Normal AB Testing

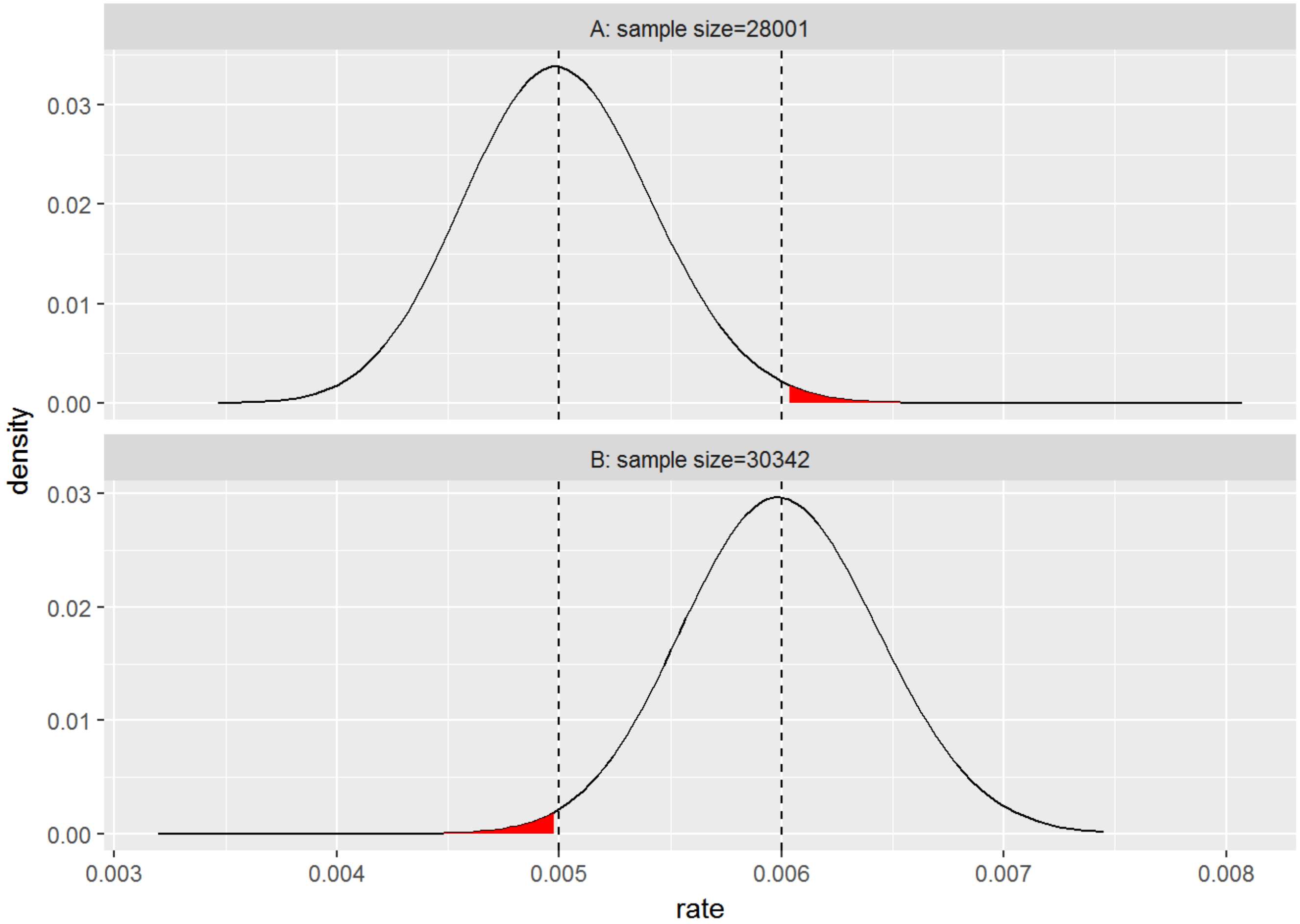
R Markdown

A/B testing R code

```
library(gtools)
library(ggplot2)

# q>p, compute the probability of a
# p-rate process measuring as q-rate
# or better in n steps
pSignificanceError <- function(p,q,n) {
  pbinom(ceiling(q*n)-1,prob=p,size=n,lower.tail=FALSE)
}

# q>p, compute the proability of a
# q-rate process measuring as p-rate
# or lower in n steps
pPowerError <- function(p,q,n) {
  pbinom(floor(p*n),prob=q,size=n,lower.tail=TRUE)
}

designExperiment <- function(pA,pB,pError,pAUpper=pB,pBLower=pA) {
   aSoln <- binsearch(
      function(k) {
         pSignificanceError(pA,pAUpper,k) - pError},
      range=c(100,1000000))
   nA <- max(aSoln$where)
   print(paste('nA',nA))

   bSoln <- binsearch(
      function(k) {
         pPowerError(pBLower,pB,k) - pError},
      range=c(100,1000000))
   nB <- max(bSoln$where)
   print(paste('nB',nB))


   low = floor(min(pA*nA,pB*nB))
   high = ceiling(max(pA*nA,pB*nB))
   width = high-low
   countRange <- (low-width):(high+width)

   dA <- data.frame(count=countRange)
   dA$group <- paste('A: sample size=',nA,sep='')
   dA$density <- dbinom(dA$count,prob=pA,size=nA)
   dA$rate <- dA$count/nA
   dA$error <- dA$rate>=pAUpper
   dB <- data.frame(count=countRange)
   dB$group <- paste('B: sample size=',nB,sep='')
   dB$density <- dbinom(dB$count,prob=pB,size=nB)
   dB$rate <- dB$count/nB
   dB$error <- dB$rate<=pBLower
   d <- rbind(dA,dB)

   plot = ggplot(data=d,aes(x=rate,y=density)) +
     geom_line() +
     geom_ribbon(data=subset(d,error),
        aes(ymin=0,ymax=density),fill='red') +
     facet_wrap(~group,ncol=1,scales='free_y') +
     geom_vline(xintercept=pAUpper,linetype=2) +
     geom_vline(xintercept=pBLower,linetype=2)
   list(nA=nA,nB=nB,plot=plot)
}


r1 <- designExperiment(pA=0.005,pB=0.006,pError=0.01)
```

```
## [1] "nA 28001"
## [1] "nB 30342"
```

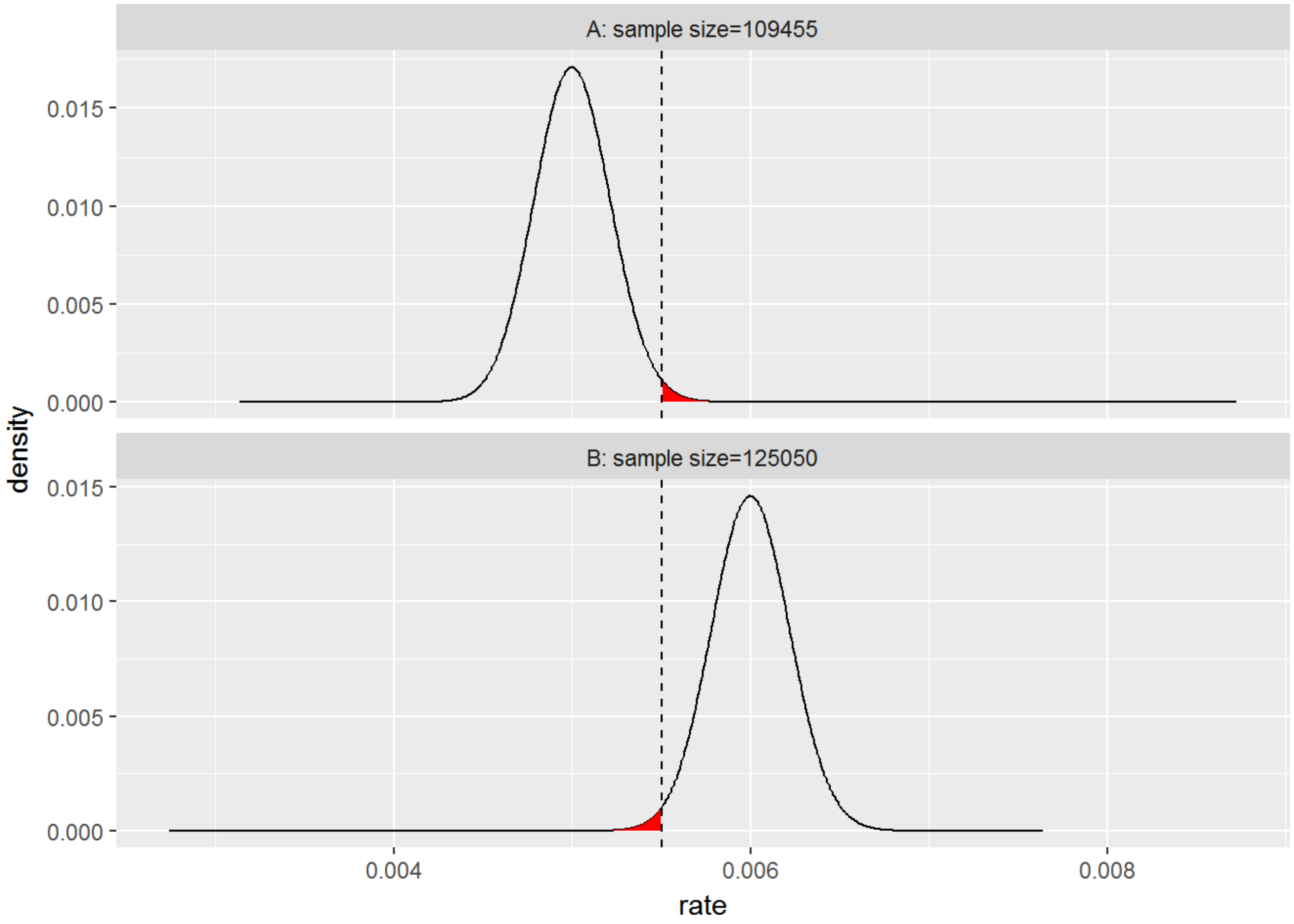
```
print(r1$plot)
```



```
r2 <- designExperiment(pA=0.005,pB=0.006,pError=0.01,pAUpper=0.0055,pBLower=0.0055)
```

```
## [1] "nA 109455"
## [1] "nB 125050"
```

```
print(r2$plot)
```



```
r3 <- designExperiment(pA=0.005,pB=0.006,pError=0.005,pAUpper=0.0055,pBLower=0.0055)
```

```
## [1] "nA 135091"
## [1] "nB 154805"
```