

Motivation

Generating Random Performance Tableaux

Raymond Bisdorff

University of Luxembourg, FSTC/CSC

Mons, April, 2009

- Provide random instances of performance tableaux for **MCDA Web services** testing and debugging
- Illustrate the use of `xmcda:methodOptions` in XMCD-2.0
- Show a new XMCD-2.0 service offered on the D3 server in Luxembourg.

Outline

Random Performance Tableau

Random Performance Tableau

- 1.1 Standard reference model
- 1.2 random performance generators
- 1.3 random thresholds

Special Performance Tableau

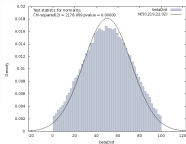
- 2.1 Cost-Benefit performance Tableau
- 2.2 Correlating the performances with three coalitions
- 2.3 Introducing random coalitions

Definition (A reference model)

- **20** decision actions; low variant: 13; high variant: 50.
- **13** criteria; low variant: 7; high variant: 20.
- All criteria are equi-significant.
- All criteria use a same cardinal scale from 0.0 to 100.0.
- Four random performance generators may be used:
 - a uniform generator ($\mathcal{U}(0.0, 100)$),
 - a truncated normal generator ($\mathcal{N}(\mu, \sigma)$),
 - a triangular generator ($\mathcal{T}(xm, r)$) with mode xm and probability repartition r ,
 - a beta generator ($\mathcal{Beta}(xm, s)$) with mode xm and standard deviation s .

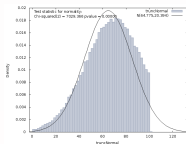
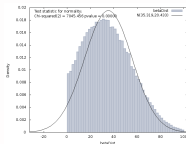
Truncated Normal Generator

- In the reference case, the mode x_m is situated in the middle (50.0) of the performance scale and the standard deviation is a fourth (25) of the scale scope.



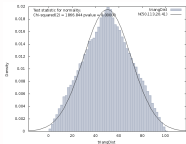
Truncated Normal Generator (continued)

- We consider two variants:
 - low performances: $x_m = 30$,
 - high performances: $x_m = 70$,



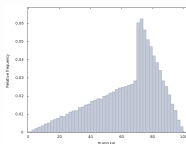
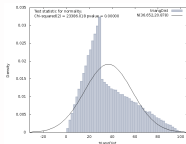
Triangular Generator

- In the reference case, the mode x_m is situated in the middle (50.0) of the performance scale and the probability is equally distributed on both sides, i.e. $r = 0.5$ and x_m represents the median performance.



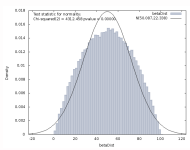
Triangular Generator (continued)

- We consider two variants with fixed repartition $r = 0.5$:
 - low performances: $x_m = 30$,
 - high performances: $x_m = 70$,



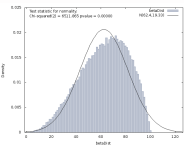
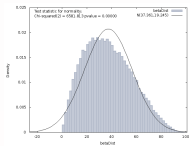
Beta Generator

- In the reference case, the mode x_m is situated in the middle (50.0) of the performance scale and the probability is equally distributed on both sides, i.e. x_m represents the median performance.



Beta Generator (continued)

- We consider two variants with equal standard deviation:
 - low performances: $x_m = 30$,
 - high performances: $x_m = 70$,



Fixed Discrimination Thresholds

On each criterion, the **default discrimination thresholds** are chosen such that the:

- indifference** threshold equals 5.0 (low: 2.5, high:10.0);
- preference** threshold equals 15.0 (low: 10.0, high:20.0);
- weak veto** threshold equals 70.0 (low: 60.0, high: 80);
- veto** threshold equals 80.0 (low: 70.0, high: 90).
- The **ordinal** criteria admit solely a preference threshold of one unit.

Example

Random performance tableau instance

Fixed Percentile Discrimination Thresholds

On each criterion, the **default discrimination thresholds** are chosen such that the:

- indifference** threshold equals the percentile 5 of all generated performance differences;
- preference** threshold equals the percentile 10 of all generated performance differences;
- weak veto** threshold equals the percentile 90 of all generated performance differences;
- veto** threshold equals the percentile 95 of all generated performance differences.
- The **ordinal** criteria admit solely a preference threshold of one unit.

Example

Random performance tableau instance

Random Cost-Benefit Performance Tableau

Definition (A reference model)

- 20 decision actions; low variant: 13; high variant: 50.
- 13 criteria; low variant: 7; high variant: 20.
- A criteria is with equal probability either to be minimized (**cost** criteria) or to be maximized (**benefit** criteria).
- All criteria either support an **ordinal** or a **cardinal** performance scale; the cost criteria being mostly cardinal (2/3) and the benefit ones mostly ordinal (2/3).
- Ordinal performances are represented on **integer** scales: $\{1, 2, \dots, 10\}$.
- Cardinal performances are represented on a **decimal** scale: $[0.0; 100.0]$ with a precision of 2 digits.

Random Cost-Benefit Performance Tableau (continued)

- In the Cost-Benefit model the decision actions are divided randomly into three categories: **cheap**, **neutral**, **advantageous**.
- An action is called:
 - **cheap** when the performances are generated with $T(xm = 30, r = 0.5)$ (reference) or $\mathcal{N}(\mu = 30, \sigma = 25)$.
 - **advantageous** when the performances are generated with $T(xm = 70, r = 0.5)$ (reference) or $\mathcal{N}(\mu = 70, \sigma = 25)$,
 - and **neutral** when the performances are generated with $T(xm = 50, r = 0.5)$ (reference) or $\mathcal{N}(\mu = 50, \sigma = 25)$.

Example

Random performance tableau instance

Correlating the performances with three coalitions

- In a first case we consider three a priori coalitions: A,B and C.
- Every criteria is affected randomly to one of the three coalitions.
- Each actions is randomly affected on each coalition to one of three performance following categories: low performance (-), medium performance (~) and high performance (+).
- When generating the performances of an alternative on a criterion, the random generator is modulated following the performance profile of the action respective to the coalition of the criterion.

Example

Random performance tableau instance

Random Criteria Coalitions

- We consider a family of n criteria.
- Every criteria is affected randomly to one of n potential coalitions.
- Each actions is randomly affected on each coalition to one of three performance categories: low performance (-), medium performance (~) and high performance (+).
- When generating the performances of an alternative on a criterion, the random generator is modulated following the performance profile of the action respective to the coalition of the criterion.

Example

Random performance tableau instance

Concluding Remarks

References I

In this communication we have presented:

- A reference model for random performance tableaux
- Four generators for random performances
- An parametric XMCD A-2.0 Web service for random performance tableaux.



R. Bisdorff (2008)

The Python digraph implementation for RuBis: User Manual.
University of Luxembourg,
<http://ernst-schroeder.uni.lu/Digraph>.



R. Bisdorff, P. Meyer, Th. Veneziano (2009)

Quick dive into XMCD A 2.0.
Decision Deck Consortium, <http://www.decision-deck.org>.