Package 'Fuzzy.p.value'

October 12, 2022

Type Package

Title Computing Fuzzy p-Value

Version 1.1

Date 2017-09-01

Author Abbas Parchami (Department of Statistics, Faculty of Mathematics and Computer, Shahid Bahonar University of Kerman, Kerman, Iran)

Maintainer Abbas Parchami <parchami@uk.ac.ir>

Description The main goal of this package is drawing the membership function of the fuzzy p-value which is defined as a fuzzy set on the unit interval for three following problems: (1) testing crisp hypotheses based on fuzzy data, (2) testing fuzzy hypotheses based on crisp data, and (3) testing fuzzy hypotheses based on fuzzy data. In all cases, the fuzziness of data or/and the fuzziness of the boundary of null fuzzy hypothesis transported via the p-value function and causes to produce the fuzzy p-value. If the p-value is fuzzy, it is more appropriate to consider a fuzzy significance level for the problem. Therefore, the comparison of the fuzzy p-value and the fuzzy significance level is evaluated by a fuzzy ranking method in this package.

License LGPL (>= 3)

Imports FuzzyNumbers

NeedsCompilation no

Repository CRAN

Date/Publication 2017-09-02 04:19:39 UTC

R topics documented:

Fuzzy.p.value-package	*Computing Fuzzy p-Value*

Description

Statistical testing hypotheses has an important rule for making decision in practical and applied problems. In traditional testing methods, data, parameters, hypotheses and other elements of problem are consider crisp. But in applied sciences such as economics, agriculture and social sciences, it may be confront with vague definitions and fuzzy concepts. In such situations, the classical methods can not solve the vague test and they need to generalized for using in fuzzy environments. The vagueness entrance in testing hypotheses problem can be done via data or/and hypotheses. Therefore, the following three major problems can be usually considered for a fuzzy environment: (1) testing crisp hypotheses based on fuzzy data, (2) testing fuzzy hypotheses based on crisp data, and (3) testing fuzzy hypotheses based on fuzzy data. Similar to the classical testing hypotheses, one can consider different procedure methods for solving the above mentioned problems such as Neyman-Pearson, Bayes, likelihood ratio, minimax and p-value. Computing Fuzzy p-Value package, i.e. `Fuzzy.p.value` package, is an open source (LGPL 3) package for R which investigate on the above three problems on the basis of fuzzy p-value approach. All formulas and given examples are match with Springer Chapter book (2016) to easily show the performance of the proposed methods.

Details

In this package, We consider only two distributions Normal and Poisson as two suitable candidates from the class of all continues and discrete distributions. This program extendable for other distributions which users can do it if needed.

Author(s)

Abbas Parchami

Maintainer: Abbas Parchami <parchami@uk.ac.ir>

References

Filzmoser, P., and Viertl, R. (2004). Testing hypotheses with fuzzy data: the fuzzy p-value. Metrika 59: 21-29.

Holena, M. (2004). Fuzzy hypotheses testing in a framework of fuzzy logic. Fuzzy Sets and Systems 145: 229-252.

Parchami, A., Taheri, S. M., and Mashinchi, M. (2010). Fuzzy p-value in testing fuzzy hypotheses with crisp data. Statistical Papers 51: 209-226.

Parchami, A., Taheri, S. M., and Mashinchi, M. (2012). Testing fuzzy hypotheses based on vague observations: a p-value approach. Statistical Papers 53: 469-484.

Wang, X., Kerre, E. E. (2001). Reasonable properties for the ordering of fuzzy quantities (II). Fuzzy Sets and Systems 118: 387-405.

Yuan, Y. (1991). Criteria for evaluating fuzzy ranking methods. Fuzzy Sets Syst 43: 139-157.

See Also

FuzzyNumbers

| B | *Class of linear fuzzy sets to construct concept "approximately bigger"* |

Description

The are several famous fuzzy sets which have meaningful means and are user friendly for introducing. Several linear functions are constructed in this package for easily inspiration to the fuzzy concepts $\ll approximately\,smaller \gg$, $\ll approximately\,bigger \gg$ and $\ll approximately\,equal \gg$. For example, function B is one of them which users can use it to construct the fuzzy concept $\ll approximately\,bigger \gg$ in fuzzy hypotheses. As presented bellow, the membership function of fuzzy set B has two parameters a and b:

$$B(a,b)(x) = \begin{cases} 1 & if & b \leq x \\ \frac{x-a}{b-a} & if & a \leq x < b \\ 0 & if & x < a \end{cases}$$

Usage

```
B(a, b)
```

Arguments

a   The first parameter of fuzzy set $B(a,b)$. For better understanding, try to draw the above membership function.

b   The second parameter of fuzzy set $B(a,b)$. For better understanding, try to draw the above membership function.

Value

This function convert a fuzzy hypothesis to its boundary which is a triangular fuzzy number. Therefore, although the input of this function is not triangular fuzzy number, but its output is triangular.

References

Parchami, A., Taheri, S. M., and Mashinchi, M. (2010). Fuzzy p-value in testing fuzzy hypotheses with crisp data. Statistical Papers 51: 209-226.

Parchami, A., Taheri, S. M., and Mashinchi, M. (2012). Testing fuzzy hypotheses based on vague observations: a p-value approach. Statistical Papers 53: 469-484.

Examples

```
# Introducing the fuzzy null hypothesis H0
H0 = B(1275,1325)


## The function is currently defined as
function (a, b)
(TriangularFuzzyNumber(a, b, b))
```

---

p_value.norm   *Testing hypotheses based on fuzzy p-value for Normal distribution*

Description

Function `p_value.norm` can draw the membership function of fuzzy p-value for the following three
major problems which can be usually considered for the following tests in a fuzzy environment: (1)
testing crisp hypotheses based on fuzzy data, (2) testing fuzzy hypotheses based on crisp data, and
(3) testing fuzzy hypotheses based on fuzzy data. Also, one can consider a fuzzy significance of
level for test by function `p_value.norm`. It must be noted that function `p_value.norm` is applied
when the test statistics has Normal distribution. This idea can be extended for other test statistics
which has other continue distribution. All needed formulas are presented in Springer chapter book
2016 and the considered examples are match with this reference.

Usage

```
p_value.norm(kind, H0, H1, t, s.d, n, sig)
```

Arguments

kind   The kind of testing hypotheses is one of three numbers 0, 1 and 2, which is
                distinguishable from the form of alternative (fuzzy/non-fuzzy) hypothesis. Set
                $kind = 0$, when the kind of alternative hypothesis H1 is non-equality (i.e., $\neq$).
                Set $kind = 1$, when the kind of H1 show "smaller than" for the parameter (i.e.,
                $<$). Set $kind = 2$, when the kind of H1 show "bigger than" for the parameter
                (i.e., $>$).

H0   The null hypothesis of the test which must introduce only via functions T, Tr, S
                and B.

H1   The alternative hypothesis of the test which must introduce only via functions T,
                Tr, S and B.

t   The observed value of the test statistic (the observed value of the mean), which
                must introduce only via functions T and Tr. Note that if it is precise and crisp
                the user must indicate it by special triangular or trapezoidal fuzzy numbers, for
                example such as $t = T(23, 23, 23)$

s.d   The standard deviation of the normal data.

n   Sample size

sig	The significance of level for the test which must introduce only via functions T and Tr. Therefore, if it is precise (crisp) it must be indicated by special triangular or trapezoidal fuzzy numbers, for example such as $sig = Tr(.05, .05, .05, .05)$

Details

In this package, the users must introduce $H0$, $H1$, t and significance level only by fuzzy numbers T, Tr, S or B. Therefore, if each of them are precise (crisp), user must introduce it by special fuzzy sets (i.e. by indicator functions). For example the crisp null hypothesis "$H_0 : \mu > 0$" must be introduced by $H0 = B(0,0)$ to function p_value.norm. For another example, if the observed value of test statistics is crisp number $t = 4.1$, it must only be introduced by $t = T(4.1, 4.1, 4.1)$ or $t = Tr(4.1, 4.1, 4.1, 4.1)$ to the function p_value.norm.

Value

The function p_value.norm is constructed for hypotheses testing in fuzzy environment where the test statistics has Normal distributions and the parameter which discussed by two hypotheses is the mean of Normal. The major target and the duty of function p_value.norm is: (1) draw the membership functions of p-value and significance level for a Normal test, (2) compute numerical measures of Δ_{SP} and Δ_{PS} (see Parchami and Mashinchi (2016) for formulas), (3) decide to accept null hypothesis or accept alternative hypothesis in the test, and finally (4) compute the degree of acceptance for the accepted hypothesis for a Normal test.

Author(s)

Abbas Parchami

References

Filzmoser, P., and Viertl, R. (2004). Testing hypotheses with fuzzy data: the fuzzy p-value. Metrika 59: 21-29.

Holena, M. (2004). Fuzzy hypotheses testing in a framework of fuzzy logic. Fuzzy Sets and Systems 145: 229-252.

Parchami, A., Taheri, S. M., and Mashinchi, M. (2010). Fuzzy p-value in testing fuzzy hypotheses with crisp data. Statistical Papers 51: 209-226.

Parchami, A., Taheri, S. M., and Mashinchi, M. (2012). Testing fuzzy hypotheses based on vague observations: a p-value approach. Statistical Papers 53: 469-484.

Wang, X., Kerre, E. E. (2001). Reasonable properties for the ordering of fuzzy quantities (II). Fuzzy Sets and Systems 118: 387-405.

Yuan, Y. (1991). Criteria for evaluating fuzzy ranking methods. Fuzzy Sets Syst 43: 139-157.

Examples

```
# Example 1 from Springer chapter book 2016
p_value.norm( kind=2, H0=T(1300,1300,1300), H1=B(1300,1300), t=T(1288,1313,1331), s.d=30,
n=10, sig=T(0,0.05,0.1) )

# Solving Example 2 from Springer chapter book 2016
```

```
p_value.norm( kind=0, H0=T(1300,1300,1300), H1=B(1300,1300), t=T(1300,1313,1321), s.d=30,
n=10, sig=T(0,0.15,0.3) )

# Solving Example 4 from Springer chapter book 2016
H0 = T(1275,1300,1325) #  Fuzzy null hypothesis
H1 = B(1275,1325) #  Fuzzy althernative hypothesis
t = T(1315,1327,1342) #  Fuzzy observed value for the test statistics
sig = T(0.0,0.15,0.3) #  Fuzzy significance level
p_value.norm( kind=2, H0, H1, t, s.d=120, n=36, sig)

# Solving Example 6 from Springer chapter book 2016
H0 = T(1275,1300,1325)
H1 = B(1275,1325)
t = T(1327,1327,1327) #  Crisp observed value for the test statistics
sig = T(0, .05, .1)
p_value.norm( kind=2, H0, H1, t, s.d=120, n=36, sig)


## The function is currently defined as
function (kind, H0, H1, t, s.d, n, sig)
{
    alpha_L = seq(0, 1, 0.01)
    alpha_U = seq(1, 0, -0.01)
    if (kind == 1) {
        p_L = pnorm(alphacut(t, alpha_L)[, "U"], alphacut(H0,
            alpha_L)[, "L"], s.d/(n^0.5))
        p_U = pnorm(alphacut(t, alpha_U)[, "L"], alphacut(H0,
            alpha_U)[, "U"], s.d/(n^0.5))
    }
    else if (kind == 2) {
        p_L = 1 - pnorm(alphacut(t, alpha_L)[, "U"], alphacut(H0,
            alpha_L)[, "L"], s.d/(n^0.5))
        p_U = 1 - pnorm(alphacut(t, alpha_U)[, "L"], alphacut(H0,
            alpha_U)[, "U"], s.d/(n^0.5))
    }
    else if (supp(t)[1] >= supp(H0)[2]) {
        p_L = 2 * (1 - pnorm(alphacut(t, alpha_L)[, "U"], alphacut(H0,
            alpha_L)[, "L"], s.d/(n^0.5)))
        p_U = 2 * (1 - pnorm(alphacut(t, alpha_U)[, "L"], alphacut(H0,
            alpha_U)[, "U"], s.d/(n^0.5)))
    }
    else if (supp(t)[2] <= supp(H0)[1]) {
        p_L = 2 * pnorm(alphacut(t, alpha_L)[, "U"], alphacut(H0,
            alpha_L)[, "L"], s.d/(n^0.5))
        p_U = 2 * pnorm(alphacut(t, alpha_U)[, "L"], alphacut(H0,
            alpha_U)[, "U"], s.d/(n^0.5))
    }
    else {
        return("The fuzzy p-value can not defined for this example, since the fuzziness of
the problem is very high. This case may be accured for the testing equality
fuzzy null hypothesis, against the non-equality fuzzy althenative hypothesis")
    }
    alpha = cbind(alpha_L, alpha_U)
```

```
    p = cbind(p_L, p_U)
    plot(p, alpha, type = "l", lwd = 3, col = 1, xlim = c(-0.04,
        1.02))
     s1 = supp(sig)[1]
     s2 = core(sig)[1]
     s3 = core(sig)[1]
     s4 = supp(sig)[2]
     lines( c(0,s1,s2,s3,s4,1), c(0,0,1,1,0,0), type='l', lty=3, lwd=2, col=2 )
# plot(sig, lty=3, lwd=2, col=2, add=TRUE) # Conflict importing 2 plot from different packages
    legend("topright", c("Fuzzy p-value", "Significance level"),
        col = c(1, 2), text.col = 1, lwd = c(3, 2), lty = c(1,
            3))
    if( class(sig) == "numeric" ){
            sig <- TriangularFuzzyNumber(sig, sig, sig)
            }
    P_L = p_L
    P_U = p_U
    knot.n = 100
    S_L = alphacut(sig, round(seq(0, 1, .01), 5))[,"L"]
    S_U = alphacut(sig, round(seq(0, 1, .01), 5))[,"U"]
    Int1 = ( P_U - S_L ) * ( P_U > S_L )
    Int2 = ( P_L - S_U ) * ( P_L > S_U )
    Arz = 1 / (knot.n - 1)  #Arze Mostatilha baraye mohasebe-ye Integral
    Integral1 <- ( sum( Int1 ) - Int1[1]/2 - Int1[length(Int1)]/2 ) *Arz
    Integral2 <- ( sum( Int2 ) - Int2[1]/2 - Int2[length(Int2)]/2 ) *Arz
    Delta_PS = Integral1 + Integral2
    Int3 = ( S_U - P_L ) * ( S_U > P_L )
    Int4 = ( S_L - P_U ) * ( S_L > P_U )
    Integral3 <- ( sum( Int3 ) - Int3[1]/2 - Int3[length(Int3)]/2 ) *Arz
    Integral4 <- ( sum( Int4 ) - Int4[1]/2 - Int4[length(Int4)]/2 ) *Arz
    Delta_SP = Integral3 + Integral4
    print("Delta_SP = ")
    print(Delta_SP)
    print("Delta_PS = ")
    print(Delta_PS)
    Degree_P_biger_than_S = Delta_PS/(Delta_PS + Delta_SP)
    Degree_S_biger_than_P = 1 - Degree_P_biger_than_S
    if (Degree_P_biger_than_S >= Degree_S_biger_than_P) {
        a = "The null hypothesis (H0) is accepted with degree D(P>S)="
        b = round(Degree_P_biger_than_S, 4)
        c = ", at  the considered significance level."
        noquote(sprintf("%s %s %s ", a, b, c))
    }
    else {
        if (Degree_P_biger_than_S < Degree_S_biger_than_P) {
            a = "The alternative hypothesis (H1) is accepted with degree D(S>P)="
            b = round(Degree_S_biger_than_P, 4)
            c = ", at  the considered significance level."
            noquote(sprintf("%s %s %s ", a, b, c))
        }
        else {
            return(noquote(paste0("Impossible case")))
        }
```

```
    }
  }
```

---

p_value.pois   *Testing hypotheses based on fuzzy p-value for Poisson distribution*

Description

Function p_value.pois can draw the membership function of fuzzy p-value for the following three major problems which can be usually considered for the following tests in a fuzzy environment: (1) testing crisp hypotheses based on fuzzy data, (2) testing fuzzy hypotheses based on crisp data, and (3) testing fuzzy hypotheses based on fuzzy data. Also, one can consider a fuzzy significance of level for each of above tests by function p_value.pois. It must be noted that function p_value.pois is applied when the test statistics has Poisson distribution. This idea can be extended for other test statistics which has other discrete distribution. All needed formulas are presented in Springer Chapter book 2016 and the considered examples are match with this reference for the readers.

Usage

```
p_value.pois(kind, H0, H1, t, n, sig)
```

Arguments

kind   The kind of testing hypotheses is one of three numbers 0, 1 and 2, which is distinguishable from the form of alternative (fuzzy/non-fuzzy) hypothesis. Set $kind = 0$, when the kind of alternative hypothesis H1 is non-equality (i.e., $\neq$). Set $kind = 1$, when the kind of H1 show "smaller than" for the parameter (i.e., $<$). Set $kind = 2$, when the kind of H1 show "bigger than" for the parameter (i.e., $>$).

H0   The null hypothesis of the test which must introduce only via functions T, Tr, S and B.

H1   The alternative hypothesis of the test which must introduce only via functions T, Tr, S and B.

t   The observed value of the test statistic (the observed value of the mean), which must introduce only via functions T and Tr. Note that if it is precise and crisp the user must indicate it by special triangular or trapezoidal fuzzy numbers, for example such as $t = T(23, 23, 23)$

n   Sample size

sig   The significance of level for the test which must introduce only via functions T and Tr. Therefore, if it is precise (crisp) it must be indicated by special triangular or trapezoidal fuzzy numbers, for example such as $sig = T(0.1, 0.1, 0.1)$

Details

In this package, the users must introduce $H0$, $H1$, t and significance level only by fuzzy numbers `T`, `Tr`, `S` or `B`. Therefore, if each of them are precise (crisp), user must introduce it by special fuzzy sets (i.e. by indicator functions). For example the crisp null hypothesis "$H_0 : \lambda > 5.2$" must be introduced by $H0 = B(5.2, 5.2)$ to function `p_value.pois`.

Value

The function `p_value.pois` is constructed for hypotheses testing in fuzzy environment where the test statistics has Poisson distributions and the parameter which discussed by two hypotheses is the mean of Poisson. The major duty of function `p_value.pois` is: (1) draw the membership functions of p-value and significance level for a Poisson test, (2) compute numerical measures of Δ_{SP} and Δ_{PS} (see Parchami and Mashinchi (2016) for formulas), (3) decide to accept null hypothesis or accept alternative hypothesis in the test, and finally (4) compute the degree of acceptance for the accepted hypothesis for a Poisson test.

Author(s)

Abbas Parchami

References

Filzmoser, P., and Viertl, R. (2004). Testing hypotheses with fuzzy data: the fuzzy p-value. Metrika 59: 21-29.

Holena, M. (2004). Fuzzy hypotheses testing in a framework of fuzzy logic. Fuzzy Sets and Systems 145: 229-252.

Parchami, A., Taheri, S. M., and Mashinchi, M. (2010). Fuzzy p-value in testing fuzzy hypotheses with crisp data. Statistical Papers 51: 209-226.

Parchami, A., Taheri, S. M., and Mashinchi, M. (2012). Testing fuzzy hypotheses based on vague observations: a p-value approach. Statistical Papers 53: 469-484.

Wang, X., Kerre, E. E. (2001). Reasonable properties for the ordering of fuzzy quantities (II). Fuzzy Sets and Systems 118: 387-405.

Yuan, Y. (1991). Criteria for evaluating fuzzy ranking methods. Fuzzy Sets Syst 43: 139-157.

Examples

```
# Solving Example 5 from Springer Chapter book 2016
# Considering the assumptions of this example, it must be note that H0 and H1 are not about
#  the parameter of the test statistics, and in fact hypotheses must be 12*H0 and 12*H1
H0 = B( 12*2.75 ,  12*3.25 )
H1 = S( 12*2.75 ,  12*3.25 )
t = T( 27, 27, 27)
sig = T(0.0,0.05,0.1)
p_value.pois( kind=1, H0, H1, t, n=12, sig)

# Solving Example 7 from Springer Chapter book 2016
H0 = B( 12*2.75 ,  12*3.25 )
H1 = S( 12*2.75 ,  12*3.25 )
```

```
t = T( 24, 27, 30)
sig = T(0.0,0.05,0.1)
p_value.pois( kind=1, H0, H1, t, n=12, sig)


## The function is currently defined as
function (kind, H0, H1, t, n, sig)
{
    alpha_L = seq(0, 1, 0.01)
    alpha_U = seq(1, 0, -0.01)
    if (kind == 1) {
        p_L = ppois(alphacut(t, alpha_L)[, "U"], alphacut(H0,
            alpha_L)[, "L"])
        p_U = ppois(alphacut(t, alpha_U)[, "L"], alphacut(H0,
            alpha_U)[, "U"])
    }
    else if (kind == 2) {
        p_L = 1 - ppois(alphacut(t, alpha_L)[, "U"], alphacut(H0,
            alpha_L)[, "L"])
        p_U = 1 - ppois(alphacut(t, alpha_U)[, "L"], alphacut(H0,
            alpha_U)[, "U"])
    }
    else if (supp(t)[1] >= supp(H0)[2]) {
        p_L = 2 * (1 - ppois(alphacut(t, alpha_L)[, "U"], alphacut(H0,
            alpha_L)[, "L"]))
        p_U = 2 * (1 - ppois(alphacut(t, alpha_U)[, "L"], alphacut(H0,
            alpha_U)[, "U"]))
    }
    else if (supp(t)[2] <= supp(H0)[1]) {
        p_L = 2 * ppois(alphacut(t, alpha_L)[, "U"], alphacut(H0,
            alpha_L)[, "L"])
        p_U = 2 * ppois(alphacut(t, alpha_U)[, "L"], alphacut(H0,
            alpha_U)[, "U"])
    }
    else {
        return("The fuzzy p-value can not defined for this example, since the fuzziness of
the problem is very high. This case may be accured for the testing equality
fuzzy null hypothesis, against the non-equality fuzzy althenative hypothesis")
    }
    alpha = cbind(alpha_L, alpha_U)
    p = cbind(p_L, p_U)
    plot(p, alpha, type = "l", lwd = 3, col = 1, xlim = c(-0.04,
        1.02))
     s1 = supp(sig)[1]
     s2 = core(sig)[1]
     s3 = core(sig)[1]
     s4 = supp(sig)[2]
     lines( c(0,s1,s2,s3,s4,1), c(0,0,1,1,0,0), type='l', lty=3, lwd=2, col=2 )
# plot(sig, lty=3, lwd=2, col=2, add=TRUE) # Conflict importing 2 plot from different packages
    legend("topright", c("Fuzzy p-value", "Significance level"),
        col = c(1, 2), text.col = 1, lwd = c(3, 2), lty = c(1,
            3))
     if( class(sig) == "numeric" ){
```

```
                    sig <- TriangularFuzzyNumber(sig, sig, sig)
                    }
          P_L = p_L
          P_U = p_U
          knot.n = 100
          S_L = alphacut(sig, round(seq(0, 1, .01), 5))[,"L"]
          S_U = alphacut(sig, round(seq(0, 1, .01), 5))[,"U"]
          Int1 = ( P_U - S_L ) * ( P_U > S_L )
          Int2 = ( P_L - S_U ) * ( P_L > S_U )
          Arz = 1 / (knot.n - 1)  #Arze Mostatilha baraye mohasebe-ye Integral
          Integral1 <- ( sum( Int1 ) - Int1[1]/2 - Int1[length(Int1)]/2 ) *Arz
          Integral2 <- ( sum( Int2 ) - Int2[1]/2 - Int2[length(Int2)]/2 ) *Arz
          Delta_PS = Integral1 + Integral2
          Int3 = ( S_U - P_L ) * ( S_U > P_L )
          Int4 = ( S_L - P_U ) * ( S_L > P_U )
          Integral3 <- ( sum( Int3 ) - Int3[1]/2 - Int3[length(Int3)]/2 ) *Arz
          Integral4 <- ( sum( Int4 ) - Int4[1]/2 - Int4[length(Int4)]/2 ) *Arz
          Delta_SP = Integral3 + Integral4
     print("Delta_SP = ")
     print(Delta_SP)
     print("Delta_PS = ")
     print(Delta_PS)
     Degree_P_biger_than_S = Delta_PS/(Delta_PS + Delta_SP)
     Degree_S_biger_than_P = 1 - Degree_P_biger_than_S
     if (Degree_P_biger_than_S >= Degree_S_biger_than_P) {
          a = "The null hypothesis (H0) is accepted with degree D(P>S)="
          b = round(Degree_P_biger_than_S, 4)
          c = ", at  the considered significance level."
          noquote(sprintf("%s %s %s ", a, b, c))
     }
     else {
          if (Degree_P_biger_than_S < Degree_S_biger_than_P) {
               a = "The althernative hypothesis (H1) is accepted with degree D(S>P)="
               b = round(Degree_S_biger_than_P, 4)
               c = ", at  the considered significance level."
               noquote(sprintf("%s %s %s ", a, b, c))
          }
          else {
               return(noquote(paste0("Impossible case")))
          }
     }
   }
```

S   *Class of linear fuzzy sets to construct concept "approximately smaller"*

Description

The are several famous fuzzy sets which have meaningful means and are user friendly for introduc-
ing. Several linear functions are constructed in this package for easily inspiration to the fuzzy con-
cepts $\ll approximately smaller \gg$, $\ll approximately bigger \gg$ and $\ll approximately equal \gg$.

For example, function S is one of them which can users can use it to construct the fuzzy concept $\ll approximately smaller \gg$ in fuzzy hypotheses. As presented bellow, the membership function of fuzzy set S has two parameters a and b:

$$S(a,b)(x) = \begin{cases} 1 & if & x \leq a \\ \frac{x-b}{a-b} & if & a < x \leq b \\ 0 & if & x > b \end{cases}$$

Usage

```
S(a, b)
```

Arguments

a   The first parameter of fuzzy set $S(a,b)$. For better understanding, try to draw the above membership function.

b   The second parameter of fuzzy set $S(a,b)$. For better understanding, try to draw the above membership function.

Value

This function convert a fuzzy hypothesis to its boundary which is a triangular fuzzy number. Therefore, although the input of this function is not triangular fuzzy number, but its output is triangular.

References

Parchami, A., Taheri, S. M., and Mashinchi, M. (2010). Fuzzy p-value in testing fuzzy hypotheses with crisp data. Statistical Papers 51: 209-226.

Parchami, A., Taheri, S. M., and Mashinchi, M. (2012). Testing fuzzy hypotheses based on vague observations: a p-value approach. Statistical Papers 53: 469-484.

Examples

```
# Introducing the fuzzy alternative hypothesis H1
H1 = S(1275,1325)

## The function is currently defined as
function (a, b)
(TriangularFuzzyNumber(a, a, b))
```

T   *Class of triangular fuzzy numbers*

Description

The are several famous fuzzy sets which have meaningful means and are user friendly for introducing. Several linear functions are constructed in this package for easily inspiration to the fuzzy concepts $\ll approximately smaller \gg$, $\ll approximately bigger \gg$ and $\ll approximately equal \gg$. For example, function T is one of them which users can use it to construct the fuzzy concept $\ll approximately equal \gg$ in fuzzy statistics and fuzzy hypotheses. As presented bellow, the membership function of fuzzy set T has three parameters a, b and c:

$$T(a,b,c)(x) = \begin{cases} \frac{x-a}{b-a} & if & a < x \leq b \\ \frac{x-c}{b-c} & if & b < x \leq c \\ 0 & if & elsewhere \end{cases}$$

Usage

```
T(a, b, c)
```

Arguments

a   Considering the introduced above membership function, the first parameter is the first point of the support of triangular fuzzy number.

b   The second parameter is the core of triangular fuzzy number.

c   The third parameter is the end point of the support of triangular fuzzy number.

Value

This function easily introduce the membership function of a triangular fuzzy number.

References

Parchami, A., Taheri, S. M., and Mashinchi, M. (2010). Fuzzy p-value in testing fuzzy hypotheses with crisp data. Statistical Papers 51: 209-226.

Parchami, A., Taheri, S. M., and Mashinchi, M. (2012). Testing fuzzy hypotheses based on vague observations: a p-value approach. Statistical Papers 53: 469-484.

Examples

```
# Introducing the membership function of triangular fuzzy number (for test statistics)
t = T(2,4,7)

## The function is currently defined as
function (a, b, c)
(TriangularFuzzyNumber(a, b, c))
```

Tr   *Class of trapezoidal fuzzy numbers*

Description

Several linear functions are constructed in this package for easily inspiration to the fuzzy concepts $\ll approximately smaller \gg$, $\ll approximately bigger \gg$ and $\ll approximately equal \gg$. For example, function Tr is one of them which can use it to construct the fuzzy concept $\ll approximately equal \gg$ for fuzzy statistics or fuzzy hypotheses. As presented bellow, the membership function of fuzzy set Tr has four parameters a, b, c and d:

$$Tr(a,b,c,d)(x) = \begin{cases} \frac{x-a}{b-a} & if & a < x \leq b \\ 1 & if & b < x \leq c \\ \frac{x-d}{c-d} & if & c < x \leq d \\ 0 & if & elsewhere \end{cases}$$

Usage

```
Tr(a, b, c, d)
```

Arguments

a   Considering the introduced above membership function, the first parameter is the first point of the support of trapezoidal fuzzy number.

b   The second parameter is the first point of the core of trapezoidal fuzzy number.

c   The third parameter is the end point of the core of trapezoidal fuzzy number.

d   The fourth parameter is the end point of the support of trapezoidal fuzzy number.

Value

This function easily introduce the membership function of a trapezoidal fuzzy number.

References

Parchami, A., Taheri, S. M., and Mashinchi, M. (2010). Fuzzy p-value in testing fuzzy hypotheses with crisp data. Statistical Papers 51: 209-226.

Parchami, A., Taheri, S. M., and Mashinchi, M. (2012). Testing fuzzy hypotheses based on vague observations: a p-value approach. Statistical Papers 53: 469-484.

Examples

```
# Introducing the membership function of trapezoidal fuzzy number (for test statistics t)
t = Tr(2,4,6,9)

## The function is currently defined as
function (a, b, c, d)
(TrapezoidalFuzzyNumber(a, b, c, d))
```

Index