MAP Prior with Binary Endpoint

Hongtao Zhang (github/squallteo)

Historical Data

The historical data used in this case study are from four studies of rheumatoid arthritis (RA) in patients who were treated with methotrexate (MTX). The primary binary outcome is the rate of patients satisfying American College of Rheumatology 50 criteria (ACR50) at week 12.

```
library(tidyverse)
library(knitr)
library(kableExtra)
histdt <- tibble(study = 1:4, r = c(33, 98, 3, 36), n = c(221, 651, 20, 214)) %>%
  mutate(ACR50 = round(r/n,3))
histdt %>% kbl(caption="Historical Data") %>%
  kable_classic(full_width = F, html_font = "Cambria", latex_options = "HOLD_position")
```

Table 1: Historical Data

study	r	n	ACR50
1	33	221	0.149
2	98	651	0.151
3	3	20	0.150
4	36	214	0.168

The ACR50 rates in the four historical trials are fairly consistent. The random effect meta-analysis yields a point estimate of 0.154. The predictive CI for current control is (0.11, 0.21).

```
with(histdt, meta::metaprop(event = r, n = n, method = "Inverse", prediction = T))
```

```
##   proportion           95%-CI %W(fixed) %W(random)
## 1     0.1493 [0.1051; 0.2033]      19.5       19.5
## 2     0.1505 [0.1239; 0.1804]      57.9       57.9
## 3     0.1500 [0.0321; 0.3789]       1.8        1.8
## 4     0.1682 [0.1207; 0.2252]      20.8       20.8
##
## Number of studies combined: k = 4
##
##                         proportion           95%-CI
## Fixed effect model          0.1538 [0.1337; 0.1763]
## Random effects model        0.1538 [0.1337; 0.1763]
## Prediction interval                [0.1127; 0.2065]
##
## Quantifying heterogeneity:
##  tau^2 = 0 [0.0000; 0.0150]; tau = 0 [0.0000; 0.1225]
##  I^2 = 0.0% [0.0%; 84.7%]; H = 1.00 [1.00; 2.56]
##
```

```
## Test of heterogeneity:
##     Q d.f. p-value
##  0.43    3  0.9337
##
## Details on meta-analytical method:
## - Inverse variance method
## - DerSimonian-Laird estimator for tau^2
## - Jackson method for confidence interval of tau^2 and tau
## - Logit transformation
## - Clopper-Pearson confidence interval for individual studies
```

MAP Prior

The original MAP prior assumes a common normal distribution for log-odds $\theta_i = \log \frac{p_i}{1-p_i}$ where subscript $i = 1, \cdots, 4$ indices the historical trials:

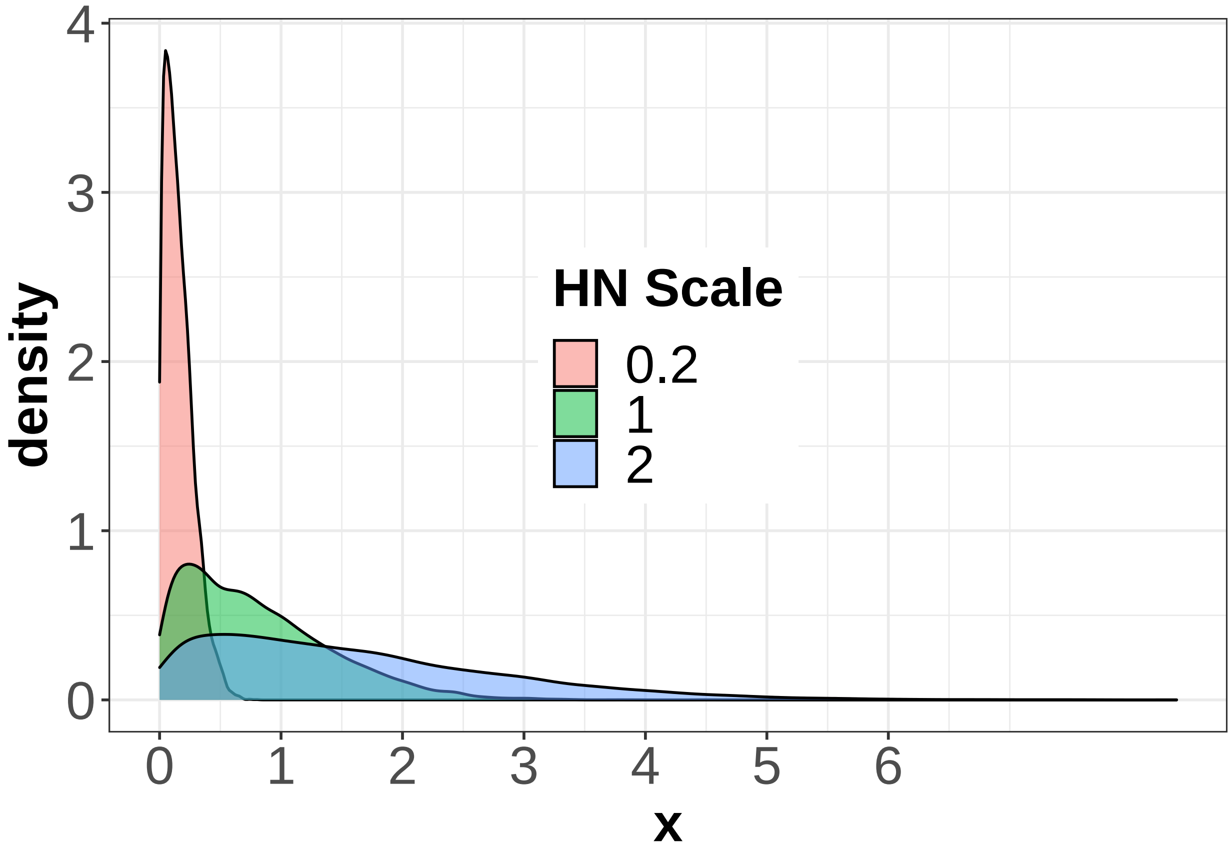
$$\theta_1, \cdots, \theta_4 \sim N(\mu_C, \sigma_C^2),$$

where the hyper-priors are

$$\mu_C \sim N(0, 2^2) \text{ and } \sigma_C \sim HN(s).$$

A standard deviation of 2 for μ_C is fairly large on the logit scale so that the hyper-prior is vague. The half-normal prior regulates the extent of borrowing and thus the scale s should be chosen carefully. One way is to examine the densities corresponding to various scales (0.2, 1, 2 in the plot). We will use $HN(s = 1)$ from hereafter.

```r
library(extraDistr)
priordt <- rbind(tibble(Sigma = 2, sample = rhnorm(10000, sigma = 2)),
                 tibble(Sigma = 1, sample = rhnorm(10000, sigma = 1)),
                 tibble(Sigma = 0.2, sample = rhnorm(10000, sigma = 0.2)))
priordt %>%
  ggplot(aes(x = sample, fill = factor(Sigma))) + geom_density(alpha = 0.5) +
  labs(fill = "HN Scale") + scale_x_continuous(breaks = seq(0, 6,  1), name = "x") +
  theme_bw() +
  theme(axis.title = element_text(face="bold",size=20),
        axis.text = element_text(size=20),
        legend.title=element_text(size=20,face="bold"),
        legend.text=element_text(size=20),
        legend.position = c(0.5, 0.5)
  )
```



Below is the command to derive the original MAP prior.

```
library(RBesT)
set.seed(1027)
options(RBesT.MC.control=list(adapt_delta=0.999))
map_c_mcmc = gMAP(cbind(r, n - r) ~ 1 | study,
                  data=histdt,
                  family=binomial,
                  beta.prior=cbind(0, 2),
                  tau.dist="HalfNormal",tau.prior=cbind(0, 1),
                  chains = 3)
```

The distribution of the MAP prior is unknown. Approximation is needed to convert it to a "workable" form. With binomial data, the MAP prior is approximated by a weighted mixture of Beta distributions. Three Beta components are used and the approximation is satisfactory as shown both numerically and visually.

```
map_c_hat = mixfit(map_c_mcmc, Nc=3)
print(map_c_hat)
```

```
## EM for Beta Mixture Model
## Log-Likelihood = 5969.913
##
## Univariate beta mixture
## Mixture Components:
##   comp1       comp2       comp3
## w   0.3893370   0.3880054   0.2226576
## a  46.5702699  72.0148367   3.5054258
## b 243.4141931 408.0689941  16.2800457
```

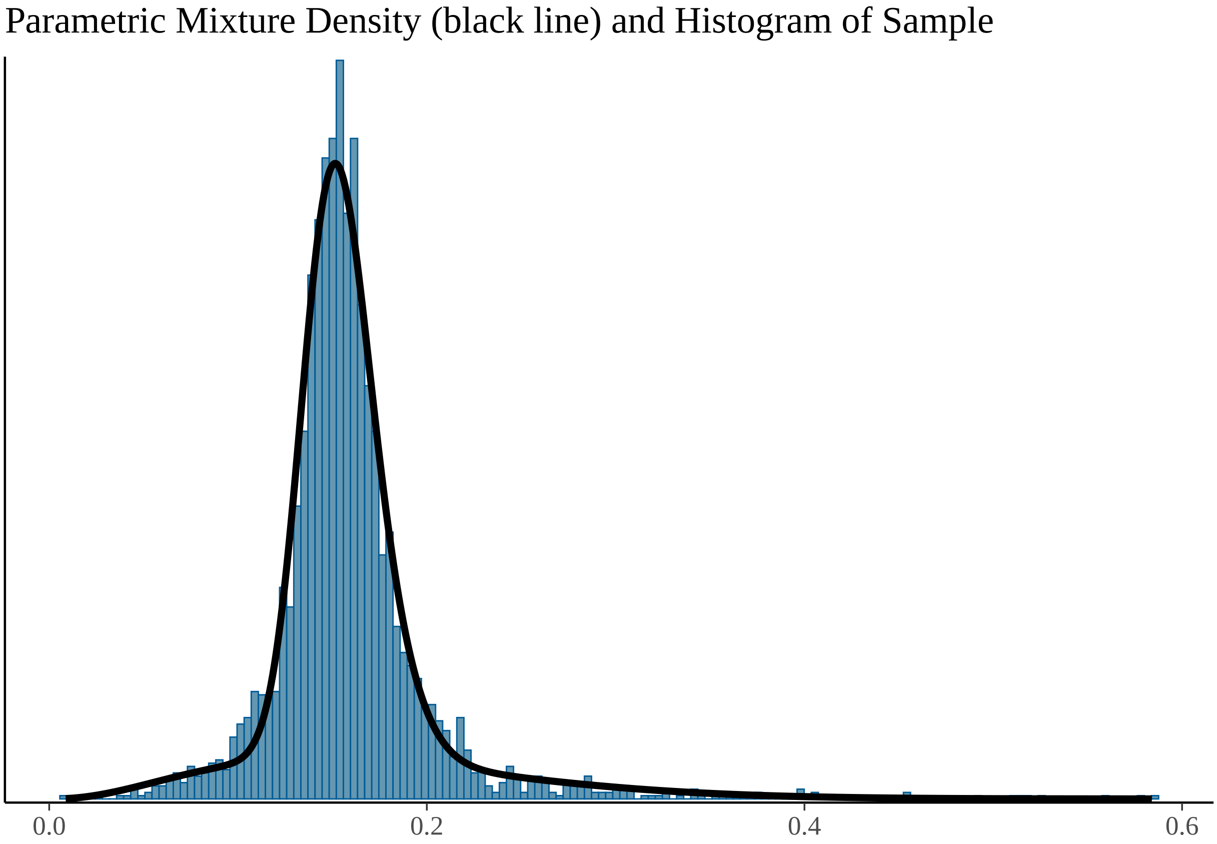
```
summary(map_c_mcmc)$theta.pred
```

```
##                       mean         sd       2.5%       50%     97.5%
## theta_resp_pred 0.1599893 0.04597546 0.08825363 0.1544131 0.2759806
```

```
summary(map_c_hat)
```

```
##       mean         sd       2.5%      50.0%      97.5%
## 0.16017713 0.04415167 0.08156843 0.15476832 0.28404156
```

```
plot(map_c_hat)$mix
```



Robust MAP Prior

The vague prior f_V to construct rMAP prior is $Beta(1, 1)$ and we form a 50-50 mixture of original MAP prior and the vague prior, that is,

$$f_{rMAP} = 0.5 * \hat{f}_{MAP} + 0.5 * f_V.$$

```
#RBesT package automatically bumps up n by 1
#see the help document of robustify command for more details
rmap_c <- robustify(map_c_hat, weight=0.5, mean=0.5, n=1)
print(rmap_c)
```

```
## Univariate beta mixture
## Mixture Components:
##   comp1       comp2       comp3       robust
## w   0.1946685   0.1940027   0.1113288   0.5000000
## a  46.5702699  72.0148367   3.5054258   1.0000000
## b 243.4141931 408.0689941  16.2800457   1.0000000
```

```
summary(rmap_c)
```

```
##       mean         sd       2.5%      50.0%      97.5%
## 0.33008857 0.26741587 0.04515052 0.18031091 0.95001226
```

New Trial Design

Suppose that we are designing a new randomized two-arm proof-of-concept study in the same RA population. The endpoint remains the same, that is, ACR50 at week 12. Randomziation ratio is 2:1 - 60 in treatment and 30 in control. The robust MAP prior derived in previous section is used to augment the current control arm. We typically don't augment the treatment arm, and therefore assign a flat prior $Beta(1, 1)$ to p_T.

```
(prior_t = mixbeta(c(1,1,1)))
```

```
## Univariate beta mixture
## Mixture Components:
##   comp1
## w 1
## a 1
## b 1
```

The operating characteristics are evaluated for current control rates between 0.11 and 0.21, guided by the predictive interval from meta-analysis. Assuming an improvement of 0.25 in ACR50 rate in the treatment arm, we may calculate the frequentist power that does not borrow historical data as a reference.

```
ctrl_range <- seq(0.11, 0.21, 0.01)
es <- 0.25
library(pwr)
FPwr20 <- pwr.2p2n.test(h = ES.h(ctrl_range, ctrl_range + es), n1 = 30, n2 = 60,
                        sig.level = 0.05, alternative = "two.sided")
```

The trial is claimed a success if the single-criterion decision rule is met:

$$Pr(p_T - p_C > 0|Y_1, ..., Y_4, Y_C, Y_T) > 0.975.$$

The type I error and power can be calculated in *RBesT* package. To do that, we first define the decision rule then the design

```
(rule_single <- decision2S(pc = c(0.975), qc = c(0), lower.tail = F))
```

```
## 2 sample decision function
## Conditions for acceptance:
## P(x1 - x2 > 0) > 0.975
## Link: identity
```

```
design <- oc2S(prior_t, rmap_c, 60, 30, rule_single)
```

On a side note, a dual-criteria decision rule can also be defined, e.g.:

$$Pr(p_T - p_C > 0|Y_1, ..., Y_4, Y_C, Y_T) > 0.975$$

and

$$Pr(p_T - p_C > 0.25|Y_1, ..., Y_4, Y_C, Y_T) > 0.6.$$

```
decision2S(pc = c(0.975, 0.6), qc = c(0, 0.25), lower.tail = F)
```

```
## 2 sample decision function
## Conditions for acceptance:
## P(x1 - x2 > 0) > 0.975
## P(x1 - x2 > 0.25) > 0.6
## Link: identity
```

The "design" object is a function that computes operating characteristics given control and treatment rates. In the results shown below, type I error corresponds to an effect size of 0, while power corresponds to an effect size of 0.25.

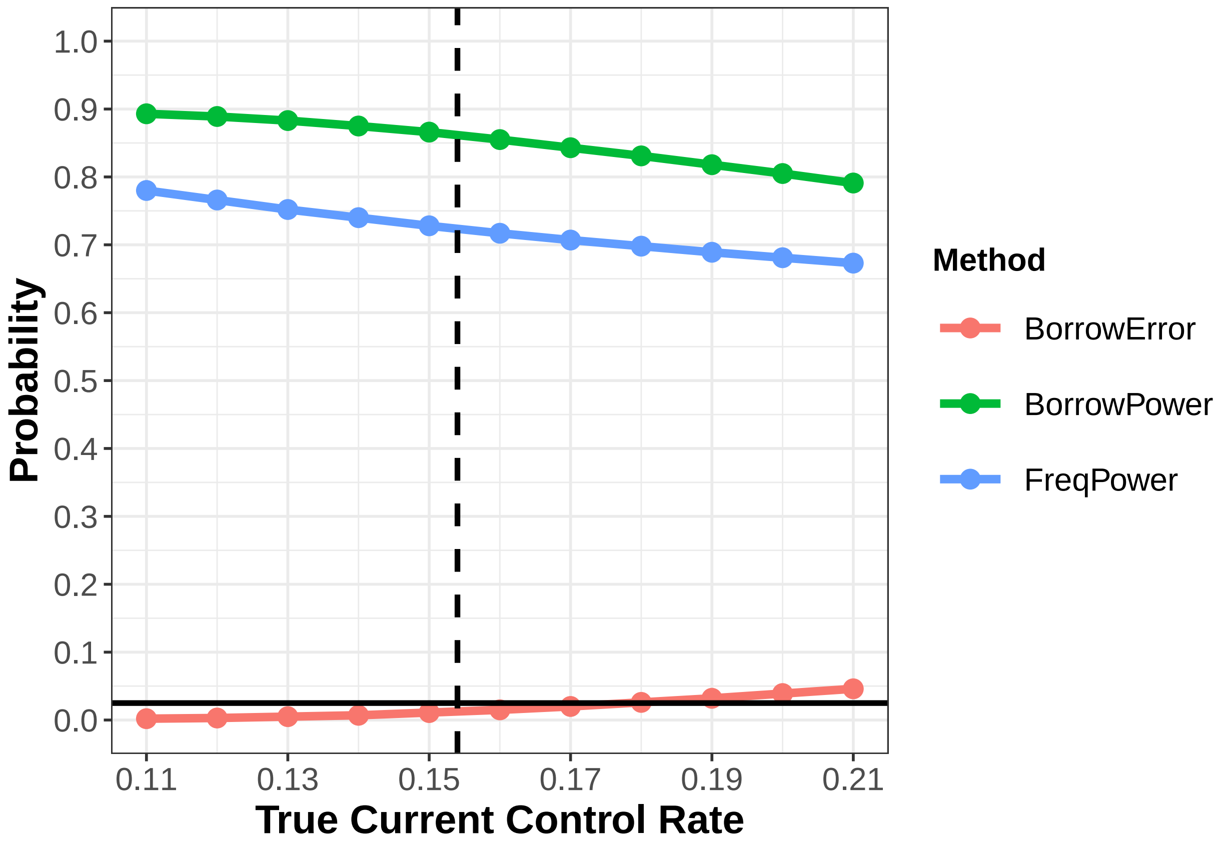
```r
#type I error
error_b <- design(ctrl_range, ctrl_range)
#power
power_b <- design(ctrl_range+es, ctrl_range)

ocdt <- tibble(CtrlRate = ctrl_range, FreqError = 0.025, BorrowError = round(error_b,3),
               FreqPower = round(FPwr20$power,3), BorrowPower = round(power_b,3))
ocdt %>% kbl(caption="Operating Characteristics") %>%
  kable_classic(full_width = F, html_font = "Cambria", latex_options = "HOLD_position")
```

Table 2: Operating Characteristics

CtrlRate	FreqError	BorrowError	FreqPower	BorrowPower
0.11	0.025	0.002	0.780	0.893
0.12	0.025	0.003	0.766	0.889
0.13	0.025	0.005	0.752	0.883
0.14	0.025	0.007	0.740	0.875
0.15	0.025	0.011	0.728	0.866
0.16	0.025	0.015	0.717	0.855
0.17	0.025	0.020	0.707	0.843
0.18	0.025	0.026	0.698	0.831
0.19	0.025	0.032	0.689	0.818
0.20	0.025	0.039	0.681	0.805
0.21	0.025	0.046	0.673	0.791

We also generate a plot that overlays the OCs. The horizontal solid line indicates the constant frequentist type I error $\alpha = 0.025$. The dashed vertical line represents the meta-analysis point estimate of historical control data. Within the range of true current control rate, the power using robust MAP prior is about 10% higher than no borrowing. In the meantime, the type I error rate is very well contained.

```r
plotdt <-
  ocdt %>% pivot_longer(!CtrlRate, names_to = "Method", values_to = "Prob") %>%
  filter(Method != "FreqError")
plotdt %>%
  ggplot(aes(x=CtrlRate,y=Prob, color=Method)) +
  geom_line(size = 1.5) + geom_point(size = 3) +
  scale_x_continuous(breaks = seq(0.11, 0.21, 0.02), name = "True Current Control Rate") +
  scale_y_continuous(limits = c(0,1), breaks = seq(0,1,0.1), name = "Probability") +
  geom_vline(xintercept = 0.154, linetype=2, size=1) +
  geom_hline(yintercept = 0.025, linetype=1, size=1) +
  theme_bw() +
  theme(axis.title = element_text(face="bold",size=15),
        axis.text = element_text(size=12),
        legend.title=element_text(size=12,face="bold"),
        legend.text=element_text(size=12),
        legend.key.size = unit(1, "cm")
  )
```



Data Analysis with Observed Current Trial Data

After the new trial is unblinded and analyzed, the robust MAP prior is updated with observed current control data. The posterior distribution is also a mixture of Beta components. Comparing with the prior mixture, the parameters of each Beta component is updated, and so is the weight w.

```
y_c <- 6
(post_c <- postmix(rmap_c, r=y_c, n=30))
```

```
## Univariate beta mixture
## Mixture Components:
##    comp1       comp2       comp3       robust
## w    0.3454043   0.3155126   0.1433055   0.1957776
## a   52.5702699  78.0148367   9.5054258   7.0000000
## b 267.4141931 432.0689941  40.2800457  25.0000000
```

```
rmap_c
```

```
## Univariate beta mixture
## Mixture Components:
##    comp1       comp2       comp3       robust
## w    0.1946685   0.1940027   0.1113288   0.5000000
## a   46.5702699  72.0148367   3.5054258   1.0000000
## b 243.4141931 408.0689941  16.2800457   1.0000000
```

Likewise, the vague prior for p_T is updated with observed current treatment data.

```
y_t <- 30
(post_t <- postmix(prior_t, r=y_t, n=60))
```

```
## Univariate beta mixture
## Mixture Components:
##   comp1
## w  1
## a 31
## b 31
```

Thanks to the *RBesT* package, the mixture distribution (and their difference) in R works in the same way as other probability distributions, in the sense that d/p/q/r functions can be used. Therefore, we can use the *pmixdiff* command to evaluate the single-criterion decision rule.

```
(prob <- pmixdiff(post_t, post_c,  0, lower.tail=FALSE))
```

```
## [1] 0.9993733
```

To verify, we can draw samples from respective posterior distributions and calculate the probability. It matches that calculated using *pmixdiff*.

```
sample_t <- rmix(post_t, 100000)
sample_c <- rmix(post_c, 100000)
mean((sample_t - sample_c) > 0)
```

```
## [1] 0.99937
```

The point estimate and 95% credible interval for the control rate can be extracted from the summary statistics of the posterior mixture distribution.

```
summary(post_c)
```

```
##       mean         sd       2.5%      50.0%      97.5%
## 0.17518995 0.04776647 0.11497612 0.16329924 0.31042833
```