Function help

This contains the help section of all the functions contained in this package.

Contents

IESDS

Help section of the IESDS.m function

```
help IESDS
```

```
  Iterated Elimination of Strictly Dominated Strategies

  Syntax
      #payoff# = IESDS(#payoff#)

  Input arguments
      #payoff# [cell(:inf x :inf)] is the payoff matrix of the game for
          which the equilibrium is queried. Each cell of #payoff#, i.e.
          #payoff#{i,j} contains a double array of size [double(1 x 2)],
          which contains the outcome (payoff) for 1st player and for 2nd
          player. The player whose strategies are the rows of the payoff
          matrix is considered as the 1st player, i.e. the number of
          strategies of the 1st player is equal to size(#payoff#,1). The
          player whose strategies are the columns of the payoff matrix is
          considered as the 2nd player, i.e. the number of strategies of
          the 2nd player is equal to size(#payoff#,2).

  Output arguments
      #payoffR# [cell(:inf x :inf)] is the reduced payoff matrix of the
          game after elimination of the dominated strategies.

  Example
      % Example 1
      payoff={[1,0],[1,2],[0,1];
          [0,3],[0,1],[2,0]};
      payoffR1 = IESDS(payoff)
      % Example 2
      payoff={[2,0],[2,5],[1,1];
          [0,3],[0,4],[2,2]};
      payoffR2 = IESDS(payoff)
      % Example 3
      payoff={[2,0],[3,5],[4,4];
          [0,3],[2,1],[5,2]};
      payoffR3 = IESDS(payoff)
      % Example 4
      payoff={[2,1],[2,2];
          [3,4],[1,2];
          [1,2],[0,3]};
      payoffR4 = IESDS(payoff)
```

```
% Example 5
payoff={[3,2],[1,1],[1,0];
    [1,3],[0,2],[0,4];
    [2,-1],[-1,3],[2,0]};
payoffR5 = IESDS(payoff)
```


NashEquilibriumMixedStr

Help section of the NashEquilibriumMixedStr.m function

```
help NashEquilibriumMixedStr
```

```
Find Nash Equilibrium in Mixed Strategies

Syntax
    [#v#,#NE#] = NashEquilibriumMixedStr(#payoff#)

Input arguments
    #payoff# [cell(:inf x :inf)] is the payoff matrix of the game for
        which the equilibrium is queried. Each cell of #payoff#, i.e.
        #payoff#{i,j} contains a double array of size [double(1 x 2)],
        which contains the outcome (payoff) for 1st player and for 2nd
        player. The player whose strategies are the rows of the payoff
        matrix is considered as the 1st player, i.e. the number of
        strategies of the 1st player is equal to size(#payoff#,1). The
        player whose strategies are the columns of the payoff matrix is
        considered as the 2nd player, i.e. the number of strategies of
        the 2nd player is equal to size(#payoff#,2).

Output arguments
    #v# [double(2 x 1)] contains the values of the game. #v#(1,1) is the
        value of the game for the 1st player, and #v#(2,1) is the value
        of the game for the 2nd player
    #NE# [double(:inf x 2)] contains the Nash equilibria. #NE#{1,1} is of
        size [double(:inf x 1)] and provides the probabilities of the
        strategies of the 1st player. #NE#{2,1} is of size [double(:inf x
        1)] and provides the probabilities of the strategies of the 2nd
        player.

Example
    % Example 1
    payoff={[0,4],[4,0],[5,3];
        [4,0],[0,4],[5,3];
        [3,5],[3,5],[6,6]};
    [v,NE] = NashEquilibriumMixedStr(payoff);
    % Example 2
    payoff={[-1,1],[1,2];
        [0,0],[0,1]};
    [v,NE] = NashEquilibriumMixedStr(payoff);
    % Example 3 (Prisoner's Dilemma)
    payoff={[-1,-1],[-9,0];
        [0,-9],[-6,-6]};
    [v,NE] = NashEquilibriumMixedStr(payoff);
```

```
    % Example 4 (Matching Pennies)
    payoff={[-1,1],[1,-1];
        [1,-1],[-1,1]};
    [v,NE] = NashEquilibriumMixedStr(payoff);
    % Example 5 (Battle of the Sexes)
    payoff={[2,1],[0,0];
        [0,0],[1,2]};
    [v,NE] = NashEquilibriumMixedStr(payoff);
    % Example 6 (Stag-hunt)
    payoff={[7,7],[0,3];
        [3,0],[3,3]};
    [v,NE] = NashEquilibriumMixedStr(payoff);
    % Example 6 (Chicken)
    payoff={[-10,-10],[5,-5];
        [-5,5],[0,0]};
    [v,NE] = NashEquilibriumMixedStr(payoff);
    % Example 7 (rock-paper-scissors)
    payoff={[0,0],[-1,-1],[1,1];[1,1],[0,0],[-1,-1];[-1,-1],[1,1],[0,0]};
    [v,NE] = NashEquilibriumMixedStr(payoff);
```


NashEquilibriumPureStr

Help section of the NashEquilibriumPureStr.m function

```
help NashEquilibriumPureStr
```

```
Find Nash Equilibrium in Pure Strategies

Syntax
    #NE# = NashEquilibriumPureStr(#payoff#)

Input arguments
    #payoff# [cell(:inf x :inf)] is the payoff matrix of the game for
        which the equilibrium is queried. Each cell of #payoff#, i.e.
        #payoff#{i,j} contains a double array of size [double(1 x 2)],
        which contains the outcome (payoff) for 1st player and for 2nd
        player. The player whose strategies are the rows of the payoff
        matrix is considered as the 1st player, i.e. the number of
        strategies of the 1st player is equal to size(#payoff#,1). The
        player whose strategies are the columns of the payoff matrix is
        considered as the 2nd player, i.e. the number of strategies of
        the 2nd player is equal to size(#payoff#,2).

Output arguments
    #NE# [double(:inf x 2)] contains the Nash equilibria. #NE#(i,:)
        provides one Nash equilibrium of the game with matrix #payoff#.
        If there are more than one Nash equilibria, then size(#NE#,1)>1.
        At the ith Nash equilibrium (#NE#(i,:)), #NE#(i,1) shows the
        strategy of the 1st player (i.e. the id of the row of #payoff#
        where the Nash equilibrium exists) whereas #NE#(i,2) shows the
        strategy of the 2nd player (i.e. the id of the column of #payoff#
        where the Nash equilibrium exists).
```

Example

```
% Example 1
payoff={[0,4],[4,0],[5,3];
    [4,0],[0,4],[5,3];
    [3,5],[3,5],[6,6]};
NE = NashEquilibriumPureStr(payoff);
% Example 2
payoff={[-1,1],[1,2];
    [0,0],[0,1]};
NE = NashEquilibriumPureStr(payoff);
% Example 3 (Prisoner's Dilemma)
payoff={[-1,-1],[-9,0];
    [0,-9],[-6,-6]};
NE = NashEquilibriumPureStr(payoff);
% Example 4 (Matching Pennies)
payoff={[-1,1],[1,-1];
    [1,-1],[-1,1]};
NE = NashEquilibriumPureStr(payoff);
% Example 5 (Battle of the Sexes)
payoff={[2,1],[0,0];
    [0,0],[1,2]};
NE = NashEquilibriumPureStr(payoff);
% Example 6 (Stag-hunt)
payoff={[7,7],[0,3];
    [3,0],[3,3]};
NE = NashEquilibriumPureStr(payoff);
% Example 6 (Chicken)
payoff={[-10,-10],[5,-5];
    [-5,5],[0,0]};
NE = NashEquilibriumPureStr(payoff);
% Example 2
% Outcome matrix for 1st player (1st player strategies as rows and
% 2nd player strategies as columns)
payoff1=round(5*rand(10));
% Outcome matrix for 2nd player (2nd player strategies as rows and
% 1st player strategies as columns)
payoff2=round(5*rand(10));
% Generate the payoff matrix in format acceptable by
% NashEquilibriumPureStr.m
for i=1:size(payoff1,1)
    for j=1:size(payoff2,1)
        payoff{i,j}=[payoff1(i,j),payoff2(j,i)];
    end
end
NE = NashEquilibriumPureStr(payoff);
```

Function code

This contains the code of all the functions contained in this package.

Contents

IESDS

Code of the IESDS.m function

```
type IESDS
```

```
function payoffR = IESDS(payoff)
%
% Iterated Elimination of Strictly Dominated Strategies
%
% Syntax
%     #payoff# = IESDS(#payoff#)
%
% Input arguments
%     #payoff# [cell(:inf x :inf)] is the payoff matrix of the game for
%         which the equilibrium is queried. Each cell of #payoff#, i.e.
%         #payoff#{i,j} contains a double array of size [double(1 x 2)],
%         which contains the outcome (payoff) for 1st player and for 2nd
%         player. The player whose strategies are the rows of the payoff
%         matrix is considered as the 1st player, i.e. the number of
%         strategies of the 1st player is equal to size(#payoff#,1). The
%         player whose strategies are the columns of the payoff matrix is
%         considered as the 2nd player, i.e. the number of strategies of
%         the 2nd player is equal to size(#payoff#,2).
%
% Output arguments
%     #payoffR# [cell(:inf x :inf)] is the reduced payoff matrix of the
%         game after elimination of the dominated strategies.
%
% Example
%     % Example 1
%     payoff={[1,0],[1,2],[0,1];
%         [0,3],[0,1],[2,0]};
%     payoffR1 = IESDS(payoff)
%     % Example 2
%     payoff={[2,0],[2,5],[1,1];
%         [0,3],[0,4],[2,2]};
%     payoffR2 = IESDS(payoff)
%     % Example 3
%     payoff={[2,0],[3,5],[4,4];
%         [0,3],[2,1],[5,2]};
%     payoffR3 = IESDS(payoff)
%     % Example 4
%     payoff={[2,1],[2,2];
%         [3,4],[1,2];
```

```
%         [1,2],[0,3]};
%     payoffR4 = IESDS(payoff)
%     % Example 5
%     payoff={[3,2],[1,1],[1,0];
%         [1,3],[0,2],[0,4];
%         [2,-1],[-1,3],[2,0]};
%     payoffR5 = IESDS(payoff)
%
% __________________________________________________________________________
% Copyright (c) 2022 by George Papazafeiropoulos
% __________________________________________________________________________


payoff1=cellfun(@(v)v(1),payoff);
payoff2=cellfun(@(v)v(2),payoff)';

sw1=true;
sw2=true;
while sw1 || sw2
    % Scan strategies of 2nd player and check the outcomes of the 1st
    % player
    sw1=false;
    i=1;
    j=2;
    s1=size(payoff1,1);
    while i<=s1
        while j<=s1
            [payoff1Sorted,ind]=sortrows([payoff1(i,:);payoff1(j,:)],1);
            if all(diff(payoff1Sorted,1,1)>0,2)
                sw1=true;
                if ind(1)==1
                    payoff1(i,:)=[];
                    payoff2(:,i)=[];
                elseif ind(1)==2
                    payoff1(j,:)=[];
                    payoff2(:,j)=[];
                end
            else
            end
            j=j+1;
            s1=size(payoff1,1);
        end
        i=i+1;
        j=i+1;
        s1=size(payoff1,1);
    end
    % Scan strategies of 1st player and check the outcomes of the 2nd
    % player
    sw2=false;
    i=1;
    j=2;
    s2=size(payoff2,1);
    while i<=s2
        while j<=s2
            [payoff1Sorted,ind]=sortrows([payoff2(i,:);payoff2(j,:)],1);
            if all(diff(payoff1Sorted,1,1)>0,2)
                sw2=true;
                if ind(1)==1
                    payoff2(i,:)=[];
                    payoff1(:,i)=[];
                elseif ind(1)==2
```

```
                payoff2(j,:)=[];
                payoff1(:,j)=[];
            end
        else
        end
        j=j+1;
        s2=size(payoff2,1);
    end
    i=i+1;
    j=i+1;
    s2=size(payoff2,1);
  end
end
% Generate the payoff matrix in its input format
payoffR=cell(size(payoff1,1),size(payoff2,1));
for i=1:size(payoff1,1)
    for j=1:size(payoff2,1)
        payoffR{i,j}=[payoff1(i,j),payoff2(j,i)];
    end
end

end
```

NashEquilibriumMixedStr

Code of the NashEquilibriumMixedStr.m function

```
type NashEquilibriumMixedStr
```

```
function [v,NE] = NashEquilibriumMixedStr(payoff)
%
% Find Nash Equilibrium in Mixed Strategies
%
% Syntax
%     [#v#,#NE#] = NashEquilibriumMixedStr(#payoff#)
%
% Input arguments
%     #payoff# [cell(:inf x :inf)] is the payoff matrix of the game for
%         which the equilibrium is queried. Each cell of #payoff#, i.e.
%         #payoff#{i,j} contains a double array of size [double(1 x 2)],
%         which contains the outcome (payoff) for 1st player and for 2nd
%         player. The player whose strategies are the rows of the payoff
%         matrix is considered as the 1st player, i.e. the number of
%         strategies of the 1st player is equal to size(#payoff#,1). The
%         player whose strategies are the columns of the payoff matrix is
%         considered as the 2nd player, i.e. the number of strategies of
%         the 2nd player is equal to size(#payoff#,2).
%
% Output arguments
%     #v# [double(2 x 1)] contains the values of the game. #v#(1,1) is the
%         value of the game for the 1st player, and #v#(2,1) is the value
%         of the game for the 2nd player
%     #NE# [double(:inf x 2)] contains the Nash equilibria. #NE#{1,1} is of
%         size [double(:inf x 1)] and provides the probabilities of the
%         strategies of the 1st player. #NE#{2,1} is of size [double(:inf x
%         1)] and provides the probabilities of the strategies of the 2nd
%         player.
```

```
%
% Example
%     % Example 1
%     payoff={[0,4],[4,0],[5,3];
%         [4,0],[0,4],[5,3];
%         [3,5],[3,5],[6,6]};
%     [v,NE] = NashEquilibriumMixedStr(payoff);
%     % Example 2
%     payoff={[-1,1],[1,2];
%         [0,0],[0,1]};
%     [v,NE] = NashEquilibriumMixedStr(payoff);
%     % Example 3 (Prisoner's Dilemma)
%     payoff={[-1,-1],[-9,0];
%         [0,-9],[-6,-6]};
%     [v,NE] = NashEquilibriumMixedStr(payoff);
%     % Example 4 (Matching Pennies)
%     payoff={[-1,1],[1,-1];
%         [1,-1],[-1,1]};
%     [v,NE] = NashEquilibriumMixedStr(payoff);
%     % Example 5 (Battle of the Sexes)
%     payoff={[2,1],[0,0];
%         [0,0],[1,2]};
%     [v,NE] = NashEquilibriumMixedStr(payoff);
%     % Example 6 (Stag-hunt)
%     payoff={[7,7],[0,3];
%         [3,0],[3,3]};
%     [v,NE] = NashEquilibriumMixedStr(payoff);
%     % Example 6 (Chicken)
%     payoff={[-10,-10],[5,-5];
%         [-5,5],[0,0]};
%     [v,NE] = NashEquilibriumMixedStr(payoff);
%     % Example 7 (rock-paper-scissors)
%     payoff={[0,0],[-1,-1],[1,1];[1,1],[0,0],[-1,-1];[-1,-1],[1,1],[0,0]};
%     [v,NE] = NashEquilibriumMixedStr(payoff);
%
% __________________________________________________________________________
% Copyright (c) 2022 by George Papazafeiropoulos
% __________________________________________________________________________


payoff1=cellfun(@(v)v(1),payoff);
payoff2=cellfun(@(v)v(2),payoff)';
v=cell(2,1);
NE=cell(2,1);
% Solve for the mixed strategy of the 1st player
M=payoff2;
[c,r] = size(M);
A = [-M, ones(c,1)];
Aeq = [ones(1,r),0];
b = zeros(c,1);
beq = 1;
Aeq=[Aeq;A];
beq=[beq;b];
A=[];
b=[];
lb = [zeros(r,1);-inf];
ub = [ones(r,1);inf];
f = [ zeros(r,1);-1];
options = optimset('Display','off');
p = linprog(f,A,b,Aeq,beq,lb,ub,[],options);
if ~isempty(p)
```

```
    v{1,1} = p(r+1);
    NE{1,1} = p(1:r);
else
    v{1,1} = [];
    NE{1,1} = zeros(r,0);
end

% Solve for the mixed strategy of the 2nd player
M=payoff1;
[c,r] = size(M);
A = [-M, ones(c,1)];
Aeq = [ones(1,r),0];
b = zeros(c,1);
beq = 1;
Aeq=[Aeq;A];
beq=[beq;b];
A=[];
b=[];
lb = [zeros(r,1);-inf];
ub = [ones(r,1);inf];
f = [ zeros(r,1);-1];
options = optimset('Display','off');
p = linprog(f,A,b,Aeq,beq,lb,ub,[],options);
if ~isempty(p)
    v{2,1} = p(r+1);
    NE{2,1} = p(1:r);
else
    v{2,1} = [];
    NE{2,1} = zeros(r,0);
end

end
```

NashEquilibriumPureStr

Code of the NashEquilibriumPureStr.m function

```
type NashEquilibriumPureStr
```

```
function NE = NashEquilibriumPureStr(payoff)
%
% Find Nash Equilibrium in Pure Strategies
%
% Syntax
%     #NE# = NashEquilibriumPureStr(#payoff#)
%
% Input arguments
%     #payoff# [cell(:inf x :inf)] is the payoff matrix of the game for
%         which the equilibrium is queried. Each cell of #payoff#, i.e.
%         #payoff#{i,j} contains a double array of size [double(1 x 2)],
%         which contains the outcome (payoff) for 1st player and for 2nd
%         player. The player whose strategies are the rows of the payoff
%         matrix is considered as the 1st player, i.e. the number of
%         strategies of the 1st player is equal to size(#payoff#,1). The
%         player whose strategies are the columns of the payoff matrix is
%         considered as the 2nd player, i.e. the number of strategies of
%         the 2nd player is equal to size(#payoff#,2).
```

```
%
% Output arguments
%     #NE# [double(:inf x 2)] contains the Nash equilibria. #NE#(i,:)
%         provides one Nash equilibrium of the game with matrix #payoff#.
%         If there are more than one Nash equilibria, then size(#NE#,1)>1.
%         At the ith Nash equilibrium (#NE#(i,:)), #NE#(i,1) shows the
%         strategy of the 1st player (i.e. the id of the row of #payoff#
%         where the Nash equilibrium exists) whereas #NE#(i,2) shows the
%         strategy of the 2nd player (i.e. the id of the column of #payoff#
%         where the Nash equilibrium exists).
%
% Example
%     % Example 1
%     payoff={[0,4],[4,0],[5,3];
%         [4,0],[0,4],[5,3];
%         [3,5],[3,5],[6,6]};
%     NE = NashEquilibriumPureStr(payoff);
%     % Example 2
%     payoff={[-1,1],[1,2];
%         [0,0],[0,1]};
%     NE = NashEquilibriumPureStr(payoff);
%     % Example 3 (Prisoner's Dilemma)
%     payoff={[-1,-1],[-9,0];
%         [0,-9],[-6,-6]};
%     NE = NashEquilibriumPureStr(payoff);
%     % Example 4 (Matching Pennies)
%     payoff={[-1,1],[1,-1];
%         [1,-1],[-1,1]};
%     NE = NashEquilibriumPureStr(payoff);
%     % Example 5 (Battle of the Sexes)
%     payoff={[2,1],[0,0];
%         [0,0],[1,2]};
%     NE = NashEquilibriumPureStr(payoff);
%     % Example 6 (Stag-hunt)
%     payoff={[7,7],[0,3];
%         [3,0],[3,3]};
%     NE = NashEquilibriumPureStr(payoff);
%     % Example 6 (Chicken)
%     payoff={[-10,-10],[5,-5];
%         [-5,5],[0,0]};
%     NE = NashEquilibriumPureStr(payoff);
%     % Example 2
%     % Outcome matrix for 1st player (1st player strategies as rows and
%     % 2nd player strategies as columns)
%     payoff1=round(5*rand(10));
%     % Outcome matrix for 2nd player (2nd player strategies as rows and
%     % 1st player strategies as columns)
%     payoff2=round(5*rand(10));
%     % Generate the payoff matrix in format acceptable by
%     % NashEquilibriumPureStr.m
%     for i=1:size(payoff1,1)
%         for j=1:size(payoff2,1)
%             payoff{i,j}=[payoff1(i,j),payoff2(j,i)];
%         end
%     end
%     NE = NashEquilibriumPureStr(payoff);
%
% __________________________________________________________________________
% Copyright (c) 2022 by George Papazafeiropoulos
% __________________________________________________________________________
```

```
payoff1=cellfun(@(v)v(1),payoff);
payoff2=cellfun(@(v)v(2),payoff)';
NE=zeros(0,2);
% ith strategy of 2nd player
for i=1:size(payoff1,2)
    a=find(payoff1(:,i)==max(payoff1(:,i)));
    best1{i}=[a,i*ones(size(a))];
end
% ith strategy of 1st player
for i=1:size(payoff2,2)
    a=find(payoff2(:,i)==max(payoff2(:,i)));
    best2{i}=[i*ones(size(a)),a];
end
for i=1:size(payoff1,2)
    for j=1:size(payoff2,2)
        a=intersect(best1{i},best2{j},'rows');
        if ~isempty(a)
            NE=[NE;a];
        end
    end
end
end
```

Examples of Iterated Elimination of Strictly Dominated Strategies

Various examples are presented where iterated elimination of strictly dominated strategies takes place, using the function IESDS.m

Contents

Example 1

Payoff matrix

```
payoff={[1,0],[1,2],[0,1];
    [0,3],[0,1],[2,0]};
celldisp(payoff)
```

```
payoff{1,1} =

     1     0


payoff{2,1} =

     0     3


payoff{1,2} =

     1     2


payoff{2,2} =

     0     1


payoff{1,3} =

     0     1


payoff{2,3} =

     2     0
```

Reduced payoff matrix after iterated elimination of strictly dominated strategies procedure

```
payoffR1 = IESDS(payoff);
celldisp(payoffR1)
```

```
payoffR1{1} =

     1     2
```

Example 2

Payoff matrix

```
payoff={[2,0],[2,5],[1,1];
    [0,3],[0,4],[2,2]};
celldisp(payoff)
```

```
payoff{1,1} =

     2     0



payoff{2,1} =

     0     3



payoff{1,2} =

     2     5



payoff{2,2} =

     0     4



payoff{1,3} =

     1     1



payoff{2,3} =
```

```
     2     2
```

Reduced payoff matrix after iterated elimination of strictly dominated strategies procedure

```
payoffR2 = IESDS(payoff);
celldisp(payoffR2)
```

```
payoffR2{1} =

     2     5
```

Example 3

Payoff matrix

```
payoff={[2,0],[3,5],[4,4];
    [0,3],[2,1],[5,2]};
celldisp(payoff)
```

```
payoff{1,1} =

     2     0



payoff{2,1} =

     0     3



payoff{1,2} =

     3     5



payoff{2,2} =

     2     1



payoff{1,3} =

     4     4
```

```
payoff{2,3} =

     5     2
```

Reduced payoff matrix after iterated elimination of strictly dominated strategies procedure

```
payoffR3 = IESDS(payoff);
celldisp(payoffR3)
```

```
payoffR3{1,1} =

     2     0


payoffR3{2,1} =

     0     3


payoffR3{1,2} =

     3     5


payoffR3{2,2} =

     2     1


payoffR3{1,3} =

     4     4


payoffR3{2,3} =

     5     2
```

Example 4

Payoff matrix

```
payoff={[2,1],[2,2];
    [3,4],[1,2];
```

```
    [1,2],[0,3]};
celldisp(payoff)
```

```
payoff{1,1} =

     2     1



payoff{2,1} =

     3     4



payoff{3,1} =

     1     2



payoff{1,2} =

     2     2



payoff{2,2} =

     1     2



payoff{3,2} =

     0     3
```

Reduced payoff matrix after iterated elimination of strictly dominated strategies procedure

```
payoffR4 = IESDS(payoff);
celldisp(payoffR4)
```

```
payoffR4{1,1} =

     2     1



payoffR4{2,1} =

     3     4
```

```
payoffR4{1,2} =

     2     2


payoffR4{2,2} =

     1     2
```

Example 5

Payoff matrix

```
payoff={[3,2],[1,1],[1,0];
    [1,3],[0,2],[0,4];
    [2,-1],[-1,3],[2,0]};
celldisp(payoff)
```

```
payoff{1,1} =

     3     2


payoff{2,1} =

     1     3


payoff{3,1} =

     2    -1


payoff{1,2} =

     1     1


payoff{2,2} =

     0     2


payoff{3,2} =

    -1     3
```

```
payoff{1,3} =

     1     0


payoff{2,3} =

     0     4


payoff{3,3} =

     2     0
```

Reduced payoff matrix after iterated elimination of strictly dominated strategies procedure

```
payoffR5 = IESDS(payoff);
celldisp(payoffR5)
```

```
payoffR5{1} =

     3     2
```

Published with MATLAB® R2017b

Examples of Nash Equilibrium for Mixed Strategies

Various examples are presented where Nash equilibrium is queried for the case of mixed strategies, using the function NashEquilibriumMixedStr.m

Contents

Example 1

Payoff matrix

```
payoff={[0,4],[4,0],[5,3];
    [4,0],[0,4],[5,3];
    [3,5],[3,5],[6,6]};
celldisp(payoff)
```

```
payoff{1,1} =

     0     4


payoff{2,1} =

     4     0


payoff{3,1} =

     3     5


payoff{1,2} =

     4     0


payoff{2,2} =

     0     4
```

```
payoff{3,2} =

     3     5


payoff{1,3} =

     5     3


payoff{2,3} =

     5     3


payoff{3,3} =

     6     6
```

Nash equilibrium for mixed strategy

```
[v,NE] = NashEquilibriumMixedStr(payoff);
celldisp(v)
celldisp(NE)
```

```
v{1} =

     []

v{2} =

     []

NE{1} =

     []

NE{2} =

     []
```

Example 2

Payoff matrix

```
payoff={[-1,1],[1,2];
    [0,0],[0,1]};
celldisp(payoff)
```

```
payoff{1,1} =

    -1     1


payoff{2,1} =

     0     0


payoff{1,2} =

     1     2


payoff{2,2} =

     0     1
```

Nash equilibrium for mixed strategy

```
[v,NE] = NashEquilibriumMixedStr(payoff);
celldisp(v)
celldisp(NE)
```

```
v{1} =

     []


v{2} =

     0


NE{1} =

     []


NE{2} =
```

```
    0.5000
    0.5000
```

Example 3 (Prisoner's Dilemma)

Payoff matrix

```
payoff={[-1,-1],[-9,0];
    [0,-9],[-6,-6]};
celldisp(payoff)
```

```
payoff{1,1} =

    -1    -1


payoff{2,1} =

     0    -9


payoff{1,2} =

    -9     0


payoff{2,2} =

    -6    -6
```

Nash equilibrium for mixed strategy

```
[v,NE] = NashEquilibriumMixedStr(payoff);
celldisp(v)
celldisp(NE)
```

```
v{1} =

     []


v{2} =

     []
```

```
NE{1} =

     []


NE{2} =

     []
```

Example 4 (Matching Pennies)

Payoff matrix

```
payoff={[-1,1],[1,-1];
    [1,-1],[-1,1]};
celldisp(payoff)
```

```
payoff{1,1} =

    -1     1



payoff{2,1} =

     1    -1



payoff{1,2} =

     1    -1



payoff{2,2} =

    -1     1
```

Nash equilibrium for mixed strategy

```
[v,NE] = NashEquilibriumMixedStr(payoff);
celldisp(v)
celldisp(NE)
```

```
v{1} =

     0
```

```
v{2} =

     0



NE{1} =

    0.5000
    0.5000



NE{2} =

    0.5000
    0.5000
```

Example 5 (Battle of the Sexes)

Payoff matrix

```
payoff={[2,1],[0,0];
    [0,0],[1,2]};
celldisp(payoff)
```

```
payoff{1,1} =

     2     1



payoff{2,1} =

     0     0



payoff{1,2} =

     0     0



payoff{2,2} =

     1     2
```

Nash equilibrium for mixed strategy

```
[v,NE] = NashEquilibriumMixedStr(payoff);
celldisp(v)
celldisp(NE)
```

```
v{1} =

    0.6667



v{2} =

    0.6667



NE{1} =

    0.6667
    0.3333



NE{2} =

    0.3333
    0.6667
```

Example 6 (Stag-hunt)

Payoff matrix

```
payoff={[7,7],[0,3];
    [3,0],[3,3]};
celldisp(payoff)
```

```
payoff{1,1} =

     7     7



payoff{2,1} =

     3     0



payoff{1,2} =
```

```
     0     3


payoff{2,2} =

     3     3
```

Nash equilibrium for mixed strategy

```
[v,NE] = NashEquilibriumMixedStr(payoff);
celldisp(v)
celldisp(NE)
```

```
v{1} =

    3.0000


v{2} =

    3.0000


NE{1} =

    0.4286
    0.5714


NE{2} =

    0.4286
    0.5714
```

Example 7 (Chicken)

Payoff matrix

```
payoff={[-10,-10],[5,-5];
    [-5,5],[0,0]};
celldisp(payoff)
```

```
payoff{1,1} =
```

```
   -10   -10


payoff{2,1} =

    -5     5


payoff{1,2} =

     5    -5


payoff{2,2} =

     0     0
```

Nash equilibrium for mixed strategy

```
[v,NE] = NashEquilibriumMixedStr(payoff);
celldisp(v)
celldisp(NE)
```

```
v{1} =

   -2.5000


v{2} =

   -2.5000


NE{1} =

    0.5000
    0.5000


NE{2} =

    0.5000
    0.5000
```

Example 8 (rock-paper-scissors)

Payoff matrix

```
payoff={[0,0],[-1,-1],[1,1];[1,1],[0,0],[-1,-1];[-1,-1],[1,1],[0,0]};
celldisp(payoff)
```

```
payoff{1,1} =

     0     0


payoff{2,1} =

     1     1


payoff{3,1} =

    -1    -1


payoff{1,2} =

    -1    -1


payoff{2,2} =

     0     0


payoff{3,2} =

     1     1


payoff{1,3} =

     1     1


payoff{2,3} =

    -1    -1


payoff{3,3} =

     0     0
```

Nash equilibrium for mixed strategy

```
[v,NE] = NashEquilibriumMixedStr(payoff);
celldisp(v)
celldisp(NE)
```

```
v{1} =

  -1.1102e-16



v{2} =

   1.1102e-16



NE{1} =

    0.3333
    0.3333
    0.3333



NE{2} =

    0.3333
    0.3333
    0.3333
```

Examples of Nash Equilibrium for Pure Strategies

Various examples are presented where Nash equilibrium is queried for the case of pure strategies, using the function NashEquilibriumPureStr.m

Contents

Example 1

Payoff matrix

```
payoff={[0,4],[4,0],[5,3];
    [4,0],[0,4],[5,3];
    [3,5],[3,5],[6,6]};
celldisp(payoff)
```

```
payoff{1,1} =

     0     4


payoff{2,1} =

     4     0


payoff{3,1} =

     3     5


payoff{1,2} =

     4     0


payoff{2,2} =

     0     4
```

```
payoff{3,2} =

     3     5


payoff{1,3} =

     5     3


payoff{2,3} =

     5     3


payoff{3,3} =

     6     6
```

Nash equilibrium for pure strategy

```
NE = NashEquilibriumPureStr(payoff)
```

```
NE =

     3     3
```

Game value

```
v=zeros(0,2);
for i=1:size(NE,1)
    v=[v;payoff{NE(i,1),NE(i,2)}];
end
```

Game value for player 1

```
v1=v(:,1)
```

```
v1 =

     6
```

Game value for player 2

```
v2=v(:,2)
```

```
v2 =

     6
```

Example 2

Payoff matrix

```
payoff={[-1,1],[1,2];
    [0,0],[0,1]};
celldisp(payoff)
```

```
payoff{1,1} =

    -1     1


payoff{2,1} =

     0     0


payoff{1,2} =

     1     2


payoff{2,2} =

     0     1
```

Nash equilibrium for pure strategy

```
NE = NashEquilibriumPureStr(payoff)
```

```
NE =

     1     2
```

Game value

```
v=zeros(0,2);
for i=1:size(NE,1)
    v=[v;payoff{NE(i,1),NE(i,2)}];
end
```

Game value for player 1

```
v1=v(:,1)
```

```
v1 =

     1
```

Game value for player 2

```
v2=v(:,2)
```

```
v2 =

     2
```

Example 3 (Prisoner's Dilemma)

Payoff matrix

```
payoff={[-1,-1],[-9,0];
    [0,-9],[-6,-6]};
celldisp(payoff)
```

```
payoff{1,1} =

    -1    -1


payoff{2,1} =

     0    -9


payoff{1,2} =

    -9     0
```

```
payoff{2,2} =

    -6    -6
```

Nash equilibrium for pure strategy

```
NE = NashEquilibriumPureStr(payoff)
```

```
NE =

     2     2
```

Game value

```
v=zeros(0,2);
for i=1:size(NE,1)
    v=[v;payoff{NE(i,1),NE(i,2)}];
end
```

Game value for player 1

```
v1=v(:,1)
```

```
v1 =

    -6
```

Game value for player 2

```
v2=v(:,2)
```

```
v2 =

    -6
```

Example 4 (Matching Pennies)

Payoff matrix

```
payoff={[-1,1],[1,-1];
    [1,-1],[-1,1]};
celldisp(payoff)
```

```
payoff{1,1} =

    -1     1


payoff{2,1} =

     1    -1


payoff{1,2} =

     1    -1


payoff{2,2} =

    -1     1
```

Nash equilibrium for pure strategy

```
NE = NashEquilibriumPureStr(payoff)
```

```
NE =

  0×2 empty double matrix
```

Game value

```
v=zeros(0,2);
for i=1:size(NE,1)
    v=[v;payoff{NE(i,1),NE(i,2)}];
end
```

Game value for player 1

```
v1=v(:,1)
```

```
v1 =
```

```
  0×1 empty double column vector
```

Game value for player 2

```
v2=v(:,2)
```

```
v2 =

  0×1 empty double column vector
```

Example 5 (Battle of the Sexes)

Payoff matrix

```
payoff={[2,1],[0,0];
    [0,0],[1,2]};
celldisp(payoff)
```

```
payoff{1,1} =

     2     1



payoff{2,1} =

     0     0



payoff{1,2} =

     0     0



payoff{2,2} =

     1     2
```

Nash equilibrium for pure strategy

```
NE = NashEquilibriumPureStr(payoff)
```

```
NE =
```

```
     1     1
     2     2
```

Game value

```
v=zeros(0,2);
for i=1:size(NE,1)
    v=[v;payoff{NE(i,1),NE(i,2)}];
end
```

Game value for player 1

```
v1=v(:,1)
```

```
v1 =

     2
     1
```

Game value for player 2

```
v2=v(:,2)
```

```
v2 =

     1
     2
```

Example 6 (Stag-hunt)

Payoff matrix

```
payoff={[7,7],[0,3];
    [3,0],[3,3]};
celldisp(payoff)
```

```
payoff{1,1} =

     7     7


payoff{2,1} =
```

```
     3     0
```

```
payoff{1,2} =

     0     3
```

```
payoff{2,2} =

     3     3
```

Nash equilibrium for pure strategy

```
NE = NashEquilibriumPureStr(payoff)
```

```
NE =

     1     1
     2     2
```

Game value

```
v=zeros(0,2);
for i=1:size(NE,1)
    v=[v;payoff{NE(i,1),NE(i,2)}];
end
```

Game value for player 1

```
v1=v(:,1)
```

```
v1 =

     7
     3
```

Game value for player 2

```
v2=v(:,2)
```

```
v2 =
```

```
     7
     3
```

Example 7 (Chicken)

Payoff matrix

```
payoff={[-10,-10],[5,-5];
    [-5,5],[0,0]};
celldisp(payoff)
```

```
payoff{1,1} =

   -10   -10


payoff{2,1} =

    -5     5


payoff{1,2} =

     5    -5


payoff{2,2} =

     0     0
```

Nash equilibrium for pure strategy

```
NE = NashEquilibriumPureStr(payoff)
```

```
NE =

     2     1
     1     2
```

Game value

```
v=zeros(0,2);
for i=1:size(NE,1)
    v=[v;payoff{NE(i,1),NE(i,2)}];
```

```
end
```

Game value for player 1

```
v1=v(:,1)
```

```
v1 =

    -5
     5
```

Game value for player 2

```
v2=v(:,2)
```

```
v2 =

     5
    -5
```

Example 8 (rock-paper-scissors)

Payoff matrix

```
payoff={[0,0],[-1,-1],[1,1];[1,1],[0,0],[-1,-1];[-1,-1],[1,1],[0,0]};
celldisp(payoff)
```

```
payoff{1,1} =

     0     0


payoff{2,1} =

     1     1


payoff{3,1} =

    -1    -1


payoff{1,2} =

    -1    -1
```

```
payoff{2,2} =

     0     0


payoff{3,2} =

     1     1


payoff{1,3} =

     1     1


payoff{2,3} =

    -1    -1


payoff{3,3} =

     0     0
```

Nash equilibrium for pure strategy

```
NE = NashEquilibriumPureStr(payoff)
```

```
NE =

     2     1
     3     2
     1     3
```

Game value

```
v=zeros(0,2);
for i=1:size(NE,1)
    v=[v;payoff{NE(i,1),NE(i,2)}];
end
```

Game value for player 1

```
v1=v(:,1)
```

```
v1 =

     1
     1
     1
```

Game value for player 2

```
v2=v(:,2)
```

```
v2 =

     1
     1
     1
```

Published with MATLAB® R2017b