

# Mémoire de recherche

## Méthode UTA

---

Elie Abi Hanna Daher  
04/09/2017

LAMSADE - M. Olivier Cailloux, M. Michel Zam

# Introduction

Introduction, contexte,  
objectifs, déroulement du  
stage

# Introduction

Mémoire de recherche à LAMSADE du 29 mai jusqu'au 1 septembre

Superviseurs : M. Olivier Cailloux, M. Michel Zam

Objectifs :

- Implémenter la méthode UTA
- Étudier la performance de la méthode en fonction de la quantité de données d'apprentissage

Travail effectué à distance, 1 rencontre par semaine avec M. Cailloux pour faire le point

# Contexte - Décision multicritère



- Un problème de décision comporte les éléments suivants :
  - Objectif du problème de la décision
  - Alternatives → actions potentielles
  - Critères → évaluer les alternatives sur les critères
- L'aide à la décision multicritère identifie les alternatives potentielles afin d'élaborer une étude pour prendre une décision.

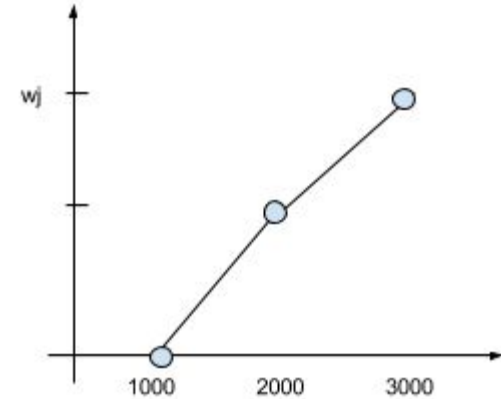
# Contexte - Fonction de valeur

La fonction de valeur est l'ensemble de fonctions de valeur partielle.

Chaque fonction de valeur partielle représente un critère avec ses coordonnées.

La somme des  $w_j$  est égale à 1.

La valeur de chaque alternative est calculée à partir de la fonction de valeur.



Fonction de valeur partielle d'un critère avec une échelle de [1000, 2000, 3000]

# Contexte - UTA



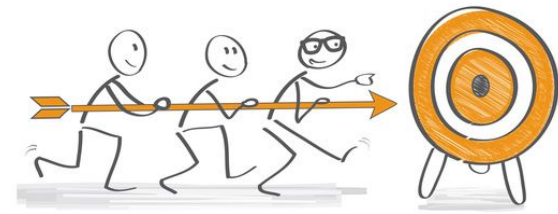
La méthode UTA résout un problème de décision multicritère en générant une fonction de valeur basée sur la préférence du DM et la solution du programme linéaire.

Créer par E. Jacquette-Lagrange et J. Siskos en 1982

L'algorithme UTA élabore un modèle de préférence qui est similaire à celui introduit par le DM

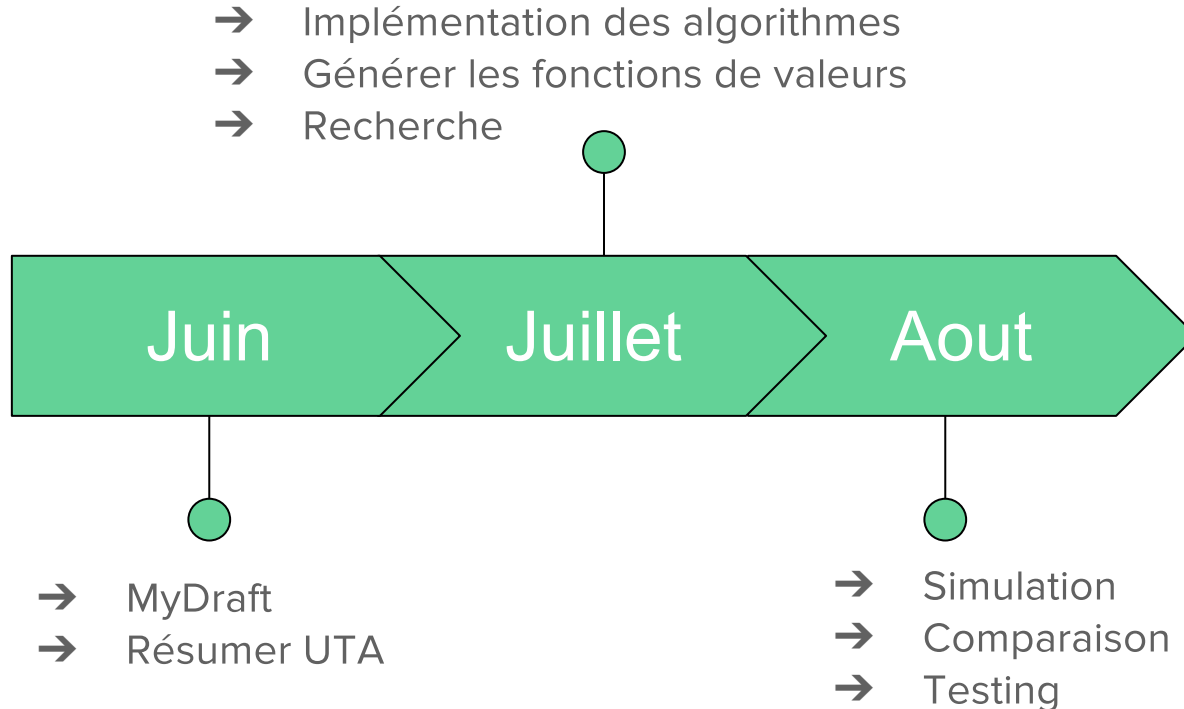
L'algorithme UTASTAR est une variante de UTA.

# Objectifs



- Résumer la méthode de décision UTA
- Rechercher des littératures sur UTA
- Générer aléatoirement des alternatives et critères : Problème de décision
- Implémenter l'algorithme UTA
- Générer aléatoirement une fonction de valeur  **$vR$**
- Générer la fonction de valeur  **$vT$**  à partir de UTA
- Comparer les deux fonctions de valeur  **$vT$**  et  **$vR$**  en effectuant des simulations

# Déroulement du stage





# Projets

MyDraft, PL Solveur,  
implémentation, recherche et  
simulation

# MyDraft

MyDraft est une application proposée par KarmicSoft. C'est une plateforme de développement d'application de gestion sur le Cloud, qui permet la gestion des données métier.

Cette plateforme permet aux non informaticiens de construire des applications seulement par des clics.

J'ai réalisé :

- Tutoriel en anglais
- Projet de StockApp - *Portfolio Management* du cours Marché Financier

# Programme Linéaire Solveur

Résolution du programme linéaire est une des étapes de la méthode UTA

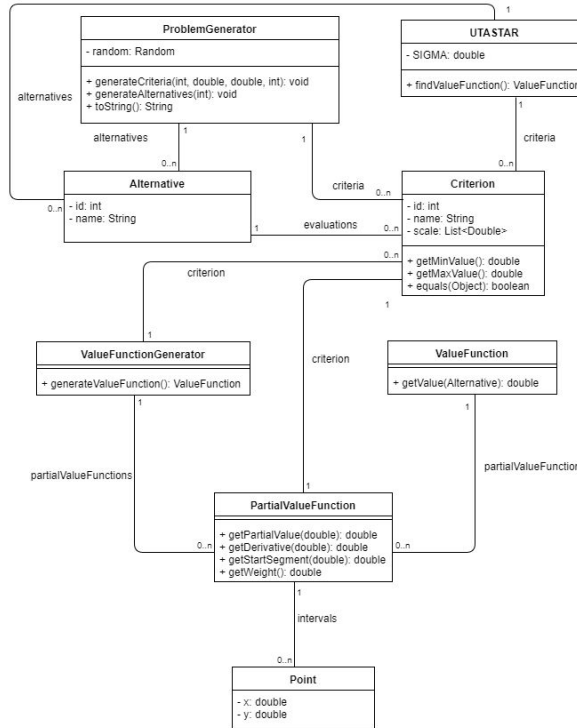
Utilisation de la librairie de google : *ortools*, qui contient 2 librairies externes de type *jar* et 1 Java Dynamic Library

# Implémentation

- Coder en Java
- Maven
- Publier sur GitHub en open source (License MIT)
- Test fonctionnel
- Bibliothèques externes : ortools, Apache Commons Math
- Utils créés : NumbersGenerator, ScaleGenerator

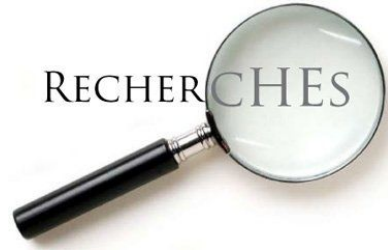


# Diagramme de classes du modèle



# Démo

Voitures	Prix	Confort	Safety	Préférence
Nissan Sentra	17,000	+++	4	1
Citroen C4	15,000	++	2	2
Peugeot 208 GT	25,000	+	3	3
Peugeot 308 Berline	18,500	0	3	4



# Recherche

- Etude de la littérature sur la méthode de décision UTA afin de la comprendre et d'implémenter l'algorithme
- Effectuer une simulation, afin de comparer la fonction de valeur  $v_R$  générer aléatoirement et la fonction de valeur  $v_T$  générer à partir de l'algorithme UTA

# Recherche - Simulation (1 / 3)

- 1- La fonction de valeur **vR** est générée aléatoirement en fixant 10,000 alternatives évaluées sur un nombre de critères (le nombre de critère varie durant la simulation)
- 2- On prend une sous liste de la liste d'alternatives appelée **exampleAlternatives** (le nombre d'exemple varie durant la simulation) et le reste des alternatives est placé sous une autre liste appelée **alternativesToCompare**



# Recherche - Simulation (2 / 3)

3- La fonction de valeur **vT** est générée à partir de l'algorithme UTA sur la liste ***exampleAlternatives***

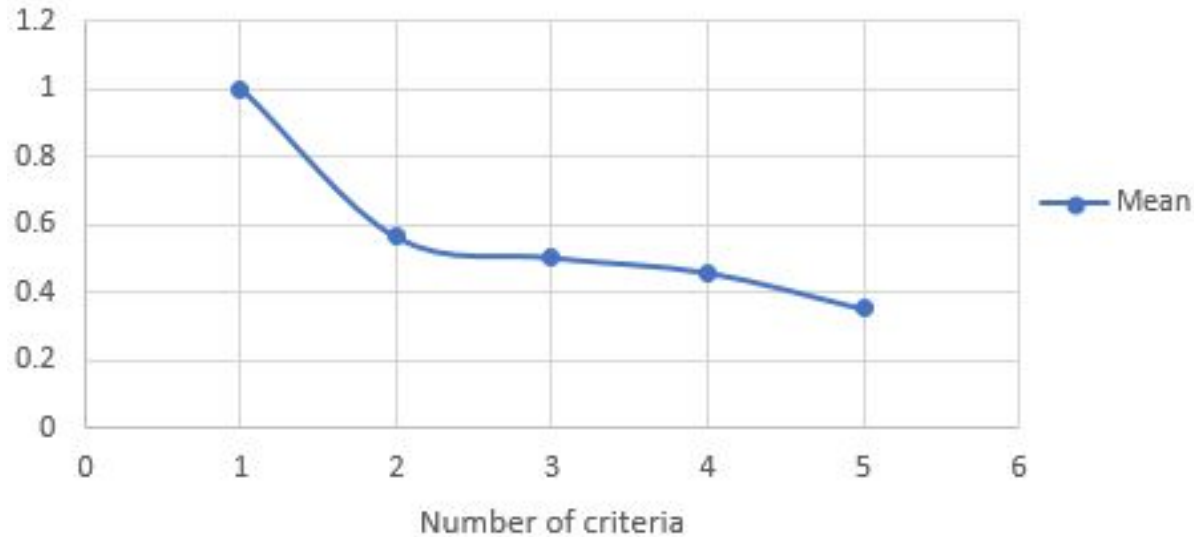
4- On prend la liste ***alternativesToCompare*** puis on ordonne cette liste sur les deux fonctions de valeur **vR** et **vT**, et on calcule le Kendall taux des deux listes ordonnées par **vR** et **vT**.

# Recherche - Simulation (3 / 3)

5- Exécution du processus cité, 500 fois en calculant la moyenne du Kendall taux.

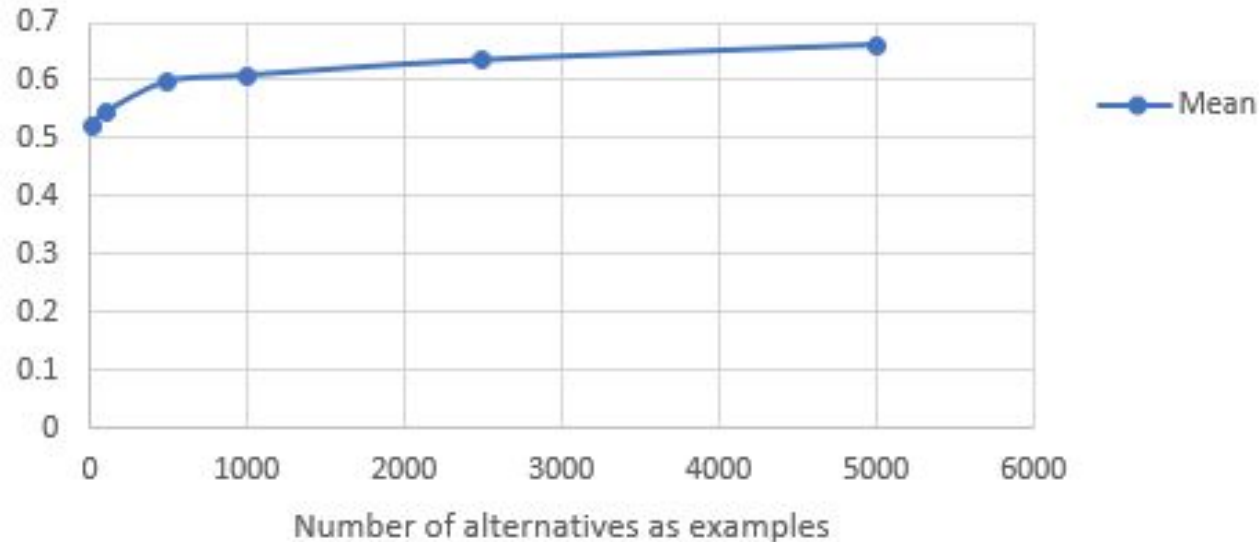
6- Exécution cette simulation en variant le nombre de critère et le nombre d'exemple d'alternatives.

# Simulation - Résultat (1 / 2)



Simulation en variant le nombre de critères, et en fixant le nombre d'exemples à 50

# Simulation - Résultat (2 / 2)



Simulation en variant le nombre d'exemples de 10 jusqu'à 1000 alternatives en fixant le nombre de critère à 2

# Conclusion

Gains personnels, difficultés  
rencontrées

# Apports personnels

- Apprendre une nouvelle méthode de décision, UTA
- Faire des recherches de littérature sur un sujet
- Refaire un projet de *Portfolio Management* du cours du Marché Financier sur MyDraft
- Création des projets en Maven
- Publier des projets sur GitHub
- Ecrire des documents en LaTeX
- Pratiquer l'anglais (rapport réalisé en anglais)

# Difficultés rencontrées

Travailler seul sans être entouré par de collègues. Quand j'avais une question ou un problème je devais attendre jusqu'à la prochaine réunion avec M. Cailloux ou à l'envoyer en mail.



Merci!