RAPPORT DE STAGE

Année universitaire 2016 / 2017

<u>Nom et Prénom de l'étudiant</u> :

Elie ABI HANNA DAHER

<u>Année d'études</u> :

☐ L3 INFO ☐ L3 INFO APPR. ☐ L3 MATHS

■ M1 MIAGE ☐ M1 INFO. DECISION ☐ M1 INFO APPR. ☐ M1 MATHS

<u>Organisme d'accueil</u> : LAMSADE

<u>Titre du rapport</u> : Internship Report – UTA Method

<u>Tuteur de stage</u> : Olivier CAILLOUX, Michel ZAM

<u>Dates du stage</u> : 29/05/2017 jusqu'à 01/09/2017

LAMSADE

M1 MIAGE

Internship

Internship Report - UTA Method

Student:
Elie Abi Hanna Daher

Supervisor:
M. Olivier Cailloux
M. Michel Zam

25th August, 2017

Abstract

This report documents the work done during the summer internship at LAMSADE Dauphine, under the supervision of Mr. Olivier Cailloux and Mr. Michel Zam. The report give an overview of the projects completed during the internship.
I have worked mainly on the UTA method which is an ordinal regression method, proposed by E. Jacquete-Lagreze and J. Siskos, which adjust a system of additive utility function.

Acknowledgment

There are a number of people that I would like to thank them for their help and contribution in the success of my internship at LAMSADE Dauphine.

First and foremost, I would like to thank Mr. Olivier Cailloux, a lecturer in computer science and researcher at LAMSADE. His advice and follow ups helped me a lot in establishing my assignments during the internship.

I also would like to thank Mr. Michel Zam, the founder of KarmicSoft and an associate professor at Paris Dauphine University, for the opportunity given to me as an intern at Lamsade

Doing my internship at LAMSADE with Mr. Cailloux and Mr. Zam was a pleasure, I have learned a lot thanks to them. I would like to them for this opportunity.

I would like to thank the internship manager, Mrs Maude Manouvrier, for all her efforts in term of follow-up as well as her availability and patience to answer my questions.

Contents

Chapter 1

Introduction

Currently I'm completing my master's degree in MIAGE at Paris Dauphine University so I had the opportunity to achieve an internship at LAMSADE (Laboratoire d'analyse et modélisation de systèmes pour l'aide à la décision). LAMSADE is a Paris-Dauphine research laboratory specialized in decision theoretic approaches. Decision theory aims at helping individuals who face decision problem by elaborating a study of the reasoning underlying agent's choices.

From 29th May, 2017 to 1st September, 2017, I completed a research internship at LAMSADE at Paris Dauphine University. My internship was made remotely, so I had a weekly meeting with Mr. Cailloux to get feedback of the work done. And I communicated with Mr. Zam via mails or calls.

This internship was very important for my professional career because it may be my last internship before I get my Masters's degree in Information systems for finance at Paris Dauphine University. Let's not also forget that this internship will allow me to practice the different courses learned throughout my academic years.

One of the objectives is to propose the UTA method, that solve a multi-criteria problem by building a utility function based on the preferences of the Decision Maker, as an open source software component. Another objective was to integrate this open source software into DecisionCloud, a software based on MyDraft a tool developed by KarmicSoft a LAMSADE spin-off. Researching similar literature and research represent an important objective in this internship.

In this internship report, I will describe the context of the project which contains an overview of the internship. Writing this reports, I will describe the projects made: UTA, LinearProgram Solver, UTA Calculator and the research realized. I will conclude by reflecting on the flow of the internship.

Chapter 2

Context of the projects

2.1 Introduction to LAMSADE

LAMSADE is a laboratory located in Paris Dauphine university. Lamsade stands for Laboratoire d'analyse et modélisation de systèmes pour l'aide à la décision. The french laboratory has been established in 1974 by Bernard Roy. The main research subjects that this center study are operational research, decision aid, artificial intelligence (AI) and decision theory.
The activities of LAMSADE are divided into 3 poles:

- Decision aid
- Combinatorial optimization and algorithmic
- Data Science

LAMSADE is well know for the origin of the multiple-criteria decision analysis (MCDA), and for its approach to the Algorithmic Decision Theory. The laboratory has established himself between the best internationally.
LAMSADE is structured in 2 dimensions:

- Scientific
- Research

Scientific dimension represent the poles, where there is a community of researches invited to exchange through out meetings and seminary. The research dimension represent the projects assembled by the researches interested in the subject in question. Both of this dimensions are neither fixed neither permanent, they can be submitted to an evaluation by the Laboratory.

2.2 Objectives

Through out the internship, I had several objectives to achieve and those objectives changed from when we began the internship.
The following list represent the objectives stated at the beginning of the internship:

- Propose an open source software component of the UTA method (Main objective)
- Integrate the UTA open software component as one of the tools in the Decision Deck community
- Document and publish the software and implement a graphical user interface to the UTA by using DecisionCloud, a tool promoted by the Decision Deck

So as stated before, the objectives changes from the beginning of the internship for different reasons. We ended up doing an open source software component of the UTA method and we published on github instead of the DecisionCloud. This gave us the opportunity to develop more the software by doing more functionalities, and allowed us to specify time for research. So the

main reason that we weren't being able to make the tools in the Decision Deck community because Mr. Zam wasn't available to assist me. So we took this opportunity to make the software more developed.
So during the internship the objectives were re-stated as follow:

- Summarize UTA
- Propose an open source software component of the UTA method
- Generate Random alternatives and criteria
- Generate Value Function
- Search literature for existing similar studies, and how to generate realistic random alternatives

2.3 Conduct of the internship

The 29th May, 2017 I started my internship. The first two weeks, was dedicated to get to know the platform MyDraft and create some basic applications. MyDraft is a platform created by KarmicSoft, it specialized in building and running data-driven Web applications. It is a platform for developing business application the easiest way without complexity. During the two weeks, I did all the tutorials available in the site and in addition I created my own project. The project was called *StockApp* where I recreated a project done during the course of *Marché Financier*: Portfolio Management Project. The project consist in collecting stock information and calculating the following information:

- Weekly Returns of assets by calculating its mean and standard deviation.
- Correlation matrix of the stock returns
- Correlation of stock returns with the stock market index
- Beta, Sharpe Ratio and Treynor Ratio
- Portfolio optimization scheme of Markowitz

I succeed in doing most of the functionalities of the project but the last point wasn't possible to do in MyDraft since it needed the Macro used in the Excel.

After learning about the platform MyDraft, I had a meeting with Mr. Cailloux the 15th June, 2017 where he introduced me a little bit about the UTA method and gave me the book: *Multiple Criteria Decision Analysis*, where the chapter 9 contains an explication about the UTA method. I had to understand this method and make a summary about it, so I took this opportunity to learn about LaTeX to create the summary using LaTeX. Till the end of the month of July, we created the summary of UTA method in Latex, and created my own problem to illustrate the method. Since one of the step of the method was to solve a Linear Program, so we created an independent project to solve a simple Linear Program (LP) by using the google ortools (Optimization tools) library. During that time I made some research about literature for existing similar studies, and how to generate realistic random alternatives research. During this time we had a weekly meeting with Mr. Cailloux.

At the end of the month of July, we had a meeting Mr. Cailloux and me to discuss about the software that will represent the UTA method. We designed the architecture, and I coded the program, I didn't took long because during the month before I had already worked on little components of the software. During the month of august our goal was to finish this document and to make simulation and testing the software.

All of the work done is available as open source on my github repository: `https://github.com/elieahd/decision-uta-method`.

So in the next chapter, we will discuss the projects done above, without including the part of MyDraft since we didn't use it in our project due to the change of objectives.

Chapter 3

Description of the projects

3.1 UTA

3.1.1 Introduction of the aid decision

In decision theory we elaborate a study of reasoning underlying the agent choices. Two branches can be broken from the decision theory: giving an advice on how to make the best decisions, or how existing agents actually make decisions.
In multiple-criteria decision analysis (MCDA), the following concepts play a fundamental role in decision-making problems:

1. object of the decision, definition of the set of potential actions(alternatives) and the determination of a problem statement
2. modeling a consistent family of criteria
3. defining a global preference model
4. decision-aid or decision support

Example: Buying a new car
Let's consider we are trying to figure out which car to buy. Using the methodology represented above, we state that the objective of this problem is **buying a new car**. After stating the objective, we can list the potential of actions that represent the list of cars that we may buy. By potential action, we designate that which constitutes the object of the decision. An action is qualified as potential when it is deemed possible to implement it or if it has some interest within the decision aiding process. So the following list represent the potential actions of this example:

- Peugeot 208 GTi
- Nissan Sentra
- Citroen C4
- Peugeot 308 berline

After listing the list of potential cars, we can define a list of criteria that we will base our decision on. A criterion is constructed to evaluate and compare potential actions according to a point of view. So when defining the list of criteria you should always remember that they must be easy to evaluate (easy to convert to a scale) and should be logical. Let's say we will base our purchase on the following criteria:

- price (in Euro)
- comfort (0, +, ++, +++) *0 being not comfortable and +++ very comfortable*
- safety (1, 2, 3, 4, 5) *1 being not safety and 5 safe*

During the decision process, we will determinate the global preferences of the potential actions:

1. Citroen C4

2. Peugeot 208 GTi

3. Peugeot 308 berline

4. Nissan Sentra

Once the global preference is defined, we can start the decision support.

One of the multi-criteria decision analysis methods is the UTA method, which was proposed by E. Jacquet-Lagrèze and J. Siskos in 1982. This method is proposed by the Multi-Attribute Utility Theory (MAUT) that build a utility function based on the DM[1] preferences.

The UTA method is used to solve a multi-criteria problem by building a utility function based on the preferences of the DM and solving a linear program (LP). It adopt the aggregation-disaggregation principles: where the model is based on a given preferences.
The UTASTAR, a variant of the UTA method, has been considered a better algorithm than UTA. Better result were found using the UTASTAR algorithm. So this is why we will focus on this method rather than the UTA method.
The aim of the UTASTAR method is to estimate a set of additive utility functions which are as consistent as possible with the decision maker's preferences.
At the beginning of the problem, the DM should present the following information

- rank of the actions
- give the criteria he want to base his decision on
- evaluate the action compared to the criterion

Once those information are presented, the UTASTAR algorithm can be executed.

[1] Decision Maker

3.1.2 Principles and Notation

Let's call $A = a, b, c, ...$ the set of potential actions and $g_1, g_2, g_3, ..., g_n$ the family of criteria. Where $g_i(a)$ represent the funtion of an action(alternative)a on the criteria g_i with $a \in A_R$.

We define g_{i*} as the least preferred criteria: $g_{i*} = min_{a \in A} g_i(a)$ and g_i^* as the most preferred criteria: $g_i^* = max_{a \in A} g_i(a)$. So the interval for each criteria g_i is: $[g_{i*}, g_i^*]$.

If we want to evaluate two actions, for example a and b, on only one criteria g_i we have the following relations:

$$\begin{cases} a \succ b \Leftrightarrow g_i(a) > g_i(b) & preference \\ a \sim b \Leftrightarrow g_i(a) = g_i(b) & indifference \end{cases} \tag{3.1}$$

The criteria aggregation model in UTASTAR has the following form:

$$v(g(a)) = \sum_{i=1}^{n} v_i(g_i(a)) \tag{3.2}$$

subject to normalization constraints:

$$\begin{cases} \sum_{i=1}^{n} v_i(g_i^*) = 1 \\ v_i(g_{i*}) = v_i(g_i^1) = 0, \forall i = 1, 2, ..., n \end{cases} \tag{3.3}$$

where $v_i, i = 1, 2, ..., n$ are non decreasing real valued function.
In UTASTAR we have

$$w_{ij} = v_i(g_i^{j+1}) - v_i(g_i^j) \geq 0 \quad \forall i \quad j \tag{3.4}$$

Which will allow us to write:

$$v_i(g_i^j) = \sum_{t=1}^{j-1} w_{it} \quad \forall i = 1, 2, ..., n \quad and \quad j = 2, 3, ..., \alpha_i - 1 \tag{3.5}$$

With the evaluation of an action a $g(a) = [g_1(a), g_2(a), ..., g_n(a)]$, we have the following relation:

$$\begin{cases} v[g(a)] > v[g(b)] \Leftrightarrow a \succ b \\ v[g(a)] \sim v[g(b)] \Leftrightarrow a = b \end{cases} \tag{3.6}$$

3.1.3 Development

The updated version of UTA, UTASTAR, propose a double error function for each action: $\sigma^+(a)$ and $\sigma^-(a)$. So the value of each alternative $a \in A_R$ can be written:

$$v^{'}[g(a)] = \sum_{i=1}^{n} v_i[g_i(a)] - \sigma^+(a) + \sigma^-(a) \quad \forall a \in A_R \tag{3.7}$$

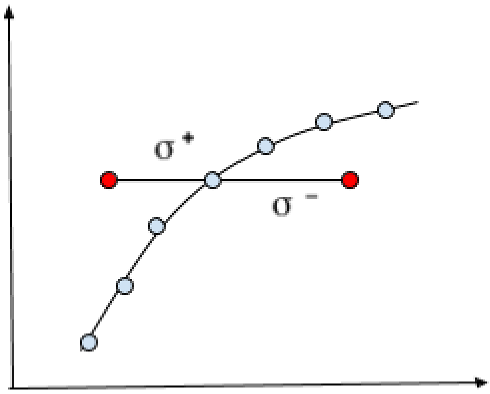

Figure 3.1 – Error function in UTASTAR

For each criteria, the interval $[g_{i*}, g_i^*]$ is cut into $(\alpha_i - 1)$ equals interval, and the end points g_i^j are given by the formula:

$$g_i^j = g_{i*} + \frac{j-1}{\alpha_i - 1}(g_i^* - g_{i*}) \forall j = 1, 2, ..., \alpha_i \tag{3.8}$$

The marginal value of an action a is calculated by a linear interpolation

$$v_i[g_i(a)] = v_i(g_i^j) + \frac{g_i(a) - g_i^j}{g_i^{j+1} - g_i^j}[v_i(g_i^{j+1}) - v_i(g_i^j)] \tag{3.9}$$

The set of reference action $A_R = a_1, a_2, ..., a_m$ is arranged where a_1 is the best action and a_m is the worst action. Which indicate that we have two possible situations:

- $a_k \succ a_{k+1}$ preference
- $a_k \sim a_{k+1}$ indifference

So if we have that $\Delta(a_k, a_{k+1}) = v^{'}[g(a_k)] - v^{'}[g(a_{k+1})]$ and δ is a small positive number we will obtain the following relations:

$$\begin{cases} \Delta(a_k, a_{k+1}) \geq \delta \\ \Delta(a_k, a_{k+1}) = 0 \end{cases} \tag{3.10}$$

The marginal value functions are estimated by means of the Linear Programm with (3.2), (3.3), (3.10) as constraints, and an objective function depending on the σ^+ and σ^-:

$$[min]z = \sum_{k=1}^{m}[\sigma^+(a_k) + \sigma^-(a_k)]$$

subject to:

$$\begin{cases} \Delta(a_k, a_{k+1}) \geq \delta \quad or \quad \Delta(a_k, a_{k+1}) = 0 \\ \sum_{i=1}^{n} \sum_{j=1}^{\alpha_i - 1} w_{ij} = 1 \\ w_{ij} \geq 0, \quad \sigma^+(a_k) \geq 0, \quad \sigma^-(a_k) \geq 0, \quad \forall i, jandk \end{cases} \tag{3.11}$$

3.1.4 Example - Buying New Car

The implementation of UTASTAR algorithm is illustrated by an example I made: **buying a new car**. Another example is available in the Appendices, Choice of transportation, this example was taken from the book: Multiple Criteria Decision Analysis.
The DM is interested only in the following criteria:

- price (in Euro)
- comfort (0, +, ++, +++) *0 being not comfortable and +++ very comfortable*
- safety $(1, 2, 3, 4, 5)$ *1 being not safety and 5 safe*

The evaluation of the previous criteria is presented in the following table:

Cars	Price	Comfort	Safety	Ranking of the DM
Nissan Sentra (ns)	17 000	+++	4	1
Citroen C4 (c4)	15 000	++	2	2
Peugeot 208 GT (p208)	25 000	+	3	3
Peugeot 308 berline (p308)	18 500	0	3	4

First of all, we should specify the scale [2] for each criteria.

- Price $\Rightarrow$ $[g_{1*}, g_1^*] = [25\,000, 20\,000, 15\,000]$
- Comfort $\Rightarrow$ $[g_{2*}, g_2^*] = [0, +, ++, +++]$
- Safety $\Rightarrow$ $[g_{3*}, g_3^*] = [1, 3, 5]$

According to this formula: $v(g(a)) = \sum_{i=1}^{n} v_i(g_i(a))$, the value of each alternative may be written:

- $v(g(ns)) = 0.4v_1(15\,000) + 0.6v_1(20\,000) + v_2(+++) + 0.5v_3(3) + 0.5v_3(5)$
- $v(g(c4)) = v_1(15\,000) + v_2(++) + 0.5v_3(1) + 0.5v_3(3) = v_1(15\,000) + v_2(++) + 0.5v_3(3)$
- $v(g(p208)) = v_1(25\,000) + v_2(+) + v_3(3) = v_2(+) + v_3(3)$
- $v(g(p308)) = 0.3v_1(15\,000) + 0.7v_1(20\,000) + v_2(0) + v_3(3) = 0.3v_1(15\,000) + 0.7v_1(20\,000) + v_3(3)$

We have that $v_1(25\,000) = v_2(0) = v_3(1) = 0$.
Since the marginal value $v_i(g_i)$ can be expressed in terms of variables w_{ij}: $v_i(g_i^j) = \sum_{t=1}^{j-1} w_{it}$, the value of each alternative can be written:

- $v(g(ns)) = w_{11} + 0.4w_{12} + w_{21} + w_{22} + w_{23} + w_{31} + 0.5w_{32}$
- $v(g(c4)) = w_{11} + w_{12} + w_{21} + w_{22} + 0.5w_{31}$
- $v(g(p208)) = w_{21} + w_{31}$
- $v(g(p308)) = w_{11} + 0.3w_{12} + w_{31}$

For each pair of consecutive alternatives, we express the difference between them:

- $\Delta(ns, c4) = -0.6w_{12} + w_{23} + 0.5w_{31} + 0.5w_{32} - \sigma_{ns}^{+} + \sigma_{ns}^{-} + \sigma_{c4}^{+} - \sigma_{c4}^{-}$
- $\Delta(c4, p208) = w_{11} + w_{12} + w_{22} - 0.5w_{31} - \sigma_{c4}^{+} + \sigma_{c4}^{-} + \sigma_{p208}^{+} - \sigma_{p208}^{-}$
- $\Delta(p208, p308) = w_{21} - w_{11} - 0.3w_{12} - \sigma_{p208}^{+} + \sigma_{p208}^{-} + \sigma_{p308}^{+} - \sigma_{p308}^{-}$

[2] the interval $[g_{i*}, g_i^*]$ is cut into equal intervals

Having $\delta = 0.05$, we can solve the following LP:

Objective:

$$Minimize \quad \sum_{a \in A} \sigma_a^+ + \sigma_a^- \tag{3.12}$$

Subject to:

$$\begin{cases} -0.6w_{12} + w_{23} + 0.5w_{31} + 0.5w_{32} - \sigma_{ns}^+ + \sigma_{ns}^- + \sigma_{c4}^+ - \sigma_{c4}^- \geq 0.05 \\ w_{11} + w_{12} + w_{22} - 0.5w_{31} - \sigma_{c4}^+ + \sigma_{c4}^- + \sigma_{p208}^+ - \sigma_{p208}^- \geq 0.05 \\ w_{21} - w_{11} - 0.3w_{12} - \sigma_{p208}^+ + \sigma_{p208}^- + \sigma_{p308}^+ - \sigma_{p308}^- \geq 0.05 \\ w_{11} + w_{12} + w_{21} + w_{22} + w_{23} + w_{31} + w_{32} = 1 \end{cases} \tag{3.13}$$

An optimal solution is $w_{12} = 0.34$, $w_{21} = 0.152$, $w_{32} = 0.51$ with $\sum_{a \in A} \sigma_a^+ + \sigma_a^- = 0$. The utilities found for each alternative are as follows:

- $v(g(ns)) = 0.543$
- $v(g(c4)) = 0.492$
- $v(g(p208)) = 0.152$
- $v(g(p308)) = 0.102$

Those utilities are consistent with the DM's preference ranking.

The UTA method build a utility function based on the preferences of the DM and it consist in solving a linear program (LP) to solve a multi-criteria problem.
This method will elaborate a model of preferences which is as similiar as possible to the DM's preferences.
The improved version of the UTA, UTASTAR, has performed better than the regular method.

3.2 Linear Program Solver

One of the steps of the UTA algorithm is solving the Linear Program. So we can complete the UTA algorithm, I created an independant java application that has the objectif of solving the LP by finding the optimal solution.
So i can achieve this goal I had to use the google ortools (Optimization tools) library. You can find those library in the github repository under src/libs.

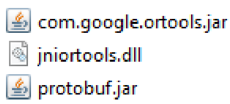

Figure 3.2 – Library used in the solver

Let's talk the following example:

Objective:

$$Maximize \quad 10x_1 + 6x_2 + 4x_3 \tag{3.14}$$

subject to:

$$\begin{cases} x_1 + x_2 + x_3 \leq 100 \\ 10x_1 + 4x_2 + 5x_3 \leq 600 \\ 2x_1 + 2x_2 + 6x_3 \leq 300 \end{cases} \tag{3.15}$$

So we can solve the LP, we need to create an instance of the solver:

```java
MPSolver solver = new MPSolver(
    "LinearProgrammingExample",
    MPSolver.OptimizationProblemType.valueOf("GLOP_LINEAR_PROGRAMMING")
);
```

Figure 3.3 – Java code to define the solver

After that, we define the 3 variables x_1, x_2 et $x_3 \quad \in \quad [0; \infty]$:

```java
MPVariable x1 = solver.makeNumVar(0.0, infinity, "x1");
MPVariable x2 = solver.makeNumVar(0.0, infinity, "x2");
MPVariable x3 = solver.makeNumVar(0.0, infinity, "x3");
```

Figure 3.4 – Java code to define the variables of the LP

We define the objective $Maximize \quad 10x_1 + 6x_2 + 4x_3$:

```java
MPObjective objective = solver.objective();
objective.setCoefficient(x1, 10);
objective.setCoefficient(x2, 6);
objective.setCoefficient(x3, 4);
objective.setMaximization();
```

Figure 3.5 – Java code to define the objective of LP

We do the same for the constraint:

```java
MPConstraint c0 = solver.makeConstraint(-infinity, 100.0);
c0.setCoefficient(x1, 1);
c0.setCoefficient(x2, 1);
c0.setCoefficient(x3, 1);

MPConstraint c1 = solver.makeConstraint(-infinity, 600.0);
c1.setCoefficient(x1, 10);
c1.setCoefficient(x2, 4);
c1.setCoefficient(x3, 5);

MPConstraint c2 = solver.makeConstraint(-infinity, 300.0);
c2.setCoefficient(x1, 2);
c2.setCoefficient(x2, 2);
c2.setCoefficient(x3, 6);
```

Figure 3.6 – Java code to define the constraints of the LP

After setting all the constraints and variables we can execute the solver:

```java
final MPSolver.ResultStatus resultStatus = solver.solve();
```

Figure 3.7 – Java code for running the solver

After executing the solver, we display the optimal value of the objective and the value of the variables:

```java
System.out.println("Optimal objective value = " + solver.objective().value());
System.out.println("x1 = " + x1.solutionValue());
System.out.println("x2 = " + x2.solutionValue());
System.out.println("x3 = " + x3.solutionValue());
```

Figure 3.8 – Java code for displaying the result of the solver

Once you run the program, you will have the following the result:

$$\text{Optimal objective value} = 733.3333333333$$
$$x1 = 33.33333333336$$
$$x2 = 66.66666666666$$
$$x3 = 0.0$$

3.3 Research

One aspect of this internship is the research. The objective of the research was to find similar studies, how to generate realistic random alternatives and studies made on UTA method.

3.3.1 Comparative analysis of UTA multicriteria methods

This paper written by Michel Beuthe and Giuseppe Scannella and it is mainly about the variants of UTA method. As discussed in this paper a method of multi-criteria analysis is chosen depending on the circumstances of the decision making.UTA makes possible the estimation of a nonlinear additive function, which is obtained by the use of a linear program and the only information required from the decision maker are the global preferences between projects.
After the execution of the algorithm we should always be possible to converse with the decision maker in order to specify the precision of the preferences stated
The value δ must not be given too high an initial value. In the basic model, it was noted that the values given to δ were to some extent arbitrary.

Simulations

The simulation are applied to the case of 353 road projects in Belgian Network during the period 1985-2010. The Center of Road Research (CRR) realized a multi criteria analysis that used 29 criteria regrouped in six main themes:

- safety on the present road
- projects socioeconomic aspects
- impact on environment
- current and future traffic
- problems of planning and urbanism
- wear state of the current road.

The 29 criteria has been established with a scale going from 1 to 5.

Variants of UTA

1. UTA
2. UTASTAR
3. UTA2
4. UTAMKEN
5. UTAMP1

6. UTAMP2
7. UTAMIME
8. UTASTARMIME
9. UTA2MIME

Conclusions made

When there is no error in the utility function estimation, $\sum_{a \in A} \sigma_a^+ + \sigma_a^- = 0$, the basic UTA method provides the most practical and efficient method of estimation.
Even when there is interdependence between criteria, the UTA approach provides good results.
In the case where the utility function estimation is positive, $\sum_{a \in A} \sigma_a^+ + \sigma_a^- > 00$, the UTASTAR appears more reliable.
The simulations results indicate that small value of δ lead to better results in case of a positive utility function estimation. But in case of no error in the utility function estimation, larger value of δ can provide better results. The use of UTAMP1 or UTAMP2 may then be used to find the practical upper bound of the values given.

3.3.2 Disadvantages of the UTA method

1. We can always question the decision marker about his preferences
2. Solution may not be the only one as in any LP we can have different solutions

3.4 Implementation - UTA Calculator

We wanted to implement the UTASTAR method. In order to make it possible we created this java program: AlternativeCriteria. So basically we will generate a decision problem and run the UTASTAR algorithm.
Another objective we were trying to achieve was to generate random alternatives and criteria, so we can generate a decision problem.
After solving the problem our goal was to get the v^R and v^T and compare them.

3.4.1 Architecture

Class Diagram

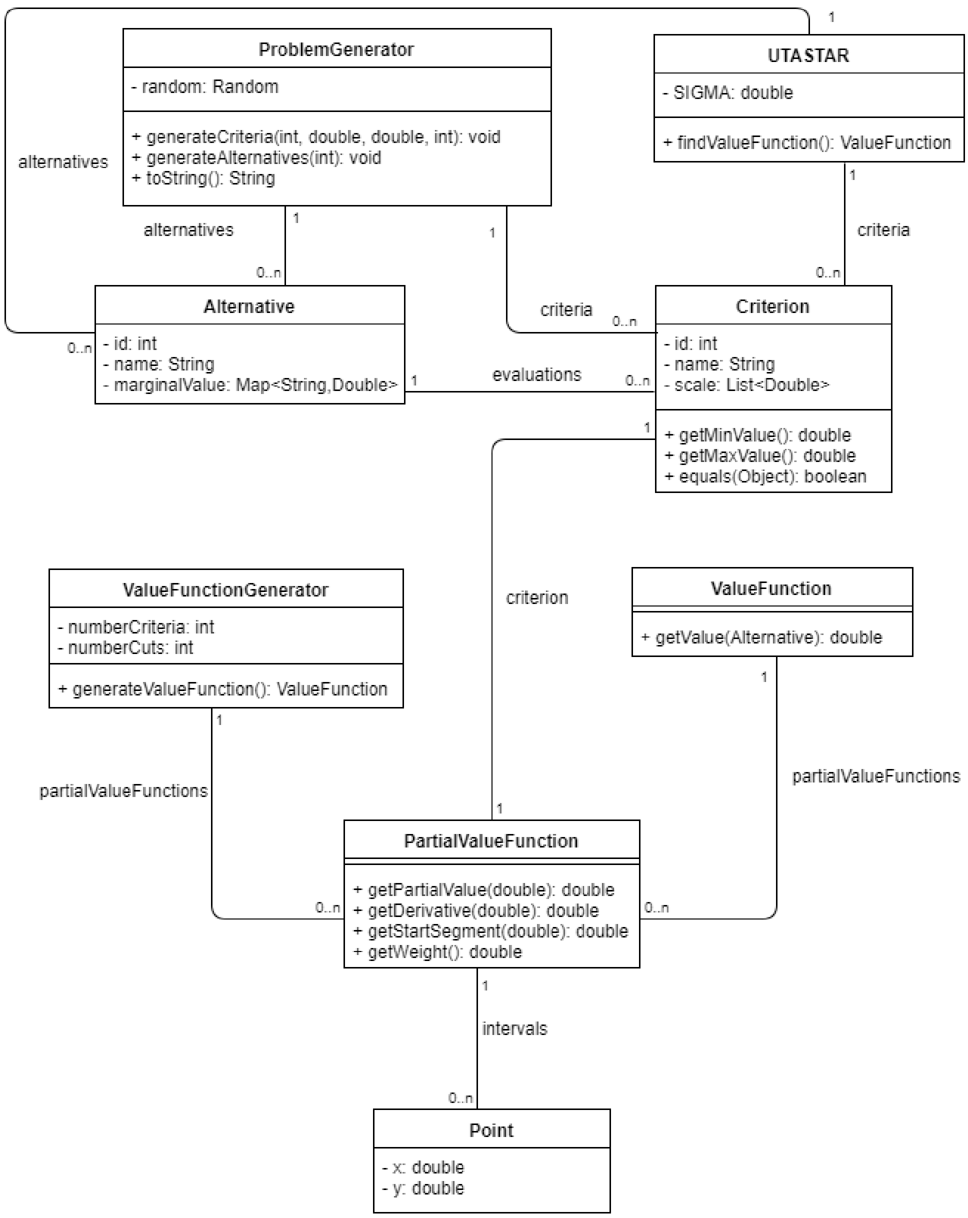

Figure 3.9 – Class Diagram

Other than the external libraries used in the resolution of a LP, no library is needed.

Maven

We figured out that we need to configure our project in Maven. Maven has been created to make a standard way to build a project and a way to share JAR files through many projects. Figuring out dependencies for small projects is not hard. But once the dependencies expand it will get out of hand.

Since we wanted to create an open source project, following the standards conventions was a must which led us to the use of Maven. Another plus of using Maven is that it provides a quality project information.

I personally prefer using the IDE Eclipse Luna to code in Java and I am currently using the version Luna 4.4.2 . So with Maven this won't be a big deal for people who don't use Eclipse, they can use NetBeans or IDEA or any other IDE that supports Maven.

Configuration of Maven:

- groupId: io.github.oliviercailloux
- artifactId: uta-calculator

3.4.2 Functionalities

The following list represent the functionalities of the program I worked on:

- Program your own decision problem
- Generate a random decision problem
- Solve the decision problem by using UTASTAR algorithm
- Elaborate a ValueFunction from a decision problem
- Generate a ValueFunction
- Available as open source

3.4.3 Utils

To make some function more generic we created a package called utils, where we can find 2 java classes:

- NumbersGenerator
- ScaleGenerator

NumbersGenerator

The goal of this util class is to generate decimals numbers that have a certain sum. For example, If you want to create 3 numbers that have the sum of 1.
To make this possible, the class NumbersGenerator has the method generate that takes the following as parameters:

1. counter (int)
2. targetSum (double)

The parameter counter represent the counter of numbers we want to generate, and the parameter targetSum represent the target sum we want.
You can specify the Random value in case you want to have the same result when doing testing.
This method will return a List of number of type double.

ScaleGenerator

The goal of this util class is to generate scale for a criterion, where the interval $[minValue, maxValue]$ is cut into equal intervals. For example, If you want the scale of criterion that have 10 as minimum value, 20 as maximum value and with 4 cuts, the function should return the following list: $[10.0, 13.333, 16.667, 20.0]$.
To make this possible, the class ScaleGenerator has the method generate that takes the following as parameters:

1. minValue (double)
2. maxValue (double)
3. cuts (int)

The parameter minValue represent the minimum value of a criterion, the parameter maxValue represent the maximum value of a criterion and the parameter cuts represent the numbers of cuts you want to cuts the interval.
This method will return a List of number of type double.

3.4.4 Simulation and Comparison

TODO: To be made after the remarks of the program.

3.4.5 Future Improvements

This program has been done during the 3 month internship. Definitely it didn't take 3 month to make this program, but to been able to make such a program I had to make a research and learn about the UTA method. Basically we had to some cuts in order of complexity and compromise in some categories to make the implementation easier. And since the program will be available as an open source on GitHub, it has the potential to grow with some improvements. The following list is some of the improvements that could be made:

1. In the current version of the program, the criterion evaluation has a minimum value and maximum value, so the evaluation of a action is made in the range $[minValue, maxValue]$ with minValue being the least preferred criterion and maxValue the most preferred. But what if we want to represent a criterion that has maxValue as the least preferred criterion and minValue as the most preferred, like the criterion price. In the current version, this won't be possible.
2. In our program, a criterion has a minimum value and maximum value, but what if the criterion is not a quantitative item, not evaluated in numbers. For example, let's say we want to evaluate the comfort, we can have the following values: 0, $+$, $++$, $+++$ or not comfortable, basic, comfortable, very comfortable. So as an improvement we can expect to be able to design all of the criterion possible.

TODO: To be continued...

Chapter 4

Conclusion

4.1 Difficulties encountered

4.2 Acquired Skills

TODO: To be made this week...

Bibliography

[1] GRECO S., EHRGOTT M., RUI FIGUEIRA J., 2004, *Multiple Criteria Decision Analysis*

[2] SISKOS J., 1979, *Cahier du LAMSADE n24 : Les programmes UTA*

[3] SISKOS J., YANNACOPOULOS D., 1982, *Cahier du LAMSADE n49 : Amélioration de la méthode UTA par introduction d'une double fonction d'erreurs*

[4] HAMMAMI A., 2003, *Thèse : Modélisation technico-économique d'une chaîne logistique dans une entreprise reseau*

[5] LUO H., SUN Z., 2014, *A study on Stock Ranking and Selection Strategy Based on UTA Method under the Condition of Inconsistence*

[6] ZOPOUNIDIS C., DOUMPOS M., 1997, *A multicriteria decision aid methodology for the assessment of country risk*

[7] EHSANIFAR M., ESHLAGHI A., KERAMATI M., NAZEMI J., 2013, *The Development of UTASTAR Method in Fuzzy Environment for Supplier Selection*

[8] SISKOS J., 1983, *Application de la méthode UTA à un problème de sélection de points de vente mettant en jeu des critères multiples*

List of Figures

Appendices

Annexe A

Compte Rendus avec M. Cailloux

A.1 CR1 - 15 juin 2017

15 juin 2017 / 15 :00 / Université Paris Dauphine

Participants

Olivier Cailloux, Elie Daher

Déroulement de la réunion

- Explication du plan du stage
- J'ai présenté les tâches effectuées
- J'avais quelques questions sur la méthode UTA et plus précisément sur un exemple "A Numerical Example" du chapitre 9 : "UTA Method" du livre : "Multiple Criteria Decision Analysis"
- Présentation des tâches à effectuer

Les tâches à faire

- Créer un repository sur github
- Ecrire un résumé sur la méthode UTA en utilisant LaTeX
- En parallèle, effectuer des recherches sur des littératures existantes

Pièces jointes

- First iteration : document présenté par M. Cailloux qui contient le plan de la première itération

La réunion a été levée à 15 :30

La prochaine réunion est prévue pour le mercredi 21 juin 2017 à 14 :00

A.2 CR2 - 21 juin 2017

21 juin 2017 / 14 :00 / Université Paris Dauphine

Participants

Olivier Cailloux, Elie Daher

Déroulement de la réunion

- Présentation du travail effectué
- Résolution du projet java sur le programme linéaire

Les tâches à faire

- Rendre le document plus détaillé et développé
- Rendre le document plus compréhensible pour les lecteurs qui n'ont pas un context décisionnel
- Fixer le programme linear programming sur Java

La réunion a été levée à 14 :40

La prochaine réunion est prévue pour le mercredi 28 juin 2017 à 11 :00

A.3 CR3 - 28 juin 2017

28 juin 2017 / 11 :00 / Université Paris Dauphine

Participants

Olivier Cailloux, Elie Daher

Déroulement de la réunion

- Présentation du travail effectué
- Citer les remarques et les modifications à réaliser sur le document summary-uta et sur le repository de github (enlever le folder qui contient les *.class)
- Présentation du travail à effectuer pour la semaine prochaine

Les tâches à faire

- Corriger les remarques sur le document
- Développer v^R et v^T
- Effectuer une recherche sur les littératures similaires déjà réalisées

La réunion a été levée à 11 :30

La prochaine réunion est prévue pour le mercredi 5 juillet 2017 à 10 :00

A.4 CR4 - 18 juillet 2017

18 juillet 2017 / 10 :00 / Université Paris Dauphine

Participants

Olivier Cailloux, Elie Daher

Déroulement de la réunion

- Présentation du travail effectué
- Explication de la fusion entre le projet Alternative-Criteria et UTA
- Apprendre comment télécharger des littératures à partir du site de la BU de Dauphine

Les tâches à faire

- Elaborer un document qui regroupe tout le travail effectué
- S'approfondir dans la recherche des littératures
- Fixer la méthode GenerateNumbers

La réunion a été levée à 10 :45

La prochaine réunion est prévue pour le mardi 25 juillet 2017 à 10 :00

A.5 CR5 - 25 juillet 2017

25 juillet 2017 / 10 :00 / Université Paris Dauphine

Participants

Olivier Cailloux, Elie Daher

Déroulement de la réunion

- Présentation du travail effectué
- Explication des modifications à faire dans le programme UTA
- Explication des modification à effectuer dans le document report

Les tâches à faire

- Ajouter les annexes des littérature dans le document
- Définir l'objectif de la méthode UTA dans le document
- Etablir les modifications sur le programme UTA
- Elaborer des résumés pour les articles trouvés

La réunion a été levée à 11 :15

La prochaine réunion est prévue pour le lundi 31 juillet 2017 à 10 :00

A.6 CR6 - 31 juillet 2017

31 juillet 2017 / 10 :00 / Café Lumière - Hôtel Scribe

Participants

Olivier Cailloux, Elie Daher

Déroulement de la réunion

- Présentation du travail effectué
- Refaire l'architecture du programme
- Présenter les modifications à faire

Les tâches à faire

- Elaborer des petits résumés pour les articles trouvés
- Compléter le rapport final
- Effectuer les modifications sur le programme

La réunion a été levée à 11 :15

Annexe B

Example of UTASTAR - Analyzing the choice of transportation

A DM wants to analyse the choice of transportation. The DM is interstered in the following criteria

1. price
2. time (min)
3. comfort (possibility to have a seat)

The evaluation of the previous criteria is presented in the following table :

Means of transportation	Price	Time	Comfort	Ranking of the DM
RER	3	10	+	1
METRO (1)	4	20	++	2
METRO (2)	2	20	0	2
BUS	6	40	0	3
TAXI	30	30	+++	4

DM's preferences : $RER \succ Metro1 \approx Metro2 \succ Bus \succ Taxi$

First of all, we should specify the scale [1] for each criteria.

- Price $\rightarrow$ $[30, 16, 2]$
- Time $\rightarrow$ $[40, 30, 20, 10]$
- Comfort $\rightarrow$ $[0, +, ++, +++]$

According to this formula : $v(g(a)) = \sum_{i=1}^{n} v_i(g_i(a))$, the value of each alternative may be written :

- $v[g(RER)] = 0.07v_1(16) + 0.93v_1(2) + v_2(10) + v_3(+)$
- $v[g(METRO1)] = 0.14v_1(16) + 0.86v_1(2) + v_2(20) + v_3(++)$
- $v[g(METRO2)] = v_1(2) + v_2(20) + v_3(0) = v_1(2) + v_2(20)$
- $v[g(BUS)] = 0.29v_1(16) + 0.71v_1(2) + v_2(40) + v_3(0) = 0.29v_1(16) + 0.71v_1(2)$
- $v[g(TAXI)] = v_1(30) + v_2(30) + v_3(+++) = v_2(30) + v_3(+++)$

We have that $v_1(30) = v_2(40) = v_3(0) = 0$.
Since the marginal value $v_i(g_i)$ can be expressed in terms of variables w_{ij} : $v_i(g_i^j) = \sum_{t=1}^{j-1} w_{it}$, the value of each alternative can be written :

- $v[g(RER)] = w_{11} + 0.93w_{12} + w_{21} + w_{22} + w_{23} + w_{31}$
- $v[g(METRO1)] = w_{11} + 0.86w_{12} + w_{21} + w_{22} + w_{31} + w_{32}$
- $v[g(METRO2)] = w_{11} + w_{12} + w_{21} + w_{22}$
- $v[g(BUS)] = w_{11} + 0.71w_{12}$
- $v[g(TAXI)] = w_{21} + w_{31} + w_{32} + w_{33}$

For each pair of consecutive alternatives, we express the difference between them :

- $\Delta(RER, METRO1) = 0.07w_{12} + w_{23} - w_{32} \geq \delta$
- $\Delta(METRO1, METRO2) = -0.14w_{12} + w_{31} + w_{32} = 0$
- $\Delta(METRO2, BUS) = 0.29w_{12} + w_{21} + w_{22} \geq \delta$
- $\Delta(BUS, TAXI) = w_{11} + 0.71w_{12} - w_{21} - w_{31} - w_{32} - w_{33} \geq \delta$

Having $\delta = 0.05$, we can solve the following LP :

Objective :

$$Minimize \quad \sum_{a \in A} \sigma_a^+ + \sigma_a^- \tag{B.1}$$

Subject to :

$$\begin{cases} 0.07w_{12} + w_{23} - w_{32} - \sigma^+_{RER} + \sigma^-_{RER} + \sigma^+_{METRO1} - \sigma^-_{METRO1} \geq \delta \\ -0.14w_{12} + w_{31} + w_{32} - \sigma^+_{METRO1} + \sigma^-_{METRO1} + \sigma^+_{METRO2} - \sigma^-_{METRO2} = 0 \\ 0.29w_{12} + w_{21} + w_{22} - \sigma^+_{METRO2} + \sigma^-_{METRO2} + \sigma^+_{BUS} - \sigma^-_{BUS} \geq \delta \\ w_{11} + 0.71w_{12} - w_{21} - w_{31} - w_{32} - w_{33} - \sigma^+_{BUS} + \sigma^-_{BUS} + \sigma^+_{TAXI} - \sigma^-_{TAXI} \geq \delta \\ w_{11} + w_{12} + w_{21} + w_{22} + w_{23} + w_{31} + w_{32} + w_{33} = 1 \end{cases} \tag{B.2}$$

So by using the com.google.ortools library, we can solve the Linear Program above with $\sigma = 0.05$.
This Linear Program solution is coded in Java class ChoiceTransportation.
By executing the class ChoiceTransportation, you will have the following result :

An optimal solution is $w_{11} = 0.5$, $w_{22} = 0.05$, $w_{23} = 0.05$, $w_{33} = 0.4$ with $\sum_{a \in A} \sigma_a^+ + \sigma_a^- = 0$.
The utilities found for each alternative are as follows :

- $v(g(RER)) = 0.6$

[1] the interval $[g_{i*}, g_i^*]$ is cut into equal intervals

- $v(g(METRO1)) = 0.55$
- $v(g(METRO2)) = 0.55$
- $v(g(BUS)) = 0.5$
- $v(g(TAXI)) = 0.4$

Those utilities are consistent with the DM's preference ranking.

Annexe C

Literature found during research

C.1 Literature used in the documentation

- Multiple Criteria Decision Analysis - Salvatore Greco, Matthias Ehrgott, Jose Rui Figueira, 2004
- Cahier du LAMSADE n24 : Les programmes UTA - J. Siskos, Octobre 1979
- Cahier du LAMSADE n49 : Amélioration de la méthode UTA par introduction d'une double fonction d'erreurs - J.Siskos, D. Yannacopoulos, octobre 1983
- Thèse : Modélisation technico-économique d'une chaîne logistique dans une entreprise reseau - Abdelkader Hammami, 2003
- Assessing a set of additive utility functions for multicriteria decision-making, the UTA method
- Assessing non-additive utility for multicriteria decision aid
- Comparative analysis of UTA multicriteria methods
- Preference disaggregation : 20 years of MCDA experience

C.2 Useful literature

- On the expressiveness of the additive value function and the Choquet integral models
- ACUTA : A novel method for eliciting additive value functions on the basis of holistic preference statements
- Stewart - Robustness of Additive Value Function Methods in MCDM (1996)
- La Conception et l'implementation d'un outil d'aide a la decision multicriteres integrant les concepts de la gestion des connaissances et du cycle de vie : application de la chaine d'approvisionnement humanitaire
- A study on Stock Ranking and Selection Strategy Based on UTA Method under the Condition of Inconsistence - Hong-chen Luo, Zhao-xu Sun, aout 2014
- The Development of UTASTAR Method in Fuzzy Environment for Supplier Selection
- A multicriteria decision aid methodology for the assessment of country risk
- Application de la méthode UTA à un problème de sélection de points de vente mettant en jeu des critères multiples