```
# ------------------------------------------------------------------------
# Immobility test v 1.0
# Based on Woronow and Love (1990) and Schedl (1998)
#
# Woronow, A. and Love, K.M., 1990, Quantifying and testing differences
#   among means of compositional data suites. Math. Geol., 22, 837-852.
# Schedl, A., 1998, Log ratio methods for establishing a reference frame
#   for chemical change. Jour. Geol., 106, 211-228.
#
# January 9, 2006
# Developed by Hiroyoshi Arai
# ------------------------------------------------------------------------


immobile <- function(group1, group2, s.level=0.05, sep="/") {

  if ((s.level <= 0) || (1 <= s.level)) {
    stop("Incorrect significance level.")
  }

  if (!is.data.frame(group1) || !is.data.frame(group2)) {
    stop("Objects are not data frames.")
  }

  pct1 <- group1/apply(group1, 1, sum)*100
  pct2 <- group2/apply(group2, 1, sum)*100

  if (ncol(pct1) != ncol(pct2)) {
    stop("The number of columns is different between groups.")
  }

  if (any(colnames(pct1) != colnames(pct2))) {
    stop("The column names are different between groups.")
  }

  if (any(is.element(c(0, NA), as.matrix(pct1)), is.element(c(0, NA), as.matrix(pct2)))) {
    stop("Data contain invalid (0 and/or null) value(s).")
  }

  nelem <- ncol(pct1)
  npairs <- choose(nelem, 2)
  logratio <- array(NA, dim=c(max(nrow(pct1), nrow(pct2)), npairs, 2))
  zratio <- array(NA, dim=c(max(nrow(pct1), nrow(pct2)), npairs, 2))
  elempairs <- data.frame(matrix(NA, nrow=npairs, ncol=12))


  # ----- Calculating logratios -----

  cat(date(), ":", "Calculating logratios...\n")
  pairnames <- NULL

  k <- 1
  for (i in 1:(nelem-1)) {
    for (j in (i+1):nelem) {
      logratio[1:nrow(pct1), k, 1] <- log(pct1[ , j]/pct1[ , i])
      logratio[1:nrow(pct2), k, 2] <- log(pct2[ , j]/pct2[ , i])
      zratio[1:nrow(pct1), k, 1] <- log((pct1[ , j]+pct1[ , i])/(100-(pct1[ , j]+pct1[ , i])))
      zratio[1:nrow(pct2), k, 2] <- log((pct2[ , j]+pct2[ , i])/(100-(pct2[ , j]+pct2[ , i])))
      elempairs[k, ] <- c(k, j, i, NA, rep(0, 8))
      pairnames <- c(pairnames, paste(colnames(pct1[j]), colnames(pct1[i]), sep=sep))
      k <- k+1
    }
  }

  colnames(elempairs) <- c("pair.no", "numerator", "denominator", "adj.r2", "r2.rank",
                           "no.cor", "no.art", "test1", "test2", "test3a", "test3b", "score")
  rownames(elempairs) <- pairnames


  # ----- Modified Test 1 -----

  cat(date(), ":", "Now running Modified Test 1 for same distribution...\n")
  shapiro1 <- apply(logratio[ ,, 1], 2, shapiro.test)
  shapiro2 <- apply(logratio[ ,, 2], 2, shapiro.test)

  for (i in 1:npairs) {
    if ((shapiro1[[i]]$p.value >= s.level) && (shapiro2[[i]]$p.value >= s.level)) {
      f.result <- var.test(logratio[ , i, 1], logratio[ , i, 2])
      if (f.result$p.value >= s.level) {
        t.result <- t.test(logratio[ , i, 1], logratio[ , i, 2], var.equal=TRUE)
        if (t.result$p.value >= s.level) {
          elempairs$test1[i] <- 1
        }
      }
    } else {
      ks.result <- ks.test(logratio[ , i, 1], logratio[ , i, 2])
      if (ks.result$p.value >= s.level) {
        w.result <- wilcox.test(logratio[ , i, 1], logratio[ , i, 2])
        if (w.result$p.value >= s.level) {
          elempairs$test1[i] <- 1
        }
      }
    }
  }


  # ----- Test 2 -----

  cat(date(), ":", "Now running Test 2 for subcompositional invariance...\n")

  for (i in 1:npairs) {
    cor.result <- cor.test(c(logratio[ , i, ]), c(zratio[ , i, ]))
    if (cor.result$p.value >= s.level) {
      elempairs$test2[i] <- 1
    }
  }


  # ----- Test 3A -----

  cat(date(), ":", "Now running Test 3A for subcompositional independence...\n")
  multi <- data.frame(matrix(NA, nrow=max(nrow(pct1), nrow(pct2))*2, ncol=nelem-2))
  attach(elempairs)
  on.exit(detach(elempairs))

  for (i in 1:npairs) {
    elem.no <- setdiff(1:nelem, elempairs[i, 2:3])
    ex.numerator <- pair.no[(numerator != numerator[i]) & (numerator != denominator[i])]
    ex.denominator <- pair.no[(denominator != numerator[i]) & (denominator != denominator[i])]
    maximum.no <- pair.no[(numerator == max(elem.no)) | (denominator == max(elem.no))]
    explanatory <- intersect(intersect(ex.numerator, ex.denominator), maximum.no)
    multi[ , 1] <- c(logratio[ , i, ])
    for (j in 1:length(explanatory)) {
      multi[ , (j+1)] <- c(logratio[ , explanatory[j], ])
    }
    lm1 <- summary(lm(X1 ~ ., data=multi))
    elempairs$adj.r2[i] <- lm1$adj.r.squared
    fvalues <- lm1$fstatistic
    if (pf(fvalues[1], fvalues[2], fvalues[3], lower.tail=FALSE) >= s.level) {
      elempairs$test3a[i] <- 1
    }
  }

  if (!is.element(1, elempairs$test3a)) {
    first.qtl <- quantile(elempairs$adj.r2, prob=0.25, names=FALSE)
    r2.order <- order(elempairs$adj.r2[elempairs$adj.r2 <= first.qtl])
    cand.test3a <- elempairs$pair.no[elempairs$adj.r2 <= first.qtl][r2.order]
  }

  elempairs$r2.rank <- rank(elempairs$adj.r2)


  # ----- Test 3B -----

  cat(date(), ":", "Now running Test 3B for subcompositional independence...\n")
  critical.pois <- qpois((1-s.level), s.level*(npairs-1))
  msg3b <- NULL

  for (i in 1:npairs) {
    for (j in 1:npairs) {
      if (j != i) {
        cor.combined <- cor.test(c(logratio[ , i, ]), c(logratio[ , j, ]))
        if (cor.combined$p.value < s.level) {
          elempairs$no.cor[i] <- elempairs$no.cor[i]+1
          cor.group1 <- cor.test(logratio[ , i, 1], logratio[ , j, 1])
          cor.sign <- cor.combined$estimate*cor.group1$estimate
          if ((cor.group1$p.value < s.level) && (cor.sign > 0)) {
            elempairs$no.art[i] <- elempairs$no.art[i]+1
          }
        }
      }
    }
    if (elempairs$no.cor[i] <= critical.pois) {
      elempairs$test3b[i] <- 1
    }
  }

  if (length(pair.no[(elempairs$test1 == 1) & (elempairs$test3b == 1)]) == 0) {
    elempairs$test3b[(elempairs$no.cor-elempairs$no.art) <= critical.pois] <- 1
    if (is.element(1, elempairs$test3b)) {
      msg3b <- "The correction for correlations in group1 has been done.\n"
    }
  }


  # ----- Calculating variance ratio & log-variance -----

  cat(date(), ":", "Calculating variance ratio and lowest log-variance...\n")
  var.ratio <- data.frame(matrix(NA, nrow=nelem, ncol=nelem+1))
  log.var <- data.frame(matrix(NA, nrow=nelem, ncol=nelem))
  colnames(var.ratio) <- c(colnames(pct1), ">=1")
  rownames(var.ratio) <- colnames(pct1)
  colnames(log.var) <- colnames(pct1)
  rownames(log.var) <- paste("1/", colnames(pct1), sep="")

  for (i in 1:nelem) {
    ratio1 <- pct1/pct1[ , i]
    ratio2 <- pct2/pct2[ , i]
    var1 <- apply(ratio1, 2, var)
    var2 <- apply(ratio2, 2, var)
    var.ratio[i, ] <- c(var2/var1, 0)
    var.ratio[i, i] <- NA
    for (j in 1:nelem) {
      if (j != i) {
        selected <- pair.no[((numerator == i) & (denominator == j)) |
                            ((numerator == j) & (denominator == i))]
        log.var[i, j] <- var(logratio[ , selected, 1], na.rm=TRUE)
        if (var.ratio[i, j] >= 1) {
          var.ratio[i, nelem+1] <- var.ratio[i, nelem+1]+1
        }
      }
    }
  }


  # ----- Result -----

  cat("\n\nResult of statistical tests for estimating immobile elemants.\n")
  cat("Date: ", date(), "\n", sep="")

  groupname <- c(deparse(substitute(group1)), deparse(substitute(group2)))

  cat("Examined data (group1): ", groupname[1], "\n", sep="")
  cat("Examined data (group2): ", groupname[2], "\n", sep="")
  cat("Significance level: ", s.level*100, "%\n\n\n", sep="")
  elempairs$score <- elempairs$test1+elempairs$test2+elempairs$test3a+elempairs$test3b
  testnames <- c("Modified Test 1", "Test 2", "Test 3A", "Test 3B")

  for (i in 1:4) {
    if (is.element(1, elempairs[ , i+7])) {
      cat("The element-pairs which passed", testnames[i], "are:\n")
      print(pairnames[elempairs$pair.no[elempairs[ ,i+7] == 1]])
      if ((i == 4) && (!is.null(msg3b))) {
        cat(msg3b)
      }
      cat("\n\n")
    } else {
      cat("No element-pair passed ", testnames[i], ".\n\n\n", sep="")
      if (i == 3) {
        cat("All the element-pairs failed an overall F-test for Test 3A.\n",
            "Following pairs (adj.R2 <=1st. quartile) are the candidates.\n",
            "Compare these candidates with the pairs which passed the other tests.\n", sep="")
        print(pairnames[cand.test3a])
        cat("\n\n")
      }
    }
  }

  if (is.element(4, elempairs$score)) {
    cat("The element-pairs which passed all the tests are:\n")
    print(pairnames[elempairs$pair.no[elempairs$score == 4]])
    cat("\n\n")
  } else {
    cat("No element-pair passed all the tests.\n\n")
  }

  elempairs <- elempairs[ , -(2:3)]
  return(invisible(list(elempairs=elempairs, var.ratio=var.ratio, log.var=log.var)))

}

# ----- End of program -----
```