Supplemental file for
“Factors that impact fragility index and their visualizations”

Appendix A. Supplemental figures for artificial datasets

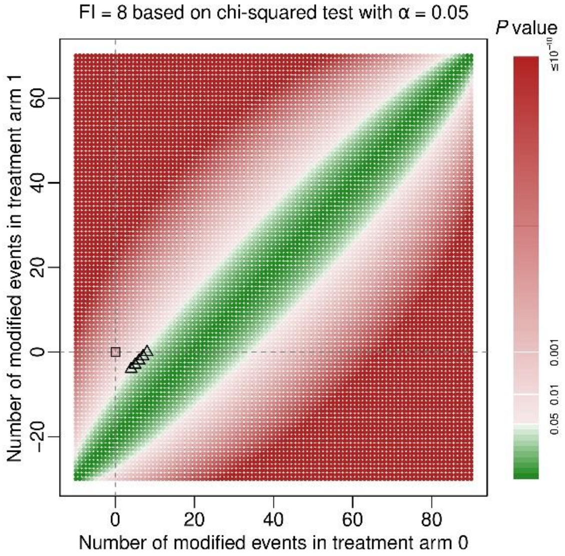


Figure S1. Fragility index of artificial dataset 1 derived using chi-squared test. Event status is modified in both treatment arms, and the significance level is 0.05. Each point represents a P-value based on specific event status modifications. P-values are presented on a base-10 logarithmic scale. Points in green indicate non-significant results, and those in red indicate significant ones. Dashed lines represent no modifications in the corresponding arms. Square points represent the original P-value, and triangle points indicate minimal modifications that alter the significance.

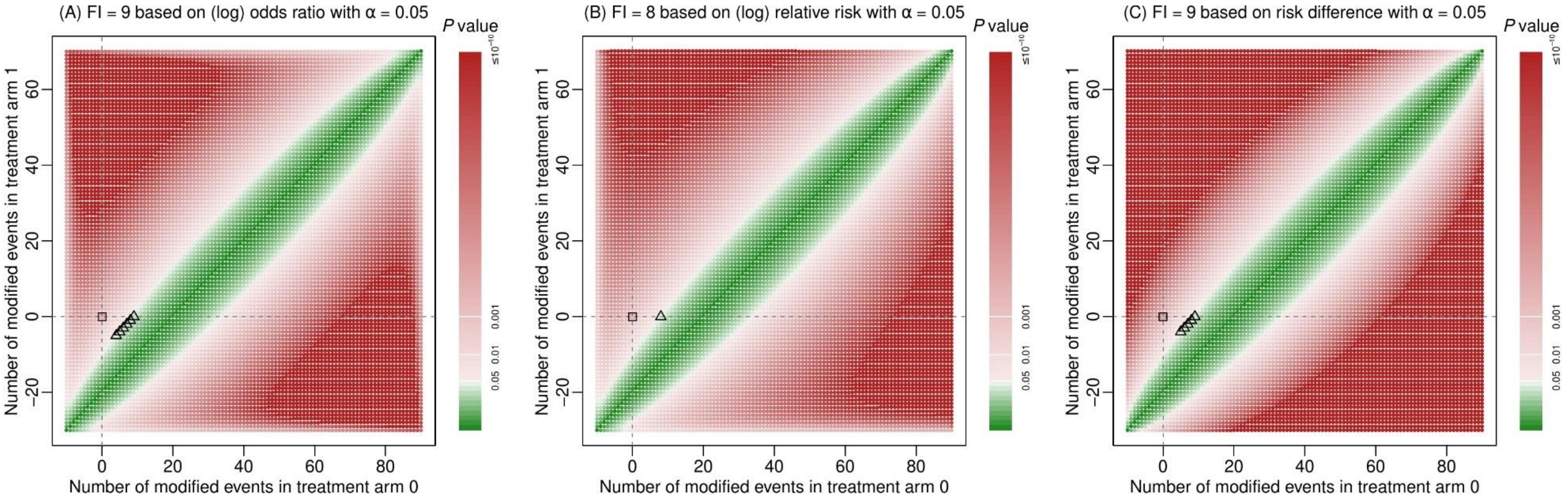


Figure S2. Fragility index of artificial dataset 1 derived using effect measures of odds ratio (A), relative risk (B), or risk difference (C). Event status is modified in both treatment arms, and the significance level is 0.05. Each point represents a P-value based on specific event status modifications. P-values are presented on a base-10 logarithmic scale. Points in green indicate non-significant results, and those in red indicate significant ones. Dashed lines represent no modifications in the corresponding arms. Square points represent the original P-value, and triangle points indicate minimal modifications that alter the significance.

Appendix B. Results of the real data analyses for the outcome of dropouts

In addition to the results for the outcome of responders (describing treatment efficacy) presented in the main content, this appendix presents the results of the antidepressant data for the outcome of dropouts (describing acceptability). A total of 347 trials with the outcome of dropouts were obtained. Among them, the median of arm-specific sample sizes was 71 with interquartile range (IQR) 35–127, and that of the arm-specific events (dropouts) was 19 with IQR 9–32. Moreover, 279 (80%) trials had sample size ratios <1.1, while the remaining 68 (20%) trials had sample size ratios ≥1.1. Figure S7 in Appendix C presents the distribution of the trials' sample size ratios.

Figure S8 presents the distributions of the 347 trials' FIs based on Fisher's exact test and chi-squared test. The FIs of NA (i.e., the cases that significance or non-significance cannot be altered) are not included in the figures and the following results. Using Fisher's exact test, 32 trials' results were significant, while 315 were non-significant. The FIs ranged from 1 to 19, with a median 6 (IQR, 3–8). Among the 32 significant results, the median FI was 3 (IQR, 1–6); among the 315 non-significant results, the median FI was 6 (IQR, 4–8). Based on the chi-squared test, the numbers of trials with significant and non-significant results were 29 and 318, respectively. The bar plot (Figure S8B) had a roughly similar trend to that produced by Fisher's exact test (Figure S8A), with an overall median FI (among all 347 trials) around 6. Nevertheless, noticeable differences existed between the FIs' distributions produced by the two different tests. The chi-squared test led to a FI of NA in one trial, whose non-significant result could never be altered to be significant. Fisher's exact test did not lead to FIs of NA.

Figure S9 presents the distributions of FIs based on effect measures of OR, RR, and RD. The numbers of trials with significant and non-significant results were 38 and 309 using the OR, 36 and 311 using the RR, and 42 and 305 using the RD, respectively. Their bar plots also had roughly similar trends to those in Figure S8. Using the OR and RR, one and two trials had FIs of NA, respectively; their non-significance could not be altered to significance.

Moreover, when restricting event status modifications to either arm 0 or arm 1, the produced FIs of some trials were strictly larger than the FIs with modifications in both arms. Based on Fisher's exact test, chi-squared test, OR, RR, and RD, the numbers of such trials were 19, 23, 10, 12, and 11, respectively. All these cases occurred for originally non-significant results. For example, Figure S10 shows the FI's visualization of the trial by Norton et al. (https://doi.org/10.1016/0165-0327(84)90051-X). Using Fisher's exact test, the FI was 5 with modifications in both arms, while it became 9 or 6 if modifications were restricted to arm 0 or arm 1, respectively.

Figure S11 presents the histograms of trial-specific average FIs based on Fisher's exact test and chi-squared test when the significance level varied from 0.005 to 0.05. Figure S12 presents those based on effect measures of OR, RR, and RD. These histograms had some noticeable differences from the bar plots of FIs in Figures S8 and S9 at the significance level 0.05; the average FIs were generally larger than the FIs.

Appendix C. Supplemental figures for real data of antidepressant drugs

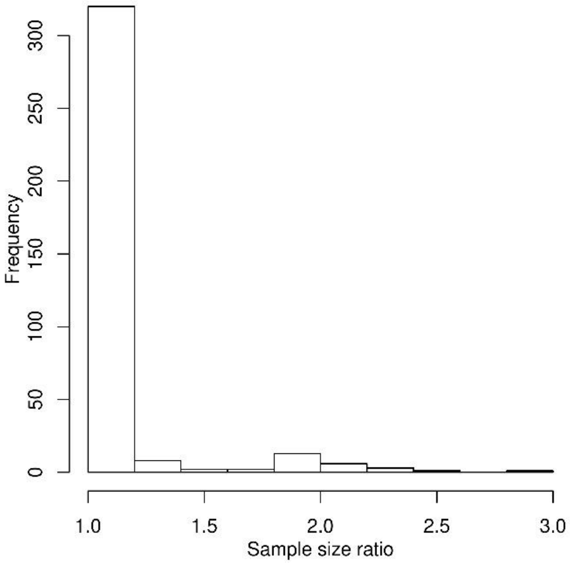


Figure S3. Histogram of sample size ratios among trials on antidepressant drugs with the outcome of responders.

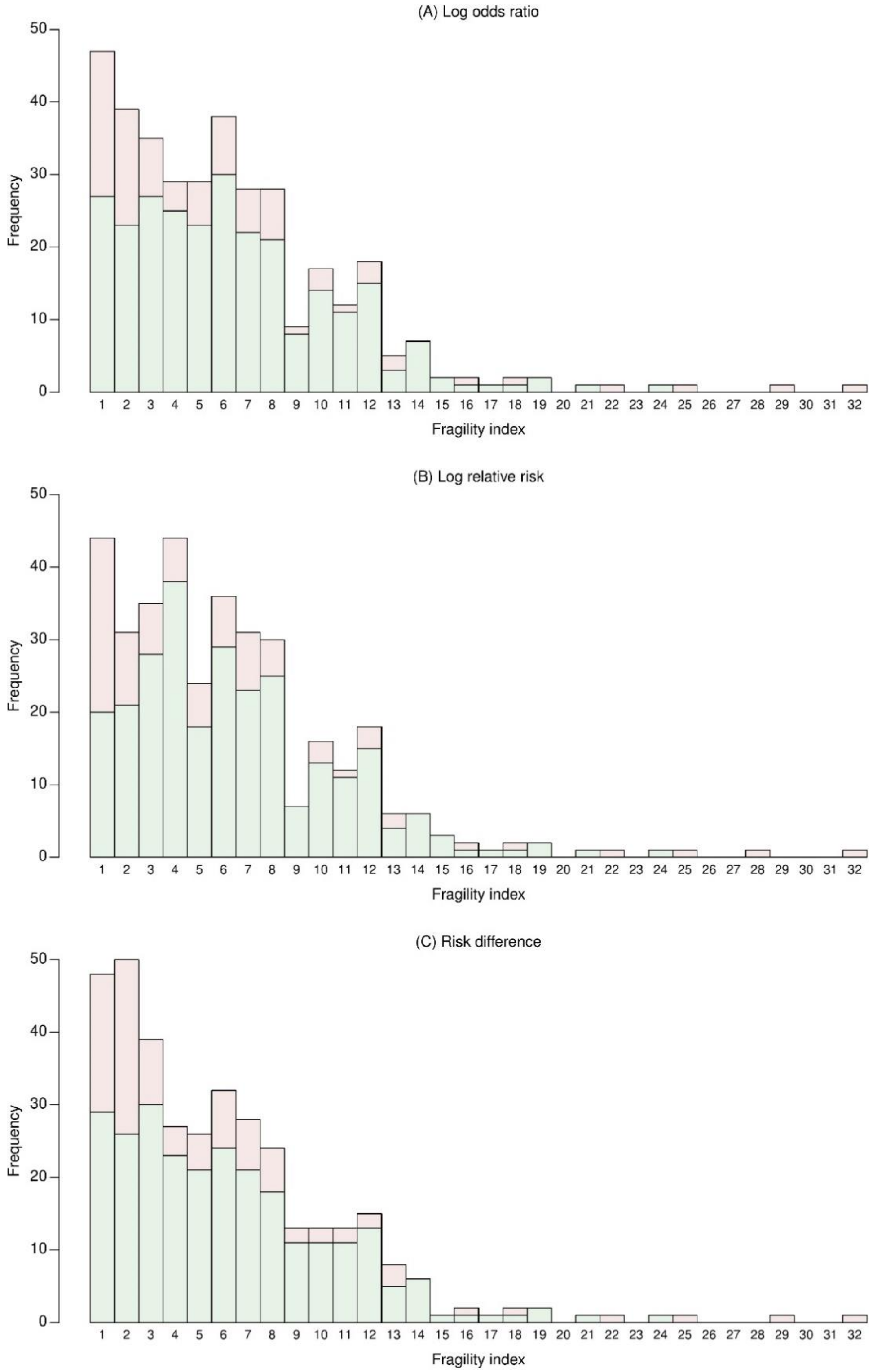
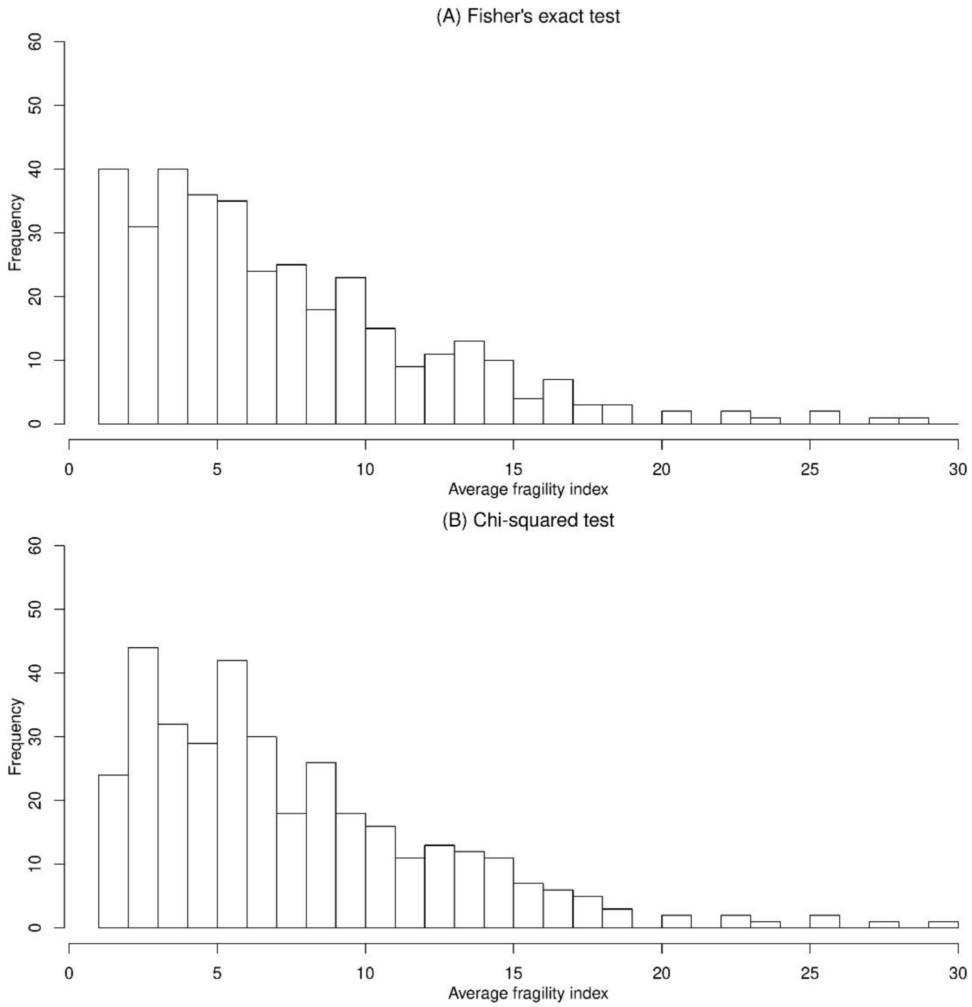


Figure S4. Bar plot of fragility indexes in the randomized controlled trials on antidepressant drugs with the outcome of responders. The fragility indexes are derived using odds ratio (A), relative risk (B), and risk difference (C) at the significance level 0.05. Bars in green represent fragility indexes that alter non-significance to significance, and those in red represent fragility indexes that alter significance to non-significance.



(A) Fisher's exact test

(B) Chi-squared test

Figure S5. Histogram of average fragility indexes of the randomized controlled trials on antidepressant drugs with the outcome of responders based on Fisher’s exact test (A) and chi-squared test (B).

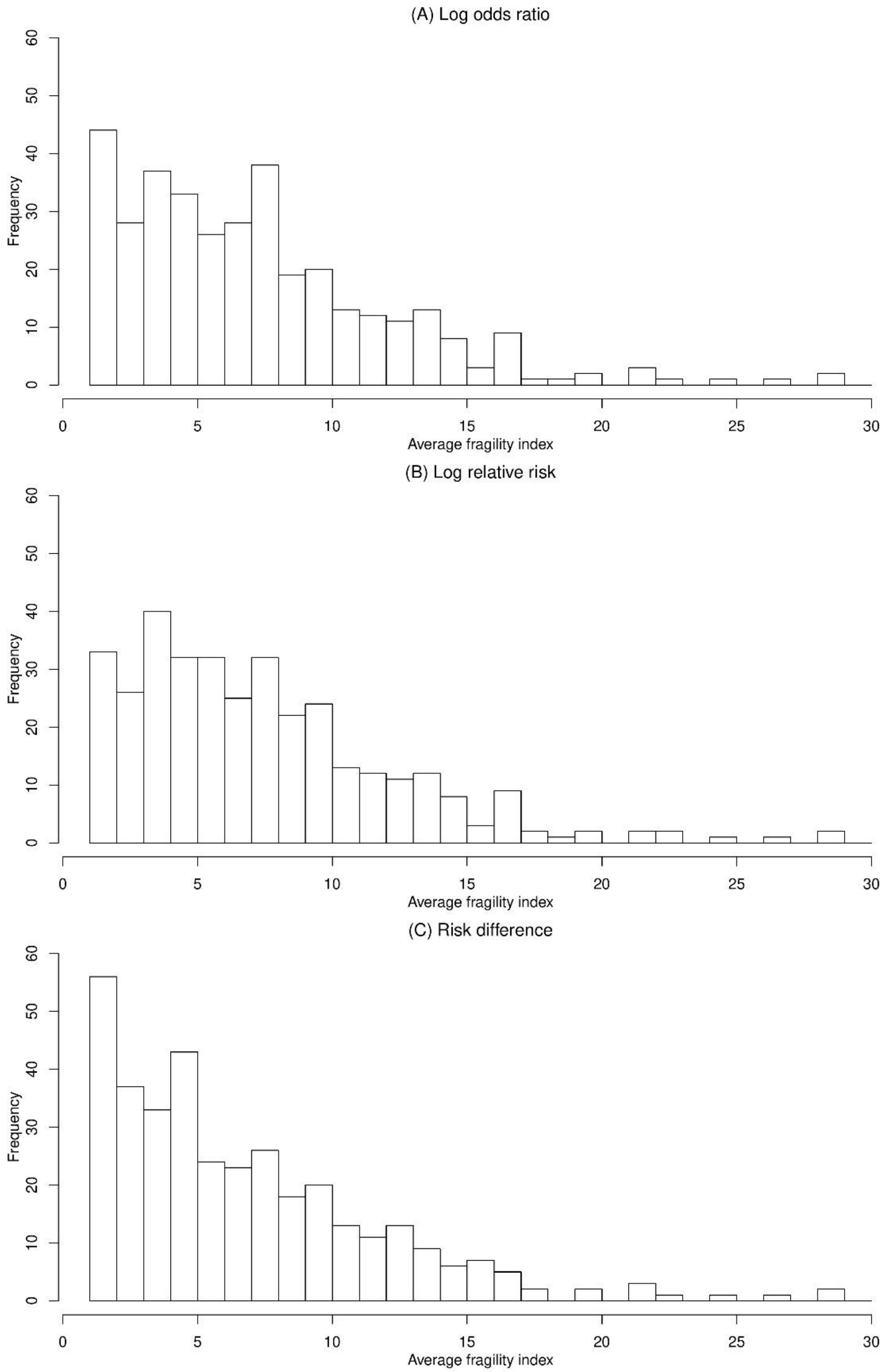


Figure S6. Histogram of average fragility indexes of the randomized controlled trials on antidepressant drugs with the outcome of responders based on odds ratio (A), relative risk (B), and risk difference (C).

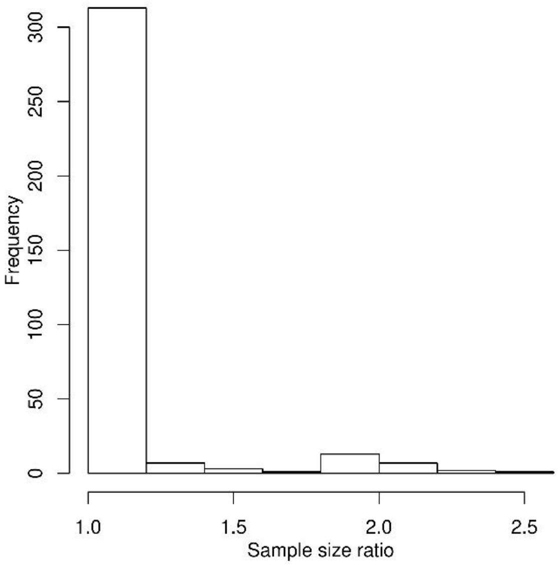


Figure S7. Histogram of sample size ratios among trials on antidepressant drugs with the outcome of dropouts.

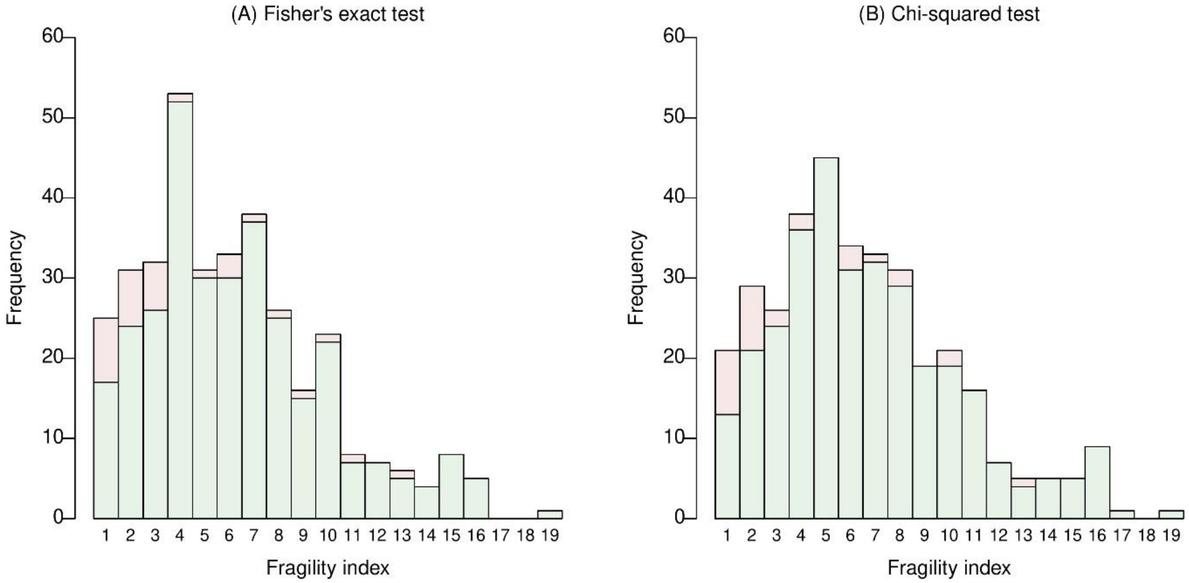


Figure S8. Bar plot of fragility indexes in the randomized controlled trials on antidepressant drugs with the outcome of dropouts. The fragility indexes are derived using Fisher’s exact test (A) and chi-squared test (B) at the significance level 0.05. Bars in green represent fragility indexes that alter non-significance to significance, and those in red represent fragility indexes that alter significance to non-significance.

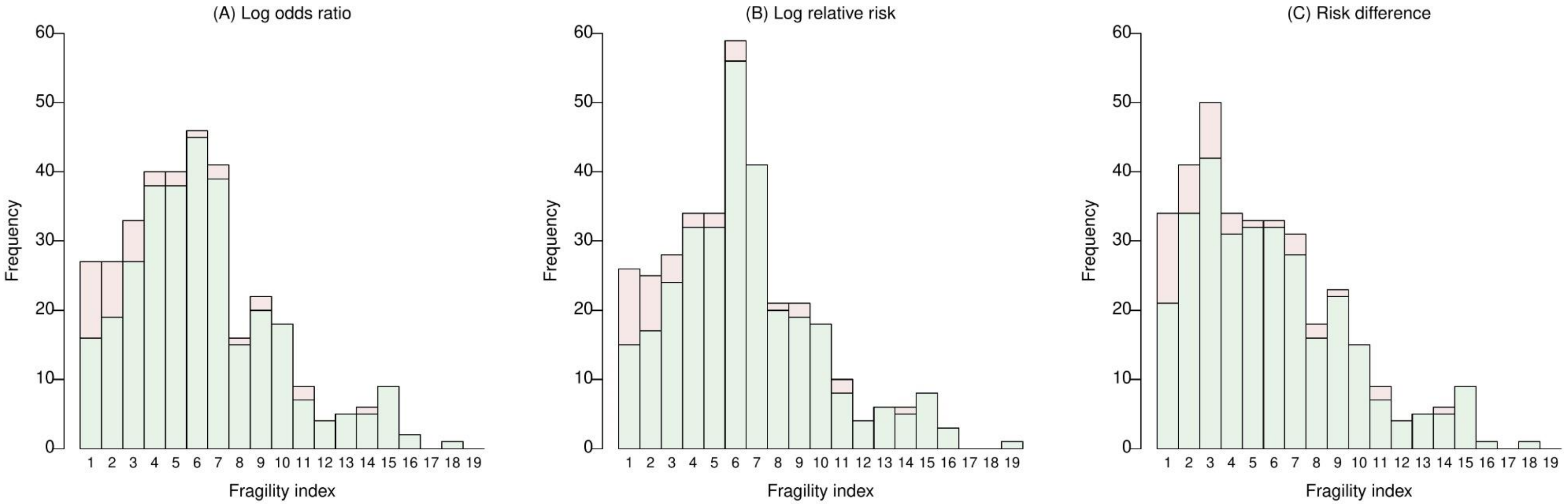


Figure S9. Bar plot of fragility indexes in the randomized controlled trials on antidepressant drugs with the outcome of dropouts. The fragility indexes are derived using odds ratio (A), relative risk (B), and risk difference (C) at the significance level 0.05. Bars in green represent fragility indexes that alter non-significance to significance, and those in red represent fragility indexes that alter significance to non-significance.

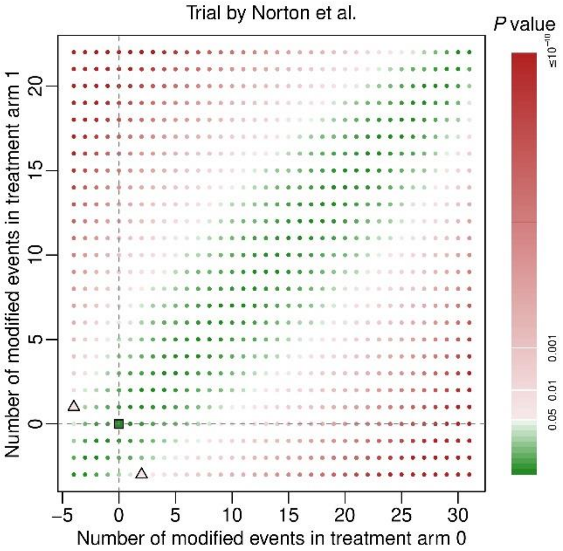


Figure S10. An example of the randomized controlled trial by Norton et al. on antidepressant drugs with the outcome of dropouts.

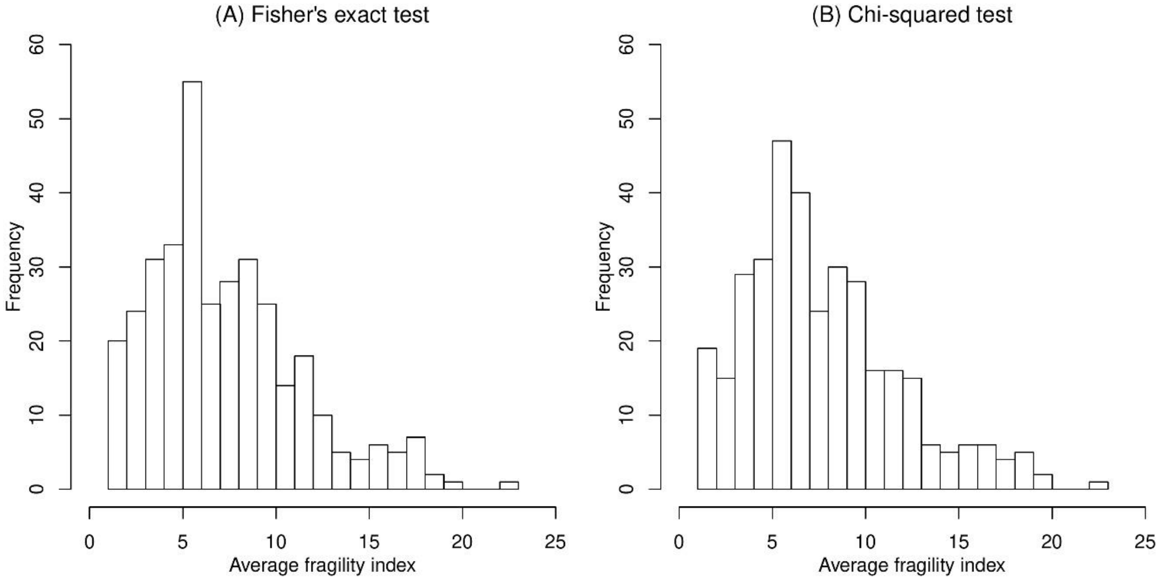


Figure S11. Histogram of average fragility indexes of the randomized controlled trials on antidepressant drugs with the outcome of dropouts based on Fisher's exact test (A) and chi-squared test (B).

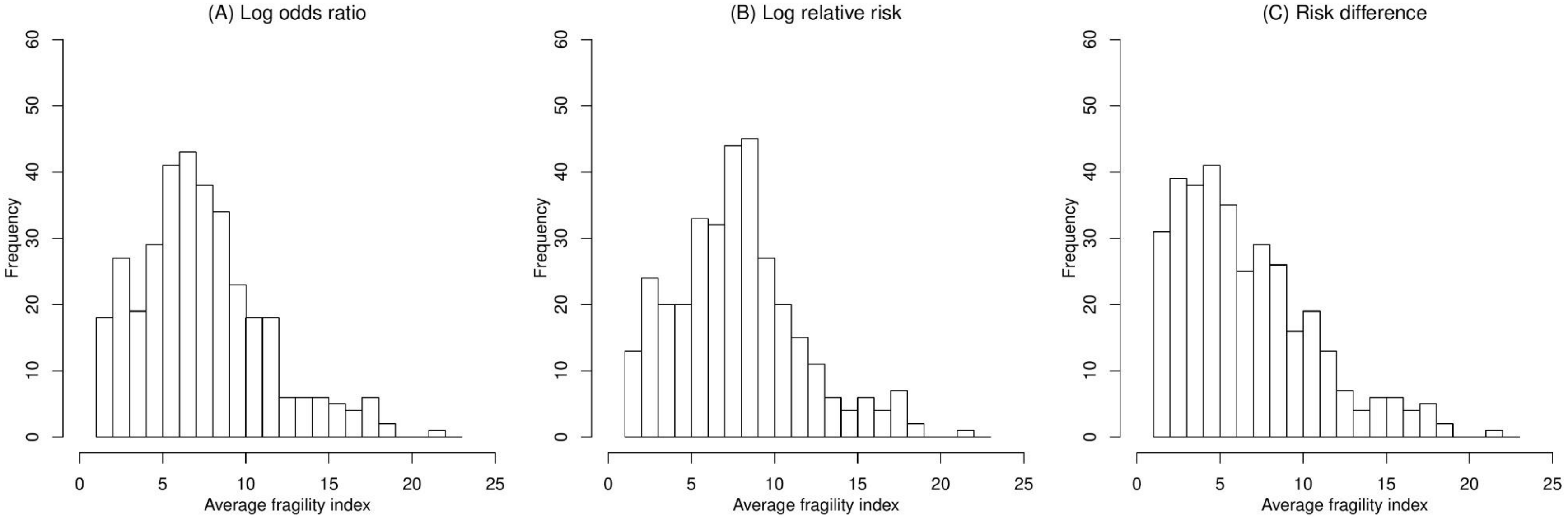


Figure S12. Histogram of average fragility indexes of the randomized controlled trials on antidepressant drugs with the outcome of dropouts based on odds ratio (A), relative risk (B), and risk difference (C).

Appendix D. R code for all data analyses

This appendix contains R code to produce the results of all artificial and real data analyses in the main content. The real datasets are also included within the code. The R code consists of three major parts: 1) R functions for deriving the fragility index and visualizations; 2) analyses of artificial datasets; and 3) analyses of real datasets (trials on antidepressant drugs with the outcomes of responders and dropouts). The proposed methods can be also implemented via our R package "fragility" (https://cran.r-project.org/package=fragility).

In the following code, there are five main R functions, i.e., `fragility.all()`, `pvals.plot()`, `fragility.iter()`, `frag.study.alpha()`, and `FIvsSignif()`. The function `fragility.all()` is used to enumerate all considered event status modifications for deriving the fragility index, and it can output complete results about fragility index. The function `pvals.plot()` produces the plots of P-values based on various event status modifications; it was used to produce Figures 1, 2, and 5A in the main content and Figures S1, S2, and S10 in Appendix C. The function `fragility.iter()` provides an efficient way to obtain the fragility index. Instead of enumerating all possible event status modifications, it derives the fragility index via an iterative algorithm, which is generally expected to be faster than obtaining the fragility index using `fragility.all()`. Nevertheless, because `fragility.iter()` does not consider all possible modifications, it cannot generate visualization plots of P-values corresponding to all possible event status modifications. The function `frag.study.alpha()` efficiently calculates fragility indexes at multiple significance levels and produces the average fragility index. The function `FIvsSignif()` plots fragility index against different significance levels, and it was used to produce Figures 3, 5B, and 5C in the main content.

The usage of these functions is provided in the R comments (after the number signs "#" in the following code) and is demonstrated using the artificial and real data analyses. Our R package "fragility" also offers a detailed manual with many reproducible examples, which is available at https://cran.r-project.org/package=fragility/fragility.pdf.

```
## Set your working directory
wd <- "..."

################################
#### Functions
################################

## arguments: e0, n0, and f0 represent event count, sample size,
##   and the number of event status modifications in arm 0;
##   e1, n1, and f1 represent those in arm 1
pval.logOR <- function(f0 = 0, f1 = 0, e0, n0, e1, n1){
  e0.m <- e0 + f0
  ne0.m <- n0 - e0.m
  e1.m <- e1 + f1
  ne1.m <- n1 - e1.m
  if((abs(e0.m) < 1e-6 & abs(e1.m) < 1e-6) |
    (abs(ne0.m) < 1e-6 & abs(ne1.m) < 1e-6)){
    pval <- 1
  }else{
    if(abs(e0.m) < 1e-6 | abs(ne0.m) < 1e-6 |
      abs(e1.m) < 1e-6 | abs(ne1.m) < 1e-6){
      e0.m <- e0.m + 0.5
      ne0.m <- ne0.m + 0.5
      e1.m <- e1.m + 0.5
      ne1.m <- ne1.m + 0.5
    }
    logOR <- log(e1.m/ne1.m) - log(e0.m/ne0.m)
    se.logOR <- sqrt(1/e0.m + 1/ne0.m + 1/e1.m + 1/ne1.m)
    pval <- 2*pnorm(-abs(logOR)/se.logOR)
  }
  return(pval)
}

pval.logRR <- function(f0 = 0, f1 = 0, e0, n0, e1, n1){
  e0.m <- e0 + f0
  ne0.m <- n0 - e0.m
  e1.m <- e1 + f1
  ne1.m <- n1 - e1.m
  if((abs(e0.m) < 1e-6 & abs(e1.m) < 1e-6) |
    (abs(ne0.m) < 1e-6 & abs(ne1.m) < 1e-6)){
    pval <- 1
  }else{
    if(abs(e0.m) < 1e-6 | abs(ne0.m) < 1e-6 |
      abs(e1.m) < 1e-6 | abs(ne1.m) < 1e-6){
      e0.m <- e0.m + 0.5
      ne0.m <- ne0.m + 0.5
      e1.m <- e1.m + 0.5
      ne1.m <- ne1.m + 0.5
    }
    logRR <- log(e1.m/(e1.m + ne1.m)) - log(e0.m/(e0.m + ne0.m))
    se.logRR <- sqrt(1/e0.m + 1/e1.m -
      1/(e0.m + ne0.m) - 1/(e1.m + ne1.m))
    pval <- 2*pnorm(-abs(logRR)/se.logRR)
  }
  return(pval)
}

pval.RD <- function(f0 = 0, f1 = 0, e0, n0, e1, n1){
  e0.m <- e0 + f0
  ne0.m <- n0 - e0.m
  e1.m <- e1 + f1
  ne1.m <- n1 - e1.m
  if((abs(e0.m) < 1e-6 & abs(e1.m) < 1e-6) |
    (abs(ne0.m) < 1e-6 & abs(ne1.m) < 1e-6)){
    pval <- 1
  }else{
    RD <- e1.m/(e1.m + ne1.m) - e0.m/(e0.m + ne0.m)
    se.RD <- sqrt(e0.m/(e0.m + ne0.m)^3*ne0.m +
      e1.m/(e1.m + ne1.m)^3*ne1.m)
    pval <- 2*pnorm(-abs(RD)/se.RD)
  }
  return(pval)
}

## fragility.all() considers all cases of event status modifications.
## arguments: methods specifies the statistical method(s) used to
##   assess the association between treatment and outcomes, which
##   can be "Fisher" (Fisher's exact test), "chisq" (Chi-squared test),
##   "logOR" (log odds ratio), "logRR" (log relative risk),
##   or "RD" (risk difference);
##   modify0 specifies the direction of event status
##   modifications in arm 0, which can be "increase" (increasing
##   event counts), "decrease" (decreasing event counts),
##   "both" (both increasing and decreasing events),
##   or "none" (no modification);
##   modify1 specifies such a direction in arm 1;
##   alpha specifies the statistical significance level (default 0.05);
##   save is a logical value, indicating whether the results from
##   all cases of event status modifications are saved (TRUE)
##   or not (FALSE).
fragility.all <- function(e0, n0, e1, n1, methods = "Fisher",
  modify0 = "both", modify1 = "both",
  alpha = 0.05, save = TRUE){
  ne0 <- n0 - e0
  ne1 <- n1 - e1
  if(modify0 == "increase") f0.range <- c(0, ne0)
  if(modify0 == "decrease") f0.range <- c(-e0, 0)
  if(modify0 == "both") f0.range <- c(-e0, ne0)
  if(modify0 == "none") f0.range <- c(0, 0)
  if(modify1 == "increase") f1.range <- c(0, ne1)
  if(modify1 == "decrease") f1.range <- c(-e1, 0)
  if(modify1 == "both") f1.range <- c(-e1, ne1)
  if(modify1 == "none") f1.range <- c(0, 0)
  f0.mods <- f0.range[1]:f0.range[2]
  f1.mods <- f1.range[1]:f1.range[2]
  tot.mods <- outer(abs(f0.mods), abs(f1.mods), "+")

  out <- list(methods = methods, alpha = alpha,
    modify0 = modify0, modify1 = modify1,
    f0.range = f0.range, f1.range = f1.range)
  if(save) out <- c(out, list(tot.mods = tot.mods))

  ## Fisher's exact test
  if(is.element("Fisher", methods)){
    temp <- function(f0, f1){
      fisher.test(rbind(c(e0 + f0, n0 - e0 - f0),
        c(e1 + f1, n1 - e1 - f1)),
        alternative = "two.sided")$p.value
    }
    temp <- Vectorize(temp)
    pvals.Fisher <- outer(f0.mods, f1.mods, temp)
    pval.Fisher.ori <- pvals.Fisher[f0.mods == 0, f1.mods == 0]
    FI.Fisher.sig2nonsig <- FI.Fisher.nonsig2sig <- NA
    mods.Fisher <- NA
    FI0.Fisher.sig2nonsig <- FI0.Fisher.nonsig2sig <-
      FI1.Fisher.sig2nonsig <- FI1.Fisher.nonsig2sig <- NA
```

```
mods0.Fisher <- mods1.Fisher <- NA
if(pval.Fisher.ori < alpha){
  if(any(pvals.Fisher >= alpha)){
    FI.Fisher.sig2nonsig <- min(tot.mods[pvals.Fisher >= alpha])
    mods.Fisher <- which(tot.mods == FI.Fisher.sig2nonsig &
       pvals.Fisher >= alpha, arr.ind = TRUE)
    mods.Fisher <- cbind(f0.mods[mods.Fisher[,1]],
      f1.mods[mods.Fisher[,2]])
    colnames(mods.Fisher) <- c("arm0", "arm1")
    if(modify0 != "none" & modify1 != "none"){
      pvals0.Fisher <- pvals.Fisher[, f1.mods == 0]
      tot.mods0 <- tot.mods[, f1.mods == 0]
      pvals1.Fisher <- pvals.Fisher[f0.mods == 0,]
      tot.mods1 <- tot.mods[f0.mods == 0,]
      if(any(pvals0.Fisher >= alpha)){
        FI0.Fisher.sig2nonsig <-
          min(tot.mods0[pvals0.Fisher >= alpha])
        mods0.Fisher <-
          f0.mods[tot.mods0 == FI0.Fisher.sig2nonsig &
          pvals0.Fisher >= alpha]
      }else{
        FI0.Fisher.sig2nonsig <- NA
      }
      if(any(pvals1.Fisher >= alpha)){
        FI1.Fisher.sig2nonsig <-
          min(tot.mods1[pvals1.Fisher >= alpha])
        mods1.Fisher <-
          f1.mods[tot.mods1 == FI1.Fisher.sig2nonsig &
          pvals1.Fisher >= alpha]
      }else{
        FI1.Fisher.sig2nonsig <- NA
      }
    }
  }else{
    FI.Fisher.sig2nonsig <- NA
  }
}else{
  if(any(pvals.Fisher < alpha)){
    FI.Fisher.nonsig2sig <- min(tot.mods[pvals.Fisher < alpha])
    mods.Fisher <- which(tot.mods == FI.Fisher.nonsig2sig &
       pvals.Fisher < alpha, arr.ind = TRUE)
    mods.Fisher <- cbind(f0.mods[mods.Fisher[,1]],
      f1.mods[mods.Fisher[,2]])
    colnames(mods.Fisher) <- c("arm0", "arm1")
    if(modify0 != "none" & modify1 != "none"){
      pvals0.Fisher <- pvals.Fisher[, f1.mods == 0]
      tot.mods0 <- tot.mods[, f1.mods == 0]
      pvals1.Fisher <- pvals.Fisher[f0.mods == 0,]
      tot.mods1 <- tot.mods[f0.mods == 0,]
      if(any(pvals0.Fisher < alpha)){
        FI0.Fisher.nonsig2sig <-
          min(tot.mods0[pvals0.Fisher < alpha])
        mods0.Fisher <-
          f0.mods[tot.mods0 == FI0.Fisher.nonsig2sig &
          pvals0.Fisher < alpha]
      }else{
        FI0.Fisher.nonsig2sig <- NA
      }
      if(any(pvals1.Fisher < alpha)){
        FI1.Fisher.nonsig2sig <-
          min(tot.mods1[pvals1.Fisher < alpha])
        mods1.Fisher <-
          f1.mods[tot.mods1 == FI1.Fisher.nonsig2sig &
```

```
            pvals1.Fisher < alpha]
        }else{
          FI1.Fisher.nonsig2sig <- NA
        }
      }
    }else{
      FI.Fisher.nonsig2sig <- NA
    }
  }
  out.Fisher <- list(pval.Fisher = pval.Fisher.ori,
    FI.Fisher.sig2nonsig = FI.Fisher.sig2nonsig,
    FI.Fisher.nonsig2sig = FI.Fisher.nonsig2sig,
    mods.Fisher = mods.Fisher)
  if(modify0 != "none" & modify1 != "none"){
    out.Fisher <- c(out.Fisher,
      list(FI0.Fisher.sig2nonsig = FI0.Fisher.sig2nonsig,
      FI0.Fisher.nonsig2sig = FI0.Fisher.nonsig2sig,
      mods0.Fisher = mods0.Fisher,
      FI1.Fisher.sig2nonsig = FI1.Fisher.sig2nonsig,
      FI1.Fisher.nonsig2sig = FI1.Fisher.nonsig2sig,
      mods1.Fisher = mods1.Fisher))
  }
  if(save){
    out.Fisher <- c(out.Fisher, list(pvals.Fisher = pvals.Fisher))
  }
  out <- c(out, out.Fisher)
}

## Chi-squared test
if(is.element("chisq", methods)){
  temp <- function(f0, f1){
    if((abs(e0 + f0) < 1e-6 & abs(e1 + f1) < 1e-6) |
      (abs(n0 - e0 - f0) < 1e-6 & abs(n1 - e1 - f1) < 1e-6)){
      temp.out <- 1
    }else{
      temp.out <- chisq.test(rbind(c(e0 + f0, n0 - e0 - f0),
        c(e1 + f1, n1 - e1 - f1)))$p.value
    }
    return(temp.out)
  }
  temp <- Vectorize(temp)
  options(warn = -1)
  pvals.chisq <- outer(f0.mods, f1.mods, temp)
  options(warn = 0)
  pval.chisq.ori <- pvals.chisq[f0.mods == 0, f1.mods == 0]
  FI.chisq.sig2nonsig <- FI.chisq.nonsig2sig <- NA
  mods.chisq <- NA
  FI0.chisq.sig2nonsig <- FI0.chisq.nonsig2sig <-
    FI1.chisq.sig2nonsig <- FI1.chisq.nonsig2sig <- NA
  mods0.chisq <- mods1.chisq <- NA
  if(pval.chisq.ori < alpha){
    if(any(pvals.chisq >= alpha)){
      FI.chisq.sig2nonsig <- min(tot.mods[pvals.chisq >= alpha])
      mods.chisq <- which(tot.mods == FI.chisq.sig2nonsig &
         pvals.chisq >= alpha, arr.ind = TRUE)
      mods.chisq <- cbind(f0.mods[mods.chisq[,1]],
        f1.mods[mods.chisq[,2]])
      colnames(mods.chisq) <- c("arm0", "arm1")
      if(modify0 != "none" & modify1 != "none"){
        pvals0.chisq <- pvals.chisq[, f1.mods == 0]
        tot.mods0 <- tot.mods[, f1.mods == 0]
        pvals1.chisq <- pvals.chisq[f0.mods == 0,]
        tot.mods1 <- tot.mods[f0.mods == 0,]
      if(any(pvals0.chisq >= alpha)){
        FI0.chisq.sig2nonsig <-
          min(tot.mods0[pvals0.chisq >= alpha])
        mods0.chisq <-
          f0.mods[tot.mods0 == FI0.chisq.sig2nonsig &
          pvals0.chisq >= alpha]
      }else{
        FI0.chisq.sig2nonsig <- NA
      }
      if(any(pvals1.chisq >= alpha)){
        FI1.chisq.sig2nonsig <-
          min(tot.mods1[pvals1.chisq >= alpha])
        mods1.chisq <-
          f1.mods[tot.mods1 == FI1.chisq.sig2nonsig &
          pvals1.chisq >= alpha]
      }else{
        FI1.chisq.sig2nonsig <- NA
      }
    }
  }else{
    FI.chisq.sig2nonsig <- NA
  }
}else{
  if(any(pvals.chisq < alpha)){
    FI.chisq.nonsig2sig <- min(tot.mods[pvals.chisq < alpha])
    mods.chisq <- which(tot.mods == FI.chisq.nonsig2sig &
       pvals.chisq < alpha, arr.ind = TRUE)
    mods.chisq <- cbind(f0.mods[mods.chisq[,1]],
      f1.mods[mods.chisq[,2]])
    colnames(mods.chisq) <- c("arm0", "arm1")
    if(modify0 != "none" & modify1 != "none"){
      pvals0.chisq <- pvals.chisq[, f1.mods == 0]
      tot.mods0 <- tot.mods[, f1.mods == 0]
      pvals1.chisq <- pvals.chisq[f0.mods == 0,]
      tot.mods1 <- tot.mods[f0.mods == 0,]
      if(any(pvals0.chisq < alpha)){
        FI0.chisq.nonsig2sig <-
          min(tot.mods0[pvals0.chisq < alpha])
        mods0.chisq <-
          f0.mods[tot.mods0 == FI0.chisq.nonsig2sig &
          pvals0.chisq < alpha]
      }else{
        FI0.chisq.nonsig2sig <- NA
      }
      if(any(pvals1.chisq < alpha)){
        FI1.chisq.nonsig2sig <-
          min(tot.mods1[pvals1.chisq < alpha])
        mods1.chisq <-
          f1.mods[tot.mods1 == FI1.chisq.nonsig2sig &
          pvals1.chisq < alpha]
      }else{
        FI1.chisq.nonsig2sig <- NA
      }
    }
  }else{
    FI.chisq.nonsig2sig <- NA
  }
}
out.chisq <- list(pval.chisq = pval.chisq.ori,
  FI.chisq.sig2nonsig = FI.chisq.sig2nonsig,
  FI.chisq.nonsig2sig = FI.chisq.nonsig2sig,
  mods.chisq = mods.chisq)
if(modify0 != "none" & modify1 != "none"){
    out.chisq <- c(out.chisq,
      list(FI0.chisq.sig2nonsig = FI0.chisq.sig2nonsig,
      FI0.chisq.nonsig2sig = FI0.chisq.nonsig2sig,
      mods0.chisq = mods0.chisq,
      FI1.chisq.sig2nonsig = FI1.chisq.sig2nonsig,
      FI1.chisq.nonsig2sig = FI1.chisq.nonsig2sig,
      mods1.chisq = mods1.chisq))
  }
  if(save){
    out.chisq <- c(out.chisq, list(pvals.chisq = pvals.chisq))
  }
  out <- c(out, out.chisq)
}

## log odds ratio
if(is.element("logOR", methods)){
  temp <- function(f0, f1){
    pval.logOR(f0 = f0, f1 = f1, e0 = e0, n0 = n0,
      e1 = e1, n1 = n1)
  }
  temp <- Vectorize(temp)
  pvals.logOR <- outer(f0.mods, f1.mods, temp)
  pval.logOR.ori <- pvals.logOR[f0.mods == 0, f1.mods == 0]
  FI.logOR.sig2nonsig <- FI.logOR.nonsig2sig <- NA
  mods.logOR <- NA
  FI0.logOR.sig2nonsig <- FI0.logOR.nonsig2sig <-
    FI1.logOR.sig2nonsig <- FI1.logOR.nonsig2sig <- NA
  mods0.logOR <- mods1.logOR <- NA
  if(pval.logOR.ori < alpha){
    if(any(pvals.logOR >= alpha)){
      FI.logOR.sig2nonsig <- min(tot.mods[pvals.logOR >= alpha])
      mods.logOR <- which(tot.mods == FI.logOR.sig2nonsig &
         pvals.logOR >= alpha, arr.ind = TRUE)
      mods.logOR <- cbind(f0.mods[mods.logOR[,1]],
        f1.mods[mods.logOR[,2]])
      colnames(mods.logOR) <- c("arm0", "arm1")
      if(modify0 != "none" & modify1 != "none"){
        pvals0.logOR <- pvals.logOR[, f1.mods == 0]
        tot.mods0 <- tot.mods[, f1.mods == 0]
        pvals1.logOR <- pvals.logOR[f0.mods == 0,]
        tot.mods1 <- tot.mods[f0.mods == 0,]
        if(any(pvals0.logOR >= alpha)){
          FI0.logOR.sig2nonsig <-
            min(tot.mods0[pvals0.logOR >= alpha])
          mods0.logOR <-
            f0.mods[tot.mods0 == FI0.logOR.sig2nonsig &
            pvals0.logOR >= alpha]
        }else{
          FI0.logOR.sig2nonsig <- NA
        }
        if(any(pvals1.logOR >= alpha)){
          FI1.logOR.sig2nonsig <-
            min(tot.mods1[pvals1.logOR >= alpha])
          mods1.logOR <-
            f1.mods[tot.mods1 == FI1.logOR.sig2nonsig &
            pvals1.logOR >= alpha]
        }else{
          FI1.logOR.sig2nonsig <- NA
        }
      }
    }else{
      FI.logOR.sig2nonsig <- NA
    }
  }else{
    if(any(pvals.logOR < alpha)){
      FI.logOR.nonsig2sig <- min(tot.mods[pvals.logOR < alpha])
      mods.logOR <- which(tot.mods == FI.logOR.nonsig2sig &
         pvals.logOR < alpha, arr.ind = TRUE)
      mods.logOR <- cbind(f0.mods[mods.logOR[,1]],
        f1.mods[mods.logOR[,2]])
      colnames(mods.logOR) <- c("arm0", "arm1")
      if(modify0 != "none" & modify1 != "none"){
        pvals0.logOR <- pvals.logOR[, f1.mods == 0]
        tot.mods0 <- tot.mods[, f1.mods == 0]
        pvals1.logOR <- pvals.logOR[f0.mods == 0,]
        tot.mods1 <- tot.mods[f0.mods == 0,]
        if(any(pvals0.logOR < alpha)){
          FI0.logOR.nonsig2sig <-
            min(tot.mods0[pvals0.logOR < alpha])
          mods0.logOR <-
            f0.mods[tot.mods0 == FI0.logOR.nonsig2sig &
            pvals0.logOR < alpha]
        }else{
          FI0.logOR.nonsig2sig <- NA
        }
        if(any(pvals1.logOR < alpha)){
          FI1.logOR.nonsig2sig <-
            min(tot.mods1[pvals1.logOR < alpha])
          mods1.logOR <-
            f1.mods[tot.mods1 == FI1.logOR.nonsig2sig &
            pvals1.logOR < alpha]
        }else{
          FI1.logOR.nonsig2sig <- NA
        }
      }
    }else{
      FI.logOR.nonsig2sig <- NA
    }
  }
  out.logOR <- list(pval.logOR = pval.logOR.ori,
    FI.logOR.sig2nonsig = FI.logOR.sig2nonsig,
    FI.logOR.nonsig2sig = FI.logOR.nonsig2sig,
    mods.logOR = mods.logOR)
  if(modify0 != "none" & modify1 != "none"){
    out.logOR <- c(out.logOR,
      list(FI0.logOR.sig2nonsig = FI0.logOR.sig2nonsig,
      FI0.logOR.nonsig2sig = FI0.logOR.nonsig2sig,
      mods0.logOR = mods0.logOR,
      FI1.logOR.sig2nonsig = FI1.logOR.sig2nonsig,
      FI1.logOR.nonsig2sig = FI1.logOR.nonsig2sig,
      mods1.logOR = mods1.logOR))
  }
  if(save){
    out.logOR <- c(out.logOR, list(pvals.logOR = pvals.logOR))
  }
  out <- c(out, out.logOR)
}

## log relative risk
if(is.element("logRR", methods)){
  temp <- function(f0, f1){
    pval.logRR(f0 = f0, f1 = f1, e0 = e0, n0 = n0,
      e1 = e1, n1 = n1)
  }
  temp <- Vectorize(temp)
  pvals.logRR <- outer(f0.mods, f1.mods, temp)
```

```
pval.logRR.ori <- pvals.logRR[f0.mods == 0, f1.mods == 0]
FI.logRR.sig2nonsig <- FI.logRR.nonsig2sig <- NA
mods.logRR <- NA
FI0.logRR.sig2nonsig <- FI0.logRR.nonsig2sig <-
  FI1.logRR.sig2nonsig <- FI1.logRR.nonsig2sig <- NA
mods0.logRR <- mods1.logRR <- NA
if(pval.logRR.ori < alpha){
  if(any(pvals.logRR >= alpha)){
    FI.logRR.sig2nonsig <- min(tot.mods[pvals.logRR >= alpha])
    mods.logRR <- which(tot.mods == FI.logRR.sig2nonsig &
       pvals.logRR >= alpha, arr.ind = TRUE)
    mods.logRR <- cbind(f0.mods[mods.logRR[,1]],
      f1.mods[mods.logRR[,2]])
    colnames(mods.logRR) <- c("arm0", "arm1")
    if(modify0 != "none" & modify1 != "none"){
      pvals0.logRR <- pvals.logRR[, f1.mods == 0]
      tot.mods0 <- tot.mods[, f1.mods == 0]
      pvals1.logRR <- pvals.logRR[f0.mods == 0,]
      tot.mods1 <- tot.mods[f0.mods == 0,]
      if(any(pvals0.logRR >= alpha)){
        FI0.logRR.sig2nonsig <-
          min(tot.mods0[pvals0.logRR >= alpha])
        mods0.logRR <-
          f0.mods[tot.mods0 == FI0.logRR.sig2nonsig &
          pvals0.logRR >= alpha]
      }else{
        FI0.logRR.sig2nonsig <- NA
      }
      if(any(pvals1.logRR >= alpha)){
        FI1.logRR.sig2nonsig <-
          min(tot.mods1[pvals1.logRR >= alpha])
        mods1.logRR <-
          f1.mods[tot.mods1 == FI1.logRR.sig2nonsig &
          pvals1.logRR >= alpha]
      }else{
        FI1.logRR.sig2nonsig <- NA
      }
    }
  }else{
    FI.logRR.sig2nonsig <- NA
  }
}else{
  if(any(pvals.logRR < alpha)){
    FI.logRR.nonsig2sig <- min(tot.mods[pvals.logRR < alpha])
    mods.logRR <- which(tot.mods == FI.logRR.nonsig2sig &
       pvals.logRR < alpha, arr.ind = TRUE)
    mods.logRR <- cbind(f0.mods[mods.logRR[,1]],
      f1.mods[mods.logRR[,2]])
    colnames(mods.logRR) <- c("arm0", "arm1")
    if(modify0 != "none" & modify1 != "none"){
      pvals0.logRR <- pvals.logRR[, f1.mods == 0]
      tot.mods0 <- tot.mods[, f1.mods == 0]
      pvals1.logRR <- pvals.logRR[f0.mods == 0,]
      tot.mods1 <- tot.mods[f0.mods == 0,]
      if(any(pvals0.logRR < alpha)){
        FI0.logRR.nonsig2sig <-
          min(tot.mods0[pvals0.logRR < alpha])
        mods0.logRR <-
          f0.mods[tot.mods0 == FI0.logRR.nonsig2sig &
          pvals0.logRR < alpha]
      }else{
        FI0.logRR.nonsig2sig <- NA
      }
```

```
        if(any(pvals1.logRR < alpha)){
          FI1.logRR.nonsig2sig <-
            min(tot.mods1[pvals1.logRR < alpha])
          mods1.logRR <-
            f1.mods[tot.mods1 == FI1.logRR.nonsig2sig &
            pvals1.logRR < alpha]
        }else{
          FI1.logRR.nonsig2sig <- NA
        }
      }
    }else{
      FI.logRR.nonsig2sig <- NA
    }
  }
  out.logRR <- list(pval.logRR = pval.logRR.ori,
    FI.logRR.sig2nonsig = FI.logRR.sig2nonsig,
    FI.logRR.nonsig2sig = FI.logRR.nonsig2sig,
    mods.logRR = mods.logRR)
  if(modify0 != "none" & modify1 != "none"){
    out.logRR <- c(out.logRR,
      list(FI0.logRR.sig2nonsig = FI0.logRR.sig2nonsig,
      FI0.logRR.nonsig2sig = FI0.logRR.nonsig2sig,
      mods0.logRR = mods0.logRR,
      FI1.logRR.sig2nonsig = FI1.logRR.sig2nonsig,
      FI1.logRR.nonsig2sig = FI1.logRR.nonsig2sig,
      mods1.logRR = mods1.logRR))
  }
  if(save){
    out.logRR <- c(out.logRR, list(pvals.logRR = pvals.logRR))
  }
  out <- c(out, out.logRR)
}

## risk difference
if(is.element("RD", methods)){
  temp <- function(f0, f1){
    pval.RD(f0 = f0, f1 = f1, e0 = e0, n0 = n0,
      e1 = e1, n1 = n1)
  }
  temp <- Vectorize(temp)
  pvals.RD <- outer(f0.mods, f1.mods, temp)
  pval.RD.ori <- pvals.RD[f0.mods == 0, f1.mods == 0]
  FI.RD.sig2nonsig <- FI.RD.nonsig2sig <- NA
  mods.RD <- NA
  FI0.RD.sig2nonsig <- FI0.RD.nonsig2sig <-
    FI1.RD.sig2nonsig <- FI1.RD.nonsig2sig <- NA
  mods0.RD <- mods1.RD <- NA
  if(pval.RD.ori < alpha){
    if(any(pvals.RD >= alpha)){
      FI.RD.sig2nonsig <- min(tot.mods[pvals.RD >= alpha])
      mods.RD <- which(tot.mods == FI.RD.sig2nonsig &
         pvals.RD >= alpha, arr.ind = TRUE)
      mods.RD <- cbind(f0.mods[mods.RD[,1]],
        f1.mods[mods.RD[,2]])
      colnames(mods.RD) <- c("arm0", "arm1")
      if(modify0 != "none" & modify1 != "none"){
        pvals0.RD <- pvals.RD[, f1.mods == 0]
        tot.mods0 <- tot.mods[, f1.mods == 0]
        pvals1.RD <- pvals.RD[f0.mods == 0,]
        tot.mods1 <- tot.mods[f0.mods == 0,]
        if(any(pvals0.RD >= alpha)){
          FI0.RD.sig2nonsig <-
            min(tot.mods0[pvals0.RD >= alpha])
        mods0.RD <-
          f0.mods[tot.mods0 == FI0.RD.sig2nonsig &
          pvals0.RD >= alpha]
      }else{
        FI0.RD.sig2nonsig <- NA
      }
      if(any(pvals1.RD >= alpha)){
        FI1.RD.sig2nonsig <-
          min(tot.mods1[pvals1.RD >= alpha])
        mods1.RD <-
          f1.mods[tot.mods1 == FI1.RD.sig2nonsig &
          pvals1.RD >= alpha]
      }else{
        FI1.RD.sig2nonsig <- NA
      }
    }

  }else{
    FI.RD.sig2nonsig <- NA
  }
}else{
  if(any(pvals.RD < alpha)){
    FI.RD.nonsig2sig <- min(tot.mods[pvals.RD < alpha])
    mods.RD <- which(tot.mods == FI.RD.nonsig2sig &
       pvals.RD < alpha, arr.ind = TRUE)
    mods.RD <- cbind(f0.mods[mods.RD[,1]],
      f1.mods[mods.RD[,2]])
    colnames(mods.RD) <- c("arm0", "arm1")
    if(modify0 != "none" & modify1 != "none"){
      pvals0.RD <- pvals.RD[, f1.mods == 0]
      tot.mods0 <- tot.mods[, f1.mods == 0]
      pvals1.RD <- pvals.RD[f0.mods == 0,]
      tot.mods1 <- tot.mods[f0.mods == 0,]
      if(any(pvals0.RD < alpha)){
        FI0.RD.nonsig2sig <-
          min(tot.mods0[pvals0.RD < alpha])
        mods0.RD <-
          f0.mods[tot.mods0 == FI0.RD.nonsig2sig &
          pvals0.RD < alpha]
      }else{
        FI0.RD.nonsig2sig <- NA
      }
      if(any(pvals1.RD < alpha)){
        FI1.RD.nonsig2sig <-
          min(tot.mods1[pvals1.RD < alpha])
        mods1.RD <-
          f1.mods[tot.mods1 == FI1.RD.nonsig2sig &
          pvals1.RD < alpha]
      }else{
        FI1.RD.nonsig2sig <- NA
      }
    }
  }else{
    FI.RD.nonsig2sig <- NA
  }
}
out.RD <- list(pval.RD = pval.RD.ori,
  FI.RD.sig2nonsig = FI.RD.sig2nonsig,
  FI.RD.nonsig2sig = FI.RD.nonsig2sig,
  mods.RD = mods.RD)
if(modify0 != "none" & modify1 != "none"){
  out.RD <- c(out.RD,
    list(FI0.RD.sig2nonsig = FI0.RD.sig2nonsig,
        FI0.RD.nonsig2sig = FI0.RD.nonsig2sig,
        mods0.RD = mods0.RD,
        FI1.RD.sig2nonsig = FI1.RD.sig2nonsig,
        FI1.RD.nonsig2sig = FI1.RD.nonsig2sig,
        mods1.RD = mods1.RD))
    }
    if(save){
      out.RD <- c(out.RD, list(pvals.RD = pvals.RD))
    }
    out <- c(out, out.RD)
  }

  return(out)
}


## pvals.plot() generates plots displaying significance of
##   different event status modifications.
## arguments: rslt.all is an object produced from the function
##   fragility.all() with save = TRUE;
##   method specifies the method ("Fisher", "chisq", "logOR",
##   "logRR", or "RD") on which the plot is based;
##   modify0 and modify1 are logical values and specify
##   the event status modifications on which arm(s) are of interest:
##   one of them being TRUE will lead to a curve of p-values against
##   the number of modified events in a single arm, while both being
##   TRUE will lead to a matrix plot of p-values;
##   trun specifies the threshold of p-values on the negative
##   log 10 scale in the plot;
##   cex.pts specifies point size;
##   main gives the plot title;
##   legend.pvals specifies the p-value cutoffs (in addition to the
##   pre-specified significance level alpha) to be plotted in
##   the plot legend;
##   cex.legend.pvals specifies the font size in the legend;
##   mar specifies plot margins if both arms are modified.
pvals.plot <- function(rslt.all, method,
  modify0 = TRUE, modify1 = TRUE,
  trun = 10, cex.pts = 0.5, main = "",
  legend.pvals = NULL, cex.legend.pvals = 0.6,
  mar = c(5, 4, 4, 2) + 0.1){
  pvals <- rslt.all[[paste0("pvals.", method)]]
  f0.range <- rslt.all$f0.range
  f1.range <- rslt.all$f1.range
  alpha <- rslt.all$alpha
  tot.mods <- rslt.all$tot.mods
  mods <- rslt.all[[paste0("mods.", method)]]
  if(modify0 + modify1 == 0){
    stop("No event status modified.")
  }
  if(modify0 + modify1 == 1){
    if(modify0){
      if(rslt.all$modify0 == "none"){
        stop("Treatment arm 0 is not modified")
      }
      pvals.pts <- pvals[,f1.range[1]:f1.range[2] == 0]
      x.pts <- f0.range[1]:f0.range[2]
      x.limit <- f0.range
      x.lab <- "Number of modified events in treatment arm 0"
      if(rslt.all$modify1 == "none"){
        mods.single <- rslt.all[[paste0("mods.", method)]]
      }else{
        mods.single <- rslt.all[[paste0("mods0.", method)]]
    }
  }
  if(modify1){
    if(rslt.all$modify1 == "none"){
      stop("Treatment arm 1 is not modified")
    }
    pvals.pts <- pvals[f0.range[1]:f0.range[2] == 0,]
    x.pts <- f1.range[1]:f1.range[2]
    x.limit <- f1.range
    x.lab <- "Number of modified events in treatment arm 1"
    if(rslt.all$modify0 == "none"){
      mods.single <- rslt.all[[paste0("mods.", method)]]
    }else{
      mods.single <- rslt.all[[paste0("mods1.", method)]]
    }
  }
  trun.pts.idx <- -log10(pvals.pts) > trun
  if(any(trun.pts.idx)){
     pvals.pts[trun.pts.idx] <- 10^(-trun)
  }
  plot(mean(x.limit), -log10(alpha),
    col = "white", xlab = x.lab, xlim = x.limit,
    ylab = expression(-log[10](italic(P))),
    ylim = c(0, max(-log10(pvals.pts))),
    axes = TRUE, mgp = c(1.5, 0.5, 0), cex.lab = 1)
  rect(xleft = x.limit[1] - 1000, ybottom = -1,
    xright = x.limit[2] + 1000, ytop = -log10(alpha),
    col = adjustcolor("forestgreen", alpha.f = 0.1),
    border = "white")
  if(any(pvals.pts < alpha)){
    rect(xleft = x.limit[1] - 1000, ybottom = -log10(alpha),
      xright = x.limit[2] + 1000, ytop = max(-log10(pvals.pts)) + 10,
      col = adjustcolor("firebrick", alpha.f = 0.1),
      border = "white")
  }
  abline(v = 0, lty = 2, col = "gray50")
  pchs <- rep(1, length(x.pts))
  pchs[x.pts == 0] <- 15
  pchs[which(x.pts %in% mods.single)] <- 17
  if(any(trun.pts.idx)){
    pchs[trun.pts.idx] <- 3
  }
  cexs <- rep(cex.pts, length(x.pts))
  cexs[x.pts == 0] <- 2 * cex.pts
  cexs[which(x.pts %in% mods.single)] <- 2 * cex.pts
  points(x.pts, -log10(pvals.pts), cex = cexs, pch = pchs)
  mtext(text = main, side = 3, line = 0.5, outer = FALSE,
    cex = 1, font = 1)
}
if(modify0 + modify1 == 2){
  if(rslt.all$modify0 == "none" | rslt.all$modify1 == "none" ){
    stop("At least one treatment arm is not modified")
  }
  layout(matrix(c(1, 2), ncol = 2, byrow = TRUE),
    widths = c(10, 1))
  par(mar = mar)
  x.pts <- f0.range[1]:f0.range[2]
  y.pts <- f1.range[1]:f1.range[2]
  x.limit <- f0.range
  x.lab <- "Number of modified events in treatment arm 0"
  y.limit <- f1.range
  y.lab <- "Number of modified events in treatment arm 1"
  plot(mean(x.limit), mean(y.limit),
    col = "white", xlab = x.lab, xlim = x.limit,
    ylab = y.lab, ylim = y.limit,
    axes = TRUE, mgp = c(1.5, 0.5, 0), cex.lab = 1)
  ori.idx <- tot.mods == 0
  FI.idx <- matrix(0, dim(pvals)[1], dim(pvals)[2])
  for(i in 1:dim(mods)[1]){
    FI.idx[x.pts == mods[i,1], y.pts == mods[i,2]] <- 1
  }
  signif <- c(pvals) < alpha
  neg.log10.pvals <- -log10(c(pvals))
  neg.log10.pvals.trun <- neg.log10.pvals
  neg.log10.pvals.trun[neg.log10.pvals.trun > trun] <- trun
  pvals.col.trans <- 0.1 +
    abs(neg.log10.pvals.trun + log10(alpha))/
    abs((c(signif)*10 + log10(alpha)))*(1 - 0.1)
  pvals.col <- ifelse(c(signif), "firebrick", "forestgreen")
  adjustcolor.vec <- Vectorize(adjustcolor)
  pchs <- rep(16, length(c(pvals)))
  cexs <- rep(cex.pts, length(c(pvals)))
  pchs[c(ori.idx) == 1] <- 15
  pchs[c(FI.idx) == 1] <- 17
  cexs[c(ori.idx) == 1 | c(FI.idx) == 1] <- 2 * cex.pts
  points(x = rep(x.pts, length(y.pts)),
    y = rep(y.pts, each = length(x.pts)),
    col = adjustcolor.vec(pvals.col, alpha.f = pvals.col.trans),
    cex = cexs, pch = pchs)
  abline(h = 0, lty = 2, col = "gray50")
  abline(v = 0, lty = 2, col = "gray50")
  points(x = rep(x.pts, length(y.pts))[c(ori.idx) == 1],
    y = rep(y.pts, each = length(x.pts))[c(ori.idx) == 1],
    col = "black",#pvals.col[c(ori.idx) == 1],
    cex = 2 * cex.pts, pch = 0)
  points(x = rep(x.pts, length(y.pts))[c(FI.idx) == 1],
    y = rep(y.pts, each = length(x.pts))[c(FI.idx) == 1],
    col = "black",#pvals.col[c(FI.idx) == 1],
    cex = 2 * cex.pts, pch = 2)
  mtext(text = main, side = 3, line = 0.5, outer = FALSE,
    cex = 1, font = 1)
  par(mar = c(mar[1], 0, mar[3], mar[4]))
  plot(1, type = "n", xlim = c(0, trun/10), ylim = c(0, trun),
    axes = FALSE, frame.plot = FALSE, xlab = "", ylab = "")
  n.nonsig <- 10
  n.sig <- round((trun + log10(alpha))/(-log10(alpha)/n.nonsig))
  ybs <- c(seq(0, -log10(alpha) * (1 - 1/n.nonsig),
    length.out = n.nonsig), seq(-log10(alpha),
    trun - (trun + log10(alpha))/n.sig, length.out = n.sig))
  yus <- c(seq(-log10(alpha)/n.nonsig, -log10(alpha),
    length.out = n.nonsig), seq(-log10(alpha) +
    (trun + log10(alpha))/n.sig, trun, length.out = n.sig))
  cols <- c(adjustcolor.vec("forestgreen", alpha.f =
    seq(1, 0.1, length.out = n.nonsig)),
    adjustcolor.vec("firebrick", alpha.f =
    seq(0.1, 1, length.out = n.sig)))
  rect(xleft = 0, ybottom = ybs, xright = trun/10, ytop = yus,
    col = cols, border = NA)
  abline(h = -log10(alpha), col = "white")
  abline(h = trun, col = "white")
  mtext(text = expression(italic(P)~value), side = 3, line = 0)
  mtext(text = format(alpha, scientific = FALSE),
    at = -log10(alpha), side = 4, line = 0,
    cex = cex.legend.pvals)
  mtext(text = bquote(""<=10^-.(trun)), at = trun, side = 4, line = 0,
    cex = cex.legend.pvals)
```

```
    if(!is.null(legend.pvals)){
      mtext(text = legend.pvals, at = -log10(legend.pvals),
        side = 4, line = 0, cex = cex.legend.pvals)
      abline(h = -log10(legend.pvals), col = "white")
    }
    resetPar <- function(){
      dev.new()
      op <- par(no.readonly = TRUE)
      dev.off()
      op
    }
    par(resetPar())
  }
}


## fragility.iter() derives the fragility index via iterations.
## arguments are similar to those in fragility.all().
fragility.iter <- function(e0, n0, e1, n1, methods = "Fisher",
  modify0 = "both", modify1 = "both", alpha = 0.05){
  ne0 <- n0 - e0
  ne1 <- n1 - e1
  if(modify0 == "none" & modify1 == "none"){
    stop("No event status modified.")
  }
  if(modify0 == "increase") f0.range <- c(0, ne0)
  if(modify0 == "decrease") f0.range <- c(-e0, 0)
  if(modify0 == "both") f0.range <- c(-e0, ne0)
  if(modify0 == "none") f0.range <- c(0, 0)
  if(modify1 == "increase") f1.range <- c(0, ne1)
  if(modify1 == "decrease") f1.range <- c(-e1, 0)
  if(modify1 == "both") f1.range <- c(-e1, ne1)
  if(modify1 == "none") f1.range <- c(0, 0)
  f0.mods <- f0.range[1]:f0.range[2]
  f1.mods <- f1.range[1]:f1.range[2]
  tot.mods <- outer(abs(f0.mods), abs(f1.mods), "+")

  out <- list(methods = methods, alpha = alpha,
    modify0 = modify0, modify1 = modify1,
    f0.range = f0.range, f1.range = f1.range)

  ## Fisher's exact test
  if(is.element("Fisher", methods)){
    temp <- function(f0, f1){
      fisher.test(rbind(c(e0 + f0, n0 - e0 - f0),
        c(e1 + f1, n1 - e1 - f1)),
        alternative = "two.sided")$p.value
    }
    pval.Fisher.ori <- temp(0, 0)
    signif.ori <- pval.Fisher.ori < alpha
    uniq.mods <- sort(unique(tot.mods))
    if(is.element(0, uniq.mods)){
      uniq.mods <- uniq.mods[uniq.mods != 0]
    }
    FI.Fisher <- NA
    signif.alter <- 0
    for(f in uniq.mods){
      idx.temp <- which(tot.mods == f, arr.ind = TRUE)
      for(i in 1:dim(idx.temp)[1]){
        pval.temp <- temp(f0 = f0.mods[idx.temp[i,1]],
          f1 = f1.mods[idx.temp[i,2]])
        signif.temp <- pval.temp < alpha
        if(signif.temp != signif.ori){
```

```
        FI.Fisher <- f
        signif.alter <- 1
        break
      }
    }
    if(signif.alter == 1) break
  }
  if(signif.alter == 0) FI.Fisher <- NA
  FI.Fisher.sig2nonsig <- FI.Fisher.nonsig2sig <- NA
  if(signif.ori){
    FI.Fisher.sig2nonsig <- FI.Fisher
  }else{
    FI.Fisher.nonsig2sig <- FI.Fisher
  }
  out <- c(out, list(pval.Fisher = pval.Fisher.ori,
    FI.Fisher.sig2nonsig = FI.Fisher.sig2nonsig,
    FI.Fisher.nonsig2sig = FI.Fisher.nonsig2sig))
}

## Chi-squared test
if(is.element("chisq", methods)){
  temp <- function(f0, f1){
    if((abs(e0 + f0) < 1e-6 & abs(e1 + f1) < 1e-6) |
      (abs(n0 - e0 - f0) < 1e-6 & abs(n1 - e1 - f1) < 1e-6)){
      temp.out <- 1
    }else{
      temp.out <- chisq.test(rbind(c(e0 + f0, n0 - e0 - f0),
        c(e1 + f1, n1 - e1 - f1)))$p.value
    }
    return(temp.out)
  }
  options(warn = -1)
  pval.chisq.ori <- temp(0, 0)
  signif.ori <- pval.chisq.ori < alpha
  uniq.mods <- sort(unique(tot.mods))
  if(is.element(0, uniq.mods)){
    uniq.mods <- uniq.mods[uniq.mods != 0]
  }
  FI.chisq <- NA
  signif.alter <- 0
  for(f in uniq.mods){
    idx.temp <- which(tot.mods == f, arr.ind = TRUE)
    for(i in 1:dim(idx.temp)[1]){
      pval.temp <- temp(f0 = f0.mods[idx.temp[i,1]],
        f1 = f1.mods[idx.temp[i,2]])
      signif.temp <- pval.temp < alpha
      if(signif.temp != signif.ori){
        FI.chisq <- f
        signif.alter <- 1
        break
      }
    }
    if(signif.alter == 1) break
  }
  if(signif.alter == 0) FI.chisq <- NA
  FI.chisq.sig2nonsig <- FI.chisq.nonsig2sig <- NA
  if(signif.ori){
    FI.chisq.sig2nonsig <- FI.chisq
  }else{
    FI.chisq.nonsig2sig <- FI.chisq
  }
  out <- c(out, list(pval.chisq = pval.chisq.ori,
    FI.chisq.sig2nonsig = FI.chisq.sig2nonsig,
    FI.chisq.nonsig2sig = FI.chisq.nonsig2sig))
  options(warn = 0)
}

## log odds ratio
if(is.element("logOR", methods)){
  temp <- function(f0, f1){
    pval.logOR(f0 = f0, f1 = f1, e0 = e0, n0 = n0,
      e1 = e1, n1 = n1)
  }
  pval.logOR.ori <- temp(0, 0)
  signif.ori <- pval.logOR.ori < alpha
  uniq.mods <- sort(unique(tot.mods))
  if(is.element(0, uniq.mods)){
    uniq.mods <- uniq.mods[uniq.mods != 0]
  }
  FI.logOR <- NA
  signif.alter <- 0
  for(f in uniq.mods){
    idx.temp <- which(tot.mods == f, arr.ind = TRUE)
    for(i in 1:dim(idx.temp)[1]){
      pval.temp <- temp(f0 = f0.mods[idx.temp[i,1]],
        f1 = f1.mods[idx.temp[i,2]])
      signif.temp <- pval.temp < alpha
      if(signif.temp != signif.ori){
        FI.logOR <- f
        signif.alter <- 1
        break
      }
    }
    if(signif.alter == 1) break
  }
  if(signif.alter == 0) FI.logOR <- NA
  FI.logOR.sig2nonsig <- FI.logOR.nonsig2sig <- NA
  if(signif.ori){
    FI.logOR.sig2nonsig <- FI.logOR
  }else{
    FI.logOR.nonsig2sig <- FI.logOR
  }
  out <- c(out, list(pval.logOR = pval.logOR.ori,
    FI.logOR.sig2nonsig = FI.logOR.sig2nonsig,
    FI.logOR.nonsig2sig = FI.logOR.nonsig2sig))
}

## log relative risk
if(is.element("logRR", methods)){
  temp <- function(f0, f1){
    pval.logRR(f0 = f0, f1 = f1, e0 = e0, n0 = n0,
      e1 = e1, n1 = n1)
  }
  pval.logRR.ori <- temp(0, 0)
  signif.ori <- pval.logRR.ori < alpha
  uniq.mods <- sort(unique(tot.mods))
  if(is.element(0, uniq.mods)){
    uniq.mods <- uniq.mods[uniq.mods != 0]
  }
  FI.logRR <- NA
  signif.alter <- 0
  for(f in uniq.mods){
    idx.temp <- which(tot.mods == f, arr.ind = TRUE)
    for(i in 1:dim(idx.temp)[1]){
      pval.temp <- temp(f0 = f0.mods[idx.temp[i,1]],
        f1 = f1.mods[idx.temp[i,2]])
        signif.temp <- pval.temp < alpha
        if(signif.temp != signif.ori){
          FI.logRR <- f
          signif.alter <- 1
          break
        }
      }
      if(signif.alter == 1) break
    }
    if(signif.alter == 0) FI.logRR <- NA
    FI.logRR.sig2nonsig <- FI.logRR.nonsig2sig <- NA
    if(signif.ori){
      FI.logRR.sig2nonsig <- FI.logRR
    }else{
      FI.logRR.nonsig2sig <- FI.logRR
    }
    out <- c(out, list(pval.logRR = pval.logRR.ori,
      FI.logRR.sig2nonsig = FI.logRR.sig2nonsig,
      FI.logRR.nonsig2sig = FI.logRR.nonsig2sig))
  }

  ## risk difference
  if(is.element("RD", methods)){
    temp <- function(f0, f1){
      pval.RD(f0 = f0, f1 = f1, e0 = e0, n0 = n0,
        e1 = e1, n1 = n1)
    }
    pval.RD.ori <- temp(0, 0)
    signif.ori <- pval.RD.ori < alpha
    uniq.mods <- sort(unique(tot.mods))
    if(is.element(0, uniq.mods)){
      uniq.mods <- uniq.mods[uniq.mods != 0]
    }
    FI.RD <- NA
    signif.alter <- 0
    for(f in uniq.mods){
      idx.temp <- which(tot.mods == f, arr.ind = TRUE)
      for(i in 1:dim(idx.temp)[1]){
        pval.temp <- temp(f0 = f0.mods[idx.temp[i,1]],
          f1 = f1.mods[idx.temp[i,2]])
        signif.temp <- pval.temp < alpha
        if(signif.temp != signif.ori){
          FI.RD <- f
          signif.alter <- 1
          break
        }
      }
      if(signif.alter == 1) break
    }
    if(signif.alter == 0) FI.RD <- NA
    FI.RD.sig2nonsig <- FI.RD.nonsig2sig <- NA
    if(signif.ori){
      FI.RD.sig2nonsig <- FI.RD
    }else{
      FI.RD.nonsig2sig <- FI.RD
    }
    out <- c(out, list(pval.RD = pval.RD.ori,
      FI.RD.sig2nonsig = FI.RD.sig2nonsig,
      FI.RD.nonsig2sig = FI.RD.nonsig2sig))
  }

  return(out)
}
```

```
## frag.study.alpha() efficiently derives the FI at
##   different significance levels and calculates the average FI
## arguments are similar to those in previous functions;
##   alpha.from and alpha.to specify the lower and upper bounds
##   of significance levels to be considered.
frag.study.alpha <- function(e0, n0, e1, n1, methods,
  modify0 = "both", modify1 = "both",
  alpha.from = 0.005, alpha.to = 0.05, alpha.breaks = 100){
  ne0 <- n0 - e0
  ne1 <- n1 - e1
  ori.data <- matrix(c(e0, ne0, e1, ne1), 2, 2, byrow = TRUE)
  rownames(ori.data) <- c("group 0", "group 1")
  colnames(ori.data) <- c("event", "no event")

  if(modify0 == "increase") f0.range <- c(0, ne0)
  if(modify0 == "decrease") f0.range <- c(-e0, 0)
  if(modify0 == "both") f0.range <- c(-e0, ne0)
  if(modify0 == "none") f0.range <- c(0, 0)
  if(modify1 == "increase") f1.range <- c(0, ne1)
  if(modify1 == "decrease") f1.range <- c(-e1, 0)
  if(modify1 == "both") f1.range <- c(-e1, ne1)
  if(modify1 == "none") f1.range <- c(0, 0)
  f0.mods <- f0.range[1]:f0.range[2]
  f1.mods <- f1.range[1]:f1.range[2]
  alphas <- seq(from = alpha.from, to = alpha.to,
    length.out = alpha.breaks)

  out <- list(data = ori.data, methods = methods, alphas = alphas,
    modify0 = modify0, modify1 = modify1,
    f0.range = f0.range, f1.range = f1.range)

  temp.Fisher <- function(f0, f1){
    fisher.test(rbind(c(e0 + f0, n0 - e0 - f0),
      c(e1 + f1, n1 - e1 - f1)))$p.value
  }
  temp.chisq <- function(f0, f1){
    if((abs(e0 + f0) < 1e-6 & abs(e1 + f1) < 1e-6) |
      (abs(n0 - e0 - f0) < 1e-6 & abs(n1 - e1 - f1) < 1e-6)){
      temp.out <- 1
    }else{
      temp.out <- suppressWarnings(
        chisq.test(rbind(c(e0 + f0, n0 - e0 - f0),
        c(e1 + f1, n1 - e1 - f1)))$p.value)
    }
    return(temp.out)
  }
  temp.logOR <- function(f0, f1){
    pval.logOR(f0 = f0, f1 = f1, e0 = e0, n0 = n0,
      e1 = e1, n1 = n1)
  }
  temp.logRR <- function(f0, f1){
    pval.logRR(f0 = f0, f1 = f1, e0 = e0, n0 = n0,
      e1 = e1, n1 = n1)
  }
  temp.RD <- function(f0, f1){
    pval.RD(f0 = f0, f1 = f1, e0 = e0, n0 = n0,
      e1 = e1, n1 = n1)
  }

  if(modify0 == "none" & modify1 == "none"){
    pval <- NULL
    for(i in 1:length(methods)){
      temp.m <- methods[i]
      temp <- get(paste0("temp.", temp.m))
      pval.temp <- temp(0, 0)
      pval <- c(pval, pval.temp)
    }
    names(pval) <- methods
    out <- c(out, list(pval = pval))
    return(out)
  }

  uniq.mods <- 1:(max(abs(f0.mods)) + max(abs(f1.mods)))

  pval <- FI <- FI.avg <- NULL
  for(i in 1:length(methods)){
    temp.m <- methods[i]
    temp <- get(paste0("temp.", temp.m))
    temp.pval <- temp(0, 0)
    temp.signif <- alphas > temp.pval
    temp.FI <- temp.check <- rep(NA, alpha.breaks)
    for(f in uniq.mods){
      tt.mods <- rbind(cbind((-f):f, f - abs((-f):f)),
        cbind((-(f - 1)):(f - 1),
        -(f - abs((-(f - 1)):(f - 1)))))
      tt.mods <- tt.mods[is.element(tt.mods[,1], f0.mods) &
        is.element(tt.mods[,2], f1.mods),]
      if(is.vector(tt.mods)){
        tt.mods <- matrix(tt.mods, ncol = 2)
      }
      for(j in 1:dim(tt.mods)[1]){
        tt.pval <- temp(f0 = tt.mods[j,1], f1 = tt.mods[j,2])
        if(any(temp.signif & tt.pval >= alphas &
          is.na(temp.check))){
          tt.idx.alpha <- which(temp.signif & tt.pval >=
            alphas & is.na(temp.check))
          temp.FI[tt.idx.alpha] <- f
          temp.check[tt.idx.alpha] <- 1
        }
        if(any(!temp.signif & tt.pval < alphas &
          is.na(temp.check))){
          tt.idx.alpha <- which(!temp.signif & tt.pval <
            alphas & is.na(temp.check))
          temp.FI[tt.idx.alpha] <- f
          temp.check[tt.idx.alpha] <- 1
        }
        if(all(!is.na(temp.check))) break
      }
      if(all(!is.na(temp.check))) break
    }
    if(any(is.na(temp.check))){
      temp.FI[is.na(temp.check)] <- NA
    }
    pval <- c(pval, temp.pval)
    FI <- cbind(FI, temp.FI)
    FI.avg <- c(FI.avg, mean(temp.FI))
  }

  FI <- cbind(alphas, FI)
  names(pval) <- names(FI.avg) <- methods
  colnames(FI) <- c("alpha", methods)
  out <- c(out, list(pval = pval, FI = FI, FI.avg = FI.avg))
  return(out)
}
```

```
## FIvsSignif() derives the fragility index with different
##   significance levels, giving plot of fragility index
##   against significance levels.
## arguments are similar to those in pvals.plot();
##   alpha.from and alpha.to specify the lower and upper bounds
##   of significance levels to be considered.
FIvsSignif <- function(e0, n0, e1, n1, method = "Fisher",
  modify0 = "both", modify1 = "both",
  alpha.from = 0.005, alpha.to = 0.05, alpha.breaks = 100,
  cex.pts = 0.5, main = NULL){

  FI.alphas <-
    frag.study.alpha(e0 = e0, n0 = n0, e1 = e1, n1 = n1,
      methods = method, modify0 = modify0, modify1 = modify1,
      alpha.from = alpha.from, alpha.to = alpha.to,
      alpha.breaks = alpha.breaks)
  alpha.pts <- seq(from = alpha.from, to = alpha.to,
    length.out = alpha.breaks)
  FI <- FI.alphas$FI[,2]
  sig2nonsig <- as.numeric(FI.alphas$pval < alpha.pts)
  FI.average <- FI.alphas$FI.avg
  if(is.null(main)){
    main <- paste("Average fragility index =", FI.average)
  }
  if(any(is.na(FI))){
    message("Note that fragility measures of NA are not plotted.")
  }
  plot(mean(alpha.pts), mean(FI), col = "white", type = "n",
    xlab = "Significance level", xlim = c(alpha.from, alpha.to),
    ylab = "Fragility index", ylim = c(min(FI), max(FI)),
    axes = TRUE, mgp = c(1.5, 0.5, 0), cex.lab = 1)
  pchs <- rep(16, alpha.breaks)
  cols <- rep("white", alpha.breaks)
  if(any(sig2nonsig == 1)) cols[sig2nonsig == 1] <- "firebrick"
  if(any(sig2nonsig == 0)) cols[sig2nonsig == 0] <- "forestgreen"
  lines(alpha.pts, FI, col = "gray50")
  points(alpha.pts, FI, cex = cex.pts, pch = pchs, col = cols)
  mtext(text = main, side = 3, line = 0.5, outer = FALSE,
    cex = 1, font = 1)
  if(FI.alphas$pval >= alpha.from & FI.alphas$pval <= alpha.to){
    abline(v = FI.alphas$pval, lty = 2, col = "gray50")
  }
  out <- FI.alphas
  return(out)
}
```

```
################################
#### Artificial data analysis
################################

setwd(wd)

artdat1 <- data.frame(e0 = 10, n0 = 100, e1 = 30, n1 = 100)
artdat2 <- data.frame(e0 = 25, n0 = 100, e1 = 30, n1 = 100)
artdat3 <- data.frame(e0 = 15, n0 = 100, e1 = 30, n1 = 100)

all.methods <- c("Fisher", "chisq", "logOR", "logRR", "RD")

## Enumerating all cases of event status modifications
## Artificial dataset 1
begin <- Sys.time()
rslt1.all <- fragility.all(e0 = artdat1$e0, n0 = artdat1$n0,
  e1 = artdat1$e1, n1 = artdat1$n1, methods = all.methods,
  modify0 = "both", modify1 = "both", alpha = 0.05, save = TRUE)
end <- Sys.time()
end - begin
#Time difference of 6.982682 secs

## Example: displaying results
## Fisher's exact test
rslt1.all$pval.Fisher
#[1] 0.0006504107
rslt1.all$FI.Fisher.sig2nonsig
#[1] 8
rslt1.all$FI.Fisher.nonsig2sig
#[1] NA
rslt1.all$mods.Fisher
#     arm0 arm1
#[1,]    4   -4
#[2,]    5   -3
#[3,]    6   -2
#[4,]    7   -1
#[5,]    8    0
## Chi-squared test
rslt1.all$pval.chisq
#[1] 0.0007829382
rslt1.all$FI.chisq.sig2nonsig
#[1] 8
rslt1.all$FI.chisq.nonsig2sig
#[1] NA
rslt1.all$mods.chisq
#     arm0 arm1
#[1,]    4   -4
#[2,]    5   -3
#[3,]    6   -2
#[4,]    7   -1
#[5,]    8    0
## Log odds ratio
rslt1.all$pval.logOR
#[1] 0.0007033007
rslt1.all$FI.logOR.sig2nonsig
#[1] 9
rslt1.all$FI.logOR.nonsig2sig
#[1] NA
rslt1.all$mods.logOR
#     arm0 arm1
#[1,]    4   -5
#[2,]    5   -4
#[3,]    6   -3
```

```
#[4,]    7   -2
#[5,]    8   -1
#[6,]    9    0
## Log relative risk
rslt1.all$pval.logRR
#[1] 0.001100977
rslt1.all$FI.logRR.sig2nonsig
#[1] 8
rslt1.all$FI.logRR.nonsig2sig
#[1] NA
rslt1.all$mods.logRR
#     arm0 arm1
#[1,]    8    0
## Risk difference
rslt1.all$pval.RD
#[1] 0.0002607296
rslt1.all$FI.RD.sig2nonsig
#[1] 9
rslt1.all$FI.RD.nonsig2sig
#[1] NA
rslt1.all$mods.RD
#     arm0 arm1
#[1,]    5   -4
#[2,]    6   -3
#[3,]    7   -2
#[4,]    8   -1
#[5,]    9    0

## Artificial dataset 1 with alpha = 0.005
begin <- Sys.time()
rslt1.all.0.005 <- fragility.all(e0 = artdat1$e0, n0 = artdat1$n0,
  e1 = artdat1$e1, n1 = artdat1$n1, methods = all.methods,
  modify0 = "both", modify1 = "both", alpha = 0.005, save = TRUE)
end <- Sys.time()
end - begin
#Time difference of 7.230364 secs

## Figure 1
## Figure 1A: artificial dataset 1; Fisher's exact test;
##  both arm modified; alpha = 0.05
pdf(file = "Fig1A.pdf", width = 5, height = 5)
pvals.plot(rslt1.all, method = "Fisher",
  modify0 = TRUE, modify1 = TRUE,
  trun = 10, cex.pts = 0.5,
  main = bquote("(A) FI ="~.(rslt1.all$FI.Fisher.sig2nonsig)~
   "based on Fisher's exact test with"~alpha~"= 0.05"),
  legend.pvals = c(0.01, 0.001), cex.legend.pvals = 0.6,
  mar = c(3, 3, 2, 1) + 0.1)
dev.off()
## Figure 1B: artificial dataset 1; Fisher's exact test;
##  both arm modified; alpha = 0.005
pdf(file = "Fig1B.pdf", width = 5, height = 5)
pvals.plot(rslt1.all.0.005, method = "Fisher",
  modify0 = TRUE, modify1 = TRUE,
  trun = 10, cex.pts = 0.5,
  main = bquote("(B) FI ="~.(rslt1.all.0.005$FI.Fisher.sig2nonsig)~
   "based on Fisher's exact test with"~alpha~"= 0.005"),
  legend.pvals = c(0.001, 0.05), cex.legend.pvals = 0.6,
  mar = c(3, 3, 2, 1) + 0.1)
dev.off()
## Figure 1C: artificial dataset 1; Fisher's exact test;
##  arm 0 modified; alpha = 0.05
pdf(file = "Fig1C.pdf", width = 5, height = 5)
```

```
par(mar = c(3, 3, 2, 1) + 0.1)
pvals.plot(rslt1.all, method = "Fisher",
  modify0 = TRUE, modify1 = FALSE,
  trun = 10, cex.pts = 0.5,
  main = bquote("(C) FI ="~.(rslt1.all$FI0.Fisher.sig2nonsig)~
  "based on Fisher's exact test with"~alpha~"= 0.05 in arm 0"))
dev.off()
## Figure 1D: artificial dataset 1; Fisher's exact test;
##  arm 1 modified; alpha = 0.05
pdf(file = "Fig1D.pdf", width = 5, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
pvals.plot(rslt1.all, method = "Fisher",
  modify0 = FALSE, modify1 = TRUE,
  trun = 10, cex.pts = 0.5,
  main = bquote("(D) FI ="~.(rslt1.all$FI1.Fisher.sig2nonsig)~
  "based on Fisher's exact test with"~alpha~"= 0.05 in arm 1"))
dev.off()
## Figure S1: artificial dataset 1; chi-squared test;
##  both arm modified; alpha = 0.05
pdf(file = "FigS1.pdf", width = 5, height = 5)
pvals.plot(rslt1.all, method = "chisq",
  modify0 = TRUE, modify1 = TRUE,
  trun = 10, cex.pts = 0.5,
  main = bquote("FI ="~.(rslt1.all$FI.chisq.sig2nonsig)~
  "based on chi\255squared test with"~alpha~"= 0.05"),
  legend.pvals = c(0.01, 0.001), cex.legend.pvals = 0.6,
  mar = c(3, 3, 2, 1) + 0.1)
dev.off()
## Figure S2A: artificial dataset 1; log odds ratio;
##  both arm modified; alpha = 0.05
pdf(file = "FigS2A.pdf", width = 5, height = 5)
pvals.plot(rslt1.all, method = "logOR",
  modify0 = TRUE, modify1 = TRUE,
  trun = 10, cex.pts = 0.5,
  main = bquote("(A) FI ="~.(rslt1.all$FI.logOR.sig2nonsig)~
  "based on (log) odds ratio with"~alpha~"= 0.05"),
  legend.pvals = c(0.01, 0.001), cex.legend.pvals = 0.6,
  mar = c(3, 3, 2, 1) + 0.1)
dev.off()
## Figure S2B: artificial dataset 1; log relative risk;
##  both arm modified; alpha = 0.05
pdf(file = "FigS2B.pdf", width = 5, height = 5)
pvals.plot(rslt1.all, method = "logRR",
  modify0 = TRUE, modify1 = TRUE,
  trun = 10, cex.pts = 0.5,
  main = bquote("(B) FI ="~.(rslt1.all$FI.logRR.sig2nonsig)~
  "based on (log) relative risk with"~alpha~"= 0.05"),
  legend.pvals = c(0.01, 0.001), cex.legend.pvals = 0.6,
  mar = c(3, 3, 2, 1) + 0.1)
dev.off()
## Figure S2C: artificial dataset 1; risk difference;
##  both arm modified; alpha = 0.05
pdf(file = "FigS2C.pdf", width = 5, height = 5)
pvals.plot(rslt1.all, method = "RD",
  modify0 = TRUE, modify1 = TRUE,
  trun = 10, cex.pts = 0.5,
  main = bquote("(C) FI ="~.(rslt1.all$FI.RD.sig2nonsig)~
  "based on risk difference with"~alpha~"= 0.05"),
  legend.pvals = c(0.01, 0.001), cex.legend.pvals = 0.6,
  mar = c(3, 3, 2, 1) + 0.1)
dev.off()

## Example: iteratively deriving the fragility index,
##  rather than enumerating all cases of modifications
begin <- Sys.time()
rslt.iter <- fragility.iter(e0 = artdat1$e0, n0 = artdat1$n0,
  e1 = artdat1$e1, n1 = artdat1$n1, methods = all.methods,
  modify0 = "both", modify1 = "both", alpha = 0.05)
end <- Sys.time()
end - begin
#Time difference of 0.8989601 secs

## Artificial dataset 2
begin <- Sys.time()
rslt2.all <- fragility.all(e0 = artdat2$e0, n0 = artdat2$n0,
  e1 = artdat2$e1, n1 = artdat2$n1, methods = all.methods,
  modify0 = "both", modify1 = "both", alpha = 0.05, save = TRUE)
end <- Sys.time()
end - begin
#Time difference of 9.384199 secs
rslt2.all$pval.Fisher
#[1] 0.5266649
rslt2.all$mods.Fisher
#     arm0 arm1
#[1,]   -8    0
#[2,]   -7    1

## Figure 2
## Figure 2A: artificial dataset 2; Fisher's exact test;
##  both arm modified; alpha = 0.05
pdf(file = "Fig2A.pdf", width = 5, height = 5)
pvals.plot(rslt2.all, method = "Fisher",
  modify0 = TRUE, modify1 = TRUE,
  trun = 10, cex.pts = 0.5,
  main = bquote("(A) FI ="~.(rslt2.all$FI.Fisher.nonsig2sig)~
   "based on Fisher's exact test with"~alpha~"= 0.05"),
  legend.pvals = c(0.01, 0.001), cex.legend.pvals = 0.6,
  mar = c(3, 3, 2, 1) + 0.1)
dev.off()
## Figure 2B: artificial dataset 2; Fisher's exact test;
##  arm 0 modified; alpha = 0.05
pdf(file = "Fig2B.pdf", width = 5, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
pvals.plot(rslt2.all, method = "Fisher",
  modify0 = TRUE, modify1 = FALSE,
  trun = 10, cex.pts = 0.5,
  main = bquote("(B) FI ="~.(rslt2.all$FI0.Fisher.nonsig2sig)~
   "based on Fisher's exact test with"~alpha~"= 0.05 in arm 0"))
dev.off()
## Figure 2C: artificial dataset 2; Fisher's exact test;
##  arm 1 modified; alpha = 0.05
pdf(file = "Fig2C.pdf", width = 5, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
pvals.plot(rslt2.all, method = "Fisher",
  modify0 = FALSE, modify1 = TRUE,
  trun = 10, cex.pts = 0.5,
  main = bquote("(C) FI ="~.(rslt2.all$FI1.Fisher.nonsig2sig)~
   "based on Fisher's exact test with"~alpha~"= 0.05 in arm 1"))
dev.off()

## Artificial dataset 3
begin <- Sys.time()
rslt3.all <- fragility.all(e0 = artdat3$e0, n0 = artdat3$n0,
  e1 = artdat3$e1, n1 = artdat3$n1, methods = all.methods,
  modify0 = "both", modify1 = "both", alpha = 0.05, save = TRUE)
end <- Sys.time()
end - begin
#Time difference of 7.153811 secs
rslt3.all$pval.Fisher
#[1] 0.01714902

## Figure 3: fragility index against significance level
## Figure 3A: artificial dataset 1
pdf(file = "Fig3A.pdf", width = 5, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
begin <- Sys.time()
rslt1.diffsignif <- FIvsSignif(e0 = artdat1$e0, n0 = artdat1$n0,
  e1 = artdat1$e1, n1 = artdat1$n1, method = "Fisher",
  modify0 = "both", modify1 = "both",
  alpha.from = 0.005, alpha.to = 0.05, alpha.breaks = 100,
  cex.pts = 0.5, main = "(A) Artificial dataset 1 (average FI = 6.20)")
end <- Sys.time()
end - begin
#Time difference of 0.5804839 secs
rslt1.diffsignif$FI.avg
# 6.2
dev.off()
## Figure 3B: artificial dataset 2
pdf(file = "Fig3B.pdf", width = 5, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
begin <- Sys.time()
rslt2.diffsignif <- FIvsSignif(e0 = artdat2$e0, n0 = artdat2$n0,
  e1 = artdat2$e1, n1 = artdat2$n1, method = "Fisher",
  modify0 = "both", modify1 = "both",
  alpha.from = 0.005, alpha.to = 0.05, alpha.breaks = 100,
  cex.pts = 0.5, main = "(B) Artificial dataset 2 (average FI = 9.80)")
end <- Sys.time()
end - begin
#Time difference of 0.5860779 secs
rslt2.diffsignif$FI.avg
# 9.8
dev.off()
## Figure 3C: artificial dataset 3
pdf(file = "Fig3C.pdf", width = 5, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
begin <- Sys.time()
rslt3.diffsignif <- FIvsSignif(e0 = artdat3$e0, n0 = artdat3$n0,
  e1 = artdat3$e1, n1 = artdat3$n1, method = "Fisher",
  modify0 = "both", modify1 = "both",
  alpha.from = 0.005, alpha.to = 0.05, alpha.breaks = 100,
  cex.pts = 0.5, main = "(C) Artificial dataset 3 (average FI = 1.73)")
end <- Sys.time()
end - begin
#Time difference of 0.07081103 secs
rslt3.diffsignif$FI.avg
# 1.73
dev.off()
```

```
################################
#### Antidepressant data analysis
## Raw data by Cipriani et al. (2018, Lancet) available at:
## http://doi.org/10.17632/83rthbp8ys.2
setwd(wd)
################################

############################
#### Outcome: responder
############################

studyid.responder <-
  c(1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6,
  7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 12,
  13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 18,
  19, 19, 20, 20, 21, 21, 22, 22, 23, 23, 24, 24,
  25, 25, 26, 26, 27, 27, 28, 28, 29, 29, 30, 30,
  31, 31, 32, 32, 33, 33, 34, 34, 35, 35, 36, 36,
  37, 37, 38, 38, 39, 39, 40, 40, 41, 41, 42, 42,
  43, 43, 44, 44, 45, 45, 46, 46, 47, 47, 48, 48,
  49, 49, 50, 50, 51, 51, 52, 52, 53, 53, 54, 54,
  55, 55, 56, 56, 57, 57, 58, 58, 59, 59, 60, 60,
  61, 61, 62, 62, 63, 63, 64, 64, 65, 65, 66, 66,
  67, 67, 68, 68, 69, 69, 70, 70, 71, 71, 72, 72,
  73, 73, 74, 74, 75, 75, 76, 76, 77, 77, 78, 78,
  79, 79, 80, 80, 81, 81, 82, 82, 83, 83, 84, 84,
  85, 85, 86, 86, 87, 87, 88, 88, 89, 89, 90, 90,
  91, 91, 92, 92, 93, 93, 94, 94, 95, 95, 96, 96,
  97, 97, 98, 98, 99, 99, 100, 100, 101, 101, 102, 102,
  103, 103, 104, 104, 105, 105, 106, 106, 107, 107, 108, 108,
  109, 109, 110, 110, 111, 111, 112, 112, 113, 113, 114, 114,
  115, 115, 116, 116, 117, 117, 118, 118, 119, 119, 120, 120,
  121, 121, 122, 122, 123, 123, 124, 124, 125, 125, 126, 126,
  127, 127, 128, 128, 129, 129, 130, 130, 131, 131, 132, 132,
  133, 133, 134, 134, 135, 135, 136, 136, 137, 137, 138, 138,
  139, 139, 140, 140, 141, 141, 142, 142, 143, 143, 144, 144,
  145, 145, 146, 146, 147, 147, 148, 148, 149, 149, 150, 150,
  151, 151, 152, 152, 153, 153, 154, 154, 155, 155, 156, 156,
  157, 157, 158, 158, 159, 159, 160, 160, 161, 161, 162, 162,
  163, 163, 164, 164, 165, 165, 166, 166, 167, 167, 168, 168,
  169, 169, 170, 170, 171, 171, 172, 172, 173, 173, 174, 174,
  175, 175, 176, 176, 177, 177, 178, 178, 179, 179, 180, 180,
  181, 181, 182, 182, 183, 183, 184, 184, 185, 185, 186, 186,
  187, 187, 188, 188, 189, 189, 190, 190, 191, 191, 192, 192,
  193, 193, 194, 194, 195, 195, 196, 196, 197, 197, 198, 198,
  199, 199, 200, 200, 201, 201, 202, 202, 203, 203, 204, 204,
  205, 205, 206, 206, 207, 207, 208, 208, 209, 209, 210, 210,
  211, 211, 212, 212, 213, 213, 214, 214, 215, 215, 216, 216,
  217, 217, 218, 218, 219, 219, 220, 220, 221, 221, 222, 222,
  223, 223, 224, 224, 225, 225, 226, 226, 227, 227, 228, 228,
  229, 229, 230, 230, 231, 231, 232, 232, 233, 233, 234, 234,
  235, 235, 236, 236, 237, 237, 238, 238, 239, 239, 240, 240,
  241, 241, 242, 242, 243, 243, 244, 244, 245, 245, 246, 246,
  247, 247, 248, 248, 249, 249, 250, 250, 251, 251, 252, 252,
  253, 253, 254, 254, 255, 255, 256, 256, 257, 257, 258, 258,
  259, 259, 260, 260, 261, 261, 262, 262, 263, 263, 264, 264,
  265, 265, 266, 266, 267, 267, 268, 268, 269, 269, 270, 270,
  271, 271, 272, 272, 273, 273, 274, 274, 275, 275, 276, 276,
  277, 277, 278, 278, 279, 279, 280, 280, 281, 281, 282, 282,
  283, 283, 284, 284, 285, 285, 286, 286, 287, 287, 288, 288,
  289, 289, 290, 290, 291, 291, 292, 292, 293, 293, 294, 294,
  295, 295, 296, 296, 297, 297, 298, 298, 299, 299, 300, 300,
  301, 301, 302, 302, 303, 303, 304, 304, 305, 305, 306, 306,
  307, 307, 308, 308, 309, 309, 310, 310, 311, 311, 312, 312,
  313, 313, 314, 314, 315, 315, 316, 316, 317, 317, 318, 318,
  319, 319, 320, 320, 321, 321, 322, 322, 323, 323, 324, 324,
  325, 325, 326, 326, 327, 327, 328, 328, 329, 329, 330, 330,
  331, 331, 332, 332, 333, 333, 334, 334, 335, 335, 336, 336,
  337, 337, 338, 338, 339, 339, 340, 340, 341, 341, 342, 342,
  343, 343, 344, 344, 345, 345, 346, 346, 347, 347, 348, 348,
  349, 349, 350, 350, 351, 351, 352, 352, 353, 353, 354, 354,
  355, 355, 356, 356)
drug.responder <-
  c('agomelatine', 'placebo', 'agomelatine', 'placebo',
  'agomelatine', 'fluoxetine', 'agomelatine', 'placebo',
  'agomelatine', 'paroxetine', 'agomelatine', 'venlafaxine',
  'agomelatine', 'paroxetine', 'agomelatine', 'fluoxetine',
  'agomelatine', 'escitalopram', 'agomelatine', 'duloxetine',
  'agomelatine', 'escitalopram', 'agomelatine', 'placebo',
  'agomelatine', 'venlafaxine', 'agomelatine', 'placebo',
  'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
  'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
  'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
  'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
  'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
  'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
  'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
  'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
  'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
  'amitriptyline', 'placebo', 'bupropion', 'placebo',
  'bupropion', 'placebo', 'bupropion', 'placebo',
  'fluoxetine', 'bupropion', 'bupropion', 'trazodone',
  'placebo', 'bupropion', 'bupropion', 'placebo',
  'bupropion', 'sertraline', 'bupropion', 'placebo',
  'bupropion', 'venlafaxine', 'placebo', 'bupropion',
  'bupropion', 'placebo', 'venlafaxine', 'placebo',
  'bupropion', 'placebo', 'bupropion', 'placebo',
  'bupropion', 'placebo', 'bupropion', 'placebo',
  'bupropion', 'paroxetine', 'bupropion', 'placebo',
  'bupropion', 'placebo', 'bupropion', 'sertraline',
  'citalopram', 'clomipramine', 'citalopram', 'amitriptyline',
  'fluvoxamine', 'citalopram', 'fluoxetine', 'citalopram',
  'fluoxetine', 'citalopram', 'citalopram', 'amitriptyline',
  'citalopram', 'placebo', 'citalopram', 'placebo',
  'reboxetine', 'citalopram', 'citalopram', 'sertraline',
  'citalopram', 'placebo', 'placebo', 'venlafaxine',
  'placebo', 'venlafaxine', 'desvenlafaxine', 'placebo',
  'desvenlafaxine', 'placebo', 'desvenlafaxine', 'placebo',
  'desvenlafaxine', 'placebo', 'placebo', 'desvenlafaxine',
  'duloxetine', 'placebo', 'duloxetine', 'placebo',
  'duloxetine', 'venlafaxine', 'duloxetine', 'placebo',
  'duloxetine', 'paroxetine', 'duloxetine', 'placebo',
  'duloxetine', 'placebo', 'duloxetine', 'placebo',
  'duloxetine', 'paroxetine', 'escitalopram', 'fluoxetine',
  'escitalopram', 'placebo', 'escitalopram', 'venlafaxine',
  'escitalopram', 'venlafaxine', 'placebo', 'escitalopram',
  'escitalopram', 'sertraline', 'escitalopram', 'placebo',
  'escitalopram', 'citalopram', 'escitalopram', 'citalopram',
  'paroxetine', 'escitalopram', 'escitalopram', 'duloxetine',
  'escitalopram', 'duloxetine', 'escitalopram', 'paroxetine',
  'escitalopram', 'citalopram', 'escitalopram', 'citalopram',
  'fluoxetine', 'escitalopram', 'escitalopram', 'citalopram',
  'escitalopram', 'citalopram', 'citalopram', 'escitalopram',
  'escitalopram', 'citalopram', 'escitalopram', 'placebo',
  'escitalopram', 'citalopram', 'escitalopram', 'placebo',
  'escitalopram', 'placebo', 'escitalopram', 'placebo',
  'escitalopram', 'placebo', 'escitalopram', 'placebo',
  'fluoxetine', 'placebo', 'fluoxetine', 'placebo',
  'fluoxetine', 'amitriptyline', 'fluoxetine', 'placebo',
  'fluoxetine', 'amitriptyline', 'fluoxetine', 'placebo',
  'placebo', 'fluoxetine', 'fluoxetine', 'amitriptyline',
  'fluoxetine', 'trazodone', 'fluoxetine', 'amitriptyline',
  'placebo', 'fluoxetine', 'trazodone', 'fluoxetine',
  'amitriptyline', 'fluoxetine', 'fluoxetine', 'clomipramine',
  'fluoxetine', 'clomipramine', 'fluoxetine', 'clomipramine',
  'fluoxetine', 'trazodone', 'fluoxetine', 'amitriptyline',
  'clomipramine', 'fluoxetine', 'fluoxetine', 'amitriptyline',
  'fluoxetine', 'amitriptyline', 'trazodone', 'fluoxetine',
  'fluoxetine', 'amitriptyline', 'fluoxetine', 'placebo',
  'fluoxetine', 'placebo', 'fluoxetine', 'placebo',
  'fluoxetine', 'placebo', 'amitriptyline', 'fluoxetine',
  'fluoxetine', 'amitriptyline', 'fluoxetine', 'amitriptyline',
  'fluoxetine', 'amitriptyline', 'fluoxetine', 'placebo',
  'fluoxetine', 'placebo', 'fluoxetine', 'placebo',
  'placebo', 'fluoxetine', 'fluoxetine', 'placebo',
  'fluoxetine', 'placebo', 'fluvoxamine', 'clomipramine',
  'fluvoxamine', 'placebo', 'fluvoxamine', 'placebo',
  'fluvoxamine', 'placebo', 'fluvoxamine', 'placebo',
  'fluvoxamine', 'placebo', 'fluvoxamine', 'placebo',
  'fluvoxamine', 'placebo', 'fluvoxamine', 'placebo',
  'amitriptyline', 'fluvoxamine', 'fluvoxamine', 'sertraline',
  'fluvoxamine', 'clomipramine', 'fluvoxamine', 'fluoxetine',
  'fluvoxamine', 'placebo', 'fluvoxamine', 'placebo',
  'fluvoxamine', 'paroxetine', 'fluvoxamine', 'amitriptyline',
  'fluvoxamine', 'fluoxetine', 'fluvoxamine', 'sertraline',
  'fluvoxamine', 'amitriptyline', 'fluvoxamine', 'placebo',
  'placebo', 'levomilnacipran', 'placebo', 'levomilnacipran',
  'placebo', 'levomilnacipran', 'milnacipran', 'amitriptyline',
  'milnacipran', 'fluoxetine', 'milnacipran', 'clomipramine',
  'fluvoxamine', 'milnacipran', 'milnacipran', 'sertraline',
  'milnacipran', 'paroxetine', 'mirtazapine', 'placebo',
  'mirtazapine', 'placebo', 'mirtazapine', 'placebo',
  'mirtazapine', 'trazodone', 'mirtazapine', 'amitriptyline',
  'mirtazapine', 'amitriptyline', 'mirtazapine', 'amitriptyline',
  'mirtazapine', 'fluoxetine', 'mirtazapine', 'citalopram',
  'mirtazapine', 'paroxetine', 'mirtazapine', 'paroxetine',
  'mirtazapine', 'fluoxetine', 'mirtazapine', 'sertraline',
  'mirtazapine', 'paroxetine', 'mirtazapine', 'fluoxetine',
  'mirtazapine', 'fluoxetine', 'mirtazapine', 'fluoxetine',
  'mirtazapine', 'venlafaxine', 'mirtazapine', 'fluvoxamine',
  'mirtazapine', 'fluvoxamine', 'mirtazapine', 'venlafaxine',
  'mirtazapine', 'paroxetine', 'mirtazapine', 'paroxetine',
  'nefazodone', 'amitriptyline', 'nefazodone', 'fluoxetine',
  'nefazodone', 'paroxetine', 'nefazodone', 'fluoxetine',
  'nefazodone', 'sertraline', 'nefazodone', 'placebo',
  'nefazodone', 'fluoxetine', 'nefazodone', 'paroxetine',
  'nefazodone', 'placebo', 'nefazodone', 'placebo',
  'nefazodone', 'placebo', 'nefazodone', 'placebo',
  'paroxetine', 'amitriptyline', 'paroxetine', 'amitriptyline',
  'paroxetine', 'placebo', 'paroxetine', 'amitriptyline',
  'paroxetine', 'amitriptyline', 'paroxetine', 'placebo',
  'paroxetine', 'placebo', 'paroxetine', 'placebo',
  'paroxetine', 'placebo', 'paroxetine', 'placebo',
  'paroxetine', 'placebo', 'paroxetine', 'placebo',
  'paroxetine', 'placebo', 'paroxetine', 'placebo',
  'paroxetine', 'placebo', 'paroxetine', 'amitriptyline',
  'paroxetine', 'placebo', 'clomipramine', 'paroxetine',
  'clomipramine', 'paroxetine', 'paroxetine', 'placebo',
  'paroxetine', 'clomipramine', 'paroxetine', 'clomipramine',
  'paroxetine', 'amitriptyline', 'paroxetine', 'clomipramine',
  'paroxetine', 'fluoxetine', 'paroxetine', 'fluoxetine',
  'paroxetine', 'fluoxetine', 'paroxetine', 'amitriptyline',
  'amitriptyline', 'paroxetine', 'paroxetine', 'amitriptyline',
  'paroxetine', 'fluvoxamine', 'paroxetine', 'fluoxetine',
  'paroxetine', 'fluoxetine', 'paroxetine', 'placebo',
  'paroxetine', 'placebo', 'paroxetine', 'fluoxetine',
  'paroxetine', 'amitriptyline', 'paroxetine', 'amitriptyline',
  'paroxetine', 'amitriptyline', 'paroxetine', 'amitriptyline',
  'paroxetine', 'clomipramine', 'paroxetine', 'placebo',
  'paroxetine', 'fluoxetine', 'paroxetine', 'trazodone',
  'paroxetine', 'amitriptyline', 'placebo', 'paroxetine',
  'paroxetine', 'fluoxetine', 'paroxetine', 'amitriptyline',
  'paroxetine', 'placebo', 'paroxetine', 'placebo',
  'paroxetine', 'placebo', 'paroxetine', 'placebo',
  'bupropion', 'paroxetine', 'paroxetine', 'placebo',
  'paroxetine', 'placebo', 'paroxetine', 'placebo',
  'reboxetine', 'placebo', 'reboxetine', 'placebo',
  'reboxetine', 'placebo', 'reboxetine', 'fluoxetine',
  'reboxetine', 'placebo', 'reboxetine', 'placebo',
  'reboxetine', 'paroxetine', 'reboxetine', 'citalopram',
  'reboxetine', 'venlafaxine', 'reboxetine', 'placebo',
  'reboxetine', 'fluoxetine', 'sertraline', 'amitriptyline',
  'sertraline', 'fluoxetine', 'sertraline', 'amitriptyline',
  'sertraline', 'clomipramine', 'sertraline', 'placebo',
  'sertraline', 'fluoxetine', 'sertraline', 'fluoxetine',
  'citalopram', 'sertraline', 'sertraline', 'placebo',
  'amitriptyline', 'sertraline', 'sertraline', 'fluoxetine',
  'sertraline', 'fluoxetine', 'paroxetine', 'sertraline',
  'sertraline', 'amitriptyline', 'sertraline', 'clomipramine',
  'venlafaxine', 'sertraline', 'venlafaxine', 'sertraline',
  'sertraline', 'placebo', 'sertraline', 'placebo',
  'sertraline', 'placebo', 'sertraline', 'placebo',
  'sertraline', 'placebo', 'sertraline', 'placebo',
  'sertraline', 'placebo', 'sertraline', 'placebo',
  'sertraline', 'placebo', 'bupropion', 'sertraline',
  'sertraline', 'placebo', 'trazodone', 'placebo',
  'trazodone', 'placebo', 'trazodone', 'placebo',
  'trazodone', 'placebo', 'amitriptyline', 'trazodone',
  'trazodone', 'placebo', 'amitriptyline', 'trazodone',
  'amitriptyline', 'trazodone', 'amitriptyline', 'trazodone',
  'paroxetine', 'trazodone', 'sertraline', 'trazodone',
  'trazodone', 'placebo', 'trazodone', 'placebo',
  'venlafaxine', 'clomipramine', 'paroxetine', 'venlafaxine',
  'venlafaxine', 'placebo', 'venlafaxine', 'placebo',
  'venlafaxine', 'fluoxetine', 'venlafaxine', 'fluoxetine',
  'venlafaxine', 'fluoxetine', 'venlafaxine', 'sertraline',
  'fluoxetine', 'venlafaxine', 'venlafaxine', 'placebo',
  'venlafaxine', 'placebo', 'fluoxetine', 'venlafaxine',
  'venlafaxine', 'placebo', 'venlafaxine', 'placebo',
  'fluoxetine', 'venlafaxine', 'fluvoxamine', 'venlafaxine',
  'fluoxetine', 'venlafaxine', 'paroxetine', 'venlafaxine',
  'venlafaxine', 'clomipramine', 'fluoxetine', 'venlafaxine',
  'sertraline', 'venlafaxine', 'fluoxetine', 'venlafaxine',
  'venlafaxine', 'amitriptyline', 'paroxetine', 'venlafaxine',
  'venlafaxine', 'amitriptyline', 'citalopram', 'venlafaxine',
  'fluoxetine', 'venlafaxine', 'venlafaxine', 'citalopram',
  'venlafaxine', 'Placebo', 'venlafaxine', 'fluoxetine',
  'fluoxetine', 'venlafaxine', 'vilazodone', 'placebo',
  'vilazodone', 'placebo', 'fluoxetine', 'placebo',
  'vilazodone', 'placebo', 'vilazodone', 'placebo',
  'placebo', 'vortioxetine', 'vortioxetine', 'venlafaxine')
samplesize.responder <-
```

```
  c(107, 105, 118, 120, 252, 263, 109, 109, 167, 168, 137, 140,
  213, 199, 314, 314, 71, 67, 202, 216, 164, 160, 151, 71,
  165, 167, 151, 71, 93, 50, 15, 18, 12, 7, 53, 23,
  95, 94, 85, 87, 20, 20, 124, 130, 55, 54, 16, 15,
  45, 55, 19, 22, 50, 50, 58, 56, 95, 94, 93, 104,
  13, 15, 50, 49, 31, 30, 13, 7, 45, 22, 112, 112,
  62, 61, 63, 61, 121, 120, 150, 154, 7, 9, 10, 9,
  171, 177, 139, 135, 212, 208, 198, 189, 56, 29, 77, 40,
  56, 52, 25, 24, 69, 71, 166, 159, 192, 187, 122, 126,
  57, 57, 24, 20, 109, 108, 184, 173, 158, 158, 179, 186,
  89, 91, 87, 91, 46, 55, 25, 26, 127, 130, 120, 127,
  125, 117, 291, 146, 218, 221, 237, 235, 227, 227, 120, 118,
  123, 122, 128, 139, 166, 166, 207, 104, 238, 240, 151, 157,
  257, 127, 261, 131, 140, 141, 102, 103, 154, 155, 101, 101,
  148, 145, 189, 191, 107, 108, 132, 135, 175, 182, 142, 152,
  159, 166, 144, 151, 140, 138, 232, 227, 28, 28, 109, 110,
  117, 123, 25, 23, 32, 32, 120, 120, 24, 24, 62, 138,
  20, 22, 140, 71, 157, 157, 84, 76, 50, 99, 69, 65,
  22, 25, 18, 24, 25, 28, 54, 58, 21, 21, 185, 169,
  29, 32, 32, 33, 22, 21, 63, 65, 59, 62, 13, 14,
  20, 20, 15, 15, 28, 26, 71, 72, 21, 17, 16, 18,
  60, 60, 20, 22, 30, 31, 61, 65, 30, 28, 46, 43,
  36, 32, 335, 336, 72, 72, 31, 35, 67, 75, 21, 21,
  77, 80, 35, 35, 49, 52, 10, 10, 43, 47, 57, 58,
  20, 26, 22, 21, 161, 150, 35, 25, 35, 31, 161, 149,
  22, 20, 31, 19, 18, 18, 18, 18, 17, 16, 49, 48,
  20, 20, 51, 49, 50, 50, 50, 50, 30, 30, 113, 122,
  90, 94, 40, 48, 35, 34, 30, 30, 281, 282, 182, 175,
  220, 222, 44, 43, 97, 93, 52, 55, 56, 57, 27, 26,
  149, 153, 45, 45, 60, 57, 45, 45, 100, 100, 56, 59,
  79, 77, 103, 104, 66, 67, 137, 133, 139, 136, 128, 126,
  66, 66, 176, 170, 99, 98, 8, 12, 18, 18, 145, 149,
  130, 128, 104, 105, 101, 102, 140, 135, 73, 75, 21, 21,
  55, 51, 22, 21, 105, 101, 37, 37, 78, 82, 61, 59,
  24, 20, 20, 20, 96, 95, 45, 44, 43, 43, 60, 60,
  9, 12, 10, 13, 25, 25, 82, 80, 29, 30, 19, 10,
  55, 56, 36, 36, 39, 38, 40, 40, 40, 40, 41, 42,
  40, 40, 40, 42, 40, 40, 12, 12, 40, 41, 39, 40,
  42, 41, 22, 25, 32, 30, 45, 47, 44, 47, 62, 58,
  89, 87, 54, 52, 60, 62, 156, 153, 32, 58, 21, 19,
  55, 65, 49, 50, 45, 45, 125, 129, 21, 21, 68, 70,
  113, 118, 109, 108, 78, 75, 71, 73, 513, 506, 72, 70,
  102, 101, 116, 109, 109, 119, 149, 154, 123, 119, 64, 65,
  125, 125, 177, 180, 28, 25, 100, 53, 40, 38, 166, 156,
  164, 161, 80, 79, 28, 28, 24, 26, 112, 112, 79, 89,
  84, 85, 107, 105, 159, 166, 183, 176, 80, 87, 26, 24,
  43, 42, 161, 80, 52, 56, 34, 34, 51, 55, 99, 101,
  142, 144, 83, 82, 200, 200, 7, 10, 79, 81, 118, 120,
  117, 119, 177, 176, 116, 124, 82, 84, 84, 79, 78, 82,
  371, 376, 93, 94, 55, 50, 43, 39, 30, 30, 19, 18,
  80, 85, 111, 116, 129, 129, 122, 126, 104, 103, 9, 9,
  13, 11, 19, 10, 19, 20, 22, 21, 10, 10, 38, 36,
  44, 112, 37, 36, 53, 55, 60, 62, 206, 206, 192, 190,
  60, 61, 81, 165, 176, 183, 83, 82, 27, 24, 156, 152,
  79, 77, 295, 293, 34, 34, 76, 78, 46, 47, 161, 153,
  78, 76, 95, 102, 170, 171, 34, 77, 186, 196, 178, 183,
  52, 50, 47, 40, 72, 75, 54, 55, 57, 59, 56, 58,
  79, 77, 75, 76, 275, 821, 28, 25, 134, 126, 15, 14,
  58, 54, 205, 205, 240, 241, 89, 95, 109, 111, 260, 258,
  300, 300, 213, 230)
responder <-
  c(52, 36, 63, 42, 177, 164, 51, 50, 106, 120, 113, 111, 127, 118,
  205, 209, 44, 36, 131, 153, 134, 128, 88, 27, 126, 115, 88, 27,
  41, 16, 11, 7, 2, 0, 43, 6, 39, 25, 49, 36, 13, 6,
  85, 44, 31, 15, 11, 5, 28, 23, 6, 6, 23, 11, 34, 20,
  51, 30, 55, 36, 10, 6, 22, 10, 12, 5, 4, 2, 27, 2,
  56, 36, 35, 37, 33, 21, 41, 52, 74, 59, 3, 4, 4, 2,
  86, 82, 52, 70, 109, 87, 127, 91, 22, 7, 27, 6, 26, 17,
  8, 7, 35, 36, 42, 36, 82, 86, 82, 85, 23, 31, 15, 13,
  31, 33, 123, 120, 109, 102, 98, 96, 27, 19, 34, 34, 28, 33,
  9, 6, 71, 51, 60, 81, 55, 66, 174, 65, 89, 72, 109, 81,
  92, 85, 34, 50, 55, 27, 62, 48, 85, 91, 75, 19, 144, 157,
  70, 55, 113, 45, 117, 44, 86, 74, 61, 63, 61, 48, 59, 47,
  113, 113, 79, 104, 75, 74, 50, 50, 104, 96, 105, 87, 111, 112,
  94, 81, 61, 52, 144, 128, 22, 19, 103, 90, 89, 94, 21, 19,
  30, 31, 87, 83, 20, 19, 29, 59, 16, 17, 97, 33, 91, 78,
  30, 22, 14, 31, 22, 18, 6, 3, 5, 7, 8, 15, 38, 12,
  12, 11, 73, 44, 4, 14, 16, 18, 9, 9, 55, 57, 9, 15,
  4, 8, 7, 8, 12, 11, 22, 18, 42, 31, 10, 13, 6, 6,
  32, 37, 12, 16, 12, 8, 34, 33, 15, 12, 25, 17, 23, 13,
  125, 91, 14, 10, 18, 22, 45, 59, 7, 6, 57, 58, 17, 9,
  25, 12, 6, 3, 15, 15, 20, 23, 11, 11, 16, 17, 77, 44,
  13, 10, 8, 8, 84, 53, 14, 5, 13, 4, 9, 5, 7, 3,
  5, 9, 27, 23, 13, 16, 30, 29, 21, 12, 23, 15, 17, 16,
  51, 54, 53, 54, 29, 28, 13, 21, 17, 11, 107, 155, 63, 67,
  63, 90, 28, 22, 32, 43, 32, 32, 32, 40, 4, 2, 86, 91,
  18, 14, 19, 15, 21, 11, 61, 51, 30, 41, 38, 41, 53, 62,
  39, 28, 116, 117, 74, 66, 72, 60, 35, 30, 117, 114, 35, 40,
  8, 6, 12, 8, 106, 104, 72, 62, 61, 70, 71, 57, 70, 56,
  40, 48, 4, 9, 21, 34, 11, 11, 58, 58, 17, 18, 42, 41,
  29, 17, 12, 8, 11, 16, 42, 29, 26, 23, 25, 15, 26, 19,
  4, 3, 2, 5, 11, 8, 30, 43, 15, 14, 12, 5, 27, 16,
  15, 8, 13, 10, 20, 5, 9, 6, 17, 14, 16, 9, 16, 12,
  12, 3, 7, 4, 15, 7, 28, 26, 25, 29, 3, 5, 22, 12,
  27, 23, 21, 24, 23, 31, 53, 54, 19, 8, 41, 35, 82, 91,
  15, 34, 7, 9, 24, 24, 25, 26, 30, 27, 56, 44, 7, 3,
  33, 32, 94, 80, 53, 51, 43, 46, 46, 48, 328, 317, 33, 20,
  67, 67, 65, 45, 42, 54, 72, 91, 50, 35, 22, 18, 57, 59,
  100, 71, 11, 6, 25, 12, 16, 15, 136, 97, 105, 76, 54, 40,
  20, 5, 4, 5, 65, 58, 59, 64, 50, 30, 42, 35, 80, 100,
  77, 98, 45, 48, 14, 9, 21, 23, 84, 40, 35, 26, 18, 17,
  28, 29, 50, 40, 73, 63, 49, 48, 136, 139, 4, 4, 40, 32,
  48, 35, 85, 84, 112, 111, 51, 71, 53, 54, 56, 56, 49, 45,
  126, 96, 52, 47, 34, 35, 25, 17, 10, 5, 10, 7, 33, 20,
  53, 50, 70, 45, 74, 73, 40, 37, 5, 0, 5, 1, 9, 0,
  2, 7, 12, 6, 7, 1, 18, 15, 29, 58, 21, 21, 48, 48,
  37, 46, 109, 84, 109, 67, 32, 41, 44, 98, 75, 83, 29, 31,
  14, 15, 86, 74, 38, 36, 174, 167, 17, 24, 49, 37, 24, 13,
  95, 107, 50, 38, 53, 30, 58, 67, 14, 48, 153, 170, 117, 130,
  30, 20, 40, 34, 41, 49, 33, 38, 43, 44, 29, 43, 30, 35,
  50, 49, 210, 612, 12, 21, 85, 67, 9, 8, 41, 38, 88, 65,
  102, 76, 30, 35, 45, 41, 146, 91, 132, 135, 139, 132)
ad.responder <-
  data.frame(studyid = studyid.responder, drug = drug.responder,
  samplesize = samplesize.responder, responder = responder)

ss.ratio.responder <-
  ad.responder$samplesize[2*(1:max(ad.responder$studyid))]/
  ad.responder$samplesize[2*(1:max(ad.responder$studyid)) - 1]
ss.ratio.responder[ss.ratio.responder < 1] <-
  1/ss.ratio.responder[ss.ratio.responder < 1]
pdf(file = "FigS3.pdf", width = 5, height = 5)
par(mar = c(3, 3, 0, 0) + 0.1, mgp = c(2, 1, 0))
hist(ss.ratio.responder, xlab = "Sample size ratio", main = "")
dev.off()
sum(ss.ratio.responder < 1.1)
```

```
#[1] 288
mean(ss.ratio.responder < 1.1)
#[1] 0.8089888

## data analysis
n.study <- max(ad.responder$studyid)
n.study
#[1] 356
quantile(ad.responder$samplesize, probs = c(0, 0.25, 0.5, 0.75, 1),
  type = 3)
#  0%  25%  50%  75% 100%
#   7   35   71  126  821
quantile(ad.responder$responder, probs = c(0, 0.25, 0.5, 0.75, 1),
  type = 3)
#  0%  25%  50%  75% 100%
#   0   15   34   63  612
totsamplesize <- ad.responder$samplesize[2*(1:356)] +
  ad.responder$samplesize[2*(1:356) - 1]
quantile(totsamplesize, probs = c(0, 0.25, 0.5, 0.75, 1), type = 3)
#  0%  25%  50%  75% 100%
#  16   72  143  248 1096

ad.pval.Fisher <- ad.pval.chisq <- ad.pval.logOR <-
  ad.pval.logRR <- ad.pval.RD <- rep(NA, n.study)
ad.FI.Fisher.sig2nonsig <- ad.FI.Fisher.nonsig2sig <-
  ad.FI.chisq.sig2nonsig <- ad.FI.chisq.nonsig2sig <-
  ad.FI.logOR.sig2nonsig <- ad.FI.logOR.nonsig2sig <-
  ad.FI.logRR.sig2nonsig <- ad.FI.logRR.nonsig2sig <-
  ad.FI.RD.sig2nonsig <- ad.FI.RD.nonsig2sig <- rep(NA, n.study)
ad.FI0.Fisher.sig2nonsig <- ad.FI0.Fisher.nonsig2sig <-
  ad.FI0.chisq.sig2nonsig <- ad.FI0.chisq.nonsig2sig <-
  ad.FI0.logOR.sig2nonsig <- ad.FI0.logOR.nonsig2sig <-
  ad.FI0.logRR.sig2nonsig <- ad.FI0.logRR.nonsig2sig <-
  ad.FI0.RD.sig2nonsig <- ad.FI0.RD.nonsig2sig <- rep(NA, n.study)
ad.FI1.Fisher.sig2nonsig <- ad.FI1.Fisher.nonsig2sig <-
  ad.FI1.chisq.sig2nonsig <- ad.FI1.chisq.nonsig2sig <-
  ad.FI1.logOR.sig2nonsig <- ad.FI1.logOR.nonsig2sig <-
  ad.FI1.logRR.sig2nonsig <- ad.FI1.logRR.nonsig2sig <-
  ad.FI1.RD.sig2nonsig <- ad.FI1.RD.nonsig2sig <- rep(NA, n.study)

all.methods <- c("Fisher", "chisq", "logOR", "logRR", "RD")
begin <- Sys.time()
for(i in 1:n.study){
  dat.temp <- ad.responder[ad.responder$studyid == i,]
  e0.temp <- dat.temp$responder[1]
  e1.temp <- dat.temp$responder[2]
  n0.temp <- dat.temp$samplesize[1]
  n1.temp <- dat.temp$samplesize[2]

  rslt.temp <- fragility.iter(e0 = e0.temp, n0 = n0.temp,
    e1 = e1.temp, n1 = n1.temp, methods = all.methods,
    modify0 = "both", modify1 = "both", alpha = 0.05)
  ad.pval.Fisher[i] <- rslt.temp$pval.Fisher
  ad.pval.chisq[i] <- rslt.temp$pval.chisq
  ad.pval.logOR[i] <- rslt.temp$pval.logOR
  ad.pval.logRR[i] <- rslt.temp$pval.logRR
  ad.pval.RD[i] <- rslt.temp$pval.RD
  ad.FI.Fisher.sig2nonsig[i] <- rslt.temp$FI.Fisher.sig2nonsig
  ad.FI.Fisher.nonsig2sig[i] <- rslt.temp$FI.Fisher.nonsig2sig
  ad.FI.chisq.sig2nonsig[i] <- rslt.temp$FI.chisq.sig2nonsig
  ad.FI.chisq.nonsig2sig[i] <- rslt.temp$FI.chisq.nonsig2sig
  ad.FI.logOR.sig2nonsig[i] <- rslt.temp$FI.logOR.sig2nonsig
  ad.FI.logOR.nonsig2sig[i] <- rslt.temp$FI.logOR.nonsig2sig
  ad.FI.logRR.sig2nonsig[i] <- rslt.temp$FI.logRR.sig2nonsig
  ad.FI.logRR.nonsig2sig[i] <- rslt.temp$FI.logRR.nonsig2sig
  ad.FI.RD.sig2nonsig[i] <- rslt.temp$FI.RD.sig2nonsig
  ad.FI.RD.nonsig2sig[i] <- rslt.temp$FI.RD.nonsig2sig

  rslt0.temp <- fragility.iter(e0 = e0.temp, n0 = n0.temp,
    e1 = e1.temp, n1 = n1.temp, methods = all.methods,
    modify0 = "both", modify1 = "none", alpha = 0.05)
  ad.FI0.Fisher.sig2nonsig[i] <- rslt0.temp$FI.Fisher.sig2nonsig
  ad.FI0.Fisher.nonsig2sig[i] <- rslt0.temp$FI.Fisher.nonsig2sig
  ad.FI0.chisq.sig2nonsig[i] <- rslt0.temp$FI.chisq.sig2nonsig
  ad.FI0.chisq.nonsig2sig[i] <- rslt0.temp$FI.chisq.nonsig2sig
  ad.FI0.logOR.sig2nonsig[i] <- rslt0.temp$FI.logOR.sig2nonsig
  ad.FI0.logOR.nonsig2sig[i] <- rslt0.temp$FI.logOR.nonsig2sig
  ad.FI0.logRR.sig2nonsig[i] <- rslt0.temp$FI.logRR.sig2nonsig
  ad.FI0.logRR.nonsig2sig[i] <- rslt0.temp$FI.logRR.nonsig2sig
  ad.FI0.RD.sig2nonsig[i] <- rslt0.temp$FI.RD.sig2nonsig
  ad.FI0.RD.nonsig2sig[i] <- rslt0.temp$FI.RD.nonsig2sig

  rslt1.temp <- fragility.iter(e0 = e0.temp, n0 = n0.temp,
    e1 = e1.temp, n1 = n1.temp, methods = all.methods,
    modify0 = "none", modify1 = "both", alpha = 0.05)
  ad.FI1.Fisher.sig2nonsig[i] <- rslt1.temp$FI.Fisher.sig2nonsig
  ad.FI1.Fisher.nonsig2sig[i] <- rslt1.temp$FI.Fisher.nonsig2sig
  ad.FI1.chisq.sig2nonsig[i] <- rslt1.temp$FI.chisq.sig2nonsig
  ad.FI1.chisq.nonsig2sig[i] <- rslt1.temp$FI.chisq.nonsig2sig
  ad.FI1.logOR.sig2nonsig[i] <- rslt1.temp$FI.logOR.sig2nonsig
  ad.FI1.logOR.nonsig2sig[i] <- rslt1.temp$FI.logOR.nonsig2sig
  ad.FI1.logRR.sig2nonsig[i] <- rslt1.temp$FI.logRR.sig2nonsig
  ad.FI1.logRR.nonsig2sig[i] <- rslt1.temp$FI.logRR.nonsig2sig
  ad.FI1.RD.sig2nonsig[i] <- rslt1.temp$FI.RD.sig2nonsig
  ad.FI1.RD.nonsig2sig[i] <- rslt1.temp$FI.RD.nonsig2sig
}
end <- Sys.time()
end - begin
#Time difference of 17.25355 mins

## Summary
ad.FI.Fisher <- ad.FI0.Fisher <- ad.FI1.Fisher <-
  ad.FI.chisq <- ad.FI0.chisq <- ad.FI1.chisq <-
  ad.FI.logOR <- ad.FI0.logOR <- ad.FI1.logOR <-
  ad.FI.logRR <- ad.FI0.logRR <- ad.FI1.logRR <-
  ad.FI.RD <- ad.FI0.RD <- ad.FI1.RD <- rep(NA, n.study)
for(i in 1:n.study){
  ad.FI.Fisher[i] <- ifelse(ad.pval.Fisher[i] < 0.05,
    ad.FI.Fisher.sig2nonsig[i], ad.FI.Fisher.nonsig2sig[i])
  ad.FI0.Fisher[i] <- ifelse(ad.pval.Fisher[i] < 0.05,
    ad.FI0.Fisher.sig2nonsig[i], ad.FI0.Fisher.nonsig2sig[i])
  ad.FI1.Fisher[i] <- ifelse(ad.pval.Fisher[i] < 0.05,
    ad.FI1.Fisher.sig2nonsig[i], ad.FI1.Fisher.nonsig2sig[i])
  ad.FI.chisq[i] <- ifelse(ad.pval.chisq[i] < 0.05,
    ad.FI.chisq.sig2nonsig[i], ad.FI.chisq.nonsig2sig[i])
  ad.FI0.chisq[i] <- ifelse(ad.pval.chisq[i] < 0.05,
    ad.FI0.chisq.sig2nonsig[i], ad.FI0.chisq.nonsig2sig[i])
  ad.FI1.chisq[i] <- ifelse(ad.pval.chisq[i] < 0.05,
    ad.FI1.chisq.sig2nonsig[i], ad.FI1.chisq.nonsig2sig[i])
  ad.FI.logOR[i] <- ifelse(ad.pval.logOR[i] < 0.05,
    ad.FI.logOR.sig2nonsig[i], ad.FI.logOR.nonsig2sig[i])
  ad.FI0.logOR[i] <- ifelse(ad.pval.logOR[i] < 0.05,
    ad.FI0.logOR.sig2nonsig[i], ad.FI0.logOR.nonsig2sig[i])
  ad.FI1.logOR[i] <- ifelse(ad.pval.logOR[i] < 0.05,
    ad.FI1.logOR.sig2nonsig[i], ad.FI1.logOR.nonsig2sig[i])
  ad.FI.logRR[i] <- ifelse(ad.pval.logRR[i] < 0.05,
    ad.FI.logRR.sig2nonsig[i], ad.FI.logRR.nonsig2sig[i])
  ad.FI0.logRR[i] <- ifelse(ad.pval.logRR[i] < 0.05,
    ad.FI0.logRR.sig2nonsig[i], ad.FI0.logRR.nonsig2sig[i])
  ad.FI1.logRR[i] <- ifelse(ad.pval.logRR[i] < 0.05,
    ad.FI1.logRR.sig2nonsig[i], ad.FI1.logRR.nonsig2sig[i])
  ad.FI.RD[i] <- ifelse(ad.pval.RD[i] < 0.05,
    ad.FI.RD.sig2nonsig[i], ad.FI.RD.nonsig2sig[i])
  ad.FI0.RD[i] <- ifelse(ad.pval.RD[i] < 0.05,
    ad.FI0.RD.sig2nonsig[i], ad.FI0.RD.nonsig2sig[i])
  ad.FI1.RD[i] <- ifelse(ad.pval.RD[i] < 0.05,
    ad.FI1.RD.sig2nonsig[i], ad.FI1.RD.nonsig2sig[i])
}

ad.FI.all <- c(ad.FI.Fisher, ad.FI.chisq, ad.FI.logOR,
  ad.FI.logRR, ad.FI.RD)
max(ad.FI.all[!is.na(ad.FI.all)])
#[1] 32
max(c(table(ad.FI.Fisher), table(ad.FI.chisq),
  table(ad.FI.logOR), table(ad.FI.logRR), table(ad.FI.RD)))
#[1] 50

sum(is.na(ad.FI.Fisher))
#[1] 0
sum(ad.pval.Fisher < 0.05)
#[1] 86
table(ad.FI.Fisher.sig2nonsig[ad.pval.Fisher < 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 15 17 21 24 28 31
#26 10  6  4  6  8  6  5  1  3  2  2  1  1  1  1  1  1  1
quantile(ad.FI.Fisher.sig2nonsig[ad.pval.Fisher < 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    1    4    7   31
sum(ad.pval.Fisher >= 0.05)
#[1] 270
table(ad.FI.Fisher.nonsig2sig[ad.pval.Fisher >= 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 18 19 20 22 24
#18 30 25 32 21 22 29 16 14 14  9 17  5  4  6  2  1  2  1  1  1
quantile(ad.FI.Fisher.nonsig2sig[ad.pval.Fisher >= 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    3    6    9   24
table(ad.FI.Fisher, useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 24 28 31
#44 40 31 36 27 30 35 21 15 17 11 19  6  4  7  2  1  1  2  1  1  1  2  1  1
quantile(ad.FI.Fisher, probs = c(0, 0.25, 0.5, 0.75, 1),
  type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    3    5    9   31

sum(is.na(ad.FI.chisq))
#[1] 0
sum(ad.pval.chisq < 0.05)
#[1] 76
table(ad.FI.chisq.sig2nonsig[ad.pval.chisq < 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 15 17 21 24 28 31
#21  8  3  7  8  6  5  3  2  2  3  2  1  1  1  1  1  1
quantile(ad.FI.chisq.sig2nonsig[ad.pval.chisq < 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    1    4    7   31
sum(ad.pval.chisq >= 0.05)
#[1] 280
table(ad.FI.chisq.nonsig2sig[ad.pval.chisq >= 0.05], useNA = "no")
```

```
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 22 25
#22 25 23 30 31 20 23 19 21 11 12 11 13  3  8  1  1  1  1  2  1  1
quantile(ad.FI.chisq.nonsig2sig[!is.na(ad.FI.chisq.nonsig2sig)],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    3    6    9   25
table(ad.FI.chisq, useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 24 25 28
#43 33 26 37 39 26 28 22 23 13 15 13 13  3  9  1  2  1  1  2  1  1  1  1  1
#31
# 1
quantile(ad.FI.chisq, probs = c(0, 0.25, 0.5, 0.75, 1),
  type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    3    5    9   31

sum(is.na(ad.FI.logOR))
#[1] 0
sum(ad.pval.logOR < 0.05)
#[1] 91
table(ad.FI.logOR.sig2nonsig[ad.pval.logOR < 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 16 18 22 25 29 32
#20 16  8  4  6  8  6  7  1  3  1  3  2  1  1  1  1  1  1
quantile(ad.FI.logOR.sig2nonsig[ad.pval.logOR < 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    2    4    7   32
sum(ad.pval.logOR >= 0.05)
#[1] 265
table(ad.FI.logOR.nonsig2sig[ad.pval.logOR >= 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 21 24
#27 23 27 25 23 30 22 21  8 14 11 15  3  7  2  1  1  1  2  1  1
quantile(ad.FI.logOR.nonsig2sig[ad.pval.logOR >= 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    3    6    9   24
table(ad.FI.logOR, useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 21 22 24 25 29 32
#47 39 35 29 29 38 28 28  9 17 12 18  5  7  2  2  1  2  2  1  1  1  1  1  1
quantile(ad.FI.logOR, probs = c(0, 0.25, 0.5, 0.75, 1),
  type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    3    5    8   32

sum(is.na(ad.FI.logRR))
#[1] 0
sum(ad.pval.logRR < 0.05)
#[1] 88
table(ad.FI.logRR.sig2nonsig[ad.pval.logRR < 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8 10 11 12 13 16 18 22 25 28 32
#24 10  7  6  6  7  8  5  3  1  3  2  1  1  1  1  1  1
quantile(ad.FI.logRR.sig2nonsig[ad.pval.logRR < 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    1    4    7   32
sum(ad.pval.logRR >= 0.05)
#[1] 268
table(ad.FI.logRR.nonsig2sig[ad.pval.logRR >= 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 21 24
#20 21 28 38 18 29 23 25  7 13 11 15  4  6  3  1  1  1  2  1  1
quantile(ad.FI.logRR.nonsig2sig[ad.pval.logRR >= 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    3    6    8   24
table(ad.FI.logRR, useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 21 22 24 25 28 32
#44 31 35 44 24 36 31 30  7 16 12 18  6  6  3  2  1  2  2  1  1  1  1  1  1
quantile(ad.FI.logRR, probs = c(0, 0.25, 0.5, 0.75, 1),
  type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    3    5    8   32

sum(is.na(ad.FI.RD))
#[1] 0
sum(ad.pval.RD < 0.05)
#[1] 99
table(ad.FI.RD.sig2nonsig[ad.pval.RD < 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 16 18 22 25 29 32
#19 24  9  4  5  8  7  6  2  2  2  2  3  1  1  1  1  1  1
quantile(ad.FI.RD.sig2nonsig[ad.pval.RD < 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    2    3    7   32
sum(ad.pval.RD >= 0.05)
#[1] 257
table(ad.FI.RD.nonsig2sig[ad.pval.RD >= 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 21 24
#29 26 30 23 21 24 21 18 11 11 11 13  5  6  1  1  1  1  2  1  1
quantile(ad.FI.RD.nonsig2sig[ad.pval.RD >= 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    3    5    9   24
table(ad.FI.RD, useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 21 22 24 25 29 32
#48 50 39 27 26 32 28 24 13 13 13 15  8  6  1  2  1  2  2  1  1  1  1  1  1
quantile(ad.FI.RD, probs = c(0, 0.25, 0.5, 0.75, 1),
  type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    2    5    8   32

## Figures: real data analysis for the responder outcome
barFI <- function(ad.FI.sig2nonsig, ad.FI.nonsig2sig,
  main = ""){
  bar.FI <- matrix(NA, nrow = 2, ncol = 32)
  for(i in 1:32){
    bar.FI[1, i] <- table(ad.FI.nonsig2sig[!is.na(
      ad.FI.nonsig2sig)])[as.character(i)]
    bar.FI[2, i] <- table(ad.FI.sig2nonsig[!is.na(
      ad.FI.sig2nonsig)])[as.character(i)]
  }
  bar.FI[is.na(bar.FI)] <- 0
  ranges <- 1:32
  x <- barplot(bar.FI, names.arg = ranges, space = 0,
    col = c(adjustcolor("forestgreen", alpha.f = 0.1),
    adjustcolor("firebrick", alpha.f = 0.1)),
    las = 1, xaxt = "n", cex.axis = 1, ylim = c(0, 50),
    yaxt = "n")
  axis(side = 2, at = seq(0, 50, by = 10), labels = FALSE)
  text(cex = 1, x = -1.4, y = seq(0, 50, by = 10),
    labels = as.character(seq(0, 50, by = 10)),
    xpd = TRUE, srt = 0, pos = 2)
  title(xlab = "Fragility index", line = 1.5)
  title(ylab = "Frequency", line = 2)
  text(cex = 0.8, x = 1:32 - 0.5, y = 0,
    labels = as.character(1:32),
    xpd = TRUE, pos = 1, font = 1)
  mtext(text = main, side = 3, line = 0.5, outer = FALSE,
    cex = 1, font = 1)
}
## Figure 4A: Bar plot; Fisher's exact test
pdf(file = "Fig4A.pdf", width = 10, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
barFI(ad.FI.Fisher.sig2nonsig, ad.FI.Fisher.nonsig2sig,
  main = "(A) Fisher's exact test")
dev.off()
## Figure 4B: Bar plot; chi-squared test
pdf(file = "Fig4B.pdf", width = 10, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
barFI(ad.FI.chisq.sig2nonsig, ad.FI.chisq.nonsig2sig,
  main = "(B) Chi\255squared test")
dev.off()
## Figure S4A: Bar plot; log odds ratio
pdf(file = "FigS4A.pdf", width = 10, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
barFI(ad.FI.logOR.sig2nonsig, ad.FI.logOR.nonsig2sig,
  main = "(A) Log odds ratio")
dev.off()
## Figure S4B: Bar plot; log relative risk
pdf(file = "FigS4B.pdf", width = 10, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
barFI(ad.FI.logRR.sig2nonsig, ad.FI.logRR.nonsig2sig,
  main = "(B) Log relative risk")
dev.off()
## Figure S4C: Bar plot; risk difference
pdf(file = "FigS4C.pdf", width = 10, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
barFI(ad.FI.RD.sig2nonsig, ad.FI.RD.nonsig2sig,
  main = "(C) Risk difference")
dev.off()

## counterexamples that the minimal event status
##  modifications may not be in a single treatment
##  arm; based on Fisher's exact test
which(ad.FI.Fisher < ad.FI0.Fisher & ad.FI.Fisher < ad.FI1.Fisher)
#[1]  42  46  99 112 122 174 230 264
sum(ad.FI.Fisher < ad.FI0.Fisher & ad.FI.Fisher < ad.FI1.Fisher,
  na.rm = TRUE)
#[1] 8
max(pmin(ad.FI0.Fisher, ad.FI1.Fisher) - ad.FI.Fisher, na.rm = TRUE)
#[1] 2
any(is.na(ad.FI0.Fisher) & is.na(ad.FI1.Fisher))
#[1] FALSE
any(ad.FI.Fisher < ad.FI0.Fisher & ad.FI.Fisher < ad.FI1.Fisher &
  ad.pval.Fisher < 0.05, na.rm = TRUE)
#[1] TRUE
which(ad.FI.Fisher < ad.FI0.Fisher & ad.FI.Fisher < ad.FI1.Fisher &
  ad.pval.Fisher < 0.05)
#[1]  46 112 264

sum(ad.FI.chisq < ad.FI0.chisq & ad.FI.chisq < ad.FI1.chisq, na.rm = TRUE)
#[1] 10
any(ad.FI.chisq < ad.FI0.chisq & ad.FI.chisq < ad.FI1.chisq &
  ad.pval.chisq < 0.05, na.rm = TRUE)
#[1] FALSE
sum(ad.FI.logOR < ad.FI0.logOR & ad.FI.logOR < ad.FI1.logOR, na.rm = TRUE)
#[1] 7
any(ad.FI.logOR < ad.FI0.logOR & ad.FI.logOR < ad.FI1.logOR &
  ad.pval.logOR < 0.05, na.rm = TRUE)
#[1] FALSE
```

```
sum(ad.FI.logRR < ad.FI0.logRR & ad.FI.logRR < ad.FI1.logRR, na.rm = TRUE)
#[1] 3
any(ad.FI.logRR < ad.FI0.logRR & ad.FI.logRR < ad.FI1.logRR &
  ad.pval.logRR < 0.05, na.rm = TRUE)
#[1] FALSE
sum(ad.FI.RD < ad.FI0.RD & ad.FI.RD < ad.FI1.RD, na.rm = TRUE)
#[1] 2
any(ad.FI.RD < ad.FI0.RD & ad.FI.RD < ad.FI1.RD &
  ad.pval.RD < 0.05, na.rm = TRUE)
#[1] TRUE
which(ad.FI.RD < ad.FI0.RD & ad.FI.RD < ad.FI1.RD &
  ad.pval.RD < 0.05)
#[1] 267

## Figure 5A: example
pdf(file = "Fig5A.pdf", width = 5, height = 5)
i <- 112
dat.temp <- ad.responder[ad.responder$studyid == i,]
dat.temp
e0.temp <- dat.temp$responder[1]
e1.temp <- dat.temp$responder[2]
n0.temp <- dat.temp$samplesize[1]
n1.temp <- dat.temp$samplesize[2]

begin <- Sys.time()
rslt.countexam.all <- fragility.all(
  e0 = e0.temp, n0 = n0.temp,
  e1 = e1.temp, n1 = n1.temp, methods = all.methods,
  modify0 = "both", modify1 = "both", alpha = 0.05, save = TRUE)
end <- Sys.time()
end - begin
#Time difference of 1.268612 secs

rslt.countexam.all$FI.Fisher.sig2nonsig
#[1] 17
rslt.countexam.all$FI0.Fisher.sig2nonsig
#[1] 18
rslt.countexam.all$FI1.Fisher.sig2nonsig
#[1] 18

pvals.plot(rslt.countexam.all, method = "Fisher",
  modify0 = TRUE, modify1 = TRUE, trun = 10, cex.pts = 0.5,
  main = "(A) Trial by Cohn and Wilcox",
  legend.pvals = c(0.01, 0.001), cex.legend.pvals = 0.6,
  mar = c(3, 3, 2, 1) + 0.1)
dev.off()

## Figures 5B and 5C: examples
which(ad.FI.Fisher.sig2nonsig == 1 & ad.pval.Fisher < 0.05)
# [1]   1  19  20  25  38  48  66  78  80  92 178 190 195 204 214 217 227 231
#[19] 252 266 279 306 308 311 328 330

i <- 48
dat.temp <- ad.responder[ad.responder$studyid == i,]
dat.temp
e0.temp <- dat.temp$responder[1]
e1.temp <- dat.temp$responder[2]
n0.temp <- dat.temp$samplesize[1]
n1.temp <- dat.temp$samplesize[2]
pdf(file = "Fig5B.pdf", width = 5, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
begin <- Sys.time()
temp.diffsignif <- FIvsSignif(e0 = e0.temp, n0 = n0.temp,
  e1 = e1.temp, n1 = n1.temp, method = "Fisher",
  modify0 = "both", modify1 = "both",
  alpha.from = 0.005, alpha.to = 0.05, alpha.breaks = 100,
  cex.pts = 0.5, main = "(B) Trial by GlaxoSmithKline (average FI = 1.10)")
end <- Sys.time()
end - begin
#Time difference of 0.06981206 secs
temp.diffsignif$FI.avg
# 1.1
dev.off()

i <- 78
dat.temp <- ad.responder[ad.responder$studyid == i,]
dat.temp
e0.temp <- dat.temp$responder[1]
e1.temp <- dat.temp$responder[2]
n0.temp <- dat.temp$samplesize[1]
n1.temp <- dat.temp$samplesize[2]
pdf(file = "Fig5C.pdf", width = 5, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
begin <- Sys.time()
temp.diffsignif <- FIvsSignif(e0 = e0.temp, n0 = n0.temp,
  e1 = e1.temp, n1 = n1.temp, method = "Fisher",
  modify0 = "both", modify1 = "both",
  alpha.from = 0.005, alpha.to = 0.05, alpha.breaks = 100,
  cex.pts = 0.5, main = "(C) Trial by Cutler et al. (average FI = 2.90)")
end <- Sys.time()
end - begin
#Time difference of 0.1621439 secs
temp.diffsignif$FI.avg
# 2.9
dev.off()

## average FI
ad.FI.Fisher.avg <- ad.FI.chisq.avg <- ad.FI.logOR.avg <-
  ad.FI.logRR.avg <- ad.FI.RD.avg <- rep(NA, n.study)
all.methods <- c("Fisher", "chisq", "logOR", "logRR", "RD")
begin <- Sys.time()
for(i in 1:n.study){
  dat.temp <- ad.responder[ad.responder$studyid == i,]
  e0.temp <- dat.temp$responder[1]
  e1.temp <- dat.temp$responder[2]
  n0.temp <- dat.temp$samplesize[1]
  n1.temp <- dat.temp$samplesize[2]

  rslt.temp <- frag.study.alpha(e0 = e0.temp, n0 = n0.temp,
    e1 = e1.temp, n1 = n1.temp, methods = all.methods,
    modify0 = "both", modify1 = "both",
    alpha.from = 0.005, alpha.to = 0.05, alpha.breaks = 100)
  ad.FI.Fisher.avg[i] <- rslt.temp$FI.avg["Fisher"]
  ad.FI.chisq.avg[i] <- rslt.temp$FI.avg["chisq"]
  ad.FI.logOR.avg[i] <- rslt.temp$FI.avg["logOR"]
  ad.FI.logRR.avg[i] <- rslt.temp$FI.avg["logRR"]
  ad.FI.RD.avg[i] <- rslt.temp$FI.avg["RD"]
}
end <- Sys.time()
end - begin
#Time difference of 1.950986 mins

sum(is.na(ad.FI.Fisher.avg))
#[1] 0
max(ad.FI.Fisher.avg, na.rm = TRUE)
#[1] 28.59
```

```
sum(is.na(ad.FI.chisq.avg))
#[1] 0
max(ad.FI.chisq.avg, na.rm = TRUE)
#[1] 29.19
sum(is.na(ad.FI.logOR.avg))
#[1] 2
max(ad.FI.logOR.avg, na.rm = TRUE)
#[1] 28.79
sum(is.na(ad.FI.logRR.avg))
#[1] 9
max(ad.FI.logRR.avg, na.rm = TRUE)
#[1] 28.66
sum(is.na(ad.FI.RD.avg))
#[1] 0
max(ad.FI.RD.avg, na.rm = TRUE)
#[1] 28.97

pdf(file = "FigS5A.pdf", width = 10, height = 5)
par(mar = c(3, 3, 1, 0) + 0.1, mgp = c(2, 1, 0))
hist(ad.FI.Fisher.avg, xlim = c(1, 30), ylim = c(0, 60),
  breaks = 1:30, main = "(A) Fisher's exact test", font.main = 1,
  xlab = "Average fragility index")
dev.off()
pdf(file = "FigS5B.pdf", width = 10, height = 5)
par(mar = c(3, 3, 1, 0) + 0.1, mgp = c(2, 1, 0))
hist(ad.FI.chisq.avg, xlim = c(1, 30), ylim = c(0, 60),
  breaks = 1:30, main = "(B) Chi\255squared test", font.main = 1,
  xlab = "Average fragility index")
dev.off()
pdf(file = "FigS6A.pdf", width = 10, height = 5)
par(mar = c(3, 3, 1, 0) + 0.1, mgp = c(2, 1, 0))
hist(ad.FI.logOR.avg, xlim = c(1, 30), ylim = c(0, 60),
  breaks = 1:30, main = "(A) Log odds ratio", font.main = 1,
  xlab = "Average fragility index")
dev.off()
pdf(file = "FigS6B.pdf", width = 10, height = 5)
par(mar = c(3, 3, 1, 0) + 0.1, mgp = c(2, 1, 0))
hist(ad.FI.logRR.avg, xlim = c(1, 30), ylim = c(0, 60),
  breaks = 1:30, main = "(B) Log relative risk", font.main = 1,
  xlab = "Average fragility index")
dev.off()
pdf(file = "FigS6C.pdf", width = 10, height = 5)
par(mar = c(3, 3, 1, 0) + 0.1, mgp = c(2, 1, 0))
hist(ad.FI.RD.avg, xlim = c(1, 30), ylim = c(0, 60),
  breaks = 1:30, main = "(C) Risk difference", font.main = 1,
  xlab = "Average fragility index")
dev.off()
```

```
############################
#### Outcome: dropout
############################

studyid.dropout <-
  c(1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6,
  7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 12,
  13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 18,
  19, 19, 20, 20, 21, 21, 22, 22, 23, 23, 24, 24,
  25, 25, 26, 26, 27, 27, 28, 28, 29, 29, 30, 30,
  31, 31, 32, 32, 33, 33, 34, 34, 35, 35, 36, 36,
  37, 37, 38, 38, 39, 39, 40, 40, 41, 41, 42, 42,
  43, 43, 44, 44, 45, 45, 46, 46, 47, 47, 48, 48,
  49, 49, 50, 50, 51, 51, 52, 52, 53, 53, 54, 54,
  55, 55, 56, 56, 57, 57, 58, 58, 59, 59, 60, 60,
  61, 61, 62, 62, 63, 63, 64, 64, 65, 65, 66, 66,
  67, 67, 68, 68, 69, 69, 70, 70, 71, 71, 72, 72,
  73, 73, 74, 74, 75, 75, 76, 76, 77, 77, 78, 78,
  79, 79, 80, 80, 81, 81, 82, 82, 83, 83, 84, 84,
  85, 85, 86, 86, 87, 87, 88, 88, 89, 89, 90, 90,
  91, 91, 92, 92, 93, 93, 94, 94, 95, 95, 96, 96,
  97, 97, 98, 98, 99, 99, 100, 100, 101, 101, 102, 102,
  103, 103, 104, 104, 105, 105, 106, 106, 107, 107, 108, 108,
  109, 109, 110, 110, 111, 111, 112, 112, 113, 113, 114, 114,
  115, 115, 116, 116, 117, 117, 118, 118, 119, 119, 120, 120,
  121, 121, 122, 122, 123, 123, 124, 124, 125, 125, 126, 126,
  127, 127, 128, 128, 129, 129, 130, 130, 131, 131, 132, 132,
  133, 133, 134, 134, 135, 135, 136, 136, 137, 137, 138, 138,
  139, 139, 140, 140, 141, 141, 142, 142, 143, 143, 144, 144,
  145, 145, 146, 146, 147, 147, 148, 148, 149, 149, 150, 150,
  151, 151, 152, 152, 153, 153, 154, 154, 155, 155, 156, 156,
  157, 157, 158, 158, 159, 159, 160, 160, 161, 161, 162, 162,
  163, 163, 164, 164, 165, 165, 166, 166, 167, 167, 168, 168,
  169, 169, 170, 170, 171, 171, 172, 172, 173, 173, 174, 174,
  175, 175, 176, 176, 177, 177, 178, 178, 179, 179, 180, 180,
  181, 181, 182, 182, 183, 183, 184, 184, 185, 185, 186, 186,
  187, 187, 188, 188, 189, 189, 190, 190, 191, 191, 192, 192,
  193, 193, 194, 194, 195, 195, 196, 196, 197, 197, 198, 198,
  199, 199, 200, 200, 201, 201, 202, 202, 203, 203, 204, 204,
  205, 205, 206, 206, 207, 207, 208, 208, 209, 209, 210, 210,
  211, 211, 212, 212, 213, 213, 214, 214, 215, 215, 216, 216,
  217, 217, 218, 218, 219, 219, 220, 220, 221, 221, 222, 222,
  223, 223, 224, 224, 225, 225, 226, 226, 227, 227, 228, 228,
  229, 229, 230, 230, 231, 231, 232, 232, 233, 233, 234, 234,
  235, 235, 236, 236, 237, 237, 238, 238, 239, 239, 240, 240,
  241, 241, 242, 242, 243, 243, 244, 244, 245, 245, 246, 246,
  247, 247, 248, 248, 249, 249, 250, 250, 251, 251, 252, 252,
  253, 253, 254, 254, 255, 255, 256, 256, 257, 257, 258, 258,
  259, 259, 260, 260, 261, 261, 262, 262, 263, 263, 264, 264,
  265, 265, 266, 266, 267, 267, 268, 268, 269, 269, 270, 270,
  271, 271, 272, 272, 273, 273, 274, 274, 275, 275, 276, 276,
  277, 277, 278, 278, 279, 279, 280, 280, 281, 281, 282, 282,
  283, 283, 284, 284, 285, 285, 286, 286, 287, 287, 288, 288,
  289, 289, 290, 290, 291, 291, 292, 292, 293, 293, 294, 294,
  295, 295, 296, 296, 297, 297, 298, 298, 299, 299, 300, 300,
  301, 301, 302, 302, 303, 303, 304, 304, 305, 305, 306, 306,
  307, 307, 308, 308, 309, 309, 310, 310, 311, 311, 312, 312,
  313, 313, 314, 314, 315, 315, 316, 316, 317, 317, 318, 318,
  319, 319, 320, 320, 321, 321, 322, 322, 323, 323, 324, 324,
  325, 325, 326, 326, 327, 327, 328, 328, 329, 329, 330, 330,
  331, 331, 332, 332, 333, 333, 334, 334, 335, 335, 336, 336,
  337, 337, 338, 338, 339, 339, 340, 340, 341, 341, 342, 342,
  343, 343, 344, 344, 345, 345, 346, 346, 347, 347)
```

```
drug.dropout <-
  c('agomelatine', 'placebo', 'agomelatine', 'placebo',
   'agomelatine', 'fluoxetine', 'agomelatine', 'placebo',
   'agomelatine', 'paroxetine', 'agomelatine', 'venlafaxine',
   'agomelatine', 'paroxetine', 'agomelatine', 'fluoxetine',
   'agomelatine', 'escitalopram', 'agomelatine', 'duloxetine',
   'agomelatine', 'escitalopram', 'agomelatine', 'placebo',
   'agomelatine', 'venlafaxine', 'agomelatine', 'placebo',
   'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
   'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
   'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
   'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
   'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
   'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
   'amitriptyline', 'placebo', 'amitriptyline', 'placebo',
   'amitriptyline', 'placebo', 'amitriptyline', 'bupropion',
   'bupropion', 'placebo', 'bupropion', 'placebo',
   'bupropion', 'placebo', 'fluoxetine', 'bupropion',
   'bupropion', 'trazodone', 'placebo', 'bupropion',
   'bupropion', 'placebo', 'bupropion', 'placebo',
   'bupropion', 'venlafaxine', 'placebo', 'bupropion',
   'bupropion', 'placebo', 'venlafaxine', 'placebo',
   'bupropion', 'placebo', 'bupropion', 'placebo',
   'bupropion', 'placebo', 'bupropion', 'placebo',
   'bupropion', 'paroxetine', 'bupropion', 'placebo',
   'bupropion', 'sertraline', 'citalopram', 'clomipramine',
   'citalopram', 'amitriptyline', 'fluvoxamine', 'citalopram',
   'fluoxetine', 'citalopram', 'fluoxetine', 'citalopram',
   'citalopram', 'amitriptyline', 'citalopram', 'placebo',
   'citalopram', 'placebo', 'reboxetine', 'citalopram',
   'citalopram', 'placebo', 'clomipramine', 'placebo',
   'placebo', 'venlafaxine', 'placebo', 'venlafaxine',
   'desvenlafaxine', 'placebo', 'desvenlafaxine', 'placebo',
   'desvenlafaxine', 'placebo', 'desvenlafaxine', 'placebo',
   'placebo', 'desvenlafaxine', 'duloxetine', 'placebo',
   'duloxetine', 'placebo', 'duloxetine', 'venlafaxine',
   'duloxetine', 'placebo', 'duloxetine', 'paroxetine',
   'duloxetine', 'placebo', 'duloxetine', 'placebo',
   'duloxetine', 'placebo', 'duloxetine', 'paroxetine',
   'escitalopram', 'fluoxetine', 'escitalopram', 'fluoxetine',
   'escitalopram', 'placebo', 'escitalopram', 'venlafaxine',
   'escitalopram', 'venlafaxine', 'placebo', 'escitalopram',
   'escitalopram', 'sertraline', 'escitalopram', 'placebo',
   'escitalopram', 'citalopram', 'escitalopram', 'citalopram',
   'paroxetine', 'escitalopram', 'escitalopram', 'duloxetine',
   'escitalopram', 'duloxetine', 'escitalopram', 'paroxetine',
   'escitalopram', 'citalopram', 'fluoxetine', 'escitalopram',
   'escitalopram', 'citalopram', 'citalopram', 'escitalopram',
   'escitalopram', 'citalopram', 'escitalopram', 'placebo',
   'escitalopram', 'placebo', 'escitalopram', 'placebo',
   'escitalopram', 'placebo', 'escitalopram', 'placebo',
   'escitalopram', 'placebo', 'fluoxetine', 'placebo',
   'fluoxetine', 'placebo', 'fluoxetine', 'amitriptyline',
   'fluoxetine', 'placebo', 'fluoxetine', 'amitriptyline',
   'fluoxetine', 'placebo', 'fluoxetine', 'amitriptyline',
   'placebo', 'fluoxetine', 'fluoxetine', 'amitriptyline',
   'fluoxetine', 'trazodone', 'fluoxetine', 'amitriptyline',
   'placebo', 'fluoxetine', 'trazodone', 'fluoxetine',
   'amitriptyline', 'fluoxetine', 'fluoxetine', 'clomipramine',
   'fluoxetine', 'trazodone', 'fluoxetine', 'amitriptyline',
   'clomipramine', 'fluoxetine', 'fluoxetine', 'amitriptyline',
   'fluoxetine', 'amitriptyline', 'trazodone', 'fluoxetine',
   'fluoxetine', 'amitriptyline', 'fluoxetine', 'placebo',
   'fluoxetine', 'placebo', 'fluoxetine', 'placebo',
   'amitriptyline', 'fluoxetine', 'fluoxetine', 'amitriptyline',
   'fluoxetine', 'amitriptyline', 'fluoxetine', 'amitriptyline',
   'fluoxetine', 'placebo', 'fluoxetine', 'placebo',
   'fluoxetine', 'amitriptyline', 'placebo', 'fluoxetine',
   'fluoxetine', 'placebo', 'fluvoxamine', 'clomipramine',
   'fluvoxamine', 'placebo', 'fluvoxamine', 'placebo',
   'fluvoxamine', 'placebo', 'fluvoxamine', 'placebo',
   'fluvoxamine', 'placebo', 'fluvoxamine', 'placebo',
   'amitriptyline', 'fluvoxamine', 'fluvoxamine', 'sertraline',
   'fluvoxamine', 'clomipramine', 'fluvoxamine', 'fluoxetine',
   'fluvoxamine', 'placebo', 'fluvoxamine', 'placebo',
   'fluvoxamine', 'paroxetine', 'fluvoxamine', 'amitriptyline',
   'fluvoxamine', 'fluoxetine', 'fluvoxamine', 'sertraline',
   'fluvoxamine', 'amitriptyline', 'fluvoxamine', 'placebo',
   'placebo', 'levomilnacipran', 'placebo', 'levomilnacipran',
   'placebo', 'levomilnacipran', 'placebo', 'levomilnacipran',
   'milnacipran', 'amitriptyline', 'milnacipran', 'fluoxetine',
   'fluvoxamine', 'milnacipran', 'milnacipran', 'sertraline',
   'milnacipran', 'paroxetine', 'mirtazapine', 'placebo',
   'mirtazapine', 'placebo', 'mirtazapine', 'placebo',
   'mirtazapine', 'trazodone', 'mirtazapine', 'amitriptyline',
   'mirtazapine', 'amitriptyline', 'mirtazapine', 'amitriptyline',
   'mirtazapine', 'fluoxetine', 'mirtazapine', 'citalopram',
   'mirtazapine', 'paroxetine', 'mirtazapine', 'paroxetine',
   'mirtazapine', 'fluoxetine', 'mirtazapine', 'sertraline',
   'mirtazapine', 'paroxetine', 'mirtazapine', 'fluoxetine',
   'mirtazapine', 'fluoxetine', 'mirtazapine', 'fluoxetine',
   'mirtazapine', 'venlafaxine', 'mirtazapine', 'fluvoxamine',
   'mirtazapine', 'fluvoxamine', 'mirtazapine', 'venlafaxine',
   'mirtazapine', 'paroxetine', 'mirtazapine', 'paroxetine',
   'nefazodone', 'amitriptyline', 'nefazodone', 'fluoxetine',
   'nefazodone', 'paroxetine', 'nefazodone', 'fluoxetine',
   'nefazodone', 'sertraline', 'nefazodone', 'placebo',
   'nefazodone', 'fluoxetine', 'nefazodone', 'paroxetine',
   'nefazodone', 'placebo', 'nefazodone', 'placebo',
   'nefazodone', 'placebo', 'nefazodone', 'placebo',
   'nefazodone', 'placebo', 'paroxetine', 'amitriptyline',
   'paroxetine', 'amitriptyline', 'paroxetine', 'amitriptyline',
   'paroxetine', 'amitriptyline', 'paroxetine', 'amitriptyline',
   'paroxetine', 'placebo', 'paroxetine', 'amitriptyline',
   'paroxetine', 'amitriptyline', 'paroxetine', 'amitriptyline',
   'paroxetine', 'placebo', 'paroxetine', 'placebo',
   'paroxetine', 'placebo', 'paroxetine', 'placebo',
   'paroxetine', 'placebo', 'paroxetine', 'placebo',
   'paroxetine', 'placebo', 'paroxetine', 'placebo',
   'paroxetine', 'placebo', 'paroxetine', 'placebo',
   'paroxetine', 'placebo', 'clomipramine', 'paroxetine',
   'clomipramine', 'paroxetine', 'paroxetine', 'placebo',
   'paroxetine', 'clomipramine', 'paroxetine', 'clomipramine',
   'paroxetine', 'amitriptyline', 'paroxetine', 'clomipramine',
   'paroxetine', 'fluoxetine', 'paroxetine', 'fluoxetine',
   'paroxetine', 'fluoxetine', 'paroxetine', 'amitriptyline',
   'amitriptyline', 'paroxetine', 'paroxetine', 'amitriptyline',
   'paroxetine', 'fluvoxamine', 'paroxetine', 'fluoxetine',
   'paroxetine', 'fluoxetine', 'paroxetine', 'placebo',
   'paroxetine', 'placebo', 'paroxetine', 'fluoxetine',
   'paroxetine', 'amitriptyline', 'paroxetine', 'amitriptyline',
   'paroxetine', 'amitriptyline', 'paroxetine', 'amitriptyline',
   'paroxetine', 'clomipramine', 'paroxetine', 'fluoxetine',
   'paroxetine', 'trazodone', 'paroxetine', 'amitriptyline',
   'placebo', 'paroxetine', 'paroxetine', 'fluoxetine',
   'paroxetine', 'amitriptyline', 'paroxetine', 'placebo',
   'paroxetine', 'placebo', 'paroxetine', 'placebo',
   'paroxetine', 'placebo', 'bupropion', 'paroxetine',
   'paroxetine', 'placebo', 'paroxetine', 'placebo',
   'amitriptyline', 'paroxetine', 'paroxetine', 'placebo',
   'paroxetine', 'placebo', 'paroxetine', 'placebo',
   'paroxetine', 'placebo', 'paroxetine', 'placebo',
   'reboxetine', 'placebo', 'reboxetine', 'placebo',
   'reboxetine', 'placebo', 'reboxetine', 'fluoxetine',
   'reboxetine', 'placebo', 'reboxetine', 'placebo',
   'reboxetine', 'paroxetine', 'reboxetine', 'citalopram',
   'reboxetine', 'placebo', 'reboxetine', 'fluoxetine',
   'sertraline', 'amitriptyline', 'sertraline', 'fluoxetine',
   'sertraline', 'amitriptyline', 'sertraline', 'clomipramine',
   'sertraline', 'fluoxetine', 'sertraline', 'fluoxetine',
   'citalopram', 'sertraline', 'amitriptyline', 'sertraline',
   'sertraline', 'fluoxetine', 'paroxetine', 'sertraline',
   'sertraline', 'amitriptyline', 'sertraline', 'clomipramine',
   'venlafaxine', 'sertraline', 'venlafaxine', 'sertraline',
   'sertraline', 'placebo', 'sertraline', 'placebo',
   'sertraline', 'placebo', 'sertraline', 'placebo',
   'sertraline', 'placebo', 'sertraline', 'placebo',
   'sertraline', 'placebo', 'sertraline', 'placebo',
   'sertraline', 'placebo', 'bupropion', 'sertraline',
   'sertraline', 'placebo', 'sertraline', 'placebo',
   'trazodone', 'placebo', 'amitriptyline', 'trazodone',
   'trazodone', 'placebo', 'trazodone', 'placebo',
   'amitriptyline', 'trazodone', 'amitriptyline', 'trazodone',
   'amitriptyline', 'trazodone', 'amitriptyline', 'trazodone',
   'paroxetine', 'trazodone', 'sertraline', 'trazodone',
   'trazodone', 'placebo', 'trazodone', 'placebo',
   'venlafaxine', 'clomipramine', 'paroxetine', 'venlafaxine',
   'paroxetine', 'venlafaxine', 'venlafaxine', 'fluoxetine',
   'venlafaxine', 'fluvoxamine', 'venlafaxine', 'placebo',
   'venlafaxine', 'placebo', 'venlafaxine', 'fluoxetine',
   'venlafaxine', 'fluoxetine', 'venlafaxine', 'fluoxetine',
   'venlafaxine', 'sertraline', 'fluoxetine', 'venlafaxine',
   'venlafaxine', 'placebo', 'venlafaxine', 'placebo',
   'fluoxetine', 'venlafaxine', 'venlafaxine', 'placebo',
   'venlafaxine', 'placebo', 'fluoxetine', 'venlafaxine',
   'fluvoxamine', 'venlafaxine', 'fluoxetine', 'venlafaxine',
   'paroxetine', 'venlafaxine', 'venlafaxine', 'clomipramine',
   'fluoxetine', 'venlafaxine', 'sertraline', 'venlafaxine',
   'fluoxetine', 'venlafaxine', 'venlafaxine', 'amitriptyline',
   'paroxetine', 'venlafaxine', 'venlafaxine', 'amitriptyline',
   'venlafaxine', 'citalopram', 'venlafaxine', 'Placebo',
   'venlafaxine', 'fluoxetine', 'vilazodone', 'placebo',
   'vilazodone', 'placebo', 'vilazodone', 'placebo',
   'vilazodone', 'placebo', 'placebo', 'vortioxetine',
   'vortioxetine', 'venlafaxine')
 samplesize.dropout <-
   c(107, 105, 118, 120, 252, 263, 109, 109, 167, 168, 137, 140,
   213, 199, 314, 314, 71, 67, 202, 216, 164, 160, 151, 71,
   165, 167, 151, 71, 93, 50, 12, 7, 95, 94, 31, 28,
   85, 87, 12, 10, 20, 20, 124, 130, 55, 54, 16, 15,
   19, 22, 7, 8, 58, 56, 50, 49, 31, 30, 41, 77,
   13, 7, 45, 22, 112, 112, 62, 61, 63, 61, 121, 120,
   150, 154, 10, 9, 171, 177, 139, 135, 212, 208, 198, 189,
   56, 29, 77, 40, 56, 52, 25, 24, 69, 71, 166, 159,
   122, 126, 57, 57, 24, 20, 109, 108, 184, 173, 158, 158,
   179, 186, 89, 91, 87, 91, 46, 55, 127, 130, 20, 18,
   120, 127, 125, 117, 291, 146, 218, 221, 237, 235, 227, 227,
   120, 118, 123, 122, 128, 139, 166, 166, 207, 104, 238, 240,
   151, 157, 249, 121, 47, 44, 140, 141, 16, 14, 102, 103,
   154, 155, 101, 101, 148, 145, 189, 191, 107, 108, 132, 135,
   175, 182, 142, 152, 159, 166, 144, 151, 140, 138, 232, 227,
   109, 110, 117, 123, 32, 32, 120, 120, 24, 24, 62, 138,
   140, 71, 157, 157, 84, 76, 50, 99, 69, 65, 22, 25,
   18, 24, 25, 28, 54, 58, 21, 21, 185, 169, 32, 32,
   29, 32, 32, 33, 22, 21, 13, 14, 59, 62, 13, 14,
   20, 20, 71, 72, 21, 17, 16, 18, 60, 60, 20, 22,
   30, 31, 61, 65, 30, 28, 46, 43, 335, 336, 72, 72,
   31, 35, 67, 75, 21, 21, 77, 80, 35, 35, 49, 52,
   11, 12, 43, 47, 57, 58, 22, 21, 22, 22, 35, 25,
   35, 31, 161, 149, 22, 20, 18, 18, 17, 16, 49, 48,
   20, 20, 51, 49, 50, 50, 50, 50, 30, 30, 113, 122,
   90, 94, 40, 48, 35, 34, 30, 30, 281, 282, 182, 175,
   220, 222, 93, 90, 44, 43, 97, 93, 56, 57, 27, 26,
   149, 153, 45, 45, 60, 57, 45, 45, 100, 100, 56, 59,
   79, 77, 103, 104, 66, 67, 137, 133, 139, 136, 128, 126,
   66, 66, 176, 170, 99, 98, 8, 12, 18, 18, 145, 149,
   130, 128, 104, 105, 101, 102, 140, 135, 73, 75, 21, 21,
   55, 51, 22, 21, 105, 101, 37, 37, 78, 82, 61, 59,
   24, 20, 20, 20, 96, 95, 45, 44, 43, 43, 60, 60,
   76, 74, 10, 12, 9, 12, 11, 10, 10, 13, 9, 8,
   25, 25, 10, 8, 82, 80, 29, 30, 19, 10, 55, 56,
   36, 36, 39, 38, 40, 40, 40, 40, 41, 42, 40, 40,
   40, 42, 40, 40, 40, 41, 39, 40, 42, 41, 22, 25,
   32, 30, 45, 47, 44, 47, 62, 58, 89, 87, 54, 52,
   60, 62, 156, 153, 32, 58, 21, 19, 55, 65, 49, 50,
   45, 45, 125, 129, 21, 21, 68, 70, 113, 118, 109, 108,
   78, 75, 71, 73, 513, 506, 102, 101, 116, 109, 109, 119,
   149, 154, 123, 119, 64, 65, 125, 125, 177, 180, 28, 25,
   100, 53, 40, 38, 166, 156, 4, 4, 14, 21, 154, 155,
   143, 145, 164, 161, 80, 79, 4, 3, 28, 28, 24, 26,
   112, 112, 79, 89, 84, 85, 107, 105, 159, 166, 183, 176,
   26, 24, 43, 42, 161, 80, 52, 56, 34, 34, 51, 55,
   142, 144, 83, 82, 200, 200, 79, 81, 117, 119, 177, 176,
   116, 124, 82, 84, 84, 79, 78, 82, 371, 376, 93, 94,
   55, 50, 43, 39, 30, 30, 19, 18, 80, 85, 111, 116,
   129, 129, 122, 126, 40, 26, 104, 103, 9, 9, 20, 20,
   19, 10, 19, 20, 22, 21, 38, 36, 44, 112, 37, 36,
   53, 55, 60, 62, 206, 206, 192, 190, 60, 61, 85, 82,
   81, 165, 131, 135, 63, 29, 176, 183, 83, 82, 27, 24,
   156, 152, 79, 77, 295, 293, 34, 34, 76, 78, 46, 47,
   161, 153, 78, 76, 95, 102, 170, 171, 34, 77, 186, 196,
   178, 183, 52, 50, 47, 40, 72, 75, 54, 55, 57, 59,
   56, 58, 79, 77, 28, 25, 134, 126, 15, 14, 205, 205,
   240, 241, 109, 111, 260, 258, 300, 300, 213, 230)
 dropout <-
   c(7, 12, 17, 18, 30, 49, 25, 19, 35, 35, 17, 26, 52, 50,
   70, 75, 6, 8, 37, 35, 20, 23, 26, 21, 16, 36, 26, 21,
   40, 20, 5, 1, 39, 38, 9, 12, 32, 29, 2, 4, 3, 5,
   34, 59, 13, 21, 2, 6, 11, 9, 3, 4, 19, 28, 22, 27,
   6, 18, 6, 18, 2, 2, 10, 7, 25, 32, 18, 16, 26, 24,
   60, 53, 47, 48, 1, 0, 72, 81, 29, 32, 49, 47, 46, 43,
   7, 5, 28, 21, 16, 10, 16, 14, 14, 18, 67, 47, 12, 12,
   12, 8, 7, 15, 29, 22, 21, 24, 45, 48, 44, 56, 43, 40,
   21, 12, 9, 14, 6, 8, 3, 5, 13, 16, 17, 25, 60, 39,
   33, 43, 22, 26, 25, 32, 22, 27, 43, 36, 50, 49, 35, 22,
   45, 24, 72, 57, 46, 33, 70, 43, 28, 25, 32, 36, 1, 2,
   36, 26, 38, 24, 29, 35, 23, 21, 29, 31, 19, 15, 36, 26,
   25, 30, 10, 25, 14, 15, 31, 37, 21, 46, 22, 34, 1, 2,
   17, 15, 1, 2, 18, 19, 2, 3, 31, 68, 12, 21, 39, 40,
   24, 26, 7, 14, 35, 29, 11, 12, 8, 10, 4, 6, 19, 42,
   1, 5, 87, 95, 7, 7, 13, 12, 9, 11, 8, 10, 2, 2,
   40, 31, 10, 4, 11, 8, 16, 24, 5, 2, 7, 8, 16, 13,
   2, 3, 6, 16, 23, 20, 7, 5, 14, 10, 72, 65, 12, 9,
   14, 6, 7, 10, 7, 10, 12, 15, 5, 12, 14, 23, 0, 1,
   22, 23, 9, 6, 0, 0, 12, 11, 4, 3, 19, 10, 59, 49,
   7, 13, 5, 6, 8, 3, 18, 7, 5, 3, 7, 8, 30, 30,
   28, 31, 11, 9, 43, 56, 24, 20, 1, 3, 11, 8, 6, 4,
   70, 57, 33, 40, 48, 59, 22, 18, 3, 5, 23, 18, 17, 15,
   15, 11, 29, 33, 17, 19, 23, 26, 16, 25, 26, 23, 13, 11,
   15, 18, 26, 27, 17, 21, 18, 8, 30, 33, 29, 39, 30, 22,
   41, 32, 40, 39, 0, 1, 2, 3, 14, 18, 39, 47, 22, 19,
   24, 22, 39, 47, 9, 7, 0, 3, 26, 12, 5, 5, 28, 28,
   8, 2, 19, 20, 30, 33, 5, 3, 5, 3, 29, 36, 30, 26,
   15, 12, 14, 22, 24, 29, 1, 2, 1, 2, 3, 8, 4, 5,
   3, 3, 8, 10, 7, 4, 36, 26, 10, 10, 7, 4, 22, 25,
   15, 18, 18, 24, 19, 22, 21, 28, 22, 22, 14, 22, 11, 11,
   24, 33, 19, 23, 12, 9, 10, 12, 10, 5, 3, 6, 10, 15,
   10, 12, 12, 19, 25, 33, 9, 9, 7, 9, 31, 20, 11, 12,
   5, 4, 16, 23, 14, 15, 8, 10, 45, 41, 10, 14, 21, 27,
   2, 7, 29, 40, 34, 30, 15, 16, 103, 142, 40, 33, 41, 44,
   53, 59, 36, 44, 50, 45, 35, 37, 64, 40, 39, 53, 4, 5,
   44, 12, 15, 10, 38, 41, 1, 1, 8, 6, 45, 46, 43, 45,
   33, 48, 16, 22, 1, 1, 4, 16, 7, 5, 23, 26, 20, 20,
   8, 10, 37, 23, 53, 33, 44, 29, 6, 7, 15, 13, 79, 41,
   17, 31, 3, 4, 4, 10, 24, 23, 14, 16, 31, 41, 14, 17,
   37, 39, 26, 33, 14, 16, 13, 19, 25, 13, 11, 19, 87, 65,
   20, 19, 22, 15, 14, 12, 6, 4, 7, 3, 17, 22, 32, 32,
   27, 45, 35, 43, 15, 13, 37, 40, 3, 9, 2, 1, 9, 7,
   6, 6, 5, 2, 9, 11, 8, 28, 5, 4, 0, 5, 8, 5,
   62, 43, 47, 43, 22, 13, 24, 28, 28, 46, 55, 43, 13, 9,
   55, 59, 35, 26, 8, 6, 43, 37, 15, 24, 92, 95, 12, 6,
   29, 21, 11, 27, 40, 38, 23, 19, 26, 41, 46, 47, 13, 18,
   18, 29, 52, 47, 18, 18, 9, 10, 12, 16, 12, 12, 9, 8,
   10, 10, 19, 20, 12, 2, 50, 42, 3, 5, 53, 51, 47, 46,
   34, 25, 48, 50, 64, 56, 38, 62)
 ad.dropout <-
   data.frame(studyid = studyid.dropout, drug = drug.dropout,
   samplesize = samplesize.dropout, dropout = dropout)

 ss.ratio.dropout <-
   ad.dropout$samplesize[2*(1:max(ad.dropout$studyid))]/
   ad.dropout$samplesize[2*(1:max(ad.dropout$studyid)) - 1]
 ss.ratio.dropout[ss.ratio.dropout < 1] <-
   1/ss.ratio.dropout[ss.ratio.dropout < 1]
 pdf(file = "FigS7.pdf", width = 5, height = 5)
 par(mar = c(3, 3, 0, 0) + 0.1, mgp = c(2, 1, 0))
 hist(ss.ratio.dropout, xlab = "Sample size ratio", main = "")
 dev.off()
 sum(ss.ratio.dropout < 1.1)
 #[1] 279
 mean(ss.ratio.dropout < 1.1)
 #[1] 0.8040346

 ## data analysis
 n.study <- max(ad.dropout$studyid)
 n.study
 #[1] 347
 quantile(ad.dropout$samplesize, probs = c(0, 0.25, 0.5, 0.75, 1),
   type = 3)
 #  0%  25%  50%  75% 100%
 #   3   35   71  127  513
 quantile(ad.dropout$dropout, probs = c(0, 0.25, 0.5, 0.75, 1),
   type = 3)
 #  0%  25%  50%  75% 100%
 #   0    9   19   32  142
```

```
totsamplesize <- ad.dropout$samplesize[2*(1:347)] +
  ad.dropout$samplesize[2*(1:347) - 1]
quantile(totsamplesize, probs = c(0, 0.25, 0.5, 0.75, 1), type = 3)
#  0%  25%  50%  75% 100%
#   7   69  143  250 1019

ad.pval.Fisher <- ad.pval.chisq <- ad.pval.logOR <-
  ad.pval.logRR <- ad.pval.RD <- rep(NA, n.study)
ad.FI.Fisher.sig2nonsig <- ad.FI.Fisher.nonsig2sig <-
  ad.FI.chisq.sig2nonsig <- ad.FI.chisq.nonsig2sig <-
  ad.FI.logOR.sig2nonsig <- ad.FI.logOR.nonsig2sig <-
  ad.FI.logRR.sig2nonsig <- ad.FI.logRR.nonsig2sig <-
  ad.FI.RD.sig2nonsig <- ad.FI.RD.nonsig2sig <- rep(NA, n.study)
ad.FI0.Fisher.sig2nonsig <- ad.FI0.Fisher.nonsig2sig <-
  ad.FI0.chisq.sig2nonsig <- ad.FI0.chisq.nonsig2sig <-
  ad.FI0.logOR.sig2nonsig <- ad.FI0.logOR.nonsig2sig <-
  ad.FI0.logRR.sig2nonsig <- ad.FI0.logRR.nonsig2sig <-
  ad.FI0.RD.sig2nonsig <- ad.FI0.RD.nonsig2sig <- rep(NA, n.study)
ad.FI1.Fisher.sig2nonsig <- ad.FI1.Fisher.nonsig2sig <-
  ad.FI1.chisq.sig2nonsig <- ad.FI1.chisq.nonsig2sig <-
  ad.FI1.logOR.sig2nonsig <- ad.FI1.logOR.nonsig2sig <-
  ad.FI1.logRR.sig2nonsig <- ad.FI1.logRR.nonsig2sig <-
  ad.FI1.RD.sig2nonsig <- ad.FI1.RD.nonsig2sig <- rep(NA, n.study)

all.methods <- c("Fisher", "chisq", "logOR", "logRR", "RD")
begin <- Sys.time()
for(i in 1:n.study){
  dat.temp <- ad.dropout[ad.dropout$studyid == i,]
  e0.temp <- dat.temp$dropout[1]
  e1.temp <- dat.temp$dropout[2]
  n0.temp <- dat.temp$samplesize[1]
  n1.temp <- dat.temp$samplesize[2]

  rslt.temp <- fragility.iter(e0 = e0.temp, n0 = n0.temp,
    e1 = e1.temp, n1 = n1.temp, methods = all.methods,
    modify0 = "both", modify1 = "both", alpha = 0.05)
  ad.pval.Fisher[i] <- rslt.temp$pval.Fisher
  ad.pval.chisq[i] <- rslt.temp$pval.chisq
  ad.pval.logOR[i] <- rslt.temp$pval.logOR
  ad.pval.logRR[i] <- rslt.temp$pval.logRR
  ad.pval.RD[i] <- rslt.temp$pval.RD
  ad.FI.Fisher.sig2nonsig[i] <- rslt.temp$FI.Fisher.sig2nonsig
  ad.FI.Fisher.nonsig2sig[i] <- rslt.temp$FI.Fisher.nonsig2sig
  ad.FI.chisq.sig2nonsig[i] <- rslt.temp$FI.chisq.sig2nonsig
  ad.FI.chisq.nonsig2sig[i] <- rslt.temp$FI.chisq.nonsig2sig
  ad.FI.logOR.sig2nonsig[i] <- rslt.temp$FI.logOR.sig2nonsig
  ad.FI.logOR.nonsig2sig[i] <- rslt.temp$FI.logOR.nonsig2sig
  ad.FI.logRR.sig2nonsig[i] <- rslt.temp$FI.logRR.sig2nonsig
  ad.FI.logRR.nonsig2sig[i] <- rslt.temp$FI.logRR.nonsig2sig
  ad.FI.RD.sig2nonsig[i] <- rslt.temp$FI.RD.sig2nonsig
  ad.FI.RD.nonsig2sig[i] <- rslt.temp$FI.RD.nonsig2sig

  rslt0.temp <- fragility.iter(e0 = e0.temp, n0 = n0.temp,
    e1 = e1.temp, n1 = n1.temp, methods = all.methods,
    modify0 = "both", modify1 = "none", alpha = 0.05)
  ad.FI0.Fisher.sig2nonsig[i] <- rslt0.temp$FI.Fisher.sig2nonsig
  ad.FI0.Fisher.nonsig2sig[i] <- rslt0.temp$FI.Fisher.nonsig2sig
  ad.FI0.chisq.sig2nonsig[i] <- rslt0.temp$FI.chisq.sig2nonsig
  ad.FI0.chisq.nonsig2sig[i] <- rslt0.temp$FI.chisq.nonsig2sig
  ad.FI0.logOR.sig2nonsig[i] <- rslt0.temp$FI.logOR.sig2nonsig
  ad.FI0.logOR.nonsig2sig[i] <- rslt0.temp$FI.logOR.nonsig2sig
  ad.FI0.logRR.sig2nonsig[i] <- rslt0.temp$FI.logRR.sig2nonsig
  ad.FI0.logRR.nonsig2sig[i] <- rslt0.temp$FI.logRR.nonsig2sig
  ad.FI0.RD.sig2nonsig[i] <- rslt0.temp$FI.RD.sig2nonsig
  ad.FI0.RD.nonsig2sig[i] <- rslt0.temp$FI.RD.nonsig2sig

  rslt1.temp <- fragility.iter(e0 = e0.temp, n0 = n0.temp,
    e1 = e1.temp, n1 = n1.temp, methods = all.methods,
    modify0 = "none", modify1 = "both", alpha = 0.05)
  ad.FI1.Fisher.sig2nonsig[i] <- rslt1.temp$FI.Fisher.sig2nonsig
  ad.FI1.Fisher.nonsig2sig[i] <- rslt1.temp$FI.Fisher.nonsig2sig
  ad.FI1.chisq.sig2nonsig[i] <- rslt1.temp$FI.chisq.sig2nonsig
  ad.FI1.chisq.nonsig2sig[i] <- rslt1.temp$FI.chisq.nonsig2sig
  ad.FI1.logOR.sig2nonsig[i] <- rslt1.temp$FI.logOR.sig2nonsig
  ad.FI1.logOR.nonsig2sig[i] <- rslt1.temp$FI.logOR.nonsig2sig
  ad.FI1.logRR.sig2nonsig[i] <- rslt1.temp$FI.logRR.sig2nonsig
  ad.FI1.logRR.nonsig2sig[i] <- rslt1.temp$FI.logRR.nonsig2sig
  ad.FI1.RD.sig2nonsig[i] <- rslt1.temp$FI.RD.sig2nonsig
  ad.FI1.RD.nonsig2sig[i] <- rslt1.temp$FI.RD.nonsig2sig
}
end <- Sys.time()
end - begin
#Time difference of 8.763304 mins

## Summary
ad.FI.Fisher <- ad.FI0.Fisher <- ad.FI1.Fisher <-
  ad.FI.chisq <- ad.FI0.chisq <- ad.FI1.chisq <-
  ad.FI.logOR <- ad.FI0.logOR <- ad.FI1.logOR <-
  ad.FI.logRR <- ad.FI0.logRR <- ad.FI1.logRR <-
  ad.FI.RD <- ad.FI0.RD <- ad.FI1.RD <- rep(NA, n.study)
for(i in 1:n.study){
  ad.FI.Fisher[i] <- ifelse(ad.pval.Fisher[i] < 0.05,
    ad.FI.Fisher.sig2nonsig[i], ad.FI.Fisher.nonsig2sig[i])
  ad.FI0.Fisher[i] <- ifelse(ad.pval.Fisher[i] < 0.05,
    ad.FI0.Fisher.sig2nonsig[i], ad.FI0.Fisher.nonsig2sig[i])
  ad.FI1.Fisher[i] <- ifelse(ad.pval.Fisher[i] < 0.05,
    ad.FI1.Fisher.sig2nonsig[i], ad.FI1.Fisher.nonsig2sig[i])
  ad.FI.chisq[i] <- ifelse(ad.pval.chisq[i] < 0.05,
    ad.FI.chisq.sig2nonsig[i], ad.FI.chisq.nonsig2sig[i])
  ad.FI0.chisq[i] <- ifelse(ad.pval.chisq[i] < 0.05,
    ad.FI0.chisq.sig2nonsig[i], ad.FI0.chisq.nonsig2sig[i])
  ad.FI1.chisq[i] <- ifelse(ad.pval.chisq[i] < 0.05,
    ad.FI1.chisq.sig2nonsig[i], ad.FI1.chisq.nonsig2sig[i])
  ad.FI.logOR[i] <- ifelse(ad.pval.logOR[i] < 0.05,
    ad.FI.logOR.sig2nonsig[i], ad.FI.logOR.nonsig2sig[i])
  ad.FI0.logOR[i] <- ifelse(ad.pval.logOR[i] < 0.05,
    ad.FI0.logOR.sig2nonsig[i], ad.FI0.logOR.nonsig2sig[i])
  ad.FI1.logOR[i] <- ifelse(ad.pval.logOR[i] < 0.05,
    ad.FI1.logOR.sig2nonsig[i], ad.FI1.logOR.nonsig2sig[i])
  ad.FI.logRR[i] <- ifelse(ad.pval.logRR[i] < 0.05,
    ad.FI.logRR.sig2nonsig[i], ad.FI.logRR.nonsig2sig[i])
  ad.FI0.logRR[i] <- ifelse(ad.pval.logRR[i] < 0.05,
    ad.FI0.logRR.sig2nonsig[i], ad.FI0.logRR.nonsig2sig[i])
  ad.FI1.logRR[i] <- ifelse(ad.pval.logRR[i] < 0.05,
    ad.FI1.logRR.sig2nonsig[i], ad.FI1.logRR.nonsig2sig[i])
  ad.FI.RD[i] <- ifelse(ad.pval.RD[i] < 0.05,
    ad.FI.RD.sig2nonsig[i], ad.FI.RD.nonsig2sig[i])
  ad.FI0.RD[i] <- ifelse(ad.pval.RD[i] < 0.05,
    ad.FI0.RD.sig2nonsig[i], ad.FI0.RD.nonsig2sig[i])
  ad.FI1.RD[i] <- ifelse(ad.pval.RD[i] < 0.05,
    ad.FI1.RD.sig2nonsig[i], ad.FI1.RD.nonsig2sig[i])
}

ad.FI.all <- c(ad.FI.Fisher, ad.FI.chisq, ad.FI.logOR,
  ad.FI.logRR, ad.FI.RD)
max(ad.FI.all[!is.na(ad.FI.all)])
```

```
#[1] 19
max(c(table(ad.FI.Fisher), table(ad.FI.chisq),
  table(ad.FI.logOR), table(ad.FI.logRR), table(ad.FI.RD)))
#[1] 59

sum(is.na(ad.FI.Fisher))
#[1] 0
sum(ad.pval.Fisher < 0.05)
#[1] 32
table(ad.FI.Fisher.sig2nonsig[ad.pval.Fisher < 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 13
# 8  7  6  1  1  3  1  1  1  1  1  1
quantile(ad.FI.Fisher.sig2nonsig[ad.pval.Fisher < 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    1    3    6   13
sum(ad.pval.Fisher >= 0.05)
#[1] 315
table(ad.FI.Fisher.nonsig2sig[ad.pval.Fisher >= 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 19
#17 24 26 52 30 30 37 25 15 22  7  7  5  4  8  5  1
quantile(ad.FI.Fisher.nonsig2sig[ad.pval.Fisher >= 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    4    6    8   19
table(ad.FI.Fisher, useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 19
#25 31 32 53 31 33 38 26 16 23  8  7  6  4  8  5  1
quantile(ad.FI.Fisher, probs = c(0, 0.25, 0.5, 0.75, 1),
  type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    3    6    8   19

sum(is.na(ad.FI.chisq))
#[1] 1
sum(is.na(ad.FI.chisq[ad.pval.chisq >= 0.05]))
#[1] 1
sum(ad.pval.chisq < 0.05)
#[1] 29
table(ad.FI.chisq.sig2nonsig[ad.pval.chisq < 0.05], useNA = "no")
# 1  2  3  4  6  7  8 10 13
# 8  8  2  2  3  1  2  2  1
quantile(ad.FI.chisq.sig2nonsig[ad.pval.chisq < 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    1    2    6   13
sum(ad.pval.chisq >= 0.05)
#[1] 318
table(ad.FI.chisq.nonsig2sig[ad.pval.chisq >= 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 19
#13 21 24 36 45 31 32 29 19 19 16  7  4  5  5  9  1  1
quantile(ad.FI.chisq.nonsig2sig[!is.na(ad.FI.chisq.nonsig2sig)],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    4    6    9   19
table(ad.FI.chisq, useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 19
#21 29 26 38 45 34 33 31 19 21 16  7  5  5  5  9  1  1
quantile(ad.FI.chisq, probs = c(0, 0.25, 0.5, 0.75, 1),
  type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    4    6    9   19
```

```
sum(is.na(ad.FI.logOR))
#[1] 1
sum(is.na(ad.FI.logOR[ad.pval.logOR >= 0.05]))
#[1] 1
sum(ad.pval.logOR < 0.05)
#[1] 38
table(ad.FI.logOR.sig2nonsig[ad.pval.logOR < 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 11 14
#11  8  6  2  2  1  2  1  2  2  1
quantile(ad.FI.logOR.sig2nonsig[ad.pval.logOR < 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    1    2    5   14
sum(ad.pval.logOR >= 0.05)
#[1] 309
table(ad.FI.logOR.nonsig2sig[ad.pval.logOR >= 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 18
#16 19 27 38 38 45 39 15 20 18  7  4  5  5  9  2  1
quantile(ad.FI.logOR.nonsig2sig[ad.pval.logOR >= 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
# 0%  25%  50%  75% 100%
#   1    4    6    8   18
table(ad.FI.logOR, useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 18
#27 27 33 40 40 46 41 16 22 18  9  4  5  6  9  2  1
quantile(ad.FI.logOR, probs = c(0, 0.25, 0.5, 0.75, 1),
  type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    3    6    8   18

sum(is.na(ad.FI.logRR))
#[1] 2
sum(is.na(ad.FI.logRR[ad.pval.logRR >= 0.05]))
#[1] 2
sum(ad.pval.logRR < 0.05)
#[1] 36
table(ad.FI.logRR.sig2nonsig[ad.pval.logRR < 0.05], useNA = "no")
# 1  2  3  4  5  6  8  9 11 14
#11  8  4  2  2  3  1  2  2  1
quantile(ad.FI.logRR.sig2nonsig[ad.pval.logRR < 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    1    2    5   14
sum(ad.pval.logRR >= 0.05)
#[1] 311
table(ad.FI.logRR.nonsig2sig[ad.pval.logRR >= 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 19
#15 17 24 32 32 56 41 20 19 18  8  4  6  5  8  3  1
quantile(ad.FI.logRR.nonsig2sig[ad.pval.logRR >= 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    4    6    8   19
table(ad.FI.logRR, useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 19
#26 25 28 34 34 59 41 21 21 18 10  4  6  6  8  3  1
quantile(ad.FI.logRR, probs = c(0, 0.25, 0.5, 0.75, 1),
  type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    4    6    8   19

sum(is.na(ad.FI.RD))
#[1] 0
sum(ad.pval.RD < 0.05)
```

```
#[1] 42
table(ad.FI.RD.sig2nonsig[ad.pval.RD < 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 11 14
#13  7  8  3  1  1  3  2  1  2  1
quantile(ad.FI.RD.sig2nonsig[ad.pval.RD < 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    1    3    5   14
sum(ad.pval.RD >= 0.05)
#[1] 305
table(ad.FI.RD.nonsig2sig[ad.pval.RD >= 0.05], useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 18
#21 34 42 31 32 32 28 16 22 15  7  4  5  5  9  1  1
quantile(ad.FI.RD.nonsig2sig[ad.pval.RD >= 0.05],
  probs = c(0, 0.25, 0.5, 0.75, 1), type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    3    5    8   18
table(ad.FI.RD, useNA = "no")
# 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 18
#34 41 50 34 33 33 31 18 23 15  9  4  5  6  9  1  1
quantile(ad.FI.RD, probs = c(0, 0.25, 0.5, 0.75, 1),
  type = 3, na.rm = TRUE)
#  0%  25%  50%  75% 100%
#   1    3    5    8   18

## Figures: real data analysis for the dropout outcome
barFI <- function(ad.FI.sig2nonsig, ad.FI.nonsig2sig,
  main = ""){
  bar.FI <- matrix(NA, nrow = 2, ncol = 19)
  for(i in 1:19){
    bar.FI[1, i] <- table(ad.FI.nonsig2sig[!is.na(
      ad.FI.nonsig2sig)])[as.character(i)]
    bar.FI[2, i] <- table(ad.FI.sig2nonsig[!is.na(
      ad.FI.sig2nonsig)])[as.character(i)]
  }
  bar.FI[is.na(bar.FI)] <- 0
  ranges <- 1:19
  x <- barplot(bar.FI, names.arg = ranges, space = 0,
    col = c(adjustcolor("forestgreen", alpha.f = 0.1),
    adjustcolor("firebrick", alpha.f = 0.1)),
    las = 1, xaxt = "n", cex.axis = 1, ylim = c(0, 60),
    yaxt = "n")
  axis(side = 2, at = seq(0, 60, by = 10), labels = FALSE)
  text(cex = 1, x = -0.7, y = seq(0, 60, by = 10),
    labels = as.character(seq(0, 60, by = 10)),
    xpd = TRUE, srt = 0, pos = 2)
  title(xlab = "Fragility index", line = 1.5)
  title(ylab = "Frequency", line = 2)
  text(cex = 0.8, x = 1:19 - 0.5, y = 0,
    labels = as.character(1:19),
    xpd = TRUE, pos = 1, font = 1)
  mtext(text = main, side = 3, line = 0.5, outer = FALSE,
    cex = 1, font = 1)
}
## Figure S8A: Bar plot; Fisher's exact test
pdf(file = "FigS8A.pdf", width = 5, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
barFI(ad.FI.Fisher.sig2nonsig, ad.FI.Fisher.nonsig2sig,
  main = "(A) Fisher's exact test")
dev.off()
## Figure S8B: Bar plot; chi-squared test
pdf(file = "FigS8B.pdf", width = 5, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
```

```
barFI(ad.FI.chisq.sig2nonsig, ad.FI.chisq.nonsig2sig,
  main = "(B) Chi\255squared test")
dev.off()
## Figure S9A: Bar plot; log odds ratio
pdf(file = "FigS9A.pdf", width = 5, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
barFI(ad.FI.logOR.sig2nonsig, ad.FI.logOR.nonsig2sig,
  main = "(A) Log odds ratio")
dev.off()
## Figure S9B: Bar plot; log relative risk
pdf(file = "FigS9B.pdf", width = 5, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
barFI(ad.FI.logRR.sig2nonsig, ad.FI.logRR.nonsig2sig,
  main = "(B) Log relative risk")
dev.off()
## Figure S9C: Bar plot; risk difference
pdf(file = "FigS9C.pdf", width = 5, height = 5)
par(mar = c(3, 3, 2, 1) + 0.1)
barFI(ad.FI.RD.sig2nonsig, ad.FI.RD.nonsig2sig,
  main = "(C) Risk difference")
dev.off()

## counterexamples that the minimal event status
##  modifications may not be in a single treatment
##  arm; based on Fisher's exact test
which(ad.FI.Fisher < ad.FI0.Fisher & ad.FI.Fisher < ad.FI1.Fisher)
# [1]   7  16  77  91  93  95 112 117 120 138 152 178 179 193 200 201
#[17] 275 300 306
sum(ad.FI.Fisher < ad.FI0.Fisher & ad.FI.Fisher < ad.FI1.Fisher,
  na.rm = TRUE)
#[1] 19
max(pmin(ad.FI0.Fisher, ad.FI1.Fisher) - ad.FI.Fisher, na.rm = TRUE)
#[1] 3
which(is.na(ad.FI0.Fisher) & is.na(ad.FI1.Fisher))
#[1] 256 262
any(ad.FI.Fisher < ad.FI0.Fisher & ad.FI.Fisher < ad.FI1.Fisher &
  ad.pval.Fisher < 0.05, na.rm = TRUE)
#[1] FALSE

sum(ad.FI.chisq < ad.FI0.chisq & ad.FI.chisq < ad.FI1.chisq, na.rm = TRUE)
#[1] 23
any(ad.FI.chisq < ad.FI0.chisq & ad.FI.chisq < ad.FI1.chisq &
  ad.pval.chisq < 0.05, na.rm = TRUE)
#[1] FALSE
sum(ad.FI.logOR < ad.FI0.logOR & ad.FI.logOR < ad.FI1.logOR, na.rm = TRUE)
#[1] 10
any(ad.FI.logOR < ad.FI0.logOR & ad.FI.logOR < ad.FI1.logOR &
  ad.pval.logOR < 0.05, na.rm = TRUE)
#[1] FALSE
sum(ad.FI.logRR < ad.FI0.logRR & ad.FI.logRR < ad.FI1.logRR, na.rm = TRUE)
#[1] 12
any(ad.FI.logRR < ad.FI0.logRR & ad.FI.logRR < ad.FI1.logRR &
  ad.pval.logRR < 0.05, na.rm = TRUE)
#[1] FALSE
sum(ad.FI.RD < ad.FI0.RD & ad.FI.RD < ad.FI1.RD, na.rm = TRUE)
#[1] 11
any(ad.FI.RD < ad.FI0.RD & ad.FI.RD < ad.FI1.RD &
  ad.pval.RD < 0.05, na.rm = TRUE)
#[1] FALSE

## Figure S10: example
pdf(file = "FigS10.pdf", width = 5, height = 5)
i <- 138
```

```
dat.temp <- ad.dropout[ad.dropout$studyid == i,]
dat.temp
e0.temp <- dat.temp$dropout[1]
e1.temp <- dat.temp$dropout[2]
n0.temp <- dat.temp$samplesize[1]
n1.temp <- dat.temp$samplesize[2]

begin <- Sys.time()
rslt.countexam.all <- fragility.all(
  e0 = e0.temp, n0 = n0.temp,
  e1 = e1.temp, n1 = n1.temp, methods = all.methods,
  modify0 = "both", modify1 = "both", alpha = 0.05, save = TRUE)
end <- Sys.time()
end - begin
#Time difference of 1.268612 secs

rslt.countexam.all$FI.Fisher.nonsig2sig
#[1] 5
rslt.countexam.all$FI0.Fisher.nonsig2sig
#[1] 9
rslt.countexam.all$FI1.Fisher.nonsig2sig
#[1] 6

pvals.plot(rslt.countexam.all, method = "Fisher",
  modify0 = TRUE, modify1 = TRUE, trun = 10, cex.pts = 0.5,
  main = "Trial by Norton et al.",
  legend.pvals = c(0.01, 0.001), cex.legend.pvals = 0.6,
  mar = c(3, 3, 2, 1) + 0.1)
dev.off()

## average FI
ad.FI.Fisher.avg <- ad.FI.chisq.avg <- ad.FI.logOR.avg <-
  ad.FI.logRR.avg <- ad.FI.RD.avg <- rep(NA, n.study)
all.methods <- c("Fisher", "chisq", "logOR", "logRR", "RD")
begin <- Sys.time()
for(i in 1:n.study){
  dat.temp <- ad.dropout[ad.dropout$studyid == i,]
  e0.temp <- dat.temp$dropout[1]
  e1.temp <- dat.temp$dropout[2]
  n0.temp <- dat.temp$samplesize[1]
  n1.temp <- dat.temp$samplesize[2]

  rslt.temp <- frag.study.alpha(e0 = e0.temp, n0 = n0.temp,
    e1 = e1.temp, n1 = n1.temp, methods = all.methods,
    modify0 = "both", modify1 = "both",
    alpha.from = 0.005, alpha.to = 0.05, alpha.breaks = 100)
  ad.FI.Fisher.avg[i] <- rslt.temp$FI.avg["Fisher"]
  ad.FI.chisq.avg[i] <- rslt.temp$FI.avg["chisq"]
  ad.FI.logOR.avg[i] <- rslt.temp$FI.avg["logOR"]
  ad.FI.logRR.avg[i] <- rslt.temp$FI.avg["logRR"]
  ad.FI.RD.avg[i] <- rslt.temp$FI.avg["RD"]
}
end <- Sys.time()
end - begin
#Time difference of 1.576995 mins

sum(is.na(ad.FI.Fisher.avg))
#[1] 2
max(ad.FI.Fisher.avg, na.rm = TRUE)
#[1] 22.04
sum(is.na(ad.FI.chisq.avg))
#[1] 2
max(ad.FI.chisq.avg, na.rm = TRUE)
```

```
#[1] 22.49
sum(is.na(ad.FI.logOR.avg))
#[1] 3
max(ad.FI.logOR.avg, na.rm = TRUE)
#[1] 21.61
sum(is.na(ad.FI.logRR.avg))
#[1] 13
max(ad.FI.logRR.avg, na.rm = TRUE)
#[1] 21.76
sum(is.na(ad.FI.RD.avg))
#[1] 0
max(ad.FI.RD.avg, na.rm = TRUE)
#[1] 21.44

pdf(file = "FigS11A.pdf", width = 5, height = 5)
par(mar = c(3, 3, 1, 0) + 0.1, mgp = c(2, 1, 0))
hist(ad.FI.Fisher.avg, xlim = c(0, 25), ylim = c(0, 60),
  breaks = 1:23, main = "(A) Fisher's exact test", font.main = 1,
  xlab = "Average fragility index")
dev.off()
pdf(file = "FigS11B.pdf", width = 5, height = 5)
par(mar = c(3, 3, 1, 0) + 0.1, mgp = c(2, 1, 0))
hist(ad.FI.chisq.avg, xlim = c(0, 25), ylim = c(0, 60),
  breaks = 1:23, main = "(B) Chi\255squared test", font.main = 1,
  xlab = "Average fragility index")
dev.off()
pdf(file = "FigS12A.pdf", width = 5, height = 5)
par(mar = c(3, 3, 1, 0) + 0.1, mgp = c(2, 1, 0))
hist(ad.FI.logOR.avg, xlim = c(0, 25), ylim = c(0, 60),
  breaks = 1:23, main = "(A) Log odds ratio", font.main = 1,
  xlab = "Average fragility index")
dev.off()
pdf(file = "FigS12B.pdf", width = 5, height = 5)
par(mar = c(3, 3, 1, 0) + 0.1, mgp = c(2, 1, 0))
hist(ad.FI.logRR.avg, xlim = c(0, 25), ylim = c(0, 60),
  breaks = 1:23, main = "(B) Log relative risk", font.main = 1,
  xlab = "Average fragility index")
dev.off()
pdf(file = "FigS12C.pdf", width = 5, height = 5)
par(mar = c(3, 3, 1, 0) + 0.1, mgp = c(2, 1, 0))
hist(ad.FI.RD.avg, xlim = c(0, 25), ylim = c(0, 60),
  breaks = 1:23, main = "(C) Risk difference", font.main = 1,
  xlab = "Average fragility index")
dev.off()
```