Ministry of Education and Science of the Russian Federation
Peter the Great St. Petersburg State Polytechnic University
Institute of Computer Sciences and Technologies
Graduate School of Cyber-Physical Systems and Control

Report 3

Tasks 1 & 2: Multi-Criteria Decision Making
Discipline: Modern Problems of Informatics and Computer Science
31 March 2017

Student Group: 13541/8

______________________ Christopher W. Blake

Professor

______________________ Rodionova E.A.

St. Petersburg
2017

Contents

Christopher W. Blake
17 March, 2017

Multi-Criteria Problem

Task Description

A decision must be made between a selected set of options. Each of these options can be described using the same vector of characteristics. Using the Saaty method, the importance of each characteristic is compared to each other characteristic. Each of the options is also compared to each other option, one characteristic at a time. All values are rated in a 1-9 system using the chart below.

Relative importance (points)	Name	Explanation
1	Equal importance	Ai and Aj have the same importance
3	Little preference	Ai is a bit more preferable than Aj
5	Significant preference	Ai is much more preferable than Aj
7	Evident preference	It is evident that Ai is more preferable than Aj
9	Absolute preference	Ai is absolutely more preferable than Aj
2,4,6,8	Intermediate points	Compromise ones

After all ratings are recorded, the weighted value of each criteria are produced using two methods. The first is a simple weighted sum and the second uses the eigen vectors, using the Nth root approximation. These weights are then combined to produce the final recommendation. Two example tasks are performed.

Task 1: Select a sport to play. (In class rankings)

Task 2: Select a travel destination.

Procedure Explanation (Task 1 Solution)

Choose Options and Characteristics

1.) Select the options to choose from
   a. Ice Skating
   b. Swimming
   c. Football
2.) Select the characteristics for comparison
   a. Location
   b. Cost per hour
   c. Safety
   d. Desire
   e. Difficulty

Compare Criteria

Each characteristic is compared, relative to each other criteria. A value over 1 represents that the row's characteristic is more important that the column's characteristic. As such, the diagonal has only values of 1, and the opposite side is the inverse.

Table 1: Criteria Comparison

Criteria	Location	Cost/Hr	Safety	Desire	Difficulty	Weight Sum	Prod	nRoot	Weight Eigen
Location	1.00	0.20	3.00	0.25	0.50	**0.09**	**0.08**	**0.60**	**0.08**
Cost/Hr	5.00	1.00	7.00	3.00	7.00	**0.43**	**735.00**	**3.74**	**0.51**
Safety	0.33	0.14	1.00	0.33	0.14	**0.04**	**0.00**	**0.30**	**0.04**
Desire	4.00	0.33	3.00	1.00	9.00	**0.32**	**36.00**	**2.05**	**0.28**
Difficulty	2.00	0.14	3.00	0.11	1.00	**0.12**	**0.10**	**0.62**	**0.09**

Weight Calculation

The Nth root is calculated using the product of the row's rankings values. These values are then weighted using the sum of all Nth Root values. The Nth root approximation is derived from the power iteration (See Appendix 1).

$$Nth\_Root_k = \left(\prod_{k=1}^{n} Rating_k\right)^{1/n}$$

$$W\_Eigen_k = \frac{Nth\_Root_k}{\sum_{k=1}^{n} Nth\_Root_k}$$

Where

$$\text{n} = \text{number of criteria}$$

Compare Options, Per Criteria

Each option is compared to each other option. However, this is considered using only one characteristic.

Table 2: Option Comparison, Per Criteria

Location	Ice Skating	Swimming	Football	Weight Sum	Prod	nRoot	Weight Eigen
Ice Skating	1.00	1.00	2.00	0.40	2.00	1.15	0.38
Swimming	1.00	1.00	2.00	0.40	2.00	1.15	0.16
Football	0.50	0.50	1.00	0.20	0.25	0.76	0.10
Cost/Hr	Ice Skating	Swimming	Football	Weight Sum	Prod	nRoot	Weight Eigen
Ice Skating	1.00	2.00	0.20	0.21	0.40	0.83	0.25
Swimming	0.50	1.00	0.25	0.12	0.13	0.66	0.20
Football	5.00	4.00	1.00	0.67	20.00	1.82	0.55
Safety	Ice Skating	Swimming	Football	Weight Sum	Prod	nRoot	Weight Eigen
Ice Skating	1.00	0.33	2.00	0.24	0.67	0.92	0.28
Swimming	3.00	1.00	5.00	0.64	15.00	1.72	0.53
Football	0.50	0.20	1.00	0.12	0.10	0.63	0.19
Desire	Ice Skating	Swimming	Football	Weight Sum	Prod	nRoot	Weight Eigen
Ice Skating	1.00	0.50	0.14	0.11	0.07	0.59	0.18
Swimming	2.00	1.00	1.00	0.27	2.00	1.15	0.36
Football	7.00	1.00	1.00	0.61	7.00	1.48	0.46
Difficulty	Ice Skating	Swimming	Football	Weight Sum	Prod	nRoot	Weight Eigen
Ice Skating	1.00	2.00	4.00	0.54	8.00	1.52	0.47
Swimming	0.50	1.00	3.00	0.34	1.50	1.08	0.34
Football	0.25	0.33	1.00	0.12	0.08	0.61	0.19

Christopher W. Blake
17 March, 2017

Compute Final Recommendation

The final recommendation uses the weights from both area. The equation below represents this process. Using both weight methods, the same result is produced. The best recommendation is "Football."

$$W_{c1} * \begin{bmatrix} W_{o1c1} \\ W_{o2c1} \\ W_{o3c1} \end{bmatrix} + \ldots + W_{cn} * \begin{bmatrix} W_{o1cn} \\ W_{o2cn} \\ W_{o3cn} \end{bmatrix}$$

where

$$W_{cn} = Weight\ for\ characteristic\ \text{"n"}$$
$$\text{Wo1cn} = \text{Weight for option "1" of characteristic "n"}$$

Table 3: Recommendation Results

Options/Criteria	Location	Cost/Hr	Safety	Desire	Difficulty	Total Score (Sum)
Ice Skating	0.04	0.09	0.01	0.04	0.06	0.24
Swimming	0.04	0.05	0.02	0.09	0.04	0.24
Football	0.02	0.29	0.00	0.20	0.01	0.52

Options/Criteria	Location	Cost/Hr	Safety	Desire	Difficulty	Total Score (Eigen)
Ice Skating	0.03	0.11	0.01	0.06	0.06	0.27
Swimming	0.01	0.09	0.02	0.12	0.04	0.27
Football	0.01	0.24	0.01	0.15	0.02	0.42

Task 2 - Solution

A homemade task was created using the same methodology. Below is a summary of the results for the task of "Select a travel destination". The results show that "Tokyo" is the best recommendation.

Table 1: Criteria Comparison

Criteria	Price	Safety	Exotic-ness	Excursions	Nightlife	Weight Sum	Prod	nRoot	Weight Eigen
Price	1	1	1/5	1/3	1/6	0.05	0.01	0.41	0.06
Safety	1	1	1/3	1	1/7	0.07	0.05	0.54	0.08
Exotic-ness	5	3	1	3	7	0.37	315.00	3.16	0.48
Excursions	3	1	1/3	1	7	0.24	7.00	1.48	0.23
Nightlife	6	7	1/7	1/7	1	0.28	0.86	0.97	0.15

Table 2: Option Comparison, Per Criteria

Price	San Francisc	Tokyo	Moscow	Weight Sum	Prod	nRoot	Weight Eigen
San Francisco	1	1/2	1/9	0.07	0.06	0.56	0.15
Tokyo	2	1	1/7	0.14	0.29	0.78	0.12
Moscow	9	7	1	0.78	63.00	2.29	0.35

Safety	San Francisc	Tokyo	Moscow	Weight Sum	Prod	nRoot	Weight Eigen
San Francisco	1	4	3	0.54	12.00	1.64	0.51
Tokyo	1/4	1	4	0.35	1.00	1.00	0.31
Moscow	1/3	1/4	1	0.11	0.08	0.61	0.19

Exotic-ness	San Francisc	Tokyo	Moscow	Weight Sum	Prod	nRoot	Weight Eigen
San Francisco	1	1/7	1/4	0.07	0.04	0.51	0.15
Tokyo	7	1	4	0.64	28.00	1.95	0.56
Moscow	4	1/4	1	0.28	1.00	1.00	0.29

Excursions	San Francisc	Tokyo	Moscow	Weight Sum	Prod	nRoot	Weight Eigen
San Francisco	1	1/7	1/3	0.08	0.05	0.54	0.16
Tokyo	7	1	5	0.70	35.00	2.04	0.58
Moscow	3	1/5	1	0.22	0.60	0.90	0.26

Nightlife	San Francisc	Tokyo	Moscow	Weight Sum	Prod	nRoot	Weight Eigen
San Francisco	1	1/5	1/3	0.10	0.07	0.58	0.18
Tokyo	5	1	3	0.61	15.00	1.72	0.52
Moscow	3	1/3	1	0.29	1.00	1.00	0.30

Table 3: Recommendation Results

Options/Criteria	Price	Safety	Exotic-ness	Excursions	Nightlife	Total Score (Sum)
San Francisco	0.00	0.04	0.03	0.02	0.03	0.11
Tokyo	0.01	0.02	0.24	0.17	0.17	0.60
Moscow	0.04	0.01	0.10	0.05	0.08	0.29

Options/Criteria	Price	Safety	Exotic-ness	Excursions	Nightlife	Total Score (Eigen)
San Francisco	0.01	0.03	0.05	0.04	0.05	0.18
Tokyo	0.01	0.02	0.21	0.14	0.14	0.52
Moscow	0.02	0.01	0.11	0.06	0.08	0.28

Christopher W. Blake

Appendix 1 – Nth Root and Power Iteration Methods

Power Iteration Method

The Power Iteration method is based on a simple iterative approach. Given a square matrix A, it will produce an eigen value (λ) such that $Av = \lambda v$.

The basic principle is to assume an eigen vector b, and iteratively divide and normalize the matrix A by the resulting vector of A*b. By repeating this process, the vector b eventually converges.

$$eigen = \lambda = b$$
$$when \lim_{n\to\infty}\left(b_{n+1} = \frac{Ab_n}{\|Ab_n\|}\right)$$

Nth Root Method

The Nth root method is essentially the power iteration method, but applied to an array instead of a matrix. Given an array (A) with k elements, there exists an eigen value (λ) which is equivalent to the product of the array elements.

Assume an initial eigen value of b equal to the product of the elements of the array A.

$$eigen = \lambda = b$$
$$when \lim_{n\to\infty}\left(b_{n+1} = \frac{Ab_n}{\|Ab_n\|}\right)$$

Repeat this procedure until the value of b converges. Because the value of b converges to a single number and it is divided many times, this can instead be represented as a root.

$$eigen = \lambda = \frac{Ab_1}{\|Ab_1\|} + \cdots + \frac{Ab_n}{\|Ab_n\|} = \frac{\prod_{k=1}^{n} A_k}{\prod_{k=1}^{n} \lambda} = \left(\prod_{k=1}^{n} A_k\right)^{1/n}$$