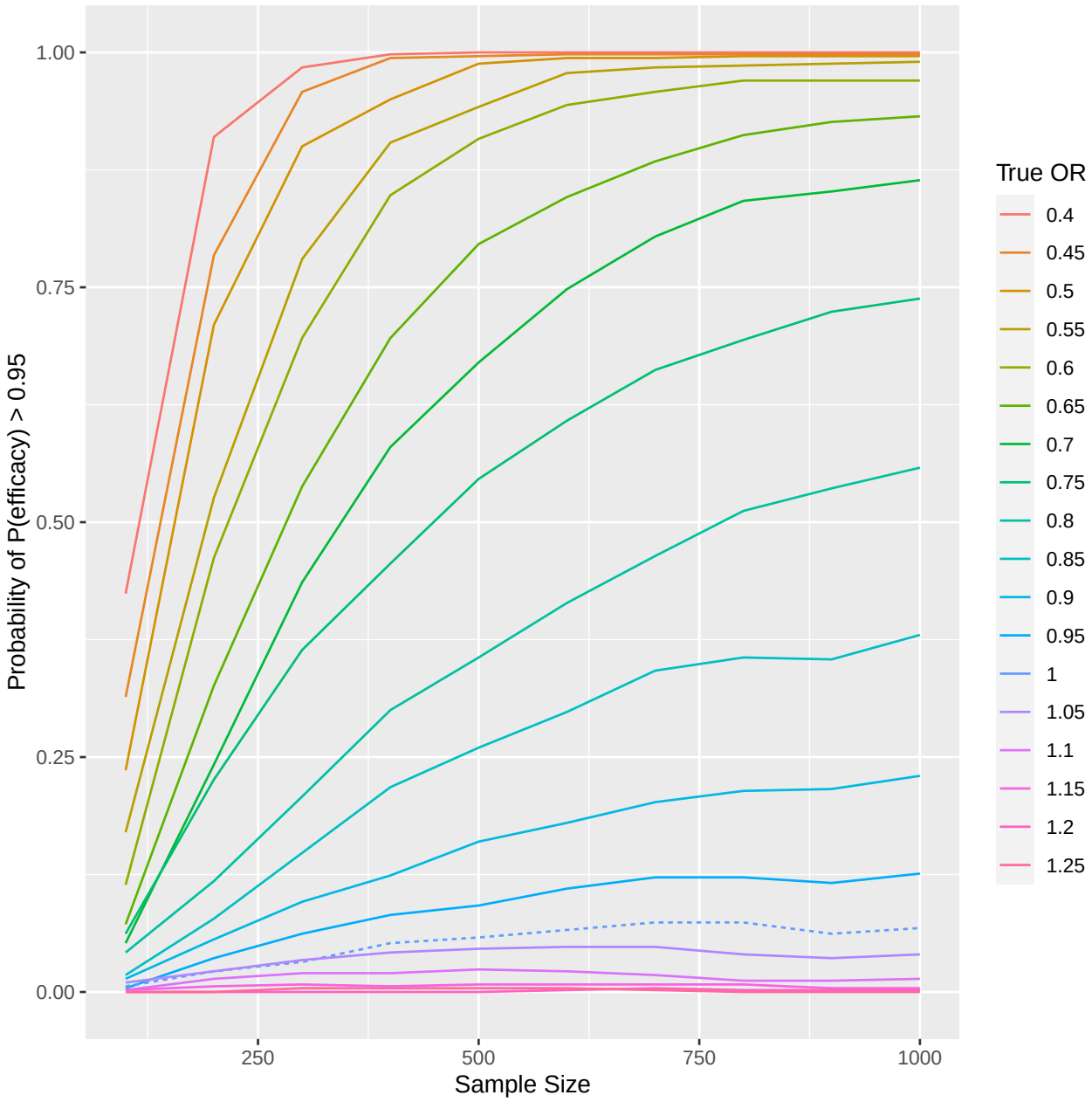

Figure 9: Bayesian operating characteristics and power when trials are allowed to stop earlier for futility, using the previously derived futility boundaries

The above estimates of the probability of declaring efficacy at any of the 10 looks, when OR=1, shows that stopping early for futility also happens to sharply limit type I probability α.

Summary

The proportional odds model was used to assess the treatment effect, with emphasize on a skeptical prior distribution. Bayesian power was estimated. This is the probability of reaching a posterior probability of efficacy > 0.95. When the true treatment effect is zero (a tall order to know without having a great deal of data), Bayesian power can be checked against frequentist type I assertion probability α, although simulations

showed that the more important quantity to evaluate is the accuracy of the posterior probability of efficacy at the moment of stopping for efficacy (not shown below). When futility is considered, we defined the futility threshold as a probability less than 0.1 that the posterior probability of efficacy will reach the 0.95 threshold at the planned end of the study (n=1000). Results are summarized below. The last column is either the probability of finding evidence for efficacy, or the probability of stopping early for futility depending on the description in the first column.

```
saveRDS(Sm, 'Sm.rds')
for(x in names(Sm)[1:5]) Sm[[x]] <- ifelse(Sm[[x]] == Lag(Sm[[x]]), '', Sm[[x]])
# LaTeX was running the last 2 columns together
if(outfmt == 'pdf') Sm$amount <- paste0('\\quad ', sprintf('%.3f', Sm$amount))
knitr::kable(Sm, col.names=rep('', length(Sm)))
```

Power to detect efficacy	almost unlimited looks	no futility stopping	n=500	OR=1	0.100
				OR=0.7	0.770
			n=1000	OR=1	0.178
				OR=0.7	0.976
	look every 100 patients		n=500	OR=1	0.062
				OR=0.7	0.674
			n=1000	OR=1	0.112
				OR=0.7	0.948
Probability of stopping early for futility			n=300	OR=1	0.500
				OR=0.7	0.078
			n=600	OR=1	0.776
				OR=0.7	0.106
Power to detect efficacy		with futility stopping	n=500	OR=1	0.058
				OR=0.7	0.670
			n=1000	OR=1	0.068
				OR=0.7	0.864

More Information

- Full R markdown script
- COVID-19 statistical resources
- Bayesian design and analysis resources

Computing Environment

To cite R in publication use:

R Core Team (2021). R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/.