```
#==============================================================================
#
# Sample Size Considerations for the ERIC trial
#
# MANUSCRIPT: Effectiveness of an intensive care telehealth programme
# to improve process quality (ERIC): a multicentre stepped wedge CRT
#
# 20.11.2022
#==============================================================================
#
# n: number of centres
# m: number of sequences
# OR: effect of infection
# m: number of patients per center
# sigma.c: standard deviation of centre random effect
# sigma.p: standard deviation of patient random effect
# p.b: baseline rate
#
#
# Allocated in three sequences.
# Sequence 1 has control in period 1 and intervention in periods 2,3,4
# Sequence 2 has control in period 1,2 and intervention in periods 3,4
# Sequence 3 has control in period 1,2,3 and intervention in periods 4
#
# Allocating m patients per period per cluster (4xm total number of patients per cluster)
# Each patient will stay for 4 days. This results in 4x4xm observations per cluster
#
# Demonstrating the random effects
#
#

p.b <- 0.6
l.b <- log(0.6 / 0.4)
l.b.c <- rnorm(100, l.b, 1.5)
p.b.c <- exp(l.b.c) / (1 + exp(l.b.c))
boxplot(p.b.c)
#
#
# Function to generate cluster data
#
m<-10
#
cluster.data.rfc <- function(cluster.ID = 1,
                             n.control = m,
                             n.interve = 3 * m,
                             p.b = 0.6,
                             OR = 1.556,
                             sigma.p = 0.5,
                             sigma.c = 1.5,
                             n = 12
                             )
{
  #
  # center specific rates under control and intervention: n centers
  #
  logit.p.b <- log(p.b / (1 - p.b))
  logit.centres.control <- logit.p.b + rnorm(n, 0, sigma.c)
  logit.centres.interve <- logit.centres.control + log(OR)
  #
  # control - taking patient specific variability under consideration
  #
  period.1.logits <- logit.centres.control[cluster.ID] + rnorm(n.control, 0, sigma.p)
  odds.1 <- exp(period.1.logits)
  probs.1 <- odds.1 / (1 + odds.1)
  #
  # individual patient IDs
  #
  ppp.1 <- (cluster.ID - 1) * 4 * m + 1:n.control
  patID.1 <- rep(ppp.1, rep(4, n.control))    # patients stay for 4 days.
  ll.1 <- length(patID.1)
  probs.1.obs <- rep(probs.1, rep(4, n.control))
  obs.1 <- rbinom(ll.1, 1, probs.1.obs)       # daily outcome per patient
  #
  # intervention
```

```
  #
  period.2.logits <- logit.centres.interve[cluster.ID] + rnorm(n.interve, 0, sigma.p)
  odds.2 <- exp(period.2.logits)
  probs.2 <- odds.2 / (1 + odds.2)
  ppp.2 <- (cluster.ID - 1) * 4 * m + n.control + (1:n.interve)
  patID.2 <- rep(ppp.2, rep(4, n.interve))
  ll.2 <- length(patID.2)
  probs.2.obs <- rep(probs.2, rep(4, n.interve))
  obs.2 <- rbinom(ll.2, 1, probs.2.obs)
  #
  patID.c.1 <- c(patID.1, patID.2)
  outco.c.1 <- c(obs.1, obs.2)
  treat.c.1 <- rep(c(0, 1), c(4 * n.control, 4 * n.interve))
  clust.c.1 <- rep(cluster.ID, 4 * 4 * m)
  res <- cbind(clust.c.1, patID.c.1, treat.c.1, outco.c.1)
  res <- data.frame(res)
  names(res) <- c("Cluster", "PatID", "Treatment", "Outcome")
  return(res)
}
#

dat.clust.simul.rfc <- function(i = 1)
{
  #
  dat.1 <- cluster.data.rfc(cluster.ID = 1, n.control = m, n.interve = 3*m)
  dat.2 <- cluster.data.rfc(cluster.ID = 2, n.control = m, n.interve = 3*m)
  dat.3 <- cluster.data.rfc(cluster.ID = 3, n.control = m, n.interve = 3*m)
  dat.4 <- cluster.data.rfc(cluster.ID = 4, n.control = m, n.interve = 3*m)
  #
  #
  dat.5 <- cluster.data.rfc(cluster.ID = 5, n.control = 2*m, n.interve = 2*m)
  dat.6 <- cluster.data.rfc(cluster.ID = 6, n.control = 2*m, n.interve = 2*m)
  dat.7 <- cluster.data.rfc(cluster.ID = 7, n.control = 2*m, n.interve = 2*m)
  dat.8 <- cluster.data.rfc(cluster.ID = 8, n.control = 2*m, n.interve = 2*m)
  #
  #
  dat.9  <- cluster.data.rfc(cluster.ID =  9, n.control = 3*m, n.interve = m)
  dat.10 <- cluster.data.rfc(cluster.ID = 10, n.control = 3*m, n.interve = m)
  dat.11 <- cluster.data.rfc(cluster.ID = 11, n.control = 3*m, n.interve = m)
  dat.12 <- cluster.data.rfc(cluster.ID = 12, n.control = 3*m, n.interve = m)
  #
  data.all <- rbind(dat.1,dat.2,dat.3,dat.4,dat.5,dat.6,dat.7,dat.8,dat.9,dat.10,dat.11,dat.12)
  return(data.all)
}
#
#
#
dat.clust <- dat.clust.simul.rfc()
table(dat.clust$Treatment, dat.clust$Outcome)
#
#
#
library(lme4)
#
#
#
glmer.1 <- glmer(Outcome ~ Treatment + (1 | Cluster) + (1 | PatID),
                 family = binomial(), data = dat.clust)
summary(glmer.1)$coefficient[2, ]


glmer.simulate.rfc <- function(i = 1, m = 30)
{
  require(lme4)
  dat.clust <- dat.clust.simul.rfc()
  glmer.1 <-
    glmer(Outcome ~ Treatment + (1 | Cluster) + (1 | PatID),
          family = binomial(), data = dat.clust)
  return(summary(glmer.1)$coefficient[2, ])
}
#
#
#
glmer.simulate.rfc()
```

```
#
#
#
ll.simul <- 1000
glmer.simulate.res <- lapply(1:ll.simul, glmer.simulate.rfc)
glmer.simulate.res <- matrix(unlist(glmer.simulate.res), byrow = T, ncol = 4)
power.res <- sum(glmer.simulate.res[, 4] < 0.00625) / ll.simul
power.res
#
#
#
# 95% CI - odds based
n.a <- ll.simul * (1 - power.res)
n.b <- ll.simul * power.res
o.power <- n.b / n.a
log.odds.95.CI <- log(o.power) + c(-1, 1) * 1.96 * sqrt(1 / n.a + 1 / n.b)
odds.95.CI <- exp(log.odds.95.CI)
power.95.CI <- odds.95.CI / (1 + odds.95.CI)
```