An Integrated Maple Package for Algebraic Interval Neutrosophic Matrices

Said Broumi1,2, Mohamed Bisher Zeina3, Mohammad Abobala4

1 Laboratory of Information Processing, Faculty of Science Ben M’Sik, University of Hassan II, Casablanca, Morocco
   broumisaid78@gmail.com
2 Regional Center for the Professions of Education and Training (C.R.M.E.F), Casablanca-Settat, Morocco.
3 Dept. of Mathematical Statistics, Faculty of Science, University of Aleppo, Aleppo, Syria
   bisher.zeina@gmail.com
4 Tishreen University, Faculty of Science, Department of Mathematics, Latakia, Syria
   mohammadabobala777@gmail.com

Abstract: In this paper, a maple code is presented to do many operations on interval valued neutrosophic matrices including entering the elements of the matrix, checking whether a given matrix is an interval valued neutrosophic matrix or not, finding complement of an interval valued neutrosophic matrix, finding score, accuracy, and certainty measures, union and intersection of two interval valued neutrosophic matrices, sum and product of two interval valued neutrosophic matrices and finding the transpose of a given interval valued of neutrosophic matrix. What distinguishes this code is its simplicity to understand and to call the functions. Many examples are presented and solved successfully.

Keywords: Maple; Neutrosophic Set; Single Valued Neutrosophic Set; Interval Valued Neutrosophic Set; Operations on Matrices.

1. Introduction

Fuzzy Sets were presented by Zadeh [1] to expand the concept of crisp sets allowing elements to belong to the sets partially (with membership degree between 0 and 1), then the last concept expanded by Atanassov [2] to what is known by intuitionistic fuzzy sets adding nonmembership component to describe elements of sets. In 1995, Smarandache [3] presented neutrosophic sets as an extension of fuzzy sets and intuitionistic fuzzy sets in which each element is described by three independent components; truth, indeterminacy and false memberships. Many other extensions to neutrosophic sets were presented including refined neutrosophic sets, interval valued neutrosophic sets, bipolar neutrosophic sets, generalized neutrosophic sets, neutrosophic vague soft expert set, fermatean neutrosophic sets, etc.

Many mathematical studies were done on neutrosophic sets and many branches of mathematics were extended to the new concept of logic including probability theory, operations research, statistics, linear algebra, abstract algebra, queueing theory, artificial intelligence and data mining.[4-14]

Since dealing with neutrosophic sets is very complex and operations on it take long time, then many researchers wrote programming packages and codes to make dealing with it more simple.
Salama et al.[15] presented an introduction to develop programming softwares to deal with neutrosophic sets. Bakro et al.[16] wrote a matlab code to neutrosophication functions and their implementation. Broumi et al.[17] wrote a matlab code to implement neutrosophic membership functions and graphing it. Bisher Zeina et al.[18] presented a maple package to do operations on single valued neutrosophic sets using \(\alpha, \beta, \gamma\)-Cuts, Broumi et al.[19] wrote a maple package to perform operations on single valued neutrosophic matrices. In this paper we generalize the code presented in [19] to deal with interval valued neutrosophic matrices and do operations on it.

2. Background on Neutrosophic Sets

**Definition 2.1**[20]

Let \( \Omega \) be a universe, we call \( A \subseteq \Omega \) a neutrosophic set if elements of \( A \) are described by their membership degree \( T_A(x) \), nonmembership degree \( F_A(x) \) and indeterminacy degree \( I_A(x) \) and we denote that by:

\[
A = \{ < T_A(x), I_A(x), F_A(x) > ; x \in \Omega \}
\]

Where:

\[
T_A(x), I_A(x), F_A(x) \in \mathbb{R} \quad \text{and} \quad 0 \leq T_A(x) + I_A(x) + F_A(x) \leq 3
\]

**Definition 2.2** [20]

Let \( \Omega \) be a universe, we call \( A \subseteq \Omega \) a single valued neutrosophic set if:

\[
A = \{ < T_A(x), I_A(x), F_A(x) > ; x \in \Omega \}
\]

Where:

\[
T_A(x), I_A(x), F_A(x) \in [0,1] \quad \text{and} \quad 0 \leq T_A(x) + I_A(x) + F_A(x) \leq 3
\]

**Definition 2.3** [21]

Let \( \Omega \) be a universe, we call \( A \subseteq \Omega \) an interval valued neutrosophic set if:

\[
A = \{ < [T_A^L(x), T_A^U(x)], [I_A^L(x), I_A^U(x)], [F_A^L(x), F_A^U(x)] > ; x \in \Omega \}
\]

Where:

\[
T_A^L(x), T_A^U(x), I_A^L(x), I_A^U(x), F_A^L(x), F_A^U(x) \in [0,1]
\]

**Definition 2.4**

Interval valued neutrosophic matrix of order \( m \times n \) is defined as follows:

\[
A = [a_{ij}, [Ta_{ij}^L, Ta_{ij}^U], [Ia_{ij}^L, Ia_{ij}^U], [Fa_{ij}^L, Fa_{ij}^U]]_{m \times n}
\]

3. Maple Package to Do Operations on Interval Valued Neutrosophic Matrices

3.1. Entering Interval Valued Neutrosophic Matrices

To enter interval valued neutrosophic matrix we call the function `IVNInput(m,n)` where \( m, n \) are numbers of rows and columns respectively and the written function is as follows:

```maple
restart;interface(warnlevel=0);
```
with(Maplets[Elements]):

IVNIInput:=proc(m::integer,n::integer)
local mat:=Matrix(m,n);
for i from 1 to m by 1 do
for j from 1 to n by 1 do
truthL:=Maplet(InputDialog['x'](cat("Enter lower truth of element ","i","j"),'onapprove'=Shutdown(['x']),'oncancel'=Shutdown()));
truthL:=Display(truthL);
truthL:=parse(op(truthL));
truthU:=Maplet(InputDialog['x'](cat("Enter upper truth of element ","i","j"),'onapprove'=Shutdown(['x']),'oncancel'=Shutdown()));
truthU:=Display(truthU);
truthU:=parse(op(truthU));
indeterminacyL:=Maplet(InputDialog['x'](cat("Enter lower indeterminacy of element ","i","j"),'onapprove'=Shutdown(['x']),'oncancel'=Shutdown()));
indeterminacyL:=Display(indeterminacyL);
indeterminacyL:=parse(op(indeterminacyL));
indeterminacyU:=Maplet(InputDialog['x'](cat("Enter upper indeterminacy of element ","i","j"),'onapprove'=Shutdown(['x']),'oncancel'=Shutdown()));
indeterminacyU:=Display(indeterminacyU);
indeterminacyU:=parse(op(indeterminacyU));
falsityL:=Maplet(InputDialog['x'](cat("Enter lower falsity of element ","i","j"),'onapprove'=Shutdown(['x']),'oncancel'=Shutdown()));
falsityL:=Display(falsityL);
falsityL:=parse(op(falsityL));
falsityU:=Maplet(InputDialog['x'](cat("Enter upper falsity of element ","i","j"),'onapprove'=Shutdown(['x']),'oncancel'=Shutdown()));
falsityU:=Display(falsityU);
falsityU:=parse(op(falsityU));
mat(i,j):=convert([[truthL,truthU],[indeterminacyL,indeterminacyU],[falsityL,falsityU]],string);
end do;
end proc:
end do;
mat;
end proc:

3.2. Checking whether the matrix is IVNM or not

We can call the function `IVNCheck(mat)` defined below to check whether matrix mat is interval valued neutrosophic matrix or not:

```maple
IVNCheck:=proc(mat)
IsMembership:=proc(num)
if num<0 or num>1 then return false else return true end if;
end proc:
m,n:=LinearAlgebra[Dimension](mat);
result:=true;
for i from 1 to m by 1 do
for j from 1 to n by 1 do
x:=parse(mat(i,j));
truth:=x[1];
indeterminacy:=x[2];
falsity:=x[3];
truthL:=truth[1];
truthU:=truth[2];
indeterminacyL:=indeterminacy[1];
indeterminacyU:=indeterminacy[2];
falsityL:=falsity[1];
falsityU:=falsity[2];
result:= IsMembership(truthL) and IsMembership(truthU) and IsMembership(indeterminacyL) and IsMembership(indeterminacyU) and IsMembership(falsityL) and IsMembership(falsityU);
if not result then break; end if;
end do;
if not result then break; end if;
```
Example 1. In this example we define an interval valued neutrosophic matrix $E$ and check whether it is right defined or not where:

$$E = \left( \begin{array}{cc}
< [1,1],[.7,.6] > & < [2,4],[.2,.9] > \\
< [.8,.9],[.5,.2] > & < [1,.2],[.5,.7] > 
\end{array} \right)$$

The interval valued neutrosophic matrix $E$ can be inputted in Maple like this:

```maple
E := Matrix(2, 2, [[[1, 1], [.7, .6]], [[2, 4], [.2, .9]], [[.8, .9], [.5, .2]], [[1, .2], [.5, .7]]]);
```

Or like this:

```maple
x := IVNIInput(2, 2);
```

Then an input box dialogue is going to appear and lead you how to input elements.

Result of checking whether matrix $E$ is Interval-Valued Neutrosophic Matrix or not can be obtained by calling the command `IVNCheck(E);`

And the result will be:

"your matrix is an interval valued neutrosophic matrix"

3.3. Finding complement of interval valued neutrosophic matrix

For a given IVNM $A = \left[ \begin{array}{c}
< a_{ij}, T_{a_{ij}}, I_{a_{ij}} > \\
< 1 - I_{a_{ij}}, 1 - L_{a_{ij}} > \\
< T_{a_{ij}}, U_{a_{ij}} > 
\end{array} \right]_{m \times n}$, the complement of $A$ is defined as follow:

$$A^c = \left[ \begin{array}{c}
< a_{ij}, FA_{ij} > \\
[1 - I_{a_{ij}}, 1 - L_{a_{ij}}, T_{a_{ij}}, U_{a_{ij}}] 
\end{array} \right]_{m \times n}$$

To find the complement of interval valued neutrosophic matrix we can call the function `IVNMComplementOf(mat)` which is defined as follow:

```maple
IVNMComplementOf := proc(mat::Matrix)
    temp := LinearAlgebra[Copy](mat);
    m, n := LinearAlgebra[Dimension](temp);
```

Said Broumi, Mohamed Bisher Zeina and Mohammad Abobala, An Integrated Maple Package for Algebraic Interval Neutrosophic Matrices
for i from 1 to m by 1 do
for j from 1 to n by 1 do
x := parse(mat(i,j));
falsity := x[1];
indeterminacy := x[2];
truth := x[3];
indeterminacyL := 1 - indeterminacy[2];
indeterminacyU := 1 - indeterminacy[1];
temp(i,j) := convert([truth, [indeterminacyL, indeterminacyU], falsity], string);
end do;
end do;
temp;
end proc:

Example 2. find the complement of matrix E in example 1.

the complement of matrix E is:

\[
E^c = \begin{pmatrix}
< [1, 6], [2, 3], [1, 1] > & < [1, 9], [2, 8], [2, 4] > \\
< [1, 2], [5, 7], [8, 9] > & < [2, 5], [3, 5], [1, 2] > 
\end{pmatrix}
\]

By calling the function

SVNMComplementOfE(E);

Same results appear:

\[
\begin{pmatrix}
"[1, 6], [2, 3], [1, 1]\" & "[1, 9], [2, 8], [2, 4]\"\\
"[1, 2], [5, 7], [8, 9]\" & "[2, 5], [3, 5], [1, 2]\"
\end{pmatrix}
\]

3.4. Finding score, accuracy and certainty matrices of interval valued neutrosophic matrices

Suppose A is an interval neutrosophic matrix, then score, accuracy and certainty measures are defined as follows: [22]

\[
S_{IWN}(x) = \frac{\mu_A^L(x) + \mu_A^U(x) + 4 - \mu_A^L(x) - \mu_A^U(x) - \mu_A^L(x) - \mu_A^U(x)}{6}
\]

\[
S_{SVN}(x) = \frac{2 \cdot \mu_A^L(x) - \mu_A^L(x) - \mu_A^U(x)}{3}
\]

\[
A_{IWN}(x) = \frac{\mu_A^L(x) + \mu_A^U(x) - \mu_A^L(x) - \mu_A^U(x)}{2}
\]

\[
\text{Said Broumi, Mohamed Bisher Zeina and Mohammad Abobala, An Integrated Maple Package for Algebraic Interval Neutrosophic Matrices}
\]
Three maple functions ScoreMatrix(), AccuracyMatrix() and CertaintyMatrix() are defined as follows:

```maple
ScoreMatrix:=proc(mat::Matrix)
m,n:=LinearAlgebra[Dimension](mat);
scoreMat:=Matrix(m,n);
for i from 1 to m by 1 do
  for j from 1 to n by 1 do
    x:=parse(mat(i,j));
    truth:=x[1];
    indeterminacy:=x[2];
    falsity:=x[3];
    truthL:=truth[1];
    truthU:=truth[2];
    indeterminacyL:=indeterminacy[1];
    indeterminacyU:=indeterminacy[2];
    falsityL:=falsity[1];
    falsityU:=falsity[2];
    score:=(4+truthL+truthU-indeterminacyL-indeterminacyU-falsityL-falsityU)/6;
    scoreMat(i,j):=score;
  end do;
end do;
scoreMat;
end proc:
AccuracyMatrix:=proc(mat::Matrix)
m,n:=LinearAlgebra[Dimension](mat);
amat:=Matrix(m,n);
for i from 1 to m by 1 do
  for j from 1 to n by 1 do
    x:=parse(mat(i,j));
    truth:=x[1];
    indeterminacy:=x[2];
    falsity:=x[3];
    truthL:=truth[1];
    truthU:=truth[2];
    indeterminacyL:=indeterminacy[1];
    indeterminacyU:=indeterminacy[2];
    falsityL:=falsity[1];
    falsityU:=falsity[2];
    a:=4+truthL+truthU-indeterminacyL-indeterminacyU-falsityL-falsityU;
    amat(i,j):=a;
  end do;
amat;
end proc:
CertaintyMatrix:=proc(mat::Matrix)
m,n:=LinearAlgebra[Dimension](mat);
certMat:=Matrix(m,n);
for i from 1 to m by 1 do
  for j from 1 to n by 1 do
    x:=parse(mat(i,j));
    truth:=x[1];
    indeterminacy:=x[2];
    falsity:=x[3];
    truthL:=truth[1];
    truthU:=truth[2];
    indeterminacyL:=indeterminacy[1];
    indeterminacyU:=indeterminacy[2];
    falsityL:=falsity[1];
    falsityU:=falsity[2];
    c:=4+truthL+truthU-indeterminacyL-indeterminacyU-falsityL-falsityU;
    certMat(i,j):=c;
  end do;
certMat;
end proc:
```

\[ \hat{A}_{SVNN}(x) = T_A(x) - F_A(x) \]

\[ \hat{C}_{IVNN}(x) = \frac{r_A(x) + r_B(x)}{2} \]

\[ \hat{C}_{SVNN}(x) = T_A(x) \]
for i from 1 to m by 1 do
    for j from 1 to n by 1 do
        x:=parse(mat(i,j));
        truth:=x[1];
        indeterminacy:=x[2];
        falsity:=x[3];
        truthL:=truth[1];
        truthU:=truth[2];
        indeterminacyL:=indeterminacy[1];
        indeterminacyU:=indeterminacy[2];
        falsityL:=falsity[1];
        falsityU:=falsity[2];
        a:=(truthL+truthU-falsityL-falsityU)/2;
        aMat(i,j):=a;
    end do;
end do;
aMat;
end proc:

CertaintyMatrix:=proc(mat::Matrix)
m,n:=LinearAlgebra[Dimension](mat);
cMat:=Matrix(m,n);
for i from 1 to m by 1 do
    for j from 1 to n by 1 do
        x:=parse(mat(i,j));
        truth:=x[1];
        indeterminacy:=x[2];
        falsity:=x[3];
        truthL:=truth[1];
        truthU:=truth[2];
        indeterminacyL:=indeterminacy[1];
        indeterminacyU:=indeterminacy[2];
        falsityL:=falsity[1];
        falsityU:=falsity[2];
        a:=(truthL+truthU-falsityL-falsityU)/2;
        aMat(i,j):=a;
    end do;
end do;
aMat;
end proc:
3.5. Computing union of two interval valued neutrosophic matrices

Union of two interval valued neutrosophic matrices $A$ and $B$ is defined as follow:

$$A \cup B = C = \langle c_{ijr}, c_{ijl}, c_{ijf} \rangle_{m \times n}$$

where

- $c_{ijr} = \left[ a_{ijL} \lor b_{ijL}, a_{ijU} \lor b_{ijU} \right]$, 
- $c_{ijl} = \left[ a_{ijL} \lor b_{ijL}, a_{ijU} \lor b_{ijU} \right]$, 
- $c_{ijf} = \left[ a_{ijL} \lor b_{ijL}, a_{ijU} \lor b_{ijU} \right]$.

And it can be evaluated by calling the function `Union(A, B)` described as follows:
Union:=proc(mat1::Matrix,mat2::Matrix)
    m1,n1:=LinearAlgebra[Dimension](mat1);
    m2,n2:=LinearAlgebra[Dimension](mat2);
    if (n1=n2) and (m1=m2) then
        m:=m1;n:=n1;
        unionMat:=Matrix(m,n);
        for i from 1 to m by 1 do
            for j from 1 to n by 1 do
                x:=parse(mat1(i,j));
                y:=parse(mat2(i,j));
                truth1:=x[1];
                indeterminacy1:=x[2];
                falsity1:=x[3];
                truthL1:=truth1[1];
                truthU1:=truth1[2];
                indeterminacyL1:=indeterminacy1[1];
                indeterminacyU1:=indeterminacy1[2];
                falsityL1:=falsity1[1];
                falsityU1:=falsity1[2];
                truth2:=y[1];
                indeterminacy2:=y[2];
                falsity2:=y[3];
                truthL2:=truth2[1];
                truthU2:=truth2[2];
                indeterminacyL2:=indeterminacy2[1];
                indeterminacyU2:=indeterminacy2[2];
                falsityL2:=falsity2[1];
                falsityU2:=falsity2[2];
    end if;
end proc;
Example 3. Say that:

\[
E = \begin{bmatrix}
[[1, 1], [7, 8], [1, 6]] & "[[2, 8], [2, 1], [1, 9]]"
& "[[2, 8], [2, 8], [2, 0]]"
\end{bmatrix}
\]

\[
F = \begin{bmatrix}
"[[0, 7], [7, 8], [4, 0]]" & "[[2, 9], [2, 0], [1, 3]]"
& "[[2, 4], [3, 5], [1, 3]]"
\end{bmatrix}
\]

So, the union of previous matrices is done by calling the function:

Union( E, F );

And the result is:

\[
E_{IVNM} \cup F_{IVNM} = \begin{bmatrix}
"[[1, 1], [7, 8], [4, 0]]" & "[[2, 9], [2, 8], [1, 9]]"
& "[[2, 8], [2, 8], [2, 8]]"
\end{bmatrix}
\]
3.6. Computing intersection of two interval valued neutrosophic matrices

The intersection of two interval valued neutrosophic matrices A and B is defined as follow:

\[ A \cap B = C = \left[ d_{ij}, d_{ij}^L, d_{ij} \right]_{m \times n} \]

Where:

\[ c_{ij}^T = \left[ a_{ij}^L \land b_{ij}^L, a_{ij}^U \land b_{ij}^U \right], \]

\[ c_{ij}^I = \left[ a_{ij}^L \lor b_{ij}^L, a_{ij}^U \lor b_{ij}^U \right], \]

\[ c_{ij}^F = \left[ a_{ij}^F \lor b_{ij}^F, a_{ij}^U \lor b_{ij}^U \right] \]

And this is done calling the function \( \text{Intersection}(A,B) \) is defined in the following manner.

```maple
Intersection:=proc(mat1::Matrix,mat2::Matrix)
m1,n1:=LinearAlgebra[Dimension](mat1);
m2,n2:=LinearAlgebra[Dimension](mat2);
if (n1=n2) and (m1=m2) then
m:=m1;n:=n1;
intersectMat:=Matrix(m,n);
for i from 1 to m by 1 do
for j from 1 to n by 1 do
x:=parse(mat1(i,j));
y:=parse(mat2(i,j));
truth1:=x[1];
indeterminacy1:=x[2];
falsity1:=x[3];
truthL1:=truth1[1];
indeterminacyL1:=indeterminacy1[1];
falsityL1:=falsity1[1];
truthU1:=truth1[2];
indeterminacyU1:=indeterminacy1[2];
falsityU1:=falsity1[2];
truth2:=y[1];
```

Said Broumi, Mohamed Bisher Zeina and Mohammad Abobala, An Integrated Maple Package for Algebraic Interval Neutrosophic Matrices
Example 4. Find intersection of interval valued neutrosophic matrices E and F presented in example 3.

Solution:
Calling the function `Intersection(E, F)` yields to the solution:

\[
\left[
\begin{array}{c}
[0.7, 1], [0.7, 0.8], [0.4, 0.6] \\
[0.2, 0.3], [0.2, 0.8], [0.1, 0.9] \\
[0.2, 0.4], [0.3, 0.5], [0.1, 0.3] \\
[0.1, 0.2], [0.5, 0.7], [0.3, 0.5]
\end{array}
\right]
\]
3.7. Addition of two interval valued neutrosophic matrices.

The Addition of two interval valued neutrosophic matrices A and B is defined as follow:

\[ A \oplus B = S = [s_{ijr}, s_{ijl}, s_{ijf}]_{m \times n} \]

Where:

\[ s_{ijr} = [a_{ijrL} + b_{ijrL} - a_{ijrL} + b_{ijrL} - a_{ijrL}, b_{ijrL} + b_{ijrL} - a_{ijrL}, b_{ijrL}, a_{ijrU} + b_{ijrU} - a_{ijrU}, b_{ijrU}] \]

\[ s_{ijl} = [a_{ijlL}, b_{ijlL}, a_{ijlL}, b_{ijlL}] \]

\[ s_{ijf} = [a_{ijfL}, b_{ijfL}, a_{ijfL}, b_{ijfL}] \]

And can be done calling the function \textbf{Addition (A, B)} which is defined as follow:

```
Addition:=proc(mat1::Matrix,mat2::Matrix)
m1,n1:=LinearAlgebra[Dimension](mat1);
m2,n2:=LinearAlgebra[Dimension](mat2);
if (n1=n2) and (m1=m2) then
m:=m1;n:=n1;
addMat:=Matrix(m,n);
for i from 1 to m by 1 do
  for j from 1 to n by 1 do
    x:=parse(mat1(i,j));
    y:=parse(mat2(i,j));
    truth1:=x[1];
    indeterminacy1:=x[2];
    falsity1:=x[3];
    truthL1:=truth1[1];
    truthU1:=truth1[2];
    indeterminacyL1:=indeterminacy1[1];
    indeterminacyU1:=indeterminacy1[2];
    falsityL1:=falsity1[1];
    falsityU1:=falsity1[2];
    truth2:=y[1];
```

Example 5. In this example we find the addition of two interval valued neutrosophic matrices E and F presented in example 3 calling the function:

\[
\text{Addition}(E,F);
\]

\[
\begin{bmatrix}
[1.0, 1, [.49, .64], [.4e-1, .36]]
& [1.36, .58], [.4e-1, .48], [.1e-1, .27]]
\end{bmatrix}
\]

3.8. Product of two interval valued neutrosophic matrices

The product of two interval valued neutrosophic matrices A and B is defined as follow:
\[ A \bigodot B = R = \begin{bmatrix} r_{ijT} & r_{ijI} & r_{ijF} \end{bmatrix} \]

where

\[ r_{ijT} = \begin{bmatrix} a_{ijT} & b_{ijT} & a_{ijT} \end{bmatrix}, \]

\[ r_{ijI} = \begin{bmatrix} a_{ijI} & b_{ijI} & a_{ijI} \end{bmatrix}, \]

\[ r_{ijF} = \begin{bmatrix} a_{ijF} & b_{ijF} & a_{ijF} \end{bmatrix}, \]

Which is simply done by the call of the function Product \((A, B)\) defined as follow:

```maple
Prod:=proc(mat1::Matrix,mat2::Matrix)
m1,n1:=LinearAlgebra[Dimension](mat1);
m2,n2:=LinearAlgebra[Dimension](mat2);
if (n1=n2) and (m1=m2) then
m:=m1;n:=n1;
prodMat:=Matrix(m,n);
for i from 1 to m by 1 do
    for j from 1 to n by 1 do
        x:=parse(mat1(i,j));
y:=parse(mat2(i,j));
        truth1:=x[1];
        indeterminacy1:=x[2];
        falsity1:=x[3];
        truthL1:=truth1[1];
        truthU1:=truth1[2];
        indeterminacyL1:=indeterminacy1[1];
        indeterminacyU1:=indeterminacy1[2];
        falsityL1:=falsity1[1];
        falsityU1:=falsity1[2];
        truth2:=y[1];
    end do;
end if;
```

Said Broumi, Mohamed Bisher Zeina and Mohammad Abobala, An Integrated Maple Package for Algebraic Interval Neutrosophic Matrices
Example 6. In this example we evaluate the product of the two interval valued neutrosophic matrices E and F presented in example 3 by calling of the command:

Product(E, F);

\[
\begin{bmatrix}
[.7, 1, .91, .96, .46, .84] & [\text{null}, \text{null}, .4e-1, .12, .36, .92, .19, .93]
\end{bmatrix}
\]

\[
\begin{bmatrix}
[.16, .36, .51, .75, .19, .44] & [\text{null}, \text{null}, \text{null}, \text{null}, .7e-1, .4e-1, .75, .91, .44, .75]
\end{bmatrix}
\]

3.9. Transpose of interval valued neutrosophic matrix
Transpose of interval valued neutrosophic matrix simply done by calling of the function\n
\textbf{Transpose(A)} defined as follow:

\begin{verbatim}
Transpose:=proc(mat::Matrix)
m,n:=LinearAlgebra[Dimension](mat);
temp:=Matrix(n,m);
for i from 1 to n by 1 do
  for j from 1 to m by 1 do
    temp(i,j):=mat(j,i);
  end do;
end do;
temp;
end proc:
\end{verbatim}

\textbf{Example 7.} In this example we evaluate the transpose of the interval valued neutrosophic matrix $E$ presented in example 3:

\[ \text{Transpose}(E); \]

\[
\begin{bmatrix}
[[1, 1], [.7, .8], [.1, .6]] & [[.8, .9], [.3, .5], [.1, .2]] \\
[[.2, .4], [.2, .8], [.1, .9]] & [[-.1, .2], [.5, .7], [.2, .5]]
\end{bmatrix}
\]

4. Conclusions

This paper proposed new Maple package to do operations on interval valued neutrosophic matrices including complement, transpose, union, intersection, addition, product, sum and product of interval valued neutrosophic matrices. This package is very useful in neutrosophic decision making operations and on neutrosophic events simulation. In future work we are looking forward to generalize this package to other neutrosophic sets like fermatean neutrosophic sets and refined neutrosophic sets.

\textbf{Funding:}

“This research received no external funding.”

\textbf{Acknowledgments:}

Authors are very grateful to the chief editor and reviewers for their comments and suggestions which improves the work.

\textbf{Conflicts of Interest:} “The authors declare no conflict of interest.”
Data Availability Statement: The authors declare that there is no Data Availability

References:


Received: July 7, 2023. Accepted: Nov 22, 2023