Critical Remarks on Reference-Scaled Average Bioequivalence

Helmut Schütz[1,2], Detlew Labes[3], Martin J Wolfsegger[4]

[1] BEBAC. Neubaugasse 36/11, 1070 Vienna, Austria
[2] Institute of Medical Statistics of the Medical University of Vienna, Austria, ORCID ID 0000-0002-1167-7880;
[3] Consultant. Lössauer Straße 18, 13055 Berlin, Germany, ORCID ID 0000-0003-2169-426X;
[4] Baxalta Innovations GmbH, a Takeda company. Donau-City-Straße 7, 1220 Vienna, Austria, ORCID ID 0000-0003-2419-4324.

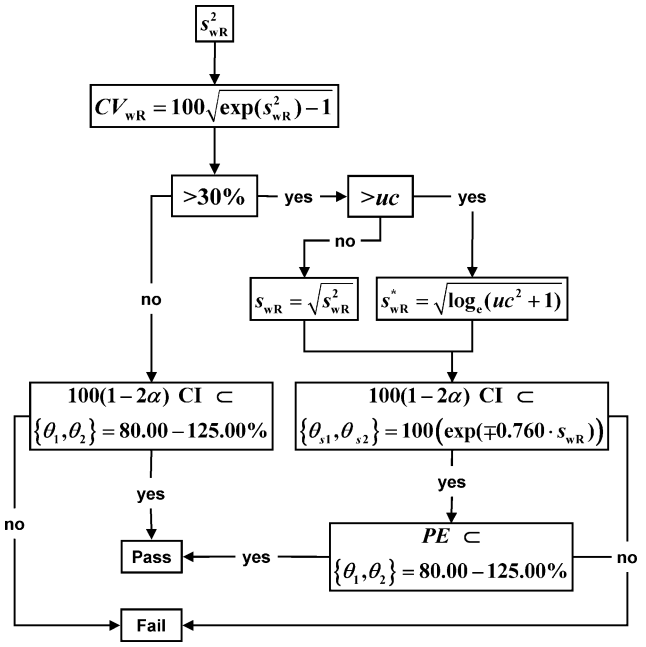

Fig. S1. Framework of Average Bioequivalence with Expanding Limits.

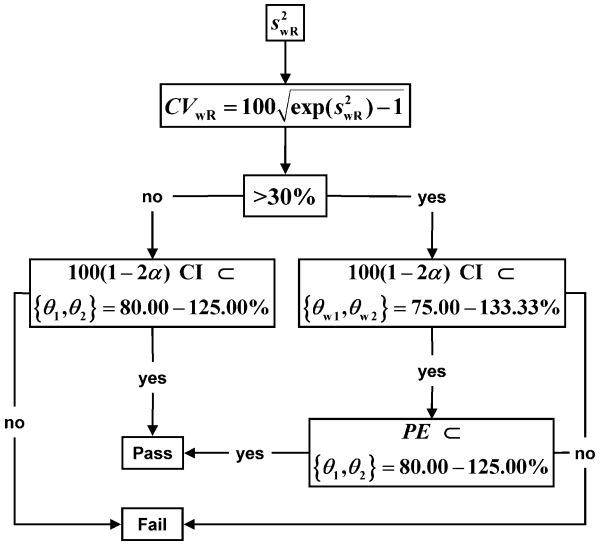

Fig. S2. Framework of the Gulf Cooperation Council.

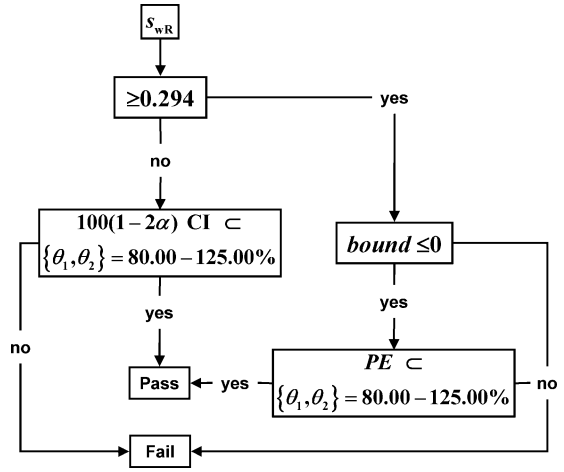

Fig. S3. Framework of Reference-Scaled Average Bioequivalence.

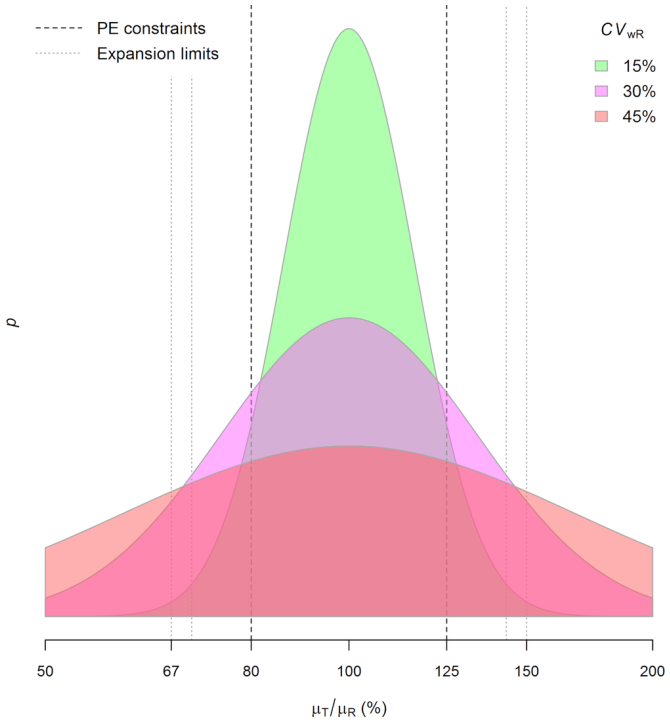

Fig. S4. Larger deviations between the geometric mean ratios are a direct consequence of higher variability. The point estimate constraint – together with the upper cap in most jurisdictions – leads to a truncated distribution whose statistical properties are essentially unknown. Hence, the test assuming the normal distribution of $\log_e$-transformed data is not correct in the strict sense.

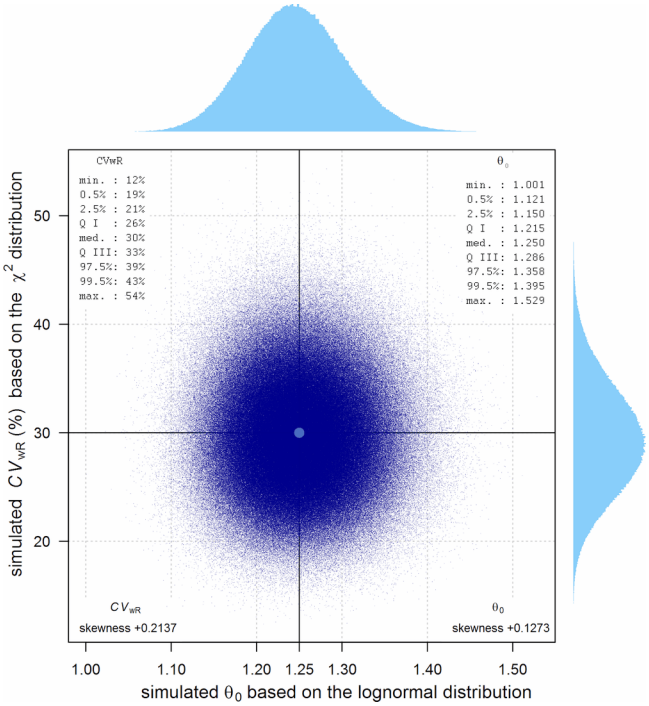

Fig. S5. Illustrative example of skewed distributions. One million simulations with true CV_{wR} = 30% and θ_0 = 1.25 (n = 24).

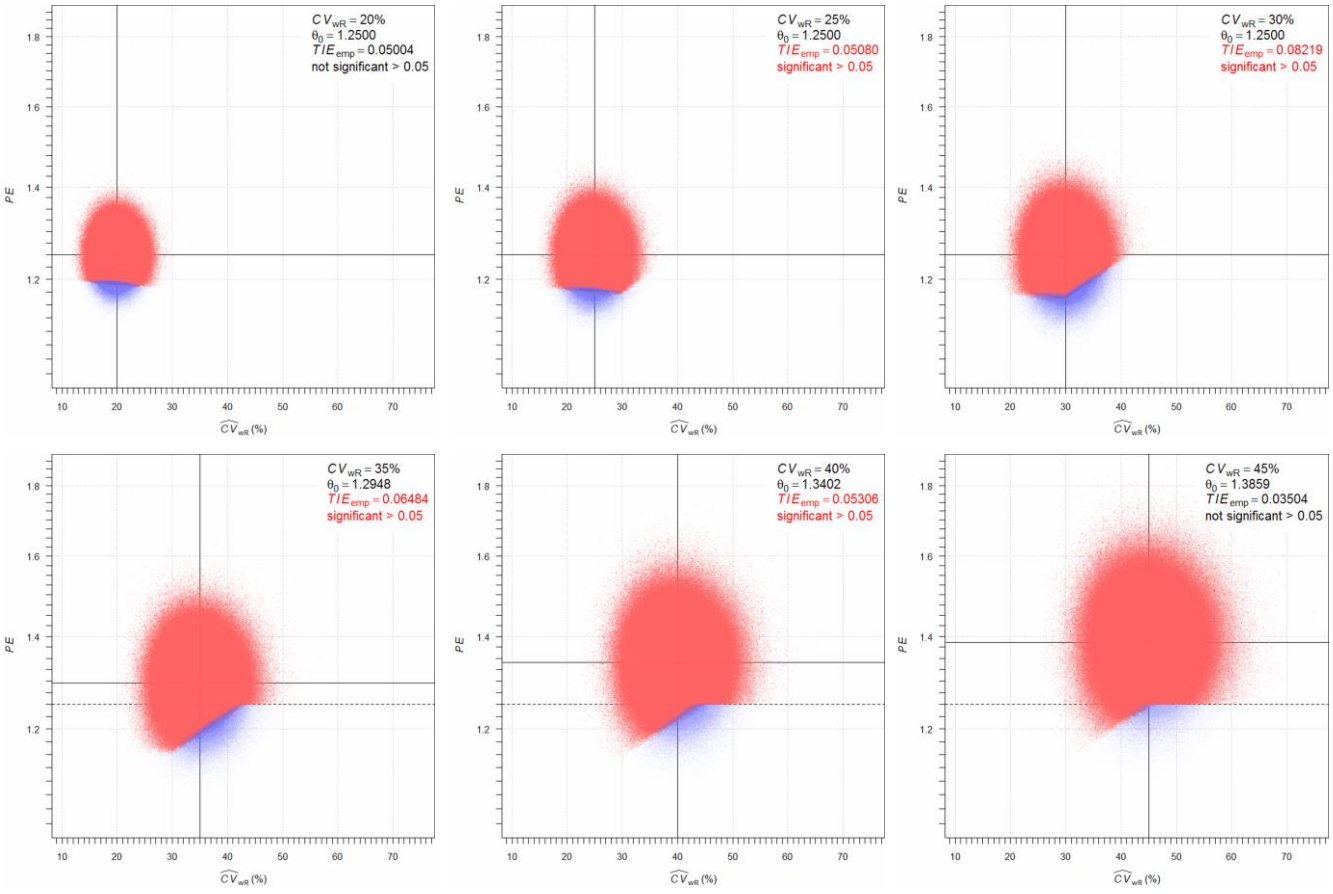

Fig. S6. One million simulations of studies in a two sequence, four period full replicate design (n = 48), assessment for ABEL (uc = 50%). True CV_{wR} = 20 – 45% → θ_0 = θ_{0s2} (solid lines). If applicable, upper PE-constraint 1.25 (dashed line). Y-axis in log-scale. Red dots represent studies which failed BE (null hypothesis correctly accepted), blue dots represent studies which passed BE (null hypothesis falsely rejected). Empiric Type I Error = number of passing studies / 10^6, significance limit of the binomial test 0. 050360.

R-script to reproduce Table 3

```
#########################################################
# Parametric 95% confidence interval of CVwR based on #
# the chi²-distribution of its associated variance.   #
#########################################################
library(PowerTOST)       # at least v1.1-0 (2013-02-08)
n    <- seq(24L, 48L, 12L)
CVwR <- seq(20L, 65L,  5L)
txt  <- "95% confidence intervals of CVwR\ndependent on the sample size.\n"
tmp  <- data.frame(n = rep(n, each = length(CVwR)), CVwR = CVwR,
                   lo = NA_real_, hi = NA_real_)
for (i in 1:nrow(tmp)) {
  tmp[i, 3:4] <- round(100*CVCL(CV = tmp$CVwR[i] / 100, df = tmp$n[i] - 2,
                                alpha = 0.05, side = "2-sided"), 1)
}
tmp <- reshape(tmp, direction = "wide", idvar = "CVwR", timevar = "n")
res <- data.frame(CVwR = CVwR, n.24 = NA_real_, n.36 = NA_real_, n.48 = NA_real_)
for (i in seq_along(CV)) {
  res$n.24[i] <- sprintf("%5.1f \u2013 %5.1f", tmp[i, 2], tmp[i, 3])
  res$n.36[i] <- sprintf("%5.1f \u2013 %5.1f", tmp[i, 4], tmp[i, 5])
  res$n.48[i] <- sprintf("%5.1f \u2013 %5.1f", tmp[i, 6], tmp[i, 7])
}
names(res)[2:4] <- paste0("    n = ", n)
cat(txt); print(res, right = FALSE, row.names = FALSE)
```

Should give:

```
   95% confidence intervals of CVwR
   dependent on the sample size.
    CVwR     n = 24        n = 36        n = 48
    20    15.4 –  28.6  16.1 –  26.4  16.6 –  25.3
    25    19.2 –  35.9  20.1 –  33.1  20.7 –  31.7
    30    23.0 –  43.4  24.1 –  39.9  24.8 –  38.2
    35    26.8 –  51.0  28.0 –  46.8  28.8 –  44.7
    40    30.5 –  58.8  31.9 –  53.9  32.9 –  51.4
    45    34.1 –  66.8  35.8 –  61.0  36.9 –  58.1
    50    37.8 –  75.1  39.6 –  68.3  40.8 –  65.0
    55    41.4 –  83.5  43.4 –  75.8  44.7 –  71.9
    60    44.9 –  92.3  47.2 –  83.4  48.6 –  79.0
    65    48.4 – 101.3  50.9 –  91.2  52.5 –  86.2
```

R-script to reproduce Table 4 (and sample sizes up to 144)

```
################################
# Empiric Type I Error at the  #
# CVwR of maximum inflation.   #
################################
library(PowerTOST)             # At least v1.5-3 (2021-01-18) for GCC
design  <- "2x2x4"             # 2-sequence 4-period full replicate
n       <- seq(24L, 144L, 12L) # Sample sizes
CV      <- c(rep(0.30, 5 * length(n)), rep(se2CV(0.25), length(n))) # True CVwR
methods <- c("ABE",            # ABE   (all jurisdictions)
             "ABEL1",          # ABEL  (EMA and others)
             "ABEL2",          # ABEL  (Health Canada)
             "ABEL3",          # ABEL  (GCC)
             "RSABE1",         # RSABE ('implied limits')
             "RSABE2")         # RSABE ('desired consumer risk model')
result  <- data.frame(method = rep(methods, each = length(n)),
                      reg = c(rep(NA, length(n)),
                              rep("EMA", length(n)),
                              rep("HC", length(n)),
                              rep("GCC", length(n)),
                              rep("FDA", 2 * length(n))),
                      CV  = CV, swR = CV2se(CV), n = n, U = 1.25,
                      TIE = NA_real_, SE = NA_real_, signif = "  no ")
limit   <- binom.test(0.05 * 1e6L, 1e6L, alternative = "less")$conf.int[[2]]
pb      <- txtProgressBar(0, 1, 0, width = NA, style = 3)
for (i in 1:nrow(result)) {
  if (result$method[i] == "ABE") {    # All jurisdictions (exact, no simulations)
    result$TIE[i] <- power.TOST(CV = result$CV[i], n = result$n[i], design = design,
                                theta0 = result$U[i])
  }
  if (result$method[i] == "ABEL1") {  # EMA and others
    result$U[i]   <- scABEL(CV = result$CV[i], regulator = result$reg[i])[["upper"]]
    result$TIE[i] <- power.scABEL(CV = result$CV[i], n = result$n[i], design = design,
                                  theta0 = result$U[i], regulator = result$reg[i], nsims = 1e6L)
  }
  if (result$method[i] == "ABEL2") {  # Health Canada
    result$U[i]   <- scABEL(CV = result$CV[i], regulator = result$reg[i])[["upper"]]
    result$TIE[i] <- power.scABEL(CV = result$CV[i], n = result$n[i], design = design,
                                  theta0 = result$U[i], regulator = result$reg[i], nsims = 1e6L)
  }
```

```
  if (result$method[i] == "ABEL3") {  # Gulf Cooperation Council
    result$U[i]   <- scABEL(CV = result$CV[i], regulator = result$reg[i])[["upper"]]
    result$TIE[i] <- power.scABEL(CV = result$CV[i], n = result$n[i], design = design,
                                  theta0 = result$U[i], regulator = result$reg[i], nsims = 1e6L)
  }
  if (result$method[i] == "RSABE1") { # FDA 'implied limits'
    result$U[i]   <- scABEL(CV = result$CV[i], regulator = result$reg[i])[["upper"]]
    result$TIE[i] <- power.RSABE(CV = result$CV[i], n = result$n[i], design = design,
                                 theta0 = result$U[i], regulator = result$reg[i], nsims = 1e6L)
    }
  if (result$method[i] == "RSABE2") { # FDA 'desired consumer risk model'
    if (result$swR[i] <= 0.25) {
      result$U[i] <- scABEL(CV = result$CV[i], regulator = result$reg[i])[["upper"]]
    } else {
      result$U[i] <- exp(result$swR[i] * log(1.25) / 0.25)
    }
    result$TIE[i] <- power.RSABE(CV = result$CV[i], n = result$n[i], design = design,
                                 theta0 = result$U[i], nsims = 1e6L)
  }
  setTxtProgressBar(pb, i / nrow(result))
}
result$signif[result$TIE > sign.limit] <- " yes "
result$SE  <- sqrt(0.5 * result$TIE / 1e6)
result$SE  <- sprintf("%.6f", result$SE)
ABE.idx    <- which(result$method == "ABE")
ABEL1.idx  <- which(result$method == "ABEL1")
ABEL2.idx  <- which(result$method == "ABEL2")
ABEL3.idx  <- which(result$method == "ABEL3")
RSABE1.idx <- which(result$method == "RSABE1")
RSABE2.idx <- which(result$method == "RSABE2")
result$CV  <- sprintf("%.3f%%", 100 * result$CV)
result$TIE <- sprintf("%.4f", result$TIE)
result$method[ABE.idx[1]]     <- "ABE (all jurisdictions)"
result$method[ABE.idx[-1]]    <- ""
result$method[ABEL1.idx[1]]   <- "ABEL (EMA and others)"
result$method[ABEL1.idx[-1]]  <- ""
result$method[ABEL2.idx[1]]   <- "ABEL (Health Canada)"
result$method[ABEL2.idx[-1]]  <- ""
result$method[ABEL3.idx[1]]   <- "ABEL (GCC)"
result$method[ABEL3.idx[-1]]  <- ""
result$method[RSABE1.idx[1]]  <- "RSABE (\u2018implied limits\u2019)"
result$method[RSABE1.idx[-1]] <- ""
result$method[RSABE2.idx[1]]  <- "RSABE (\u2018desired consumer risk model\u2019)"
result$method[RSABE2.idx[-1]] <- ""
result                        <- result[, -c(2, 4, 6)]
names(result)[2:6]            <- c("  CVwR", " n", " TIE", "   SE", "sign.")
print(result, row.names = FALSE, right = FALSE)
```

Should give:

```
   method                              CVwR   n   TIE      SE    sign.
   ABE (all jurisdictions)           30.000%  24 0.0500 0.000158   no
                                     30.000%  36 0.0500 0.000158   no
                                     30.000%  48 0.0500 0.000158   no
                                     30.000%  60 0.0500 0.000158   no
                                     30.000%  72 0.0500 0.000158   no
                                     30.000%  84 0.0500 0.000158   no
                                     30.000%  96 0.0500 0.000158   no
                                     30.000% 108 0.0500 0.000158   no
                                     30.000% 120 0.0500 0.000158   no
                                     30.000% 132 0.0500 0.000158   no
                                     30.000% 144 0.0500 0.000158   no
   ABEL (EMA and others)             30.000%  24 0.0804 0.000200  yes
                                     30.000%  36 0.0819 0.000202  yes
                                     30.000%  48 0.0823 0.000203  yes
                                     30.000%  60 0.0827 0.000203  yes
                                     30.000%  72 0.0831 0.000204  yes
                                     30.000%  84 0.0836 0.000204  yes
                                     30.000%  96 0.0834 0.000204  yes
                                     30.000% 108 0.0839 0.000205  yes
                                     30.000% 120 0.0838 0.000205  yes
                                     30.000% 132 0.0843 0.000205  yes
                                     30.000% 144 0.0840 0.000205  yes
```

```
   ABEL (Health Canada)                  30.000%  24 0.0841 0.000205  yes
                                         30.000%  36 0.0846 0.000206  yes
                                         30.000%  48 0.0846 0.000206  yes
                                         30.000%  60 0.0852 0.000206  yes
                                         30.000%  72 0.0851 0.000206  yes
                                         30.000%  84 0.0855 0.000207  yes
                                         30.000%  96 0.0855 0.000207  yes
                                         30.000% 108 0.0855 0.000207  yes
                                         30.000% 120 0.0855 0.000207  yes
                                         30.000% 132 0.0857 0.000207  yes
                                         30.000% 144 0.0857 0.000207  yes
   ABEL (GCC)                            30.000%  24 0.1493 0.000273  yes
                                         30.000%  36 0.1931 0.000311  yes
                                         30.000%  48 0.2324 0.000341  yes
                                         30.000%  60 0.2621 0.000362  yes
                                         30.000%  72 0.2664 0.000365  yes
                                         30.000%  84 0.2668 0.000365  yes
                                         30.000%  96 0.2670 0.000365  yes
                                         30.000% 108 0.2684 0.000366  yes
                                         30.000% 120 0.2683 0.000366  yes
                                         30.000% 132 0.2689 0.000367  yes
                                         30.000% 144 0.2688 0.000367  yes
   RSABE ('implied limits')              30.000%  24 0.1335 0.000258  yes
                                         30.000%  36 0.1536 0.000277  yes
                                         30.000%  48 0.1708 0.000292  yes
                                         30.000%  60 0.1871 0.000306  yes
                                         30.000%  72 0.2006 0.000317  yes
                                         30.000%  84 0.2134 0.000327  yes
                                         30.000%  96 0.2245 0.000335  yes
                                         30.000% 108 0.2336 0.000342  yes
                                         30.000% 120 0.2418 0.000348  yes
                                         30.000% 132 0.2494 0.000353  yes
                                         30.000% 144 0.2551 0.000357  yes
   RSABE ('desired consumer risk model') 25.396%  24 0.0663 0.000182  yes
                                         25.396%  36 0.0629 0.000177  yes
                                         25.396%  48 0.0600 0.000173  yes
                                         25.396%  60 0.0576 0.000170  yes
                                         25.396%  72 0.0557 0.000167  yes
                                         25.396%  84 0.0542 0.000165  yes
                                         25.396%  96 0.0532 0.000163  yes
                                         25.396% 108 0.0523 0.000162  yes
                                         25.396% 120 0.0517 0.000161  yes
                                         25.396% 132 0.0514 0.000160  yes
                                         25.396% 144 0.0508 0.000159  yes
```

R-script to reproduce the example

```
################################################################################
# AUC data from Reference (36) were evaluated in the R-package replicateBE v1.1.3 #
# (ABEL: WHO, GCC) and Phoenix WinNonlin v8.3.4 (ABEL: HC, FDA: RSABE).        #
################################################################################
library(PowerTOST) # at least v1.5-3 (2021-01-18)
data <- data.frame(subject = c(rep(1L:2L, each = 4),   rep(3L, 2),  rep(4L:13L, each = 4),
                               rep(15L:26L, each = 4), rep(27L, 2), rep(28L:40L, each = 4),
                               rep(42L:50L, each = 4), rep(52L:57L, each = 4)),
                   period = c(rep(1L:4L, 2), 1L:2L, rep(1L:4L, 10), rep(1L:4L, 12),
                              1L:2L, rep(1L:4L, 13), rep(1L:4L, 9), rep(1L:4L, 6)),
                   sequence = c(rep("RTRT", 4), rep("TRTR",  4), rep("RTRT",  2), rep("TRTR",  4),
                                rep("RTRT", 8), rep("TRTR", 12), rep("RTRT",  4), rep("TRTR",  4),
                                rep("RTRT", 4), rep("TRTR",  4), rep("RTRT",  4), rep("TRTR",  4),
                                rep("RTRT", 8), rep("TRTR",  8), rep("RTRT",  4), rep("TRTR",  8),
                                rep("TRTR", 8), rep("TRTR",  6), rep("RTRT",  8), rep("TRTR",  4),
                                rep("RTRT", 4), rep("TRTR",  8), rep("RTRT",  8), rep("TRTR", 12),
                                rep("RTRT", 8), rep("TRTR",  8), rep("RTRT",  4), rep("TRTR",  4),
                                rep("RTRT", 4), rep("TRTR",  4), rep("RTRT", 12), rep("TRTR",  4),
                                rep("RTRT", 8), rep("TRTR",  8), rep("RTRT",  4)),
                   treatment = NA_character_,
                   AUC = c(812.6, 1173.7, 889.1, 620.1, 216.3, 338.0,  502.8, 398.6, 545.1, 542.9,
                           632.6,  520.0, 716.7, 860.4, 400.0, 223.8,  173.7, 289.7, 102.1, 185.3,
                            42.0,   88.3, 596.0, 659.3, 543.8, 662.9,  402.4, 359.8, 590.8, 444.3,
                           456.7,  378.4, 477.5, 407.9, 304.5, 351.5,  520.2, 335.7, 500.7, 323.0,
                           416.3,  525.1, 176.1, 710.7, 409.5, 645.5,  160.6, 218.0, 170.1, 124.6,
                           562.4,  490.4, 504.7, 675.9, 756.0, 606.8,  477.4, 626.8, 207.5, 271.6,
                           173.7,  240.5, 571.3, 705.2, 619.0, 633.6,  511.9, 549.7, 388.2, 141.0,
                           124.0,   91.9, 113.3,  59.5, 536.1, 595.2,  445.5, 521.5, 239.7, 265.1,
                           445.9,  433.2, 609.6, 371.6, 511.3, 432.7,  449.9, 860.4, 606.8, 577.2,
                           192.5,  220.1, 233.1, 227.0, 764.4, 508.8,  757.8, 449.4, 151.9, 194.8,
                           568.1,  321.1, 338.3, 403.6, 735.6, 634.5, 1244.2, 641.9, 429.1, 391.8,
                           316.9,  335.1, 307.4, 481.8, 346.6, 369.7,  409.0, 514.6, 763.1, 406.5,
                           271.0,  221.0, 296.5, 463.7, 292.9, 431.0,  448.5, 267.8, 217.2, 332.2,
                           103.0,  127.5, 290.8, 208.6, 243.7, 489.7,  297.2, 502.0, 320.4, 334.3,
```

```
                          163.8,  232.1, 636.9, 434.9, 368.3, 292.6,  446.1, 222.3, 193.7, 202.8,
                          255.2,  244.3, 534.1, 243.1, 418.4, 441.9,  355.1, 415.2, 382.7, 334.0,
                          102.0,  282.5, 245.6, 286.2, 320.5, 233.9,  331.7, 260.5, 223.6, 645.4,
                          349.0,  507.4, 504.5, 289.9, 550.7, 244.2,  615.8, 732.1, 620.9, 665.2,
                          898.4,  924.9, 398.3, 828.3, 410.4, 329.2,  449.4, 442.1, 237.0, 505.0,
                          496.3,  580.6, 332.4, 273.6, 525.3, 293.3,  185.2, 222.9, 182.1, 194.1,
                          246.9,  620.9, 678.3, 752.2, 235.4, 190.4,  318.3, 248.4, 180.6, 174.7,
                          102.9,  117.0))
for (i in 1:nrow(data)) {      # extract treatments from sequences and periods
  treatments        <- unlist(strsplit(data$sequence[i], split = ""))
  data$treatment[i] <- treatments[data$period[i]]
}
nsims      <- 1e6L             # number of simulations for empiric Type I Error
sign.limit <- binom.test(0.05 * nsims, n = nsims, alternative = "less)$conf.int[[2]]
txt        <- paste0("Significance limit based on the binomial test for ",
                     formatC(1e6L, digits = 0, big.mark = ","), " simulations = ",
                     sprintf("%.6f", sign.limit), ".\n")
method     <- c("ABEL  (WHO)", # ANOVA (EMA 'Method A')
                "ABEL  (GCC)", # ANOVA
                "ABEL  (HC) ", # mixed-effects model
                "RSABE (FDA)") # ANDA guidance
regulator  <- c("EMA", "GCC", "HC", "FDA")
result     <- data.frame(method  = method, regulator = regulator,
                         CVwR    = c(rep(0.355648, 3), 0.353968),
                         L       = NA_real_, U = NA_real_,
                         lower   = c(rep(1.017529, 2), 1.016937, NA_real_),
                         upper   = c(rep(1.174621, 2), 1.175704, NA_real_),
                         PE      = c(rep(1.093257, 2), 1.093257, 1.097784),
                         Delta.r = NA_real_,
                         bound   = c(rep(NA_real_, 3), -0.0555212),
                         BE      = "fail", TIE = NA_real_, signif = "",
                         stringsAsFactors = FALSE) # last line required only for R <4.0
for (i in seq_along(method)) {
  if (method[i] == "ABEL  (WHO)") {
    result[i, 4:5] <- scABEL(CV = result$CVwR[i], regulator = regulator[i])
    result$Delta.r[i] <- 1 - result$L[i]
    if (result$lower[i] >= result$L[i] & result$upper[i] >= result$U[i] & # check CI inclusion
        result$PE[i] >= 0.80 & result$PE[i] <= 1.25)                      # check PE constraint
      result$BE <- "pass"
    result$TIE[i]  <- power.scABEL(CV = result$CVwR[i], n = 54, design = "2x2x4",
                                   theta0 = result$U[i], regulator = regulator[i], nsims = nsims)
    if (result$TIE[i] > limit) result$signif[i] <- "*  "                  # check significant TIE
  }
  if (method[i] == "ABEL  (GCC)") {
    result[i, 4:5] <- scABEL(CV = result$CVwR[i], regulator = regulator[i])
    result$Delta.r[i] <- 1 - result$L[i]
    if (result$lower[i] >= result$L[i] & result$upper[i] >= result$U[i] & # check CI inclusion
        result$PE[i] >= 0.80 & result$PE[i] <= 1.25)                      # check PE constraint
      result$BE <- "pass"
    result$TIE[i]  <- power.scABEL(CV = result$CVwR[i], n = 54, design = "2x2x4",
                                   theta0 = result$U[i], regulator = regulator[i], nsims = nsims)
    if (result$TIE[i] > limit) result$signif[i] <- "*  "                  # check significant TIE
  }
  if (method[i] == "ABEL  (HC) ") {
    result[i, 4:5] <- scABEL(CV = result$CVwR[i], regulator = result$regulator[i])
    result$Delta.r[i] <- 1 - result$L[i]
    if (result$lower[i] >= result$L[i] & result$upper[i] >= result$U[i] & # check CI inclusion
        result$PE[i] >= 0.80 & result$PE[i] <= 1.25)                      # check PE constraint
      result$BE <- "pass"
    result$TIE[i]   <- power.scABEL(CV = result$CVwR[i], n = 54, design = "2x2x4",
                                    theta0 = result$U[i], regulator = regulator[i], nsims = nsims)
    if (result$TIE[i] > limit) result$signif[i] <- "*  "                  # check significant TIE
  }
  if (method[i] == "RSABE (FDA)") {
    result[i, 4:5] <- scABEL(CV = result$CVwR[i], regulator = regulator[i])
    result$Delta.r[i] <- 1 - result$L[i]
    if (result$bound[i] <= 0 &                                            # check bound
        result$PE[i] >= 0.80 & result$PE[i] <= 1.25)                      # check PE constraint
      result$BE <- "pass"
    result$TIE[i]  <- power.RSABE(CV = result$CVwR[i], n = 54, design = "2x2x4",
                                  theta0 = result$U[i], regulator = regulator[i], nsims = nsims)
    if (result$TIE[i] > limit) result$signif[i] <- "*  "                  # check significant TIE
  }
}
result         <- result[, -which(names(result) == "regulator")]
result$CVwR    <- sprintf("%.2f%%", 100 * result$CVwR)
result$L       <- sprintf("%.2f%%", 100 * result$L)
result$U       <- sprintf("%.2f%%", 100 * result$U)
result$lower   <- sprintf("%.2f%%", 100 * result$lower)
result$upper   <- sprintf("%.2f%%", 100 * result$upper)
result$PE      <- sprintf("%.2f%%", 100 * result$PE)
result$Delta.r <- sprintf("%.2f%%", 100 * result$Delta.r)
result$bound   <- signif(result$bound, 4)
result$TIE     <- signif(result$TIE, 3)
result[1:3, 9] <- "\u2013     "
result[4, 5:6] <- "\u2013   "
```

```
121 cat(txt); print(result, row.names = FALSE)
```

Should give:

```
Significance limit based on the binomial test for 1,000,000 simulations = 0.050360.
      method   CVwR      L       U   lower   upper      PE Delta.r    bound   BE    TIE signif
 ABEL  (WHO) 35.56% 76.93% 129.99% 101.75% 117.46% 109.33%  23.07%   -      pass 0.0643    *
 ABEL  (GCC) 35.56% 75.00% 133.33% 101.75% 117.46% 109.33%  25.00%   -      pass 0.0459
 ABEL  (HC)  35.56% 76.93% 129.99% 101.69% 117.57% 109.33%  23.07%   -      pass 0.0651    *
 RSABE (FDA) 35.40% 73.59% 135.89%    -       -    109.78%  26.41% -0.05552 pass 0.0232
```