Computing assignment 3

Alyssa Vanderbeek (amv2187)

4 November 2019

```
# N = total sample size
# block_size = block size for randomization
# p.A = response probability for arm A
# p.B = response probability for arm B
# attenuation = attenuation value for Thompson's law (default = 1)

block_design = function(N, block_size, p.A, p.B, attenuation = 1){

  n_blocks = N/block_size

  Y = list(arm = c(),
           response = c())

  # randomization probability for block 1
  rand.A = 0.5

  for (i in 1:n_blocks) {
    # randomize subjs based on current randomization probability
    rand = rbinom(block_size, 1, p = rand.A) # 1 = A, 0 = B
    n.A = sum(rand)
    n.B = block_size - sum(rand)

    # generate responses on each arm based on true reponse probabilities
    A.response = rbinom(n.A, 1, p.A)
    B.response = rbinom(n.B, 1, p.B)

    # record observed data
    Y$arm = append(Y$arm, values = c(rep("A", n.A),
                                     rep("B", n.B)))
    Y$response = append(Y$response, values = c(A.response, B.response))

    # Thompson's rule; update randomization probabilities based on observed outcomes
    a.A = 1 + sum(A.response)
    b.A = 1 + n.A - sum(A.response)
    Th.A = rbeta(1e4, a.A, b.A)

    a.B = 1 + sum(B.response)
    b.B = 1 + n.B - sum(B.response)
    Th.B = rbeta(1e4, a.B, b.B)

    r = mean(Th.A > Th.B)
    rand.A = r^attenuation / ((r^attenuation) + ((1 - r)^attenuation))
  }

  tbl = table(Y$response, Y$arm)
  test = chisq.test(tbl)
  return(test$p.value)
}
```

Problem 1

```
sims = 1e3

set.seed(1)
trials.1a = sapply(1:sims, block_design, N = 120, block_size = 30, p.A = 0.5, p.B = 0.5)

rej = mean(trials.1a < 0.05)
```

Under the scenario where $p_A = p_B = 0.5$, the block design with chi-square test rejects the null in favor of the alternative that there is treatment-dependent response 1.5% of the time.

Problem 2

```
set.seed(1)
trials.2a = sapply(1:sims, block_design, N = 120, block_size = 30, p.A = 0.3, p.B = 0.6)

rej = mean(trials.2a < 0.05)
```

(a) Under the scenario where $p_A = 0.3$ and $p_B = 0.6$, the block design with chi-square test rejects the null in favor of the alternative that there is treatment-dependent response 61% of the time.

```
nonadaptive_rand <- function(n.A, n.B, p.A, p.B){

  Y = list(arm = c(),
           response = c())

  # generate responses on each arm based on true reponse probabilities
  A.response = rbinom(n.A, 1, p.A)
  B.response = rbinom(n.B, 1, p.B)

  # record observed data
  Y$arm = append(Y$arm, values = c(rep("A", n.A),
                                   rep("B", n.B)))
  Y$response = append(Y$response, values = c(A.response, B.response))

  tbl = table(Y$response, Y$arm)
  test = chisq.test(tbl)
  return(test$p.value)
}

set.seed(1)
trials.2b = sapply(1:sims, function(i){ nonadaptive_rand(n.A = 60, n.B = 60, p.A = 0.3, p.B = 0.6) } )

pow = mean(trials.2b < 0.05)
```

(b) The non-adaptive RCT has power of 0.897 under the scenario where $p_A = 0.3$ and $p_B = 0.6$.

Problem 3

```r
data <-
  array(c(3, 1, 7, 9,
          8, 1, 15, 6,
          16, 2, 10, 2,
          33, 1, 6, 0),
        dim = c(2, 2, 4),
        dimnames = list(
          Arm = c("0", "1"),
          Response = c("Response", "No response"),
          Block = c("1", "2", "3", "4")))
mantelhaen.test(data)
```

```
##
##  Mantel-Haenszel chi-squared test with continuity correction
##
## data:  data
## Mantel-Haenszel X-squared = 1.1423, df = 1, p-value = 0.2852
## alternative hypothesis: true common odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.6953671 8.3525664
## sample estimates:
## common odds ratio
##              2.41
```

Problem 4

```r
# N = total sample size
# block_size = block size for randomization
# p.A = response probability for arm A
# p.B = response probability for arm B
# attenuation = attenuation value for Thompson's law (default = 1)

block_design_v2 = function(N, block_size, attenuation = 1, test){

  n_blocks = N/block_size

  Y = list(block = c(),
           arm = c(),
           response = c())

  # randomization probability for block 1
  rand.A = 0.5

  for (i in 1:n_blocks) {
    if (i %in% c(1, 2)) {
      p.A = p.B = 0.3
    } else {
      p.A = p.B = 0.6
    }
    # randomize subjs based on current randomization probability
    rand = rbinom(block_size, 1, p = rand.A) # 1 = A, 0 = B
    n.A = sum(rand)
    n.B = block_size - sum(rand)

    # generate responses on each arm based on true reponse probabilities
    A.response = rbinom(n.A, 1, p.A)
    B.response = rbinom(n.B, 1, p.B)

    # record observed data
    Y$block = append(Y$block, values = c(rep(i, block_size)))
    Y$arm = append(Y$arm, values = c(rep("A", n.A),
                                     rep("B", n.B)))
    Y$response = append(Y$response, values = c(A.response, B.response))

    # Thompson's rule; update randomization probabilities based on observed outcomes
    a.A = 1 + sum(A.response)
    b.A = 1 + n.A - sum(A.response)
    Th.A = rbeta(1e4, a.A, b.A)

    a.B = 1 + sum(B.response)
    b.B = 1 + n.B - sum(B.response)
    Th.B = rbeta(1e4, a.B, b.B)

    r = mean(Th.A > Th.B)
    rand.A = r^attenuation / ((r^attenuation) + ((1 - r)^attenuation))
  }

  return(Y)

}
```

```r
set.seed(1)
trials.4 = lapply(1:sims, block_design_v2, N = 120, block_size = 30, test = "Chi")

Chi = sapply(trials.4, function(i){
  tbl = table(i$response, i$arm)
  test = chisq.test(tbl)
  return(test$p.value)
})

rej = mean(Chi < 0.05)
```

In conducting a block design with a chi-square test, we reject the null in favor of the alternative that there is a treatment-dependent difference in response 53.1% of the time.

```r
MH = sapply(trials.4, function(i){
  data <- table(i$arm, i$response, i$block)
  test = mantelhaen.test(data)
  return (test$p.value)
})

rej = mean(MH < 0.05)
```

By contrast, under a Mantel-Haenzel test, we reject the null 1.3% of the time. This stark contrast to the Chi-square test under the same scenario shows just how poorly Chi-square is at accounting for time trends.