Beta-binomial responder rate

Decision based on posterior probability of ORR>15%

mgrafit

2019-05-26

Exact 95%CI

Assuming an early clinical trial, with 1 responder out of 24 patients. The 95% confidence interval for the responder rate can be calculated exactly, and is known as the Clopper-Pearson exact method.

```
exaci<-PropCIs::exactci(x=1, n=24, conf.level=0.95)
round(exaci$conf.int, 4)
```

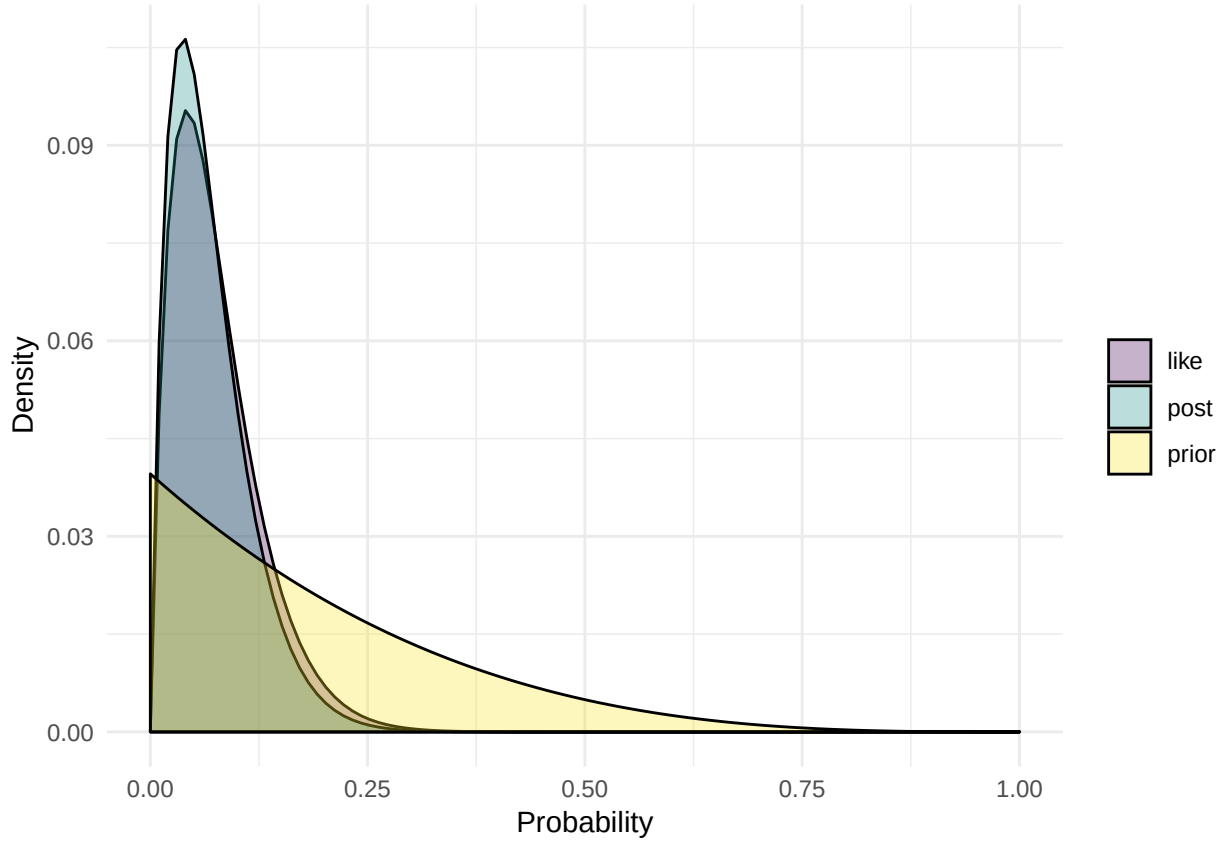
```
## [1] 0.0011 0.2112
## attr(,"conf.level")
## [1] 0.95
```

Using a Bayesian approach

The beta-binomial distribution (https://www.wikiwand.com/en/Beta-binomial_distribution) is the binomial distribution in which the probability of success at each trial is fixed but randomly drawn from a beta distribution prior to n Bernoulli trials.

Assuming the responder/non-responder outcome has been registered in the below order, we want to see how our prior belief is updated as evidence accumulates during a Ph1 oncology clinical trial.

```
#' total sample size (n); successes (Y)
n=24;Y=1;
#' beta prior
a=1; b=4
#' prop. grid
grid   <- seq(0,1,length.out=100)
#' elements
like <- dbinom(Y,n,grid); like=like/sum(like) #standardize
prior<- dbeta(grid,a,b); prior=prior/sum(prior) #standardize
post <- like*prior; post<-post/sum(post)

df<-data.frame(grid=rep(grid, 3),
               probas=c(prior, like, post),
               value=rep(c("prior", "like", "post"), each=100))

ggplot(df, aes(x=grid, y=probas))+
  geom_density(aes(fill=as.factor(value)), alpha=.3, stat="identity") +
  ylab("Density")+
  scale_x_continuous("Probability")+
  scale_fill_viridis_d("") +
  theme_minimal()
```



Calculation of the posterior mean and 95% interval:

```
n=24;Y=1;
#' beta prior
a=1; b=4
round((Y+a)/(n+a+b), 4)
```

```
## [1] 0.069
```

```
round(qbeta(c(0.025, 0.975), Y+a, n-Y+b), 4)
```

```
## [1] 0.0088 0.1835
```

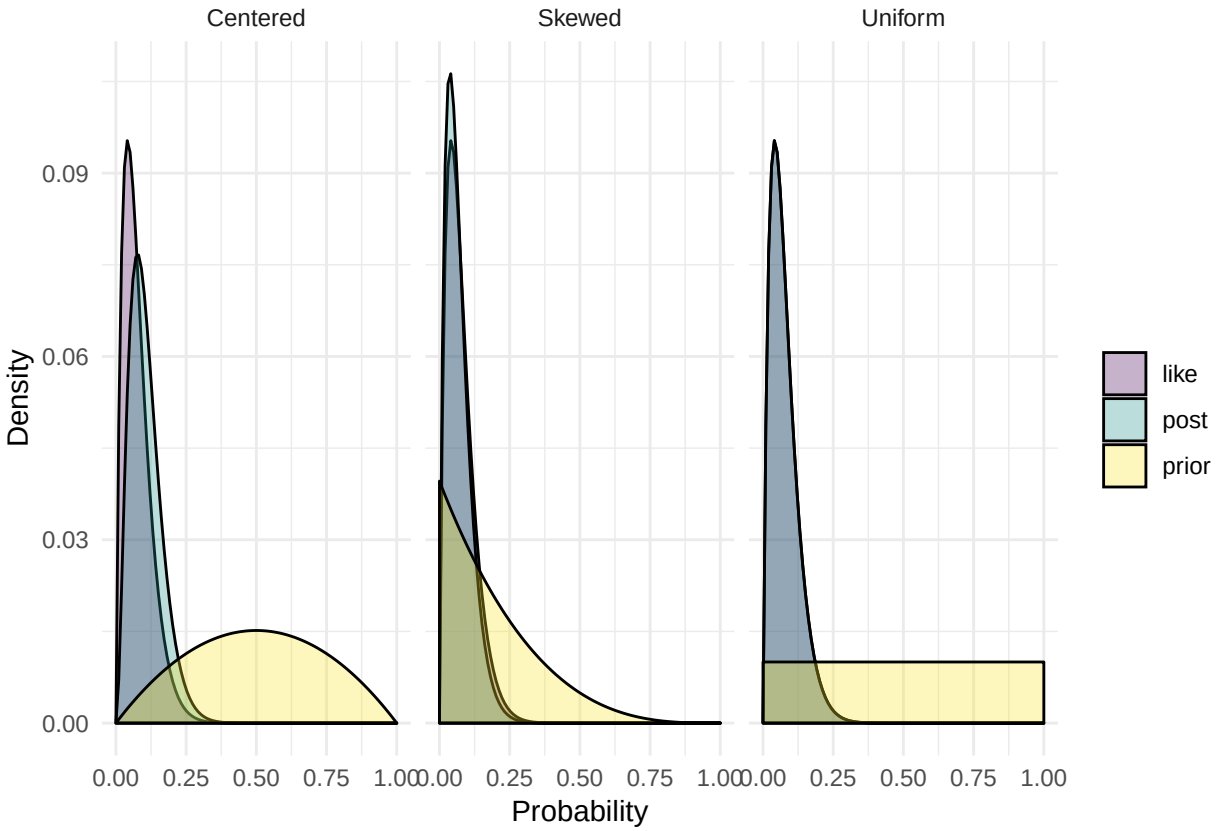
Calculation of the posterior probability that p>0.15:

```
n=24;Y=1;
#' beta prior
a=1; b=4
round(1-pbeta(0.15, Y+a, n-Y+b), 4)
```

```
## [1] 0.0627
```

Had we chosen other priors ...

```r
#' total sample size (n); successes (Y)
n=24;Y=1;
#' beta prior
a1=1; b1=1 #uniform
a2=2; b2=2 #weakly inf centered on 0.5
a3=1; b3=4 #weakly inf centered on 0.2
#' prop. grid
grid<- seq(0,1,length.out=100)
#' elements
like <- dbinom(Y,n,grid); like=like/sum(like) #standardize
prior1<- dbeta(grid,a1,b1); prior1=prior1/sum(prior1) #standardize
prior2<- dbeta(grid,a2,b2); prior2=prior2/sum(prior2) #standardize
prior3<- dbeta(grid,a3,b3); prior3=prior3/sum(prior3) #standardize
post1 <- like*prior1; post1<-post1/sum(post1)
post2 <- like*prior2; post2<-post2/sum(post2)
post3 <- like*prior3; post3<-post3/sum(post3)

df<-data.frame(grid=rep(rep(grid, 3), 3),
               probas=c(c(prior1, like, post1), c(prior2, like, post2), c(prior3, like, post3)),
               value=rep(rep(c("prior", "like", "post"), each=100), 3),
               scenario=rep(c("Uniform", "Centered", "Skewed"), each=300))

ggplot(df, aes(x=grid, y=probas))+
  geom_density(aes(fill=as.factor(value)), alpha=.3, stat="identity") +
  facet_wrap(~scenario, nrow=1)+
  ylab("Density")+
  scale_x_continuous("Probability")+
  scale_fill_viridis_d("") +
  theme_minimal()
```



The 95% HDI can be calculated as the highest density interval for posterior samples. Unlike equal-tailed intervals that exclude 2.5% from each tail of the distribution, the HDI is not equal-tailed and therefore always includes the mode(s) of posterior distributions.

```
n=24;Y=1;
#' beta prior
a=1; b=4
#' prop. grid
grid<- seq(0,1,length.out=100)
zhdi<-hdi(qbeta(grid, Y+a, n-Y+b), 0.95)
round(zhdi, 4)
```

```
##  lower  upper
## 0.0000 0.1663
## attr(,"credMass")
## [1] 0.95
```

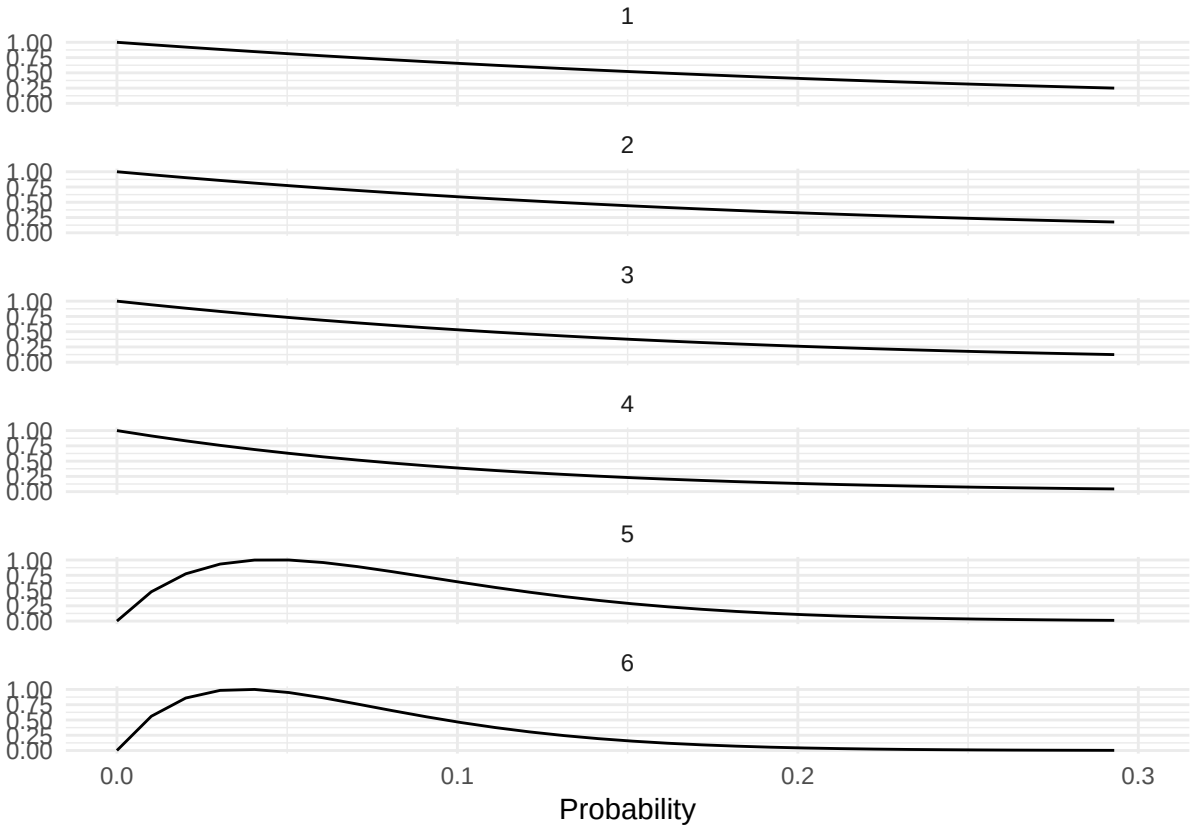
Sequential updates

As cohorts are providing staggered (with increasing dose levels), the prior and posterior distributions can actually be updated as time goes.

```
d<-data.frame(Cohort=c(1, 2, 3, 4, 5, 6),
          Dose=c(65, 160, 400, 400, 800, 1200),
          ncum=c(1, 2, 3, 6, 19, 24),
```

```
          Ycum=c(0, 0, 0, 0, 1, 1),
          a=1, b=4) %>%
  mutate(shap1=Ycum+a, shap2=ncum-Ycum+b)

set.seed(123)

#Use of pmap from https://github.com/jennybc/row-oriented-workflows/blob/master/ex08_nesting-is-good.md
my_dbeta <- function(...) {
  l <- list(...)
  dbeta(x=grid, shape1=l$shap1, shape2=l$shap2)
}

set.seed(123)
lbeta<-pmap(d, my_dbeta)
lbeta2<-map(lbeta, function(x) x/max(x))
dfbeta<-as.data.frame(t(do.call(rbind, lbeta2))) %>%
gather(VCohort, probas) %>%
mutate(grid=rep(grid, 6), Cohort=as.character(substr(VCohort, 2, 2)))

ggplot(dfbeta, aes(x=grid, y=probas))+
  geom_line()+
  facet_wrap(~Cohort, ncol=1)+
  scale_x_continuous("Probability", limits=c(0, .3))+
  ylab("")+
  theme_minimal()
```

```
## Warning: Removed 70 rows containing missing values (geom_path).
```



Related posts include: https://cran.r-project.org/web/packages/rstanarm/vignettes/betareg.html http://www.sumsar.net/blog/2018/12/visualizing-the-beta-binomial/ https://www4.stat.ncsu.edu/~reich/ABA/notes/BetaBinom.pdf https://en.wikipedia.org/wiki/Beta-binomial_distribution http://www.koenbro.com/the-beta-distribution/#highest-density-interval