Decision Support

An action learning approach for assessing the consistency of pairwise comparison data

Kweku-Muata Osei-Bryson *

Department of Information Systems & The Information Systems Research Institute, Virginia Commonwealth University, Richmond, VA 23284, USA

Received 24 July 2003; accepted 20 January 2005
Available online 23 May 2005

Abstract

Pairwise comparison data are used in various contexts including the generation of weight vectors for multiple criteria decision making problems. If this data is not sufficiently consistent, then the resulting weight vector cannot be considered to be a reliable reflection of the evaluator’s opinion. Hence, it is necessary to measure its level of inconsistency. Different approaches have been proposed to measuring the level of inconsistency, but they are often based on ‘rules of thumb’ and/or randomly generated matrices, and are not interpretable. In this paper we present an action learning approach for assessing the consistency of the input pairwise comparison data that offer interpretable consistency measures.

© 2005 Published by Elsevier B.V.

Keywords: Pairwise comparisons; Consistency indicators; Inconsistency; AHP; Interpretability; Multiple criteria decision making; Goal programming

1. Introduction

Pairwise comparison data are used in various contexts such as the generation of weight vectors for multiple criteria decision making problems (e.g. Ngai, 2003; Saaty, 1980), subjective probabilities for expert systems (e.g. Monti and Carenini, 2000), and Dempster–Shafer belief functions (e.g. Bryson and Mobolurin, 1999). The pairwise comparison information is represented numerically using an $N \times N$ matrix $A = \{a_{ij}\}$, where $a_{ij}$ is a non-negative rational number (e.g. 0.90) that is the numerical equivalent of the comparison...
between objects “i” and “j”. The weight vector \( w \) may then be obtained from the comparison matrix \( A \) using a variety of techniques (e.g. Saaty and Vargas, 1984; Fichtner, 1986; Bryson, 1995), the most popular of which are the right eigenvector method (EM) of Saaty, and the logarithmic least squares method (LLSM), although a logarithmic goal programming method was also proposed by Bryson (1995). The pairwise comparison matrix \( A \) is said to be consistent if for each triplet of objects \((i, j, k)\) the equality \( a_{ij} = a_{ik}a_{kj} \) holds; otherwise it is said to be inconsistent. As noted by Obata et al. (1999), “When a pairwise comparison matrix contains seriously inconsistent comparisons, the priority weights calculated from such a wrong matrix are not reliable”. Now because the matrix \( A \) is often inconsistent, it is necessary to measure its level of consistency in order to determine if the resulting weight vector \( w \) will be meaningful.

Consistency measures have been proposed by various researchers (e.g. Saaty, 1980; Golden and Wang, 1989; Salo and Hämäläinen, 1997; Aguaron and Moreno-Jimenez, 2003), but a major limitation of these consistency measures is that they are not interpretable. The more popular of these measures, Saaty’s consistency ratio (CR), is defined as \( CR = CI/RI \), where \( CI = (\lambda_{\text{Max}} - N)/(N - 1) \), \( \lambda_{\text{Max}} \) is the largest eigenvalue of \( A \), and RI is similar to CI but based on random matrices, each with the same dimension as \( A \). Using Saaty’s original ‘rule of thumb’ the pairwise comparison matrix is deemed to be inconsistent only if \( CR > 0.10 \). Salo and Hämäläinen’s consistency measure CM takes its values from the closed interval \([0, 1]\), with increasing values indicating decreasing consistency. Aguaron and Moreno-Jimenez’ (2003) measure (i.e. Geometric Consistency Index) applies to situations when the row averaging method is used instead of Saaty’s right eigenvector method. For each of these measures, apart from the end-points of the \([0, 1]\) interval, the values of the measure are not interpretable. Further, because the thresholds associated with these measures are based on ‘rules of thumb’ and/or randomly generated matrices in some situations these measures do not appear to be appropriate. For example, in a recent study that applied the Analytic Hierarchy Process (AHP) to elicit subjective probabilities from human experts, Monti and Carenini (2000) noted that “…the manifest inconsistency showed by the expert’s assessments based on different elicitation techniques provided us with evidence that the 0.10 CR was not appropriate”.

In this paper we present an action learning approach for assessing the consistency of the input pairwise comparison data that offers interpretable consistency measures.

2. Overview on the logarithmic goal programming method

Bryson (1995) proposed a logarithmic goal programming method for generating weight vectors. Given an input pairwise comparison matrix \( A = \{a_{ij}\} \), the objective is to generate a weight vector \( w = (w_1, \ldots, w_N) \) such that for each pair of objects “i” and “j”, the ratio \( w_{ij} = (w_i/w_j) \) is as close as possible to the evaluator’s specified \( a_{ij} \). Now let there be positive real numbers \( p_{ij} \leq 1, n_{ij} \leq 1 \) such that \( (w_i/w_j) \) * \( (n_{ij}/p_{ij}) = a_{ij} \) where \( p_{ij} \) and \( n_{ij} \) cannot both be less than 1. Then \( p_{ij} = n_{ij} = 1 \) implies that \( (w_i/w_j) = a_{ij} \); \( n_{ij} < 1 \) implies that \( (w_i/w_j) > a_{ij} \); and \( p_{ij} < 1 \) implies that \( (w_i/w_j) < a_{ij} \). Therefore, if \( p_{ij} = n_{ij} = 1 \) for each pair of objects “i” and “j”, then the set of point estimates provided by the evaluator is consistent. Otherwise the data in \( A \) are inconsistent, and our objective is to generate a weight vector \( w \) that is closest to the evaluator’s specifications. Thus the problem is to maximize the product \( H_{ij} \forall i, j \) \( p_{ij}n_{ij} \). This translates to solving the following linear goal programming problem:

\[
P_{\text{GPG}}: \log s_{\text{Avg}} = (1/|IJ|)\text{Max} \sum_{(ij) \in IJ} (\log p_{ij} + \log n_{ij})
\]

s.t. \( \log v_i - \log v_j - \log p_{ij} + \log n_{ij} = \log a_{ij} \ \forall (i, j) \in IJ \ (1 \leq i < j \leq N) \),

where \( IJ \) is the set of possible pairwise comparisons; \(|IJ|\) is the cardinality of \( IJ \); \( \log a_{ij} = \log(a_{ij}) \); \( \log s_{\text{Avg}} = \log(s_{\text{Avg}}) \); \( \log v_i = \log(v_i) \); \( \log v_j = \log(v_j) \); \( \log p_{ij} = \log(p_{ij}) \); \( \log n_{ij} = \log(n_{ij}) \); and all \( \log p_{ij} \) and \( \log n_{ij} \) are
non-positive. The solution of this problem will result in the unnormalized vector \( v = (v_1, \ldots, v_N) \) where \( (v_i/v_j) = (w_i/w_j) \). The vector \( v \) is then normalized to give the vector \( w \). The reader may note that our \( \log p_{ij} \) and \( \log n_{ij} \) have non-positive constraints rather than the traditional non-negative constraints of LP problems. We adopted this approach in order to ensure that \( p_{ij}, n_{ij} \) take their values from the \((0, 1)\) interval.

3. An action learning approach for assessing consistency

3.1. Assessing consistency

Bryson (1995) suggested that it is reasonable to consider as a consistency indicator \( s_{\text{Avg}} \), the average value that each entry in \( A \) would have to be multiplied or divided by in order the make the set of values consistent. The reader may note that \( s_{\text{Avg}} \in (0, 1) \). If the evaluator’s estimates were consistent we would have \( \log(s_{\text{Avg}}) = 0 \) and \( s_{\text{Avg}} = 1 \); otherwise we would have \( \log(s_{\text{Avg}}) < 0 \) and \( s_{\text{Avg}} < 1 \). The indicator \( s_{\text{Avg}} \) provides us with a measure of the average consistency of the pairwise comparisons, but has the disadvantage that it is an average its value is sensitive to the presence of outliers.

Let \( C = \{c_{ij} = (w_i/w_j)\} \), \( s_{ij} = \text{Min}\{a_{ij}, c_{ij}, d_{ij}\} \), Then \( s_{ij} \in (0, 1) \) provides a measure of the agreement between \( a_{ij} \) and \( c_{ij} \), with higher values indicating greater agreement and \( s_{ij} = 1 \) indicating perfect agreement. Let \( \tau \ (0 < \tau \leq 1) \) be a tolerance parameter such that \( \tau \) is the minimum amount that we are tolerating \( a_{ij} \) to be multiplied by or divided by in order for it to considered to be approximately equal to \( c_{ij} \). Then \( s_{ij} \geq \tau \) indicates that \( a_{ij} \) and \( c_{ij} \) are considered to be approximately equal at degree of approximation \( \tau \). Also let \( \rho \) be the proportion of the \( a_{ij} \)'s that are considered to be approximately equal to \( c_{ij} \)'s at degree of approximation \( \tau \). The indicator \( \rho \) is defined as follows:

\[
\rho = \sum_{(i,j) \in I} \Phi(i,j)/|IJ|,
\]

where \( \Phi(i,j) = 1 \) if \( s_{ij} \geq \tau \); \( \Phi(i,j) = 0 \) otherwise. Thus \( \rho \in [0, 1] \), with \( \rho = 1.00 \) indicating perfect consistency and \( \rho = 0.00 \) indicating total inconsistency at approximation degree \( \tau \).

If matrix \( A \) is very consistent then we would expect corresponding elements of \( A \) and \( C \) to be very similar, which would imply that each \( s_{ij} \) would be close to \( 1 \). If the pairwise comparison matrix \( A \) was perfectly consistent then \( s_{ij} = 1 \) for all \( (i,j) \in IJ \). Given that matrix \( A \) may be acceptable even if it is not perfect, then it would be useful to determine the threshold value \( \tau \) such that \( s_{ij} \geq \tau \) would indicate that the corresponding elements are sufficiently similar. Bryson (1995) discussed the issue of ‘single outlier neutralization’, where if all elements of the pairwise comparison matrix except one are consistent then the weight vector generation technique should still be used to generate the correct weight vector. In such a situation the matrix \( A \) should still be considered to be highly consistent despite the presence of an outlier entry. This suggests that one aspect of the consistency of the \( A \) matrix is the proportion of its entries that are sufficiently similar to corresponding elements of the output \( C \) matrix (i.e. \( s_{ij} \geq \tau \)) for the threshold \( \tau \). From this perspective, one interesting question is what is the maximum value of \( \tau \) for which \( \rho = 1.00 \). However, exploration of other values of \( \rho \) may also be useful in order to assess the consistency of the input pairwise comparison matrix. We will therefore look at the thresholds \( \tau \) associated with different values of \( \rho \). It should be noted that both \( \tau \) and \( \rho \) are interpretable measures.

Let \( k \) be the number of elements that are in violation (i.e. \( s_{ij} < \tau \)). Then \( k_{\text{In}} = |IJ|(1 - \rho_{\text{In}}) \), where \( \lfloor \cdot \rfloor \) denotes the integer part of the relevant number. We can generate the threshold \( \tau \) by solving the following mixed integer programming problem \( P_{\text{MILGP}}(p_{\text{In}}) \) for different feasible values of \( k_{\text{In}} \):
where \( \varepsilon \) is a very small positive number; \( k_{\text{in}} = \lceil |IJ| (1 - \rho_{\text{in}}) \rceil \) is the maximum number of pairwise comparisons that are permitted to violate the similarity threshold (i.e. \( s_{ij} < \tau \) for those pairwise comparisons); \( \lg s_{ij} = \log s_{ij}; \lg \tau = \log \tau; x_{ij} = 1 \text{ if } \lg s_{ij} < \lg \tau, \text{ and } x_{ij} = 0 \text{ otherwise}; M \) is a large number such that if there is a violation (i.e. \( s_{ij} < \tau \)) then constraint 6a will hold if \( x_{ij} = 1 \); \( fp_{ij} = 1 \) if \( \lg p_{ij} < 0 \), and \( fp_{ij} = 0 \) otherwise; \( fn_{ij} = 1 \text{ if } \lg n_{ij} < 0, \text{ and } fn_{ij} = 0 \text{ otherwise}; \) and that the following always hold: \( (\lg p_{ij} + M) \geq 0 \) and \( (\lg n_{ij} + M) \geq 0 \). Constraints (1a), (2a) are used to identify both the values of the weight vector and the similarity values \( s_{ij} \); constraints (3a)–(5a) and (10a)–(11a) are used to ensure that \( \lg p_{ij} \cdot \lg n_{ij} = 0 \) (see Observation 1 in the Appendix A); constraint (6a) is used to identify pairwise comparisons that are in violation based on the threshold \( \tau \); constraints (6a) and (9a) ensure that \( x_{ij} = 1 \) if the associated pairwise comparison violates the threshold \( \tau \); constraint (7a) ensures that no more than \( k_{\text{in}} \) pairwise comparisons violate the threshold \( \tau \); constraints (8a) ensure that \( \tau \) takes its values from the \([0, 1]\) interval. In order to ensure that \( x_{ij} = 1 \) only if the associated pairwise comparison violates the threshold \( (\lg s_{ij} - \lg \tau < 0) \), we include a penalty (i.e. \( - \varepsilon \sum_{(i,j) \in U} x_{ij} \)) in the objective function.

The reader may have observed that constraint (7a) is expressed as an inequality. The reason is that for a given pairwise comparison matrix it might not be possible to have \( k_{\text{in}} \) pairwise comparisons that are simultaneously in violation of any threshold (i.e. \( \tau \leq 1 \)). One such situation in which this could occur is in the case of a perfectly consistent pairwise comparison matrix where it would be impossible to have even one pairwise comparison that is in violation (i.e. \( k_{\text{in}} = 1 \)) of any feasible threshold \( \tau \). Solution of this problem for a given feasible value of \( (\rho_{\text{in}} = 1 - (k_{\text{in}}/|IJ|)) \) provides \( \tau \), and \( \rho_{\text{out}} = \sum_{(i,j) \in U} x_{ij} \Rightarrow \rho_{\text{out}} = 1 - k_{\text{out}}/|IJ| \).

A related problem to \( P_{\text{MILGPb}}(\rho_{\text{in}}) \) is problem \( P_{\text{MILGPa}}(\tau) \) which can be stated as

\[
P_{\text{MILGPa}}(\rho_{\text{in}}) \quad \begin{aligned}
\text{Max} & \quad \lg \tau - \varepsilon \sum_{(i,j) \in U} x_{ij} \\
\text{s.t.} & \quad \lg v_i - \lg v_j - \lg p_{ij} + \lg n_{ij} = \lg a_{ij} \quad \forall (i, j) \in IJ, \\
& \quad \lg p_{ij} + \lg n_{ij} - \lg s_{ij} = 0 \quad \forall (i, j) \in IJ, \\
& \quad \lg p_{ij} + Mfp_{ij} \geq 0 \quad \forall (i, j) \in IJ, \\
& \quad \lg n_{ij} + Mn_{ij} \geq 0 \quad \forall (i, j) \in IJ, \\
& \quad fp_{ij} + fn_{ij} \leq 1 \quad \forall (i, j) \in IJ, \\
& \quad \sum_{(i,j) \in U} x_{ij} \leq \lfloor |IJ|(1 - \rho_{\text{in}}) \rfloor, \\
& \quad \lg \tau \leq 0, \\
& \quad x_{ij} \in \{0, 1\} \quad \forall (i, j) \in IJ, \\
& \quad fp_{ij} \in \{0, 1\}, fn_{ij} \in \{0, 1\} \quad \forall (i, j) \in IJ, \\
& \quad \lg p_{ij} \leq 0, \lg n_{ij} \leq 0, \lg s_{ij} \leq 0 \quad \forall (i, j) \in IJ, \\
\end{aligned}
\]
\[
\sum_{(i, j) \in IJ} x_{ij} - k = 0, \quad (7b)
\]

\[k \text{ is integer valued,} \quad (8b)\]

\[x_{ij} \in \{0, 1\} \quad \forall (i, j) \in IJ, \quad (9b)\]

\[f p_{ij}, f n_{ij} \in \{0, 1\} \quad \forall (i, j) \in IJ, \quad (10b)\]

\[\lg p_{ij} \leq 0, \lg n_{ij} \leq 0, \lg s_{ij} \leq 0, \quad \forall (i, j) \in IJ. \quad (11b)\]

This problem is similar to problem \(P_{\text{MILGPa}}(\rho_{\text{In}})\) in that they share the same constraints but for problem \(P_{\text{MILGPa}}(\tau)\) \(k\) is a variable, \(\tau\) is a constant, and the objective function is Min \(k\). If the user has a value for \(\tau\) with which he/she is comfortable then problem \(P_{\text{MILGPa}}(\tau)\) might be the appropriate problem to solve as it would return the value of \(\rho\) that corresponds to the value \(\tau\). If the user is in an explanatory mode then solution of a sequence instantiations of \(P_{\text{MILGPa}}(\rho_{\text{In}})\) for different values of \(\rho_{\text{In}}\) would be appropriate.

Given that Qualitative/Numeric transformation tables (e.g. Table 1) are often used to facilitate the elicitation of pairwise comparison data from evaluators, it is reasonable to assume that the evaluator could provide a value of \(\tau\) for which s/he would be comfortable. The lower bound of the numeric interval associated with the linguistic term(s) that are considered equivalent to “approximately equal” could be used to provide the value of \(\tau\). For example, if the range of linguistic terms “Almost Equal” through “Very Close” is considered equivalent to “approximately equal” then \(\tau = 0.90\); while if the range of linguistic term “Almost Equal” is considered equivalent to “approximately equal” then \(\tau = 0.95\).

3.2. Generating the associated weight vector

Assuming that the value of the tuple \((\rho_{\text{Out}}, \tau)\) which was obtained from the solution of \(P_{\text{MILGPa}}(\rho_{\text{In}})\) reflects a satisfactory level of consistency, then a reliable weight vector could be generated from the pairwise comparison matrix \(A\) using those entries that are deemed to be acceptable (i.e. \(s_{ij} \geq \tau x_{ij} = 0\)). Below we define the set of constraints that are relevant to tuple \((\rho_{\text{Out}}, \tau)\):

\[\lg v_i - \lg v_j - \lg p_{ij} + \lg n_{ij} = \lg a_{ij} \quad \forall (i, j) \in IJ, \quad (1c)\]

\[\lg p_{ij} + \lg n_{ij} - \lg s_{ij} = 0 \quad \forall (i, j) \in IJ, \quad (2c)\]

\[\lg p_{ij} + M f p_{ij} \geq 0 \quad \forall (i, j) \in IJ, \quad (3c)\]

\[\lg n_{ij} + M f n_{ij} \geq 0 \quad \forall (i, j) \in IJ, \quad (4c)\]

\[f p_{ij} + f n_{ij} \leq 1 \quad \forall (i, j) \in IJ, \quad (5c)\]

\[\lg s_{ij} + M x_{ij} \geq \lg \tau \quad \forall (i, j) \in IJ, \quad (6c)\]

\[\sum_{(i, j) \in IJ} x_{ij} = |IJ|/\lfloor 1 - \rho_{\text{Out}} \rfloor, \quad (7c)\]

\[\lg \tau \leq 0, \quad (8c)\]

\[x_{ij} \in \{0, 1\} \quad \forall (i, j) \in IJ, \quad (9c)\]

\[f p_{ij}, f n_{ij} \in \{0, 1\} \quad \forall (i, j) \in IJ, \quad (10c)\]

\[\lg p_{ij} \leq 0, \lg n_{ij} \leq 0, \lg s_{ij} \leq 0 \quad \forall (i, j) \in IJ. \quad (11c)\]

Given \((\rho_{\text{Out}}, \tau)\), we are interested in \(IJ_1\), the index set of pairwise comparison entries in \(A\) for which it is possible that \(s_{ij} \geq \tau (\equiv x_{ij} = 0\)). Fortunately some of these entries would have been identified when problem \(P_{\text{MILGPa}}(\rho_{\text{In}})\) was solved. For the other \((i_0, j_0) \in IJ\), determine \(x_{i_0j_0(\text{Min})} = \min\{x_{ij}(1c)-(11c)\}\). Let \(IJ_1 = \{(i, j) : x_{ij(\text{Min})} = 0\}\).
For \( k = 0 \), \(|JI_1|\) corresponds to a complete pairwise comparison matrix, and for \( k \geq 1 \) \(|JI_1|\) may correspond to an incomplete pairwise comparison matrix. Thus for \( k = 0 \) any weight vector generation technique could be applied, for \( k \geq 1 \) (i.e. \(|JI_1| < |JI|\)) the weight generation technique that is selected must be able to accommodate an incomplete pairwise comparison matrix. Various techniques are available for generating a suitable weight vector for an incomplete pairwise comparison matrix (e.g. Harker, 1987; Bryson, 1995; Choo and Wedley, 2004). Choo and Wedley (2004) discuss strengths and weakness of various weight vector generation techniques.

In our illustrative example we will use a modified version of Bryson’s LGPM for generating the associated weight vector. Choo and Wedley (2004) in their experimental study on the effectiveness of various weight vector generation techniques, observed that for their sample if the “errors” (i.e. deviation from consistency) in the pairwise comparison matrix were all small (i.e. \( s_{ij} \)’s are large), then weight vector generation techniques that minimized the largest error (i.e. maximized the smallest \( s_{ij} \)) gave the best performance. They also observed that when the “errors” in the pairwise comparison matrix \( A \) are few (e.g. most \( s_{ij} \)’s are greater than \( \tau \)) but large, methods using absolute deviations had the best performance. Our approach for generating the weight vector utilizes these observations using a prioritized goal programming approach in which the first objective is to maximize the smallest \( s_{ij} \), and the second objective is to maximize the geometric average \( s_{Avg} \). It should be noted that in some cases solution of problem \( P_{MILGPM}(\rho_{LB}) \) is equivalent to minimizing the worst “error” for the pairwise comparisons in \( IJ_1 \).

3.3. Description of the action learning procedure:

Let \( \tau_{LB} \) and \( \rho_{LB} \) be lower bounds on \( \tau \) and \( \rho \) respectively. Since measures \( \tau \) and \( \rho_{Out} \) are interpretable, specifying lower bounds \( \tau_{LB} \) (e.g. “Fairly Close” \( \Rightarrow \tau_{LB} = 0.80 \)) and \( \rho_{LB} \) (e.g. at least 75% of the input pairwise comparison entries should be acceptable \( \Rightarrow \rho_{LB} = 0.75 \)) should not be a major challenge for users. Given the tuple \((\rho_{LB}, \tau_{LB})\), a pairwise comparison matrix \( A \) would be considered to be inconsistent if it is not possible to obtain any tuple \((\rho_{Out}, \tau)\) such that \((\rho_{Out}, \tau) \geq (\rho_{LB}, \tau_{LB})\).

In computing the weight vector that is associated with a given \((\rho_{Out}, \tau)\) we will be using only those pairwise comparisons in \( IJ_1 \), and so we require that each object be included in at least one of these pairwise comparisons. If \( A \) is a complete pairwise comparison matrix, then there are \((N-1)\) pairwise comparisons for each object. One way of guaranteeing that each object is included in at least one of the pairwise comparisons in \( IJ_1 \) is to require that \( k_{Out} \leq (N-2) \). This would imply that \( \rho_{LB} \geq 1 - (N-2)/(N(N-1)/2) = 1 - (2N - 4)/(N^2 - N) \). The following table displays the lower bound on \( \rho_{LB} \) for selected values of \( N \):

<table>
<thead>
<tr>
<th>( N )</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \rho_{LB} \geq )</td>
<td>0.667</td>
<td>0.667</td>
<td>0.700</td>
<td>0.733</td>
<td>0.762</td>
<td>0.786</td>
<td>0.806</td>
<td>0.822</td>
</tr>
</tbody>
</table>

Table 1
Qualitative/numeric transformation table of Mobolurin and Bryson (1993)

<table>
<thead>
<tr>
<th>Qualitative category</th>
<th>Numerical interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>Almost equal</td>
<td>(0.95, 1.00)</td>
</tr>
<tr>
<td>Very close</td>
<td>(0.90, 0.95)</td>
</tr>
<tr>
<td>Fairly close</td>
<td>(0.80, 0.90)</td>
</tr>
<tr>
<td>Moderately inferior</td>
<td>(0.70, 0.80)</td>
</tr>
<tr>
<td>Very inferior</td>
<td>(0.60, 0.70)</td>
</tr>
<tr>
<td>Extremely inferior</td>
<td>(0.50, 0.60)</td>
</tr>
<tr>
<td></td>
<td>(0.40, 0.50)</td>
</tr>
<tr>
<td></td>
<td>(0.30, 0.40)</td>
</tr>
<tr>
<td></td>
<td>(0.20, 0.30)</td>
</tr>
<tr>
<td></td>
<td>(0.10, 0.20)</td>
</tr>
</tbody>
</table>
It should also be noted that we could include additional constraints in our mixed integer programming problem in order to ensure that each object is included in at least one of these pairwise comparisons \( IJ_1 \) (e.g. 
\[
\sum_{(i,h) \in IJ} x_{ih} + \sum_{(h,j) \in IJ} x_{hj} \geq 1 \quad \forall h \in I = \{1, 2, \ldots, N\}.
\]

**Step 1: Initialization**

- Specify the tuple \((\rho_{LB}, \tau_{LB})\).
- Set \(q_{In} = \rho_{LB}\)
  \[
  k_{In} = [|IJ| * (1 - \rho_{In})].
  \]

**Step 2:**

(a) Solve problem \( P_{MILGPa} (q_{In}) \).
(b) If the corresponding \( \tau < \tau_{LB} \) then
   Terminate (since when \( \rho_{Out} \) is increasing then \( \tau \) is non-increasing, and so for any larger value of \( \rho_{Out} \) the corresponding \( \tau \) will also be less than \( \tau_{LB} \))
else
   Given the optimal solution of problem \( P_{MILGPa} (\rho_{In}) \).
   Set \( k_{Out} = \sum_{(i,j) \in IJ} x_{ij} \).
   Determine the associated \( IJ_1 \) and \( A_1 = \{ a_{ij}; (i,j) \in IJ_1 \} \)
   Apply an appropriate Weight Vector Generation Technique on \( A_1 \) and store the resulting weight vector as \( w^{k_{Out}} \).
(c) If \( \tau = 1 \) or \( k_{Out} = 0 \) then Terminate; otherwise go to Step 3.

**Step 3:**

(a) Set \( k_{In} = k_{Out} - 1 \).
(b) Set \( \rho_{In} = (|IJ| - k_{In})/|IJ| \).
(c) Go to step 2.

**4. Illustrative examples**

For both of the examples below we will assume that for measures \( \tau \) and \( \rho \), the evaluator specified that at least 75\% of the input pairwise comparison entries should be acceptable (i.e. \( \rho_{LB} = 0.75 \)) at an approximation degree that is at least “Fairly Close” (i.e. \( \tau_{LB} = 0.80 \)).

**4.1. Example 1**

For our first example, consider the following pairwise comparison matrix that was obtained from Golden and Wang (1989) and which involves four objects. Table 2 contains the contents of the \( A \) matrix of pairwise comparison information.

The weight vector computed by the EM is \((0.150, 0.545, 0.046, 0.259)\). The corresponding CR value is 0.11688, which would suggest that this pairwise comparison matrix should be considered to be highly inconsistent using the Saaty’s cut-off value of 0.08 for \( N = 4 \). Golden and Wang’s \( G \) value is 0.31535 which is well beyond their cut-off value of 0.2032 for \( N = 4 \), indicating that this pairwise comparison matrix is highly inconsistent. It should be noted that the weight vector generated by LLSM is \((0.154, 0.540, 0.047, 0.259)\) which is just slightly different from that generated by EM.
Solving problem $P_{\text{MILGPa}}(\rho_{\text{In}})$ for $\rho_{\text{In}} = 1.00, 0.83, 0.67, 0.50$ (i.e. $k = 0, 1, 2, 3$) respectively yields the results displayed in Table 3. These results indicate that in order to have an interpretation that corresponding elements of the input pairwise comparison matrix $A$ and the output consistent pairwise comparison matrix $C$ are perfectly consistent (i.e. $q_{\text{Out}} = 1$), the similarity threshold $\tau$ could be no higher than 0.5844, which would be “Extremely Inferior” based on our Qualitative/Numeric Transformation Table (i.e. Table 1). Further, even for $q_{\text{Out}} = 0.8333$, the similarity threshold $\tau$ could be no higher than 0.7368, which is “Very Inferior”. More reasonable similarity thresholds of “Almost Equal” (i.e. $\tau = 0.90$) or “Very Close” (i.e. $\tau = 0.95$) would each result in $q_{\text{Out}} = 0.667$ (i.e. only two-thirds of the entries are acceptable). These results also suggest that this pairwise comparison matrix is highly inconsistent.

The reader might observe that $q_{\text{Out}}$ is a non-increasing function of $\tau$ and vice versa, and that for each value of $q_{\text{Out}}$ there is an exclusive interval of $\tau$ values. For example, the data in Table 3 implies that $\rho_{\text{Out}} = 1.00$ for $\tau \in [0, 0.584]$, $\rho_{\text{Out}} = 0.83$ for $\tau \in (0.584, 0.737]$, $\rho_{\text{Out}} = 0.67$ for $\tau \in (0.737, 0.962]$ and $\rho_{\text{Out}} = 0.50$ for $\tau \in (0.962, 1.000]$. Thus given that $\rho_{\text{LB}} = 0.75$ and $\tau_{\text{LB}} = 0.80$, there is no tuple $(\rho_{\text{Out}}, \tau) \geq (\rho_{\text{LB}}, \tau_{\text{LB}}) = (0.75, 0.80)$ and so the consistency of this pairwise comparison data would be considered to be unacceptable. Therefore no weight vector is generated for this case.

### 4.2. Example 2

For our second example, consider the following $A$ matrix displayed in Table 4. Solving problem $P_{\text{MILGPa}}(\rho_{\text{In}})$ for $\rho_{\text{In}} = 1.00, 0.83, 0.67, 0.50$ (i.e. $k = 0, 1, 2, 3$) respectively. yields the results displayed in Table 5. These results indicate that in order to have an interpretation that corresponding elements of the input matrix $A$ and the output consistent pairwise comparison matrix $C$ are perfectly consistent (i.e. $\rho_{\text{Out}} = 1$), the similarity threshold $\tau$ could be “Very Close” since $0.90 \leq 0.925 \leq 0.95$. Further even for
Out = 0.8333, the similarity threshold \( s \) could be “Almost Equal” since than 0.90 < 0.967 < 1, which is also very high. This suggests that this pairwise comparison matrix is highly consistent. It should also be noted that for this example, when \( q_{in} = 0.50 \) (i.e. \( k_{in} = 3 \)) it is not possible to have three (3) entries that are in violation and so \( P_{MILGPa2}(p_{in}) \) returns the value \( \tau = 1 \) and \( k_{out} = 2 \) indicating that this value of \( \tau \) is associated with 2 violations.

The data in Table 5 indicates that \( q_{out} = 1.00 \) for \( \tau \in [0, 0.925] \); \( q_{out} = 0.83 \) for \( \tau \in (0.925, 0.967] \); \( q_{out} = 0.67 \) for \( \tau \in (0.967, 1.000] \). Thus given that \( p_{LB} = 0.75 \) and \( \tau_{LB} = 0.80 \), there are two tuples (i.e. (1.00, 0.925) and (0.83, 0.967)) for which \((p_{out}, \tau) = (p_{LB}, \tau_{LB}) = (0.75, 0.80)\) and so the consistency of this pairwise comparison data would be considered to be acceptable. The corresponding weights vectors could thus be considered to be reliable.

### 5. Conclusion

In this paper we have presented an action learning approach for assessing the consistency of pairwise comparison data. Rather than focusing on the removing the most inconsistent element (e.g. Saaty, 1994), we identify multiple elements whose differences are acceptable and those whose differences are unacceptable. This is based on an interpretable tolerance parameter \( \tau \) and similarity measure \( s_{ij} = \min\{a_{ij}/(w_i/w_j), (w_i/w_j)/a_{ij}\} \), where for the element \((i,j)\) the difference is considered to be acceptable if \( s_{ij} \geq \tau \) and unacceptable if \( s_{ij} < \tau \). For a given value of \( \tau \), we can determine the maximum proportion \( p \) of the input pairwise comparisons \( a_{ij} \) that could be considered to have either no difference or acceptable difference with the corresponding \((w_i/w_j)\). Similarly for \( q_{in} \), the target proportion of acceptable pairwise comparisons, we can determine the maximum value of the tolerance parameter \( \tau \). Rather than have the user specify a specific value for \( \tau \) or \( q_{in} \), the user can explore ranges of values for these parameters (i.e. \((q_{out}, \tau) = (q_{LB}, \tau_{LB})\)) and be presented either with the information that the input pairwise comparison data is perfectly consistent (i.e. \((q_{out}, \tau) = (1.00, 1.00)\)), is acceptable though not perfectly consistent (i.e. \((p_{LB}, \tau_{LB}) \leq (p_{out}, \tau) < (1.00, 1.00)\)), or is unacceptably inconsistent (i.e. \((p_{out}, \tau) < (p_{LB}, \tau_{LB})\)). Rather than using an adjusted pairwise comparison matrix (i.e. the most different \( a_{ij} \) is replaced with \((w_i/w_j)\)) to generate the weight vector, if the consistency of the pairwise comparison matrix could be considered to be acceptable then we generate a weight vector for each \((q_{out}, \tau) \geq (p_{LB}, \tau_{LB})\).

Although some techniques for assessing the consistency of the pairwise comparison data are tightly coupled with techniques for generating weight vectors (e.g. Saaty’s right eigenvector method), it is possible to decouple them. We believe that what is most important here is that given pairwise comparison data with an acceptable level of consistency, the weight vector generation technique should be able to generate an appropriate weight vector. Our procedure thus allows for the use of various weight vector generation techniques for generating the weight vector.

The approach to assessing consistency that was presented in this paper could require more involvement on the part of the user than those of Saaty (1980), Golden and Wang (1989), and Salo and Hämäläinen (1997) because there are no standard threshold values for our parameters. However, the standard threshold values associated with these other measures do not have an objective basis. Therefore, it might be more
prudent for the user to adopt an action learning process using naturally meaningful indicators, eventually resulting in the development of his/her personal cut-off values (i.e. $\rho_{LB}$, $\tau_{LB}$). It should also be noted that the approach suggested in this paper is appropriate for both complete and incomplete pairwise comparison matrices.

Acknowledgements

Thanks to the anonymous referees for their very constructive comments which have helped me to improve the quality of this paper.

Appendix A

Observation 1. Let $\log p_{ij}$ and $\log n_{ij}$ be non-positive variables, and $M$ be large number such that the following constraints always hold: $\log p_{ij} + M \geq 0$; $\log n_{ij} + M \geq 0$. Let $f_{pij}$ and $fn_{ij}$ be 0–1 binary variables. Then $\log p_{ij} \cdot \log n_{ij} = 0$ if the following constraints hold:

\[
\begin{align*}
\log p_{ij} + Mf_{pij} &\geq 0, \\
\log n_{ij} + Mfn_{ij} &\geq 0, \\
f_{pij} + fn_{ij} &\leq 1.
\end{align*}
\]

Proof

Case 1: $\log p_{ij} = 0$. This is the trivial case for which it is obvious that $\log p_{ij} \cdot \log n_{ij} = 0$.

Case 2: $\log p_{ij} = 0$.

Given constraint (A.1) then $\log p_{ij} < 0 \Rightarrow f_{pij} = 1$.

Given constraint (A.3) then $f_{pij} = 1 \Rightarrow fn_{ij} = 0$.

Given constraint (A.2) then $fn_{ij} = 0 \Rightarrow \log n_{ij} \geq 0 \Rightarrow \log n_{ij} = 0$.

$\Rightarrow \log p_{ij} \cdot \log n_{ij} = 0$. $\square$

References


