“By Example” introduction into the Mathematica PairwiseComparisons` Package

Konrad Kułakowski (C) 2014 - 2017
e-mail: konrad@kulakowski.org

To install the package on the computer start Mathematica and choose File/Install..., then select PairwiseComparisons.m as the package source and press OK.

To install the package at the Raspberry PI just copy them into /opt/Wolfram/WolframEngine/10.0/AddOns/ExtraPackages and restart the application

Introduction

Loading the package

In[39]:=
```
<< PairwiseComparisons`;
```

Printing function usage

In[40]:=
```
? GeometricRank
```

GeometricRank[M] returns rank list given as geometric means of rows of the matrix M

Printing full (with implementation) function usage

In[41]:=
```
?? GeometricRank
```

GeometricRank[M] returns rank list given as geometric means of rows of the matrix M

```
GeometricRank[PairwiseComparisons`Private`matrix_] :=
 (GeometricMean[#1] &) /@ PairwiseComparisons`Private`matrix
```

GeometricRank[M] returns rank list given as geometric means of rows of the matrix M

```
GeometricRank[PairwiseComparisons`Private`matrix_] :=
 (GeometricMean[#1] &) /@ PairwiseComparisons`Private`matrix
```

GeometricRank[M] returns rank list given as geometric means of rows of the matrix M

```
GeometricRank[PairwiseComparisons`Private`matrix_] :=
 (GeometricMean[#1] &) /@ PairwiseComparisons`Private`matrix
```

```
GeometricRank[PairwiseComparisons`Private`matrix_] :=
 (GeometricMean[#1] &) /@ PairwiseComparisons`Private`matrix
```

Print all the public functions in the package

In[42]:=
```
? PairwiseComparisons`*
```

PairwiseComparisons`

AHP	KendallTauDistance
AIJadd	KoczkodajConsistentTriad
AIJgeom	KoczkodajIdx
ConsistentMatrixFromRank	KoczkodajIdxRecip
COP1Check	KoczkodajImproveMatrixStep
COP1ViolationList	KoczkodajTheMostInconsistentTriad
COP2Check	KoczkodajTheWorstTriad
COP2ViolationList	KoczkodajTheWorstTriads
DeleteColumns	KoczkodajTriadIdx
DeleteRows	KoczkodajTriadInconsistency
DeleteRowsAndColumns	ListOfPossibleRationalEntries
EigenvalueRank	LocalDiscrepancyMatrix
EigenvalueRankSym	NormalizedKendallTauDistance
ErrorMatrix	PrincipalEigenValue
FundamentalScale	PrincipalEigenValueSym
GCI	PrincipalEigenVector
GeometricRank	PrincipalEigenVectorSym
GeometricRescaledRank	RandomBoolean
GetMatrixEntry	RandomMatrix
GlobalDiscrepancy	RandomMatrixEntry
HarkerMatrix	RandomRankingPattern
HREConstantTermVector	RandomRationalMatrix
HREFullRank	RandomRationalMatrixEntry
HREGeomConstantTermVector	RandomRationalUniformRankingPattern
HREGeomFullRank	RankOrder
HREGeomIntermediateRank	RankToVector
HREGeomMatrix	RationalScale
HREGeomPartialRank	RecreatePCMatrix
HREGeomRescaledRank	SaatyIdx
HREMatrix	SaatyIdxSym
HREPartialRank	SetDiagonal
HRERescaledRank	VersionPC

In[43]:=

Assign the matrix to the variable M

In[44]:= $M = \begin{pmatrix} 1 & 2 & 3 \\ \frac{1}{2} & 1 & 4 \\ \frac{1}{3} & \frac{1}{4} & 1 \end{pmatrix}$

Out[44]= $\left\{\{1, 2, 3\}, \left\{\frac{1}{2}, 1, 4\right\}, \left\{\frac{1}{3}, \frac{1}{4}, 1\right\}\right\}$

Eigenvalue based method

Calculate the principal eigenvalue of the matrix M

In[45]:= `PrincipalEigenValue[M]`

Out[45]= `3.10785`

Calculate the principal eigenvector of the matrix M

In[46]:= `PrincipalEigenVector[M]`

Out[46]= PrincipalEigenVector$\left[\left\{\{1, 2, 3\}, \left\{\frac{1}{2}, 1, 4\right\}, \left\{\frac{1}{3}, \frac{1}{4}, 1\right\}\right\}\right]$

Calculate the value of the Saaty (eigenvalue based) inconsistency index of M

In[47]:= `SaatyIdx[M]`

Out[47]= `0.0539237`

Calculate the eigenvalue based ranking based on M

In[48]:= `EigenvalueRank[M]`

Out[48]= `{0.517134, 0.35856, 0.124306}`

Calculate rank for the three different criteria using AHP (see Belton and Gear, On a short-coming of Saaty's method of analytic hierarchies, 1983)

In[49]:= $A_1 = \begin{pmatrix} 1 & 1/9 & 1 \\ 9 & 1 & 9 \\ 1 & 1/9 & 1 \end{pmatrix};$

In[50]:= $A_2 = \begin{pmatrix} 1 & 9 & 9 \\ 1/9 & 1 & 1 \\ 1/9 & 1 & 1 \end{pmatrix};$

In[51]:= $A_3 = \begin{pmatrix} 1 & 8/9 & 8 \\ 9/8 & 1 & 9 \\ 1/8 & 1/9 & 1 \end{pmatrix};$

In[52]:= $A_{criteria} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix};$

then the result is:

In[53]:= $\text{AHP}[A_{criteria}, A_1, A_2, A_3]$

Out[53]= `{0.451178, 0.469697, 0.0791246}`

In[54]:=

Geometric Mean method (Logarithmic Least Square Method)

Calculate the geometric mean based ranking based on M

In[55]:= `N[GeometricRank[M]]`

Out[55]= `{1.81712, 1.25992, 0.43679}`

Calculate the geometric mean based ranking based on M rescaled so that the sum of entries is 1

In[56]:=
```
N[GeometricRescaledRank[M]]
```
Out[56]=
```
{0.517134, 0.35856, 0.124306}
```

Please note that if you want to have numeric (not symbolic) output the input data also should be "numeric".
Thus, it is not a bad idea to add a numeric conversion N@ to the input matrix. For example:

In[57]:=
```
N[GeometricRank[N@M]]
```
Out[57]=
```
{1.81712, 1.25992, 0.43679}
```

The above applies to all the methods in the package, except eigenvalue based methods such as PrincipalEigenValue,
PrincipalEigenVector, SaatyIdx and EigenvalueRank, which are "by design" numeric.

The geometric consisteny index (GCI) as defined in "Aguaron & Moreno-Jimenez, The geometric consistency index: Approximated thresholds" can be computed by calling:

In[58]:=
```
N@GCI[M, GeometricRank[M], 10]
```
Out[58]=
```
0.0604831
```

Heuristic Rating Estimation Method (additive)

Calculate the HRE Matrix for the given matrix M where the unknown concepts are $\{c_1, c_2, c_3\}$ and the known concepts are $\{c_4 = 5, c_5 = 9\}$. It is assumed that the unknown concepts has value 0 whilst the known concepts have the values greater than 0.

Further references could be found in papers :
* Konrad Kułakowski,
Heuristic Rating Estimation Approach to The Pairwise Comparisons Method
http : // arxiv.org/abs/1309.0386
* Konrad Kułakowski, A heuristic rating estimation algorithm for the pairwise comparisons method
http : // dx.doi.org/10.1007/s10100 – 013 – 0311 – x

In[59]:=

$$\text{HREMatrix}\left[\begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} & m_{1,4} & m_{1,5} \\ m_{2,1} & m_{2,2} & m_{2,3} & m_{2,4} & m_{2,5} \\ m_{3,1} & m_{3,2} & m_{3,3} & m_{3,4} & m_{3,5} \\ m_{4,1} & m_{4,2} & m_{4,3} & m_{4,4} & m_{4,5} \\ m_{5,1} & m_{5,2} & m_{5,3} & m_{5,4} & m_{5,5} \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 5 \\ 9 \end{pmatrix}\right]$$

Out[59]=

$$\left\{\left\{1, -\frac{m_{1,2}}{4}, -\frac{m_{1,3}}{4}\right\}, \left\{-\frac{m_{2,1}}{4}, 1, -\frac{m_{2,3}}{4}\right\}, \left\{-\frac{m_{3,1}}{4}, -\frac{m_{3,2}}{4}, 1\right\}\right\}$$

Calculate the HRE constant term vector for M
where C_U equals $\{c_1, c_2, c_3\}$ and C_K equals $\{c_4 = 5, c_5 = 9\}$

In[60]:=

$$\text{HREConstantTermVector}\left[\begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} & m_{1,4} & m_{1,5} \\ m_{2,1} & m_{2,2} & m_{2,3} & m_{2,4} & m_{2,5} \\ m_{3,1} & m_{3,2} & m_{3,3} & m_{3,4} & m_{3,5} \\ m_{4,1} & m_{4,2} & m_{4,3} & m_{4,4} & m_{4,5} \\ m_{5,1} & m_{5,2} & m_{5,3} & m_{5,4} & m_{5,5} \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 5 \\ 9 \end{pmatrix}\right]$$

Out[60]=

$$\left\{\left\{\frac{5\, m_{1,4}}{4} + \frac{9\, m_{1,5}}{4}\right\}, \left\{\frac{5\, m_{2,4}}{4} + \frac{9\, m_{2,5}}{4}\right\}, \left\{\frac{5\, m_{3,4}}{4} + \frac{9\, m_{3,5}}{4}\right\}\right\}$$

Auxiliary function that transform an upper triangle matrix into a full and reciprocal matrix

In[61]:= M = RecreatePCMatrix[$\begin{pmatrix} 1 & \frac{3}{5} & \frac{4}{7} & \frac{5}{8} & \frac{5}{10} \\ 0 & 1 & \frac{5}{7} & \frac{5}{2} & \frac{10}{3} \\ 0 & 0 & 1 & \frac{7}{2} & 4 \\ 0 & 0 & 0 & 1 & \frac{4}{3} \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$]

Out[61]= $\{\{1, \frac{3}{5}, \frac{4}{7}, \frac{5}{8}, \frac{1}{2}\}, \{\frac{5}{3}, 1, \frac{5}{7}, \frac{5}{2}, \frac{10}{3}\}, \{\frac{7}{4}, \frac{7}{5}, 1, \frac{7}{2}, 4\}, \{\frac{8}{5}, \frac{2}{5}, \frac{2}{7}, 1, \frac{4}{3}\}, \{2, \frac{3}{10}, \frac{1}{4}, \frac{3}{4}, 1\}\}$

In[62]:=

Defining known and unknown alternatives. It is assumed that c_2 and c_3 are known and equal 5 and 7 correspondingly

In[63]:= mk = $\begin{pmatrix} 0 \\ 5 \\ 7 \\ 0 \\ 0 \end{pmatrix}$

Out[63]= {{0}, {5}, {7}, {0}, {0}}

Calculate the HRE ranking vector for unknown alternatives i.e. for c_1, c_4 and c_5 only

```
In[64]:= mu = N[HREPartialRank[M, mk]]
Out[64]= {{2.52765}, {2.88338}, {2.61696}}
```

Calculate the full HRE ranking for the given input matrix M and the vector mk

```
In[65]:= mu = N[HREFullRank[M, mk]]
Out[65]= {2.52765, 5., 7., 2.88338, 2.61696}
```

Calculate the full HRE ranking rescaled so that all its entries sum up to 1

```
In[66]:= mu = N[HRERescaledRank[M, mk]]
Out[66]= {0.126206, 0.249651, 0.349511, 0.143967, 0.130665}
```

Heuristic Rating Estimation Method (multiplicative/geometric)

Further references could be found in papers:
* Konrad Kułakowski, Grobler-Dębska Katarzyna, Wąs Jarosław, Heuristic rating estimation - geometric approach, http://arxiv.org/abs/1404.6981

In[67]:= **HREGeomMatrix**[$\begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} & m_{1,4} & m_{1,5} \\ m_{2,1} & m_{2,2} & m_{2,3} & m_{2,4} & m_{2,5} \\ m_{3,1} & m_{3,2} & m_{3,3} & m_{3,4} & m_{3,5} \\ m_{4,1} & m_{4,2} & m_{4,3} & m_{4,4} & m_{4,5} \\ m_{5,1} & m_{5,2} & m_{5,3} & m_{5,4} & m_{5,5} \end{pmatrix}$, $\begin{pmatrix} 0 \\ 0 \\ 0 \\ q_4 \\ q_5 \end{pmatrix}$]

Out[67]= {{4, -1, -1}, {-1, 4, -1}, {-1, -1, 4}}

Calculate the HRE constant term vector

In[68]:= **HREGeomConstantTermVector**$\left[\begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} & m_{1,4} & m_{1,5} \\ m_{2,1} & m_{2,2} & m_{2,3} & m_{2,4} & m_{2,5} \\ m_{3,1} & m_{3,2} & m_{3,3} & m_{3,4} & m_{3,5} \\ m_{4,1} & m_{4,2} & m_{4,3} & m_{4,4} & m_{4,5} \\ m_{5,1} & m_{5,2} & m_{5,3} & m_{5,4} & m_{5,5} \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ q_4 \\ q_5 \end{pmatrix}\right]$

Out[68]= $\left\{\left\{\frac{\text{Log}[q_4\, q_5\, m_{1,2}\, m_{1,3}\, m_{1,4}\, m_{1,5}]}{\text{Log}[10]}\right\}, \left\{\frac{\text{Log}[q_4\, q_5\, m_{2,1}\, m_{2,3}\, m_{2,4}\, m_{2,5}]}{\text{Log}[10]}\right\}, \left\{\frac{\text{Log}[q_4\, q_5\, m_{3,1}\, m_{3,2}\, m_{3,4}\, m_{3,5}]}{\text{Log}[10]}\right\}\right\}$

Calculate the HRE geometric ranking vector for unknown alternatives i.e. for c_1, c_4 and c_5 only

In[69]:= `mu = N[HREGeomPartialRank[M, mk]]`

Out[69]= `{{2.11273}, {2.49035}, {2.13344}}`

Calculate the full HRE geometric ranking for the given input matrix M and the vector mk

In[70]:= `mu = N[HREGeomFullRank[M, mk]]`

Out[70]= `{2.11273, 5., 7., 2.49035, 2.13344}`

Calculate the full HRE geometric ranking, rescaled so that all its entries sum up to 1

In[71]:= `mu = N[HREGeomRescaledRank[M, mk]]`

Out[71]= `{0.11276, 0.266858, 0.373602, 0.132914, 0.113866}`

Show intermediate (before raisng up to the power) HRE geometric partial rank vector

In[72]:= **`N[HREGeomIntermediateRank[M, mk]]`**

Out[72]= `{{0.324843}, {0.396261}, {0.329081}}`

Bana e Costa and Vansnick's Condition of Order Preservation test

Let calculate the eigenvalue ranking

In[73]:= `rank = EigenvalueRank[M]`

Out[73]= `{0.119092, 0.27476, 0.356526, 0.130954, 0.118669}`

.... and check whether the first Bana e Costa and Vansnick condition “condition of order preservation - COP” is satisfied

In[74]:= `COP1Check[M, mu]`

Out[74]= `True`

then check whether the second Bana e Costa and Vansnick condition (preserving intensity of preferences postulate) is satisfied

In[75]:= `COP2Check[M, mu]`

Out[75]= `False`

Prints the list of pairs for which the 1st COP is not satisfied

In[76]:= `COP1ViolationList[M, mu]`

Out[76]= `{}`

Prints the list of pairs of pairs for which the 2nd COP is not satisfied

In[77]:=
```
COP2ViolationList[M, mu]
```

Out[77]=
```
{{False, {{1, 2}, {1, 5}}}, {False, {{1, 2}, {4, 2}}}, {False, {{1, 2}, {5, 2}}},
 {False, {{1, 3}, {1, 5}}}, {False, {{1, 3}, {4, 2}}}, {False, {{1, 3}, {4, 3}}},
 {False, {{1, 3}, {5, 2}}}, {False, {{1, 3}, {5, 3}}}, {False, {{1, 4}, {1, 5}}},
 {False, {{1, 4}, {2, 3}}}, {False, {{1, 5}, {1, 2}}}, {False, {{1, 5}, {1, 3}}},
 {False, {{1, 5}, {1, 4}}}, {False, {{1, 5}, {2, 3}}}, {False, {{1, 5}, {5, 4}}},
 {False, {{2, 1}, {2, 4}}}, {False, {{2, 1}, {2, 5}}}, {False, {{2, 1}, {5, 1}}},
 {False, {{2, 3}, {1, 4}}}, {False, {{2, 3}, {1, 5}}}, {False, {{2, 4}, {2, 1}}},
 {False, {{2, 4}, {3, 1}}}, {False, {{2, 5}, {2, 1}}}, {False, {{2, 5}, {3, 1}}},
 {False, {{3, 1}, {2, 4}}}, {False, {{3, 1}, {2, 5}}}, {False, {{3, 1}, {3, 4}}},
 {False, {{3, 1}, {3, 5}}}, {False, {{3, 1}, {5, 1}}}, {False, {{3, 2}, {4, 1}}},
 {False, {{3, 2}, {5, 1}}}, {False, {{3, 4}, {3, 1}}}, {False, {{3, 5}, {3, 1}}},
 {False, {{4, 1}, {3, 2}}}, {False, {{4, 1}, {5, 1}}}, {False, {{4, 2}, {1, 2}}},
 {False, {{4, 2}, {1, 3}}}, {False, {{4, 3}, {1, 3}}}, {False, {{4, 5}, {5, 1}}},
 {False, {{5, 1}, {2, 1}}}, {False, {{5, 1}, {3, 1}}}, {False, {{5, 1}, {3, 2}}},
 {False, {{5, 1}, {4, 1}}}, {False, {{5, 1}, {4, 5}}}, {False, {{5, 2}, {1, 2}}},
 {False, {{5, 2}, {1, 3}}}, {False, {{5, 3}, {1, 3}}}, {False, {{5, 4}, {1, 5}}}}
```

Koczkodaj's Iterative Inconsistency Reduction algorithm

Calculate the value of the Koczkodaj inconsistency index

In[78]:=
```
N[KoczkodajIdx[M]]
```

Out[78]=
```
0.78125
```

Prints the worst Koczkodaj triad in M. As we can see it is $m_{5,3} == \frac{1}{4}$, $m_{3,1} == \frac{7}{4}$, $m_{5,1} == 2$. The value of inconsistency introduced by this triad is $\frac{25}{32}$

In[79]:=
```
KoczkodajTheWorstTriad[M]
```

Out[79]= $\left\{\{5, 3, 1\}, \left\{\frac{1}{4}, \frac{7}{4}, 2\right\}, \frac{25}{32}\right\}$

Perform one step of the Koczkodaj inconsistency reduction algorithm. On the output there is a new slightly modified matrix M2 that is expected to be more consistent than M

In[80]:=
```
M2 = N[KoczkodajImproveMatrixStep[M]]
```

Out[80]=
```
{{1., 0.6, 0.344306, 0.625, 0.829827},
 {1.66667, 1., 0.714286, 2.5, 3.33333}, {2.90439, 1.4, 1., 3.5, 2.41014},
 {1.6, 0.4, 0.285714, 1., 1.33333}, {1.20507, 0.3, 0.414913, 0.75, 1.}}
```

In[81]:=
```
KoczkodajIdx[M2]
```

Out[81]=
```
0.585087
```

Aggregation of Individual Judgments (AIJ)

Let us consider three different PC matrices *X*, *Y* and *Z* that come from three different experts

In[82]:= $X = \begin{pmatrix} 1 & x_{12} & x_{13} & x_{14} \\ x_{21} & 1 & x_{23} & x_{24} \\ x_{31} & x_{32} & 1 & x_{34} \\ x_{41} & x_{42} & x_{43} & 1 \end{pmatrix};$

In[83]:= $Y = \begin{pmatrix} 1 & y_{12} & y_{13} & y_{14} \\ y_{21} & 1 & y_{23} & y_{24} \\ y_{31} & y_{32} & 1 & y_{34} \\ y_{41} & y_{42} & y_{43} & 1 \end{pmatrix}$;

In[84]:= $Z = \begin{pmatrix} 1 & z_{12} & z_{13} & z_{14} \\ z_{21} & 1 & z_{23} & z_{24} \\ z_{31} & z_{32} & 1 & z_{34} \\ z_{41} & z_{42} & z_{43} & 1 \end{pmatrix}$;

Then it is possible to aggregate the results using appropriate functions.

Aggregate Individual Judgments additively (AIJadd):

In[85]:= **AIJadd[X, Y, Z] // MatrixForm**

Out[85]//MatrixForm=

$$\begin{pmatrix} 1 & \frac{1}{3}(x_{12}+y_{12}+z_{12}) & \frac{1}{3}(x_{13}+y_{13}+z_{13}) & \frac{1}{3}(x_{14}+y_{14}+z_{14}) \\ \frac{1}{3}(x_{21}+y_{21}+z_{21}) & 1 & \frac{1}{3}(x_{23}+y_{23}+z_{23}) & \frac{1}{3}(x_{24}+y_{24}+z_{24}) \\ \frac{1}{3}(x_{31}+y_{31}+z_{31}) & \frac{1}{3}(x_{32}+y_{32}+z_{32}) & 1 & \frac{1}{3}(x_{34}+y_{34}+z_{34}) \\ \frac{1}{3}(x_{41}+y_{41}+z_{41}) & \frac{1}{3}(x_{42}+y_{42}+z_{42}) & \frac{1}{3}(x_{43}+y_{43}+z_{43}) & 1 \end{pmatrix}$$

Aggregate Individual Judgments geometrically (AIJgeom)

In[86]:= **AIJgeom[X, Y, Z] // MatrixForm**

Out[86]//MatrixForm=

$$\begin{pmatrix} 1 & (x_{12}\,y_{12}\,z_{12})^{1/3} & (x_{13}\,y_{13}\,z_{13})^{1/3} & (x_{14}\,y_{14}\,z_{14})^{1/3} \\ (x_{21}\,y_{21}\,z_{21})^{1/3} & 1 & (x_{23}\,y_{23}\,z_{23})^{1/3} & (x_{24}\,y_{24}\,z_{24})^{1/3} \\ (x_{31}\,y_{31}\,z_{31})^{1/3} & (x_{32}\,y_{32}\,z_{32})^{1/3} & 1 & (x_{34}\,y_{34}\,z_{34})^{1/3} \\ (x_{41}\,y_{41}\,z_{41})^{1/3} & (x_{42}\,y_{42}\,z_{42})^{1/3} & (x_{43}\,y_{43}\,z_{43})^{1/3} & 1 \end{pmatrix}$$

Note that these two functions works also for result lists i.e.:

In[87]:= **AIJgeom[{x_1, x_2, x_3, x_4}, {y_1, y_2, y_3, y_4}]**

Out[87]= $\{\sqrt{x_1\,y_1}, \sqrt{x_2\,y_2}, \sqrt{x_3\,y_3}, \sqrt{x_4\,y_4}\}$

In[88]:= **AIJadd[{x_1, x_2, x_3, x_4}, {y_1, y_2, y_3, y_4}]**

Out[88]= $\{\frac{1}{2}(x_1+y_1), \frac{1}{2}(x_2+y_2), \frac{1}{2}(x_3+y_3), \frac{1}{2}(x_4+y_4)\}$

Incomplete Pairwise Comparisons Matrix - Harker Method

The idea comes from P. T. Harker, Alternative Modes of Questioning in The Analytic Hierarchy Process, Math Modeling, 1987.

Let M be an incomplete pairwise comparisons matrix.

In[89]:= $M = \begin{pmatrix} 1 & 2 & \square \\ 1/2 & 1 & 2 \\ \square & 1/2 & 1 \end{pmatrix}$;

Thus, to compute the ranking based on M we need to compute matrix A (hereinafter referred to as Harker matrix)

In[90]:= **A = HarkerMatrix[M];**

In[91]:= **A // MatrixForm**

Out[91]//MatrixForm=

$$\begin{pmatrix} 2 & 2 & 0 \\ \frac{1}{2} & 1 & 2 \\ 0 & \frac{1}{2} & 2 \end{pmatrix}$$

Then compute the eigenvalue based ranking and inconsistency index as usuall:

In[92]:= **EigenvalueRank[A]**

Out[92]= {0.571429, 0.285714, 0.142857}

In[93]:= **SaatyIdx[A]**

Out[93]= 0

The ranking errors and the ranking discrepancies

Let the w = {w_1, w_2, w_3} be the ranking vector whilst M a PC matrix. An error is defined as $e_{ij} = m_{ij}(w_j / w_i)$ and it corresponds to the “discrepancy” between individual judgment m_{ij} and the ranking result. Hence, the error matrix is just E = [e_{ij}]. E.g.:

In[94]:= M = RecreatePCMatrix[$\begin{pmatrix} 1 & \frac{3}{5} & \frac{4}{7} & \frac{5}{8} & \frac{5}{10} \\ 0 & 1 & \frac{5}{7} & \frac{5}{2} & \frac{10}{3} \\ 0 & 0 & 1 & \frac{7}{2} & 4 \\ 0 & 0 & 0 & 1 & \frac{4}{3} \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$];

In[95]:= **ErrorMatrix[M, EigenvalueRank[M]] // MatrixForm**

Out[95]//MatrixForm=

$$\begin{pmatrix} 1. & 0.722398 & 0.584559 & 1.45507 & 2.00712 \\ 1.38428 & 1. & 1.07892 & 0.839258 & 0.694602 \\ 1.71069 & 0.92685 & 1. & 0.777866 & 0.751091 \\ 0.687253 & 1.19153 & 1.28557 & 1. & 0.827639 \\ 0.498227 & 1.43967 & 1.3314 & 1.20826 & 1. \end{pmatrix}$$

The error matrix is not symmetric. An attempt to symmetrization of error matrix leads to the local discrepancy matrix D = (d_{ij}) where $d_{ij} = \max\{e_{ij} - 1, 1/e_{ij} - 1\}$. Another interesting property of the local discrepancy matrix is the fact that wherever $d_{ij} = 0$ this means that the local discrepancy (error) is 0. If $d_{ij} = X$ this means that the local judgment m_{ij} differs from the ratio $\frac{w_i}{w_j}$ by 100%*X

In[96]:= **Chop@LocalDiscrepancyMatrix[M, EigenvalueRank[M]] // MatrixForm**

Out[96]//MatrixForm=

$$\begin{pmatrix} 0 & 0.384278 & 0.71069 & 0.455068 & 1.00712 \\ 0.384278 & 0 & 0.0789237 & 0.191529 & 0.439673 \\ 0.71069 & 0.0789237 & 0 & 0.285569 & 0.331397 \\ 0.455068 & 0.191529 & 0.285569 & 0 & 0.208257 \\ 1.00712 & 0.439673 & 0.331397 & 0.208257 & 0 \end{pmatrix}$$

The greatest entry of the local discrepancy matrix can be found by using the GlobalDiscrepancy function

In[97]:= **GlobalDiscrepancy[M, EigenvalueRank[M]]**

Out[97]= 1.00712

Usefull methods built into Mathematica

Spearman Rank Correlation (compare with the example from https://en.wikipedia.org/wiki/Spearman%27s_rank_correlation_coefficient)

```
In[98]:= Needs["MultivariateStatistics`"];

In[99]:= N@SpearmanRankCorrelation[{86, 97, 99, 100, 101, 103, 106, 110, 112, 113},
           {0, 20, 28, 27, 50, 29, 7, 17, 6, 12}]

General::obsfun: The functionSpearmanRankCorrelationis now obsoleteand has been supersededby SpearmanRho.

Out[99]= -0.175758

In[100]:= rs = N@SpearmanRankCorrelation[{1.1, 1.57, 0.51, 1.1, 1.1}, {1.2, 1, 2.3, 1, 18}]

Out[100]= -0.573539

In[101]:= n = Length@{1.1, 1.57, 0.51, 1.1, 1.1};
```

In[102]:= t = N[r_s * Sqrt[$\frac{n-2}{1-r_s^2}$]]

```
Out[102]= -1.21268
```

Kendall Tau Corellation

```
In[103]:= KendallTau[{86, 97, 99, 100, 101, 103, 106, 110, 112, 113},
           {0, 20, 28, 27, 50, 29, 7, 17, 6, 12}]
```

Out[103]= $-\frac{1}{9}$

Rank Order

The use of RankOrder[ranking] is a way to shift from cardinal ranking as produced by EVM or GMM to the ordinal ranking. Hence, the result is a list of alternatives ordered according to the outrank relation $\prec$ introduced by the cardinal ranking. I.e. for two alternatives c_i and c_j it will holds that $c_i \prec c_j$ if $w(c_i) < w(c_j)$ and, of course $c_i \sim c_j$ if $w(c_i) = w(c_j)$. For example:

```
In[104]:= ranking = EigenvalueRank[M]

Out[104]= {0.119092, 0.27476, 0.356526, 0.130954, 0.118669}

In[105]:= RankOrder[ranking]

Out[105]= {3, 2, 4, 1, 5}
```

The above result means that the highest priority has 3rd alternative, then goes 2nd alternative, 4th, 1st and 5th at the end. It may happen, however, that some alternatives have the same priority. E.g.

In[106]:= ranking = EigenvalueRank[$\begin{pmatrix} 1 & 1 & 1 & 1 & 3 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1/3 & 1 & 1 & 1 & 1 \end{pmatrix}$]

```
Out[106]= {0.256633, 0.194146, 0.194146, 0.194146, 0.16093}
```

In such a case RankOrder[] groups the alternatives with the similar priorities

```
In[107]:= RankOrder[ranking]

Out[107]= {1, 3, 2, 4, 5}
```

The above result means that the alternatives 2, 3, and 4 took ex aequo the second position.

In[108]:=

Monte Carlo PC Matrices

Sometimes it is useful to generate random PC matrix to verify some properties of such matrix statistically
For example let us create some real random ranking of 7 alternatives with values from interval [1/15, 15]:

In[109]:=
```
ranking = RandomRankingPattern[7, 15]
```

Out[109]=
```
{3.43545, 6.96391, 0.626052, 0.965588, 14.7538, 2.02032, 0.0998182}
```

Then we may create fully consistent matrix based on this ranking by setting $c_{ij} = w_i / w_j$

In[110]:=
```
M = RandomMatrix[7, ranking, 1];
```

In[111]:=
```
M // MatrixForm
```

Out[111]//MatrixForm=

1.	0.493322	5.48749	3.55789	0.232852	1.70045	34.4171
2.02707	1.	11.1235	7.21209	0.472007	3.44694	69.766
0.182233	0.0898995	1.	0.648364	0.0424332	0.309878	6.27193
0.281066	0.138656	1.54234	1.	0.0654466	0.477939	9.67347
4.29458	2.11861	23.5665	15.2796	1.	7.30274	147.807
0.588078	0.290112	3.22707	2.09232	0.136935	1.	20.24
0.0290553	0.0143336	0.159441	0.103375	0.00676557	0.0494072	1.

In[112]:=
```
SaatyIdx[M]
```

Out[112]= 0

We may also disturb every entry by a factor chosen from the range [1/1.3, 1.3]. This makes the random matrix a bit inconsistent:

In[113]:=
```
M = RandomMatrix[7, ranking, 1.3];
```

In[114]:=
```
M // MatrixForm
```

Out[114]//MatrixForm=

1.	0.597671	4.6991	2.77945	0.242081	1.41311	27.7146
1.67316	1.	13.6706	9.09286	0.516014	2.65234	64.8967
0.212807	0.0731499	1.	0.741982	0.049473	0.29772	5.86368
0.359784	0.109976	1.34774	1.	0.0791945	0.415534	9.1086
4.13085	1.93793	20.213	12.6271	1.	9.3523	183.218
0.707661	0.377025	3.35887	2.40654	0.106926	1.	23.587
0.0360821	0.0154091	0.170541	0.109786	0.00545798	0.0423963	1.

In[115]:=
```
SaatyIdx[M]
```

Out[115]= 0.0126507

In[116]:=
```
KoczkodajIdx[M]
```

Out[116]= 0.48856

One may want to consider disturbed rational matrix. To this end first we create rational ranking pattern of 7 values unifirmly distributed over the scale 1/15 to 15:

In[117]:=
```
ranking = RandomRationalUniformRankingPattern[7, 15]
```

Out[117]= $\left\{\frac{1}{13}, \frac{1}{5}, 5, 2, 13, \frac{1}{11}, \frac{1}{2}\right\}$

Then we create fully consistent Rational PC matrix over this ranking

In[118]:=
```
M = RandomRationalMatrix[7, 15, ranking, 1];
```

In[119]:=
```
M // MatrixForm
```

Out[119]//MatrixForm=

$$\begin{pmatrix} 1 & \frac{5}{13} & \frac{1}{65} & \frac{1}{26} & \frac{1}{169} & \frac{11}{13} & \frac{2}{13} \\ \frac{13}{5} & 1 & \frac{1}{25} & \frac{1}{10} & \frac{1}{65} & \frac{11}{5} & \frac{2}{5} \\ 65 & 25 & 1 & \frac{5}{2} & \frac{5}{13} & 55 & 10 \\ 26 & 10 & \frac{2}{5} & 1 & \frac{2}{13} & 22 & 4 \\ 169 & 65 & \frac{13}{5} & \frac{13}{2} & 1 & 143 & 26 \\ \frac{13}{11} & \frac{5}{11} & \frac{1}{55} & \frac{1}{22} & \frac{1}{143} & 1 & \frac{2}{11} \\ \frac{13}{2} & \frac{5}{2} & \frac{1}{10} & \frac{1}{4} & \frac{1}{26} & \frac{11}{2} & 1 \end{pmatrix}$$

In[120]:=
```
SaatyIdx[M]
```

Out[120]= 0

In[121]:=
```
KoczkodajIdx[M]
```

Out[121]= 0

As previously we may also disturb every entry by the factor chosen from the range [1/1.3, 1.3], however, this time we have also to find the closest rational number within the set of numbers in the form $\frac{p}{q}$ where p,q belongs to $\{1,2,3,...,15^2\}$. To get this matrix we have to call once again:

In[122]:=
```
M = RandomRationalMatrix[7, 15, ranking, 1.3];
```

In[123]:=
```
M // MatrixForm
```

Out[123]//MatrixForm=

$$\begin{pmatrix} 1 & \frac{72}{205} & \frac{3}{196} & \frac{3}{88} & \frac{1}{174} & \frac{168}{191} & \frac{41}{220} \\ \frac{205}{72} & 1 & \frac{5}{137} & \frac{23}{180} & \frac{3}{166} & \frac{211}{82} & \frac{74}{223} \\ \frac{196}{3} & \frac{137}{5} & 1 & \frac{223}{110} & \frac{47}{118} & 52 & \frac{193}{20} \\ \frac{88}{3} & \frac{180}{23} & \frac{110}{223} & 1 & \frac{35}{176} & \frac{131}{5} & \frac{48}{11} \\ 174 & \frac{166}{3} & \frac{118}{47} & \frac{176}{35} & 1 & 131 & \frac{223}{8} \\ \frac{191}{168} & \frac{82}{211} & \frac{1}{52} & \frac{5}{131} & \frac{1}{131} & 1 & \frac{23}{123} \\ \frac{220}{41} & \frac{223}{74} & \frac{20}{193} & \frac{11}{48} & \frac{8}{223} & \frac{123}{23} & 1 \end{pmatrix}$$

but of course this time there is some small inconsistency

In[124]:=
```
SaatyIdx[M]
```

Out[124]= 0.00646102

In[125]:=
```
N@KoczkodajIdx[M]
```

Out[125]= 0.420963

Both methods for generating random PC matrices has its short forms for default scales and patterns. They need only two parameters: size of the matrix and disturbance level

In[126]:=
```
M = RandomMatrix[5, 1.5];
```

In[127]:= **M // MatrixForm**

Out[127]//MatrixForm=

$$\begin{pmatrix} 1. & 0.169105 & 0.120294 & 0.131173 & 0.223008 \\ 5.91349 & 1. & 0.765685 & 0.932853 & 1.01421 \\ 8.31297 & 1.30602 & 1. & 1.23107 & 1.80808 \\ 7.6235 & 1.07198 & 0.812302 & 1. & 1.79561 \\ 4.48414 & 0.98599 & 0.553072 & 0.556915 & 1. \end{pmatrix}$$

In[128]:= **M = RandomRationalMatrix[5, 1.5];**

In[129]:= **M // MatrixForm**

Out[129]//MatrixForm=

$$\begin{pmatrix} 1 & \frac{33}{52} & 26 & \frac{79}{50} & \frac{55}{3} \\ \frac{52}{33} & 1 & 81 & \frac{67}{30} & 50 \\ \frac{1}{26} & \frac{1}{81} & 1 & \frac{1}{26} & \frac{40}{73} \\ \frac{50}{79} & \frac{30}{67} & 26 & 1 & \frac{43}{4} \\ \frac{3}{55} & \frac{1}{50} & \frac{73}{40} & \frac{4}{43} & 1 \end{pmatrix}$$

For example is we want to check what is the average inconsistency of the rational matrices 5 by 5 over the fundamental scale disturbed by the factor from [1/2, 2] we may just call:

In[130]:= **Mean@Table[SaatyIdx[RandomRationalMatrix[5, 2]], {100}]**

Out[130]= 0.049591

The above result was computed based on 100 randomly choosen rational random PC matrices.